

**RULE-BASED GRAPH THEORY TO ENABLE EXPLORATION OF THE
SPACE SYSTEM ARCHITECTURE DESIGN SPACE**

A Dissertation
Presented to
The Academic Faculty

By

Dale Curtis Arney

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

Georgia Institute of Technology

August 2012

Copyright © 2012 by Dale Arney

**RULE-BASED GRAPH THEORY TO ENABLE EXPLORATION OF THE
SPACE SYSTEM ARCHITECTURE DESIGN SPACE**

Approved by:

Dr. Alan W. Wilhite, Chairman
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Trina M. Chytka
Space Mission Analysis Branch
NASA Langley Research Center

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

Dr. Ryan P. Russell
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Vitali V. Volovoi
School of Aerospace Engineering
Georgia Institute of Technology

Date Approved: June 27, 2012

To my wife Jennifer

*Two roads diverged in a wood,
And apparently, we took the longer one.*

TABLE OF CONTENTS

LIST OF FIGURES	vi
LIST OF TABLES	xiii
NOMENCLATURE.....	xv
SUMMARY	xix
CHAPTER 1 INTRODUCTION.....	1
1.1. Motivation.....	1
1.2. Research Goals and Objectives.....	9
1.3. Problem Statement.....	11
1.4. Dissertation Overview	18
CHAPTER 2 BACKGROUND.....	20
2.1 Space System Architecture Modeling.....	20
2.1.1. EXAMINE	20
2.1.2. Object Process Network (OPN).....	23
2.1.3. Logistics Network.....	26
2.1.4. Overview.....	27
2.2. Graph Theory.....	27
2.3. Performance Modeling.....	32
2.4. Cost Estimation.....	36
2.5. Architecture Optimization Methods	39
2.6. System Architecture Evaluation Criteria	44
CHAPTER 3 METHODOLOGY	51
3.1. Applying Graph Theory to a Space System Architecture.....	51
3.2. Graph Generation Model	53
3.3. Design Space Exploration.....	59
3.3.1. System Map Overview.....	59
3.3.2. Rule-Based Graph Traversal.....	62
3.3.3. System Hierarchy.....	66
3.4.2. System Modeling	70

3.5. Exploration of the Space System Architecture Design Space	74
3.5.1. Optimization Method	75
3.5.2. Selection Criterion	78
CHAPTER 4 VALIDATION	80
4.1. Lunar Mission Design Space	80
4.2. Architecture Definition	81
4.3. Analysis Results and Validation	88
4.3.1. Analysis Results	89
4.3.2. Validation	98
CHAPTER 5 FLEXIBLE PATH DESIGN SPACE EXPLORATION	101
5.1. Flexible Path Design Space	101
5.1.1. System Architecture Design Space Representation	102
5.1.2. Baseline System Architectures	106
5.2. Design Space Exploration Results	110
5.2.1. GEO System Architecture Results	111
5.2.2. Lunar System Architecture Results	116
5.2.3. NEO System Architecture Results	121
5.2.4. Evolutionary Exploration Program	125
5.3. Design Space Implications	128
5.3.1. Launch Vehicle Selection	128
5.3.2. Propellant Depots and On-Orbit Refueling	137
5.3.3. Aggregation Strategy	144
5.3.4. Comparison between IMLEO and RNPV	150
5.3.5. Summary	153
CHAPTER 6 CONCLUSIONS AND FUTURE WORK	155
6.1. Conclusions	155
6.2. Future Work	158
APPENDIX A	162
APPENDIX B	168
APPENDIX C	175
C.1. Launch Vehicles	175

C.2. Propellant Depot.....	187
C.3. Aggregation Strategy.....	191
APPENDIX D	200
D.1. Crew	200
D.2. Crew Capsule	200
D.3. Lunar Descent Stage	203
D.4. Lunar Ascent Stage	205
D.5. Launch Vehicle	207
D.6. Propulsive Stage.....	208
D.7. Propellant Depot	211
D.8. Surface Habitat.....	212
D.9. In-Space Habitat.....	213
REFERENCES.....	215

LIST OF FIGURES

Figure 1: Notional Cost, Freedom, and Knowledge in the Design Process [17],[18],[19].	4
Figure 2: Design Reference Architecture 5.0 Trade Tree [16]	6
Figure 3: Possible Sequences in which to Visit Flexible Path Destinations [1]	12
Figure 4: Solution to Evolutionary Mission Sequence Using Graph Theory Model.....	17
Figure 5: iBAT Graphical User Interface [22].....	21
Figure 6: DSM of EXAMINE Framework from <i>Komar et al. (2008)</i> [22].....	23
Figure 7: OPN Example [23]	24
Figure 8: Specified Flights within OPN [23].....	24
Figure 9: Fixed System Sizing Hierarchy for a System Architecture [35].....	25
Figure 10: Example Graph.....	28
Figure 11: Adjacency and Incidence Matrix Representations	29
Figure 12: Regression on Existing Liquid Hydrogen Tanks [41].....	33
Figure 13: Inert Mass Comparison between Simplified Models and Existing Vehicles ..	35
Figure 14: Gross Mass Comparison between Simplified Models and Existing Vehicles	36
Figure 15: Transcost CER for Expendable, Liquid-Propulsion Launch Vehicle Stage [49]	38
Figure 16: Information Flow for Embedded Optimization [21]	41
Figure 17: Figures of Merit from ESAS [7].....	46
Figure 18: DRA 5.0 Architecture Selection Criteria [16].....	47
Figure 19: Procedure for Modeling a Space System Architecture Using Graph Theory .	52
Figure 20: Lunar System Architecture Design Space as a Graph.....	54
Figure 21: Recursively Defined Lunar Architecture Graph with Multiple Flights	55
Figure 22: Adjacency Matrix for Lunar System Architecture Graph	57
Figure 23: Partial Incidence Matrix for Lunar System Architecture Graph	58
Figure 24: Matrix Representation of a Graph Traversal (System Map)	59
Figure 25: ESAS Baseline System Architecture Concept of Operations [7].....	61
Figure 26: System Map for ESAS Baseline System Architecture	62
Figure 27: Algorithm for Rule-Based Graph Traversal	65

Figure 28: System Hierarchy for Lunar Architecture Selected in ESAS [7].....	69
Figure 29: Adjacency Matrix for the System Hierarchy of the ESAS Baseline Architecture.....	69
Figure 30: Incidence Matrix for the System Hierarchy of the ESAS Baseline Architecture	70
Figure 31: Summary of ESAS Baseline Sizing and Cost Estimation (Part 1).....	73
Figure 32: Summary of ESAS Baseline Sizing and Cost Estimation (Part 2).....	74
Figure 33: ACO Algorithm for System Architecture Design Space Exploration.....	76
Figure 34: ESAS Lunar Mission Mode Taxonomy [7]	81
Figure 35: Comparison of CEV SM (left) and EDS (right) Configuration (not to scale) [7]	83
Figure 36: LOR-LOR Lunar System Architecture Option [7].....	84
Figure 37: 1.5-Launch EOR-LOR System Architecture Option [7].....	85
Figure 38: EOR-Direct System Architecture Option [7]	86
Figure 39: Modified EOR-Direct (SM Remains in LLO) System Architecture Option [7]	87
Figure 40: EOR-LOR System Architecture with On-Orbit Refueling and Commercial Launch Vehicles [66]	88
Figure 41: Mass and RNPV Summary Plot for ESAS Mode Analysis	94
Figure 42: Relative DDT&E NPV and Flight Unit NPV Summary Plot for ESAS Mode Analysis.....	95
Figure 43: Loss of Crew (LOC) FOM Comparison from ESAS Mission Modes [7].....	99
Figure 44: GEO System Architecture Design Space as a Graph	103
Figure 45: NEO System Architecture Design Space with HEO Departure as a Graph..	105
Figure 46: NEO System Architecture Design Space with LEO Departure as a Graph ..	105
Figure 47: Baseline GEO System Architecture Concept of Operations	107
Figure 48: Baseline Lunar System Architecture Concept of Operations [7]	108
Figure 49: Baseline NEO System Architecture Concept of Operations	109
Figure 50: Results from GEO System Architecture Design Space Exploration.....	112
Figure 51: GEO System Architecture Design Points that Improve RNPV over the Baseline.....	113

Figure 52: Best GEO System Architecture Concept of Operations.....	116
Figure 53: Results from Lunar System Architecture Design Space Exploration	117
Figure 54: Lunar System Architecture Design Points that Improve RNPV over the Baseline.....	119
Figure 55: Best Lunar System Architecture Concept of Operations	121
Figure 56: Results from NEO System Architecture Design Space Exploration.....	122
Figure 57: NEO System Architecture Design Points that Improve RNPV over the Baseline.....	123
Figure 58: Best NEO System Architecture Concept of Operations.....	125
Figure 59: Evolutionary Exploration Program Capability Development	126
Figure 60: RNPV of GEO System Architectures that Utilize Falcon Heavy Launch Vehicles.....	129
Figure 61: RNPV of GEO System Architectures that Utilize Delta IV-H Launch Vehicles	129
Figure 62: RNPV of GEO System Architectures that Utilize 70 mt HLLVs	130
Figure 63: RNPV of GEO System Architectures that Utilize 100 mt HLLVs	131
Figure 64: RNPV of GEO System Architectures that Utilize 130 mt HLLVs	131
Figure 65: Box and Whisker Plot of RNPV for GEO Architectures that Exclusively Use a Given Launch Vehicle	133
Figure 66: Box and Whisker Plot of RNPV for Lunar System Architectures by Launch Vehicle Type.....	135
Figure 67: Box and Whisker Plot of RNPV for NEO System Architectures that Exclusively Use a Certain Launch Vehicle	136
Figure 68: RNPV of GEO System Architectures that Use Propellant Depots	138
Figure 69: Box and Whisker Plot of RNPV for GEO System Architectures that Use a Propellant Depot	139
Figure 70: RNPV of Lunar System Architectures that Use Propellant Depots	140
Figure 71: Box and Whisker Plot of RNPV for Lunar System Architectures that Use a Propellant Depot	141
Figure 72: RNPV of NEO System Architectures that Use Propellant Depots	142

Figure 73: Box and Whisker Plot of RNPV for Lunar System Architectures that Use a Propellant Depot	143
Figure 74: RNPV of GEO System Architectures for Different Pre-Deploy Strategies ..	145
Figure 75: Box and Whisker Plot of GEO System Architectures for Different Pre-Deploy Strategies.....	146
Figure 76: Box and Whisker Plot for Lunar System Architecture for Different Pre-Deploy Strategies.....	148
Figure 77: Box and Whisker Plot of Lunar System Architectures for Location of Assets during a Surface Mission	149
Figure 78: Box and Whisker Plot of NEO System Architectures for Different Pre-Deploy Strategies.....	150
Figure 79: Comparison of IMLEO and RNPV for the GEO System Architecture Design Space.....	152
Figure 80: Comparison of IMLEO and RNPV for the Lunar System Architecture Design Space.....	152
Figure 81: Comparison of IMLEO and RNPV for the NEO System Architecture Design Space.....	153
Figure 82: Notional Block Diagram of Current Modeling Framework Capability	159
Figure 83: Notional Block Diagram of Future Modeling Framework Capability	160
Figure B-1: Crew Capsule CER for DDT&E Cost (Top) and Flight Unit Cost (Bottom)	168
Figure B-2: Surface Habitat CER for DDT&E Cost (Top) and Flight Unit Cost (Bottom)	169
Figure B-3: In-Space Habitat CER for DDT&E Cost (Top) and Flight Unit Cost (Bottom)	170
Figure B-4: Cryogenic Lunar Ascent Stage CER for DDT&E Cost (Top) and Flight Unit Cost (Bottom).....	171
Figure B-5: Cryogenic Lunar Descent Stage CER for DDT&E Cost (Top) and Flight Unit Cost (Bottom).....	172
Figure B-6: Cryogenic Propulsive Stage CER for DDT&E Cost (Top) and Flight Unit Cost (Bottom).....	173

Figure B-7: Propellant Depot CER for DDT&E Cost (Top) and Flight Unit Cost (Bottom)	174
Figure C-1: Box and Whisker Plot of DDT&E RNPV for GEO System Architectures that Exclusively Use a Certain Launch Vehicle	175
Figure C-2: Box and Whisker Plot of Flight Unit RNPV for GEO System Architectures that Exclusively Use a Certain Launch Vehicle	176
Figure C-3: RNPV of Lunar System Architectures that Utilize a Certain Launch Vehicle	176
Figure C-4: RNPV of Lunar System Architectures that Utilize a Certain Launch Vehicle	177
Figure C-5: RNPV of Lunar System Architectures that Utilize a Certain Launch Vehicle	177
Figure C-6: RNPV of Lunar System Architectures that Utilize a Certain Launch Vehicle	178
Figure C-7: RNPV of Lunar System Architectures that Utilize a Certain Launch Vehicle	178
Figure C-8: Box and Whisker Plot of RNPV for Lunar System Architectures that Exclusively Use a Certain Launch Vehicle	179
Figure C-9: Box and Whisker Plot of DDT&E RNPV for GEO System Architectures that Exclusively Use a Certain Launch Vehicle	180
Figure C-10: Box and Whisker Plot of Flight Unit RNPV for GEO System Architectures that Exclusively Use a Certain Launch Vehicle	181
Figure C-11: RNPV of NEO System Architectures that Utilize a Certain Launch Vehicle	182
Figure C-12: RNPV of NEO System Architectures that Utilize a Certain Launch Vehicle	182
Figure C-13: RNPV of NEO System Architectures that Utilize a Certain Launch Vehicle	183
Figure C-14: RNPV of NEO System Architectures that Utilize a Certain Launch Vehicle	183

Figure C-15: RNPV of NEO System Architectures that Utilize a Certain Launch Vehicle	184
Figure C- 16: Box and Whisker Plot of DDT&E RNPV for NEO System Architectures that Exclusively Use a Certain Launch Vehicle	185
Figure C-17: Box and Whisker Plot of DDT&E RNPV for NEO System Architectures that Exclusively Use a Certain Launch Vehicle	186
Figure C-18: Box and Whisker Plot of DDT&E RNPV for GEO System Architectures that Use a Propellant Depot	187
Figure C-19: Box and Whisker Plot of Flight Unit RNPV for GEO System Architectures that Use a Propellant Depot	188
Figure C-20: Box and Whisker Plot of DDT&E RNPV for Lunar System Architectures that Use a Propellant Depot	188
Figure C-21: Box and Whisker Plot of Flight Unit RNPV for Lunar System Architectures that Use a Propellant Depot	189
Figure C-22: Box and Whisker Plot of Flight Unit RNPV for NEO System Architectures that Use a Propellant Depot	189
Figure C-23: Box and Whisker Plot of DDT&E RNPV for NEO System Architectures that Use a Propellant Depot	190
Figure C-24: Box and Whisker Plot of DDT&E RNPV for GEO System Architectures that Use Different Aggregation Strategies	191
Figure C-25: Box and Whisker Plot of Flight Unit RNPV for GEO System Architectures that Use Different Aggregation Strategies	192
Figure C- 26: Box and Whisker Plot of DDT&E RNPV for Lunar System Architectures that Use Different Aggregation Strategies	193
Figure C-27: Box and Whisker Plot of Flight Unit RNPV for Lunar System Architectures that Use Different Aggregation Strategies	194
Figure C-28: Box and Whisker Plot of DDT&E RNPV for Lunar System Architectures for Location of Assets during Surface Mission	195
Figure C-29: Box and Whisker Plot of Flight Unit RNPV for Lunar System Architectures for Location of Assets during Surface Mission	196

Figure C-30: RNPV of NEO System Architectures for Different Aggregation Strategies	197
Figure C-31: Box and Whisker Plot of DDT&E RNPV for NEO System Architectures that Use Different Aggregation Strategies	198
Figure C-32: Box and Whisker Plot of Flight Unit RNPV for NEO System Architectures that Use Different Aggregation Strategies	199
Figure D-1: Configuration of Block 2 Lunar Crew Exploration Vehicle [7]	201
Figure D-2: Screenshot of Crew Capsule Sizing Spreadsheet.....	202
Figure D-3: Configuration of ESAS LSAM Cryogenic Descent Stage [7]	203
Figure D-4: Inert Mass Fraction Estimation for a Cryogenic Lunar Descent Stage.....	203
Figure D-5: Configuration of Apollo Lunar Excursion Module Hypergolic Descent Stage (Image: NASA).....	204
Figure D-6: Inert Mass Fraction Estimation for a Storable Lunar Descent Stage	205
Figure D-7: Configuration of ESAS LSAM Lunar Ascent Stage and Surface Habitat [7]	206
Figure D-8: Screenshot of Lunar Ascent Stage Sizing Spreadsheet.....	206
Figure D-9: Comparison of Launch Vehicles Used in System Architecture Analysis...	207
Figure D-10: Regression of Propulsive Stage Inert Mass Fraction for Different Propellants.....	210
Figure D-11: Configuration of Propellant Depot (Derived from Propulsive Stage)	211
Figure D-12: Screenshot of Surface Habitat Sizing Spreadsheet	213
Figure D-13: Configuration of Deep Space Habitat [11].....	214
Figure D-14: Screenshot of In-Space Habitat Sizing Spreadsheet	214

LIST OF TABLES

Table 1: Design Space Options for Lunar, NEO, and Mars System Architectures	3
Table 2: Comparison of Architecture Modeling Frameworks	8
Table 3: Comparison of Mathematical Frameworks	15
Table 4: Comparison Summary of Architecture Modeling Frameworks	27
Table 5: Data Used to Develop Regression for Hydrogen Tanks [41]	34
Table 6: Comparison of Stochastic Optimization Methods	44
Table 7: Set of Criteria for a Mars Architecture [58]	45
Table 8: Nominal Discount and Inflation Rates for Future Years	49
Table 9: Node Definition for Lunar System Architecture Graph	54
Table 10: Metadata within Edges for Each Edge Type	56
Table 11: System List for ESAS Baseline Architecture	61
Table 12: Rule-Based Traversal of Architecture Graph	63
Table 13: Hierarchy of System Types	67
Table 14: Overview of System Models	70
Table 15: CER Coefficients for Each System Type	71
Table 16: Launch Vehicle Cost Model Results Overview.....	72
Table 17: Overview of Systems Used in ESAS Mission Modes Comparison	82
Table 18: Overview of Architecture Options (Bold Text Indicates Baseline).....	89
Table 19: Results from ESAS Mode Analysis.....	90
Table 20: System Architecture Design Space Options with Alternatives	102
Table 21: Node Definition for GEO System Architecture Graph.....	103
Table 22: Node Definition for NEO System Architecture Graph with HEO Departure	105
Table 23: Node Definition for NEO System Architecture Graph with LEO Departure.	105
Table 24: Cost and Mass Estimates for Baseline GEO System Architecture.....	107
Table 25: Cost and Mass Estimates for Baseline Lunar System Architecture	108
Table 26: Cost and Mass Estimates for Baseline NEO System Architecture.....	110
Table 27: Summary of Cost, NPV, and IMLEO for Baseline System Architectures	110
Table 28: Description of Improved System Architectures from GEO Design Space.....	114

Table 29: Description of Improved System Architectures from Lunar Design Space ...	119
Table 30: Description of Improved System Architectures from NEO Design Space.....	124
Table A-1: GEO System Architecture Design Space Graph Definition.....	162
Table A-2: Lunar System Architecture Design Space Graph Definition	163
Table A-3: NEO System Architecture (HEO Aggregation) Design Space Graph Definition	165
Table A-4: NEO System Architecture (LEO Aggregation) Design Space Graph Definition	166
Table D-1: Overview of Existing and Designed Propulsive Stages [7],[44].....	209
Table D-2: Overview of Propellant Depot System Sizing Relationships.....	212

NOMENCLATURE

a	= Proportional Coefficient in a CER
A	= CDF Distribution Parameter
$A(i,j)$	= Adjacency Matrix with Row Index i and Column Index j
ABMS	= Agent Based Modeling and Simulation
ACO	= Ant Colony Optimization
A_{new}/A_{bl}	= Area Ratio in Photographic Scaling
ATM	= Architecture Trade Manager
b	= Power Coefficient in a CER
B	= CDF Distribution Parameter
b_0	= Constant in an RSE
b_i	= First Order Coefficient for Design Variable i in an RSE
b_{ii}	= Second Order Coefficient for Design Variable i in an RSE
b_{ij}	= Cross Term Coefficient for Design Variables i and j in an RSE
C	= Subsystem Cost
CDF	= Cumulative Distribution Function
CM	= Command Module
CER	= Cost Estimating Relationship
CEV	= Crew Exploration Vehicle
CPS	= Cryogenic Propulsive Stage
DAG	= Directed Acyclic Graph
DES	= Discrete Event Simulation
DDT&E	= Design, Development, Testing and Evaluation
DRA	= Design Reference Architecture
DSM	= Design Structure Matrix
EDL	= Entry, Descent, and Landing
EDS	= Earth Departure Stage
EOI	= Earth Orbit Insertion
EOR	= Earth Orbit Rendezvous

ESAS	= Exploration Systems Architecture Study
EXAMINE	= Exploration Architecture Model for In-Space and Earth-to-orbit
f_{inert}	= Inert Mass Fraction ($m_{inert}/(m_{inert} + m_{propellant})$), also IMF
FOM	= Figure of Merit
FV	= Future Value
FY12	= Fiscal Year 2012
g_0	= Gravitational Acceleration at Earth, $9.80665 \text{ m}^2/\text{s}^2$
GA	= Genetic Algorithm
GEO	= Geosynchronous Earth Orbit
GOR	= Geosynchronous Orbit Rendezvous
HEO	= High Earth Orbit
HLLV	= Heavy Lift Launch Vehicle
i	= Nominal Discount Rate
IMLEO	= Initial Mass in Low Earth Orbit
Isp	= Specific Impulse, seconds
ISS	= International Space Station
JSC	= Johnson Space Center
k	= Number of Independent Variables in an RSE
k	= Complexity Factor in a CER
LCC	= Life Cycle Cost, \$
L/D	= Lift-to-Drag Ratio
LEO	= Low-Earth Orbit
LLO	= Low-Lunar Orbit
LM	= Lunar Module
LMO	= Low-Mars Orbit
LOC	= Loss of Crew
LOI	= Lunar Orbit Insertion
LOR	= Lunar Orbit Rendezvous
LOX/CH ₄	= Liquid Oxygen/Liquid Methane
LOX/LH ₂	= Liquid Oxygen/Liquid Hydrogen
LOX/RP-1	= Liquid Oxygen/Rocket Propellant-1

LSAM	= Lunar Surface Access Module
LVSS	= Launch Vehicle and Spacecraft Synthesis System
$M(i,j)$	= Incidence Matrix with Row Index i and Column Index j
m_{inert}	= Inert Mass, kg
$m_{initial}$	= Initial Mass, kg
m_{final}	= Final Mass, kg
MMH	= Monomethylhydrazine
MPCV	= Multi-Purpose Crew Vehicle
$m_{propellant}$	= Propellant Mass, kg
MR	= Mass Ratio ($m_{initial}/m_{final} = \exp(\Delta V/(g_0 \cdot Isp))$)
MR_{max}	= Maximum Mass Ratio
n	= Number of Years between Future Date and Present Date
NAFCOM	= NASA/Air Force Cost Model
NEI	= Near Earth Objects
$N_{flights}$	= Number of Flights
NPV	= Net Present Value
NTO	= Nitrogen Tetroxide
OMB	= Office of Management and Budget
OPN	= Object Process Network
$Ph(i,j)$	= Pheromone Matrix with Row Index i and Column Index j
PSO	= Particle Swarm Optimization
PV	= Present Value
REDSTAR	= Resource Data Storage and Retrieval
RNPV	= Relative Net Present Value
RSE	= Response Surface Equation
SA	= Simulated Annealing
SM	= Service Module
T	= Time Fraction of a DDT&E Period
$T(i,j)$	= System Map Matrix with Edge Index i and System Index j
TEI	= Trans-Earth Injection
TLI	= Trans-Lunar Injection

TNI	= Trans-NEO Injection
<i>TOF</i>	= Time of Flight, days
TSP	= Traveling Salesman Problem
<i>t_{stay}</i>	= Stay Time, days
<i>T/W</i>	= Thrust-to-Weight Ratio
ULA	= United Launch Alliance
<i>V_{entry}</i>	= Entry Velocity (m/s)
<i>V_{new}/V_{bl}</i>	= Volume Ratio in Photographic Scaling
VRP	= Vehicle Routing Problem
<i>W</i>	= Subsystem Weight in a CER
<i>x_i</i>	= In an RSE, Independent Variable <i>i</i>
<i>y</i>	= Dependent Variable in an RSE
ΔV	= Ideal Change in Velocity, m/s
ε	= In an RSE, Error Associated with Neglecting Higher Order Terms
ε	= In ACO, Evaporation Percentage

SUMMARY

Developing a space exploration program to send humans beyond low-Earth orbit is a complicated problem that contains several complex and interconnected options. For the Apollo mission to the Moon, NASA used over one million man-hours starting in 1961 to select the lunar orbit rendezvous architecture based on performance requirements, reliability, cost, and the probability of completing the first mission by the end of the decade. NASA's current plans are to develop an evolutionary exploration program based on a steady increase in capability to explore cis-lunar space, the Moon, near Earth asteroids, and eventually Mars. There are countless options for the development of an exploration program: transportation systems (launch vehicles, in-space vehicles, and planetary descent/ascent modules), utilization of in-situ lunar and planetary resources and/or pre-positioned propellant depots, the technologies and capabilities supporting the space system architecture, and the evolutionary sequence of the missions from cis-lunar to the Mars surface. Exploring these options and selecting the best sequence of system architectures for each destination is crucial to develop an affordable exploration program. In an environment that emphasizes a fiscally responsible civil space program, selecting such a program is critical to mission success.

The primary goal of this research is to improve upon system architecture modeling in order to enable the exploration of these design space options. A system architecture is the description of the functional and physical allocation of elements and the relationships, interactions, and interfaces between those elements necessary to satisfy a set of constraints and requirements. The functional allocation defines the functions that

each system (element) performs, and the physical allocation defines the systems required to meet those functions. Trading the functionality between systems leads to the architecture-level design space that is available to the system architect.

The research presents a methodology that enables the modeling of complex space system architectures using a mathematical framework. To accomplish the goal of improved architecture modeling, the framework meets five goals: technical credibility, adaptability, flexibility, intuitiveness, and exhaustiveness. The framework is technically credible, in that it produces an accurate and complete representation of the system architecture under consideration. The framework is adaptable, in that it provides the ability to create user-specified locations, steady states, and functions. The framework is flexible, in that it allows the user to model system architectures to multiple destinations without changing the underlying framework. The framework is intuitive for user input while still creating a comprehensive mathematical representation that maintains the necessary information to completely model complex system architectures. Finally, the framework is exhaustive, in that it provides the ability to explore the entire system architecture design space.

After an extensive search of the literature, graph theory presents a valuable mechanism for representing the flow of information or vehicles within a simple mathematical framework. Graph theory has been used in developing mathematical models of many transportation and network flow problems in the past, where nodes represent physical locations and edges represent the means by which information or vehicles travel between those locations. In space system architecting, expressing the physical locations (low-Earth orbit, low-lunar orbit, etc.) and steady states (interplanetary

trajectory) as nodes and the different means of moving between the nodes (propulsive maneuvers, etc.) as edges formulates a mathematical representation of this design space.

The selection of a given system architecture using graph theory entails defining the paths that the systems take through the space system architecture graph. A path through the graph is defined as a list of edges that are traversed, which in turn defines functions performed by the system. A structure to compactly represent this information is a matrix, called the system map, in which the column indices are associated with the systems that exist and row indices are associated with the edges, or functions, to which each system has access.

With the system map defined, the mass and cost of each system can be determined so that different system architecture options can be compared. Using topological sort within graph theory, a directed acyclic graph represents the relationships between systems and the order in which those systems are sized. This methodology allows for a flexible system hierarchy that is automatically generated for each system map and can be used to explore a vast system architecture design space.

Trading different architecture options equates to the manipulation of the path that each system takes through the system architecture graph subject to a set of rules to ensure feasibility. An ant colony optimization algorithm was chosen from a number of methods to automatically explore the system architecture design space subject to these rules.

By developing this modeling framework, several contributions have been added to the state of the art in space system architecture analysis. The framework adds the capability to rapidly explore the design space without the need to limit trade options or the need for user interaction during the exploration process. The unique mathematical

representation of a system architecture, through the use of the adjacency, incidence, and system map matrices, enables automated design space exploration using stochastic optimization processes. The innovative rule-based graph traversal algorithm ensures functional feasibility of each system architecture that is analyzed, and the automatic generation of the system hierarchy eliminates the need for the user to manually determine the relationships between systems during or before the design space exploration process. Finally, the rapid evaluation of system architectures for various mission types enables analysis of the system architecture design space for multiple destinations within an evolutionary exploration program.

To demonstrate the functionality of this modeling framework, this research presents the system architecture design space exploration of three missions within an evolutionary exploration program (geosynchronous-Earth orbit, the lunar surface, and a near Earth asteroid). Each system architecture design space is represented as a graph. Alternative system architectures, which have significant reductions in cost over the baseline architectures, are produced for each mission, and a gradual capability development strategy is presented that reduces cost over the evolutionary exploration program. Utilizing common launch vehicles and systems across multiple destination missions reduces the development cost significantly for the exploration program as a whole.

The unique insight resulting from this exploration of the design space reveals that the launch vehicle selection is the primary driver in reducing the cost of a given system architecture. Other considerations, such as propellant type, staging location, and aggregation strategy provide less impact on the cost of a system architecture. The use of

commercial launch vehicles reduces development and flight unit cost, and when feasible, is the lowest cost option for performing a given mission regardless of the other selected architecture options.

The detriment for using commercial launch vehicles with lower payloads is the increased number of flights required to deliver the in-space hardware. This increase in number of flights reduces the probability of mission success due to the increased operational complexity and increased launch failure risk. One solution to this issue is to develop a heavy lift launch vehicle to reduce the number of required flights. The high development and flight unit cost of this option will, however, increase the overall cost of the system architecture by an order of magnitude over a commercial option. Alternatively, a propellant depot could reduce the number of critical launches that carry flight hardware and still use commercial launch vehicles to reduce the overall cost. However, these decisions require considerations beyond cost, such as development risk, reliability, sustainability, and launch availability, which were not quantified in this research.

CHAPTER 1

INTRODUCTION

The purpose of this research is to improve the methods used to model space system architectures in order to enable design space exploration at the architecture level. The research provides a methodology to model space system architectures using a mathematical framework that enables exploration and optimization of the architecture-level design space. This framework is then used to explore a space system architecture design space within NASA's human space exploration program.

1.1. Motivation

NASA's current plans are to develop an evolutionary series of missions based on systematic technology development to return to the Moon for testing the viability of long-term human outposts, intercepting asteroids for science and planetary defense, and eventually exploring Mars and the outer planets [1],[2]. There are countless options for the development of an exploration program: transportation systems (launch vehicles, in-space vehicles, and planetary descent/ascent modules), utilization of in-situ lunar and planetary resources and/or pre-positioned propellant depots, the technologies and capabilities supporting the space system architecture, and the evolutionary sequence of the missions from near Earth to the outer planets.

A system architecture is the description of the functional and physical allocation of elements and the relationships, interactions, and interfaces between those elements necessary to satisfy a set of constraints and requirements [3],[4],[5]. The functional allocation defines the functions that each system (element) performs, and the physical

allocation defines the systems required to perform those functions. Trading the functionality between systems leads to the architecture-level design space that is available to the system architect.

System architecture design spaces of three potential human exploration destinations are presented in Table 1, which contains both architecture-level trade options as well as the most common options for implementing advanced propulsion technologies. These design spaces define possible architecture and technology options that have been developed through previous mission architecture studies. Potential architecture options for lunar-surface missions have been studied since Apollo and in recent years due to the Vision for Space Exploration of 2004 [6],[7],[8]. It is challenging to form an architecture design space for a mission to a Near Earth Object (NEO) due to the uniqueness of each NEO and the long periods between departure opportunities, but NEO architecture studies have identified a significant amount of architecture options available in the design space for a mission to a given NEO [9],[10],[11]. The Mars architecture design space defines possible architecture and technology options that have been developed for sending humans to Mars, as derived from previous Mars architecture studies as far back as 1952 [12],[13],[14],[15],[16].

In the architecture studies that have analyzed missions to these destinations, many architecture and technology options were not considered, reducing the scope of those system architecture analyses in order to meet constraints on the available design and analysis resources. Limiting the options in these lunar, NEO, and Mars architecture design spaces during this early design phase, however, may limit the eventual effectiveness of the mission.

Table 1: Design Space Options for Lunar, NEO, and Mars System Architectures

Characteristic	Mission Destination		
	Lunar Surface	NEO	Mars
Architecture/Concept of Operations			
Infrastructure Development	<ul style="list-style-type: none"> • Earth Orbit Rendezvous (EOR) • Low Lunar Orbit (LLO) Rendezvous (LOR) • EOR-LOR • Lunar Surface Rendezvous • Direct 	<ul style="list-style-type: none"> • Low Earth Orbit (LEO) Rendezvous • L1 Rendezvous • Direct 	<ul style="list-style-type: none"> • LEO Rendezvous • Low Mars Orbit (LMO) Rendezvous • Mars Surface Rendezvous • LMO & Mars Surface Rendezvous • Direct
Launch Operations	<ul style="list-style-type: none"> • NASA 1-Launch • NASA 1.5-Launch • NASA 2-Launch • Commercial 	<ul style="list-style-type: none"> • NASA 1-Launch • NASA 1.5-Launch • NASA 2-Launch • Commercial 	<ul style="list-style-type: none"> • NASA 1.5-Launch • NASA Multi-Launch • Commercial
LEO Operations	<ul style="list-style-type: none"> • Assembly • Propellant Refueling • None 	<ul style="list-style-type: none"> • Assembly • Propellant Refueling • None 	<ul style="list-style-type: none"> • Assembly • Propellant Refueling • None
Trajectory Options	<ul style="list-style-type: none"> • High Thrust • Low Thrust 	<ul style="list-style-type: none"> • High Thrust • Low Thrust 	<ul style="list-style-type: none"> • High Thrust, Conjunction • High Thrust, Fast Transit • Low Thrust
Surface Architecture	<ul style="list-style-type: none"> • Outpost • Sortie 	<ul style="list-style-type: none"> • Flyby • Lander/ Rendezvous • Orbit 	<ul style="list-style-type: none"> • Commuter • Mobile Home • Telecommuter • Short Stay
Destination Operations	<ul style="list-style-type: none"> • Asset Rendezvous • Propellant Refueling • None 	<ul style="list-style-type: none"> • Asset Rendezvous • Propellant Refueling • None 	<ul style="list-style-type: none"> • Asset Rendezvous • Propellant Refueling • Phobos/Deimos • None
Technology Options			
Propulsion Technology	<ul style="list-style-type: none"> • Cryogenic • Hypergolic • Nuclear • Electric (Solar or Nuclear) 	<ul style="list-style-type: none"> • Cryogenic • Hypergolic • Nuclear • Electric (Solar or Nuclear) 	<ul style="list-style-type: none"> • Cryogenic • Hypergolic • Nuclear • Electric (Solar or Nuclear)

Figure 1 provides a notional representation of the cost, design freedom, and the knowledge about the performance throughout the design process [17],[18],[19]. In the earliest design phase, architecture creation, the Life Cycle Cost (LCC) is being

committed even though knowledge about the behavior of the architecture is still limited and little money has been spent to analyze the options. Also, the freedom to make changes to the physical systems decreases rapidly during architecture creation, fixing the eventual performance of the architecture. Therefore, the architecture selection process has a larger impact on both the overall feasibility and viability of the mission than subsequent design phases. Analyzing several architecture options rapidly in this early design phase is the preferred method to increase the knowledge of the performance of the architecture so that mission objectives are accomplished and the committed cost is minimized [17]. In an environment that emphasizes a fiscally responsible civil space program [1],[7], selecting such an architecture is critical to mission success.

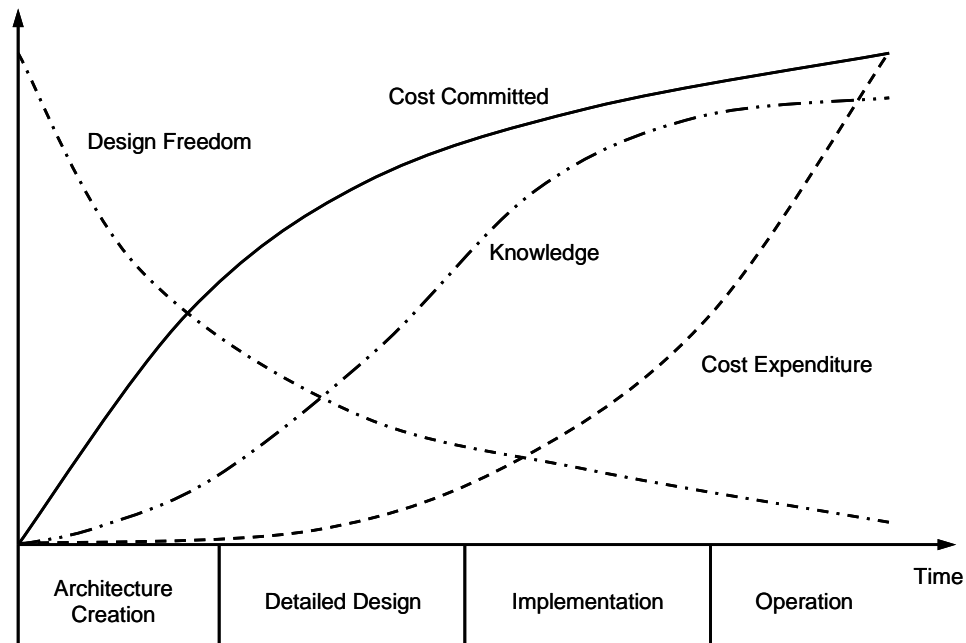


Figure 1: Notional Cost, Freedom, and Knowledge in the Design Process [17],[18],[19]

Although the system architecture definition has the greatest impact on the eventual performance and cost, selecting an optimal architecture is a difficult task due to the lack of methods to adequately explore the architecture design space. The current state

of the art in the field of multidisciplinary design optimization lacks the ability to effectively design and optimize systems that are a part of a larger architecture [20].

Contemporary attempts at defining space system architectures have identified this need to optimize space architectures. *Taylor (2007)* identifies that the optimization of both vehicle design (analogous to the physical decomposition) and path definition (analogous to the functional decomposition) in air and space logistics problems yield improvement over the optimization of just one of the two [21]. There exists coupling between the architecture path and the vehicle design that must be captured to effectively explore the design space. Also, NASA's latest Mars reference mission, Design Reference Architecture (DRA) 5.0, identified an architecture-level design space shown in Figure 2, but could not consider all possible options due to "limited scope and time allocated for this study," even though over 185 people from various NASA centers, academia, and industry worked on the study for several months [16]. Ultimately, two architecture paths were selected for further analysis based on qualitative assessments and the results from previous architecture designs. The systems were then optimized to operate within those two system architectures [16].

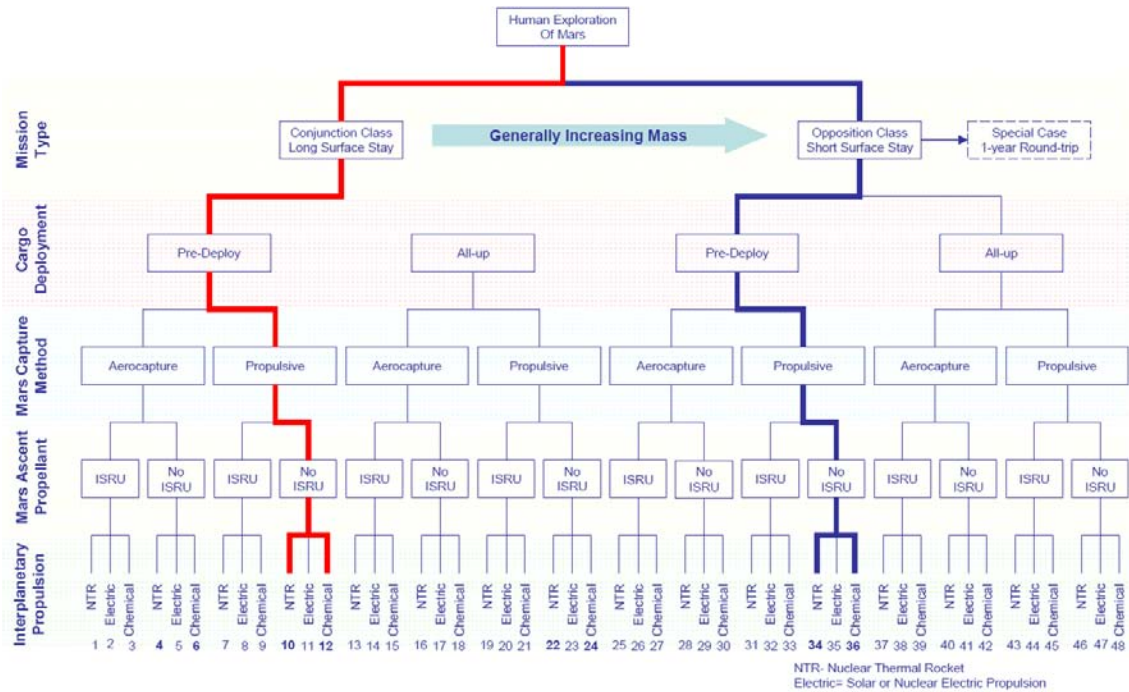


Figure 2: Design Reference Architecture 5.0 Trade Tree [16]

System architecting is a “necessary but incompletely understood step in creating [complex systems] [5],” such as a space system architecture, primarily due to the fact that there are “no algorithmic procedures for creating architectures [5].” The current state of the art frameworks for modeling space system architectures either are without a mathematical basis or over-simplify the problem which eliminates potential options from the design space [22],[23]. Therefore, before system architectures can be optimized, research must uncover a set of principles, methods, and tools to model system architectures that will help system architects make a sound decision [5].

Current space system architecture modeling frameworks utilize different methods to generate their architecture definitions and system models. The EXploration Architecture Model for IN-space and Earth-to-orbit (EXAMINE) tool, which contains the Architecture Trade Manager (ATM) architecture definition tool, is under development at

NASA Langley Research Center and is used to conceptually analyze end-to-end space system architectures [22],[24]. EXAMINE uses the ATM to define an architecture and manage the data within the framework of the tool. ATM contains the iBAT tool, which allows the user to manually define a system architecture using a graphical user interface. The user can define waypoints at various physical locations, such as LEO or LMO, and stack vehicles at them. These stacks then move from one waypoint to another along paths that have user defined properties. This is a manual process that requires a significant amount of user interaction to create a system architecture to flow down into a sophisticated set of vehicle sizing algorithms. Using this tool requires a significant amount of user interaction to define each system architecture, which is not conducive to rapid, automated exploration of the system architecture design space [22].

The Object Process Network (OPN) was developed to define lunar and Mars architectures, enabling architecture-level trades [23]. OPN is a modeling language that defines a system architecture as a network of objects and processes. Tokens travel between objects by way of the processes. When a decision must be made, the token splits and takes all possible paths. These tokens retain the information on the path that each has taken through the network, which is used to drive a set of simple sizing models. Vehicles in this framework have specified functionality and travel on predetermined flights defined by their final destination (LEO, LMO, etc.). This methodology removes the possibility of multifunction elements and reduces the flexibility by constraining the paths available to each system.

Finally, a logistics model has been developed to optimize the space logistics associated with human space exploration [21]. The logistics network develops a method

to optimize a logistics problem where commodities or cargo travels through a network on vehicles that are allowed to scale up and down to meet the demand. The emphasis of this tool is for use in developing a logistics support infrastructure for an existing human establishment [21]. Representing the supply and demand of goods using a network is a valuable analogy for developing a space system architecture, where systems must be delivered to certain locations along specified paths. However, there is a significant increase in complexity between a logistics network and a space system architecture. There are only two types of systems that travel through the logistics network: cargo and vehicles. The logistics focus of this model introduces assumptions that would not be applicable to a space transportation architecture. One such assumption is the independence between the vehicles/cargo and the paths on which they travel [21]. This cannot be true in general for a space system architecture as the path that is traveled defines the technology usage, operation time, propulsive requirements, and can even preclude the existence of certain systems. A summary of these architecture modeling frameworks with their benefits and detriments is presented in Table 2.

Table 2: Comparison of Architecture Modeling Frameworks

Framework	Pros	Cons
EXAMINE	<ul style="list-style-type: none"> • Flexible (multiple mission types can be modeled) • User-defined options • Detailed, bottoms-up sizing 	<ul style="list-style-type: none"> • Manual architecture definition makes optimization difficult • Requires significant user interaction
OPN	<ul style="list-style-type: none"> • Quickly explores options • Can be used for multiple mission types 	<ul style="list-style-type: none"> • Vehicles and paths are fixed • Limited design space
Logistics Network	<ul style="list-style-type: none"> • Mathematical model for space logistics • Network enables flexibility 	<ul style="list-style-type: none"> • Logistics-related assumptions oversimplify the problem • Limited flexibility in system types • Relationships between cargo and vehicles are predefined

While each framework is valuable to the field, each suffers from one of two drawbacks that prohibit architecture-level design space exploration: they are either too manual or too limited. The tools that have the capability to explore a large design space require a significant amount of manual input or manipulation, which prohibits rapid, automated exploration. The tools capable of automated exploration impose constraints such that much of the design space is excluded. These drawbacks reveal the inability of current tools to rapidly explore and optimize the architecture-level design space.

1.2. Research Goals and Objectives

The primary goal of this research is to improve upon these architecture modeling frameworks. The research presents a methodology to model complex space system architectures using a mathematical framework for design space exploration. This framework must meet five goals: technical credibility, adaptability, flexibility, intuitiveness, and exhaustiveness. The framework must be technically credible, in that it produces an accurate and complete representation of the system architecture under consideration. The framework must be adaptable, in that it provides the ability to create user-specified locations and functions available to each system. The framework must be flexible, in that it allows the user to model any type of architecture (Moon, NEO, Mars, etc.) within NASA's exploration program without changing the underlying model. The framework must be intuitive for user input (i.e. a visual representation) while still creating a comprehensive mathematical representation that maintains the necessary information to completely model complex architectures and explore the alternatives. Finally, the framework must be exhaustive, in that it provides the ability to explore the entire design space for use in eventual optimization.

There are several research objectives that will fulfill this goal of improved system architecture modeling. These objectives are as follows:

- 1. Develop a mathematical representation that adequately models a space system architecture applicable to multiple mission types.**

A mathematical system architecture modeling framework enables exploration of the design space rapidly by using currently available optimization algorithms to explore system architectures within the design space. In order to meet the intuitiveness goal for the modeling framework, a visual representation is preferred to enable system architects to easily create and manipulate the design space and the individual system architectures within that design space.

- 2. Determine a method to incorporate constraints, requirements, and interrelationships between the systems and functions to ensure feasible architectures are considered.**

When developing the system architecture design space, certain combinations of functions, locations, and/or systems may not be compatible. For instance, the crew cannot exist without a habitat system, or a low thrust propulsive system cannot perform an impulsive burn. Therefore, an algorithm must be developed to effectively explore the design space while satisfying all of these rules and constraints.

- 3. Develop a method to flexibly link the functional and physical description of the system architecture with the order in which each system is sized and the information flow between the system sizing tools.**

One of the primary drivers for sizing each system is the dependencies that exist between the systems. For instance, the payload defined for each of the propulsive systems is a collection of other systems. The system architecture defines which systems are included in this collection for each propulsive maneuver as well as the propulsive requirements to perform the maneuver. This process cannot require user manipulation at each iteration if the system architecture design space is to be explored because of the many different relationships between systems that could exist within a single design space.

- 4. Determine a selection criterion that captures the system architecture decision drivers, applicable to multiple mission types.**

In order to select promising system architectures from the design space, a selection criterion must be established against which the different system architectures can be compared. Many decision drivers exist for an exploration program, such as performance, cost, extensibility, and reliability. A single criterion must be developed to capture each of these decision drivers to select the most promising alternatives from the design space.

1.3. Problem Statement

The plan for human exploration of the Solar System involves an evolutionary progression of capability, moving from destinations within the Earth-Moon system to the

Martian surface. In 2009, the *Review of U.S. Human Spaceflight Plans Committee Final Report* recommended the Flexible Path option, which explores incrementally more difficult destinations in the inner Solar System with the final destination of landing a human on Mars [1]. Several strategies for the order in which these various destinations are visited within this evolutionary strategy are presented in Figure 3.

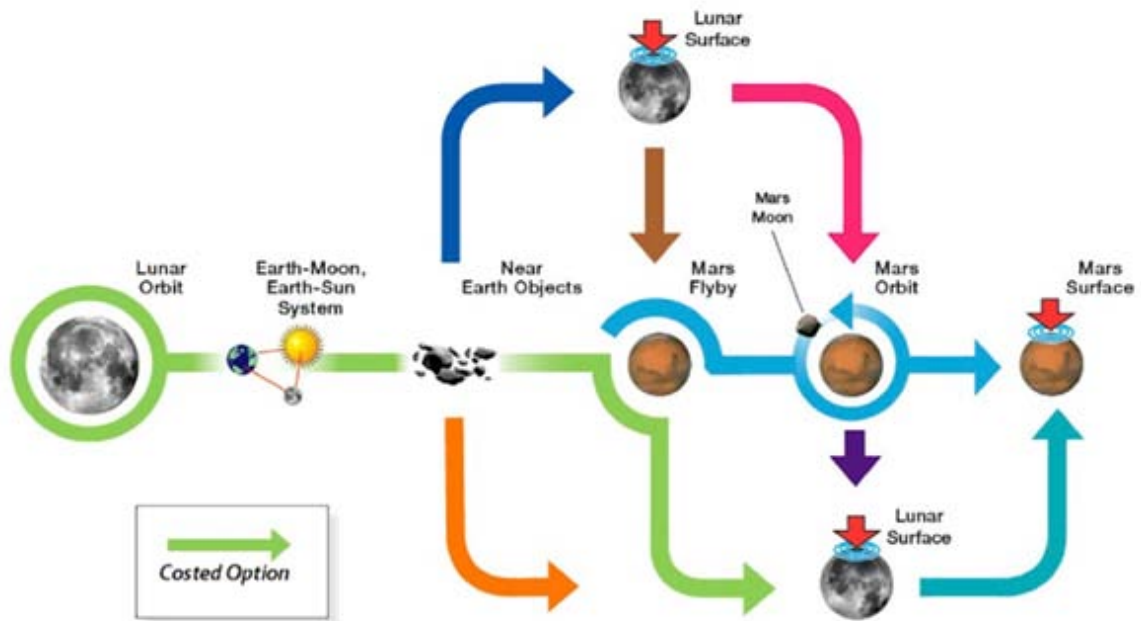


Figure 3: Possible Sequences in which to Visit Flexible Path Destinations [1]

Because conventional system architecture analysis is resource intensive, analyzing each of the potential mission destinations' architecture design spaces would be prohibitive. The solution to this problem in the past has been to qualitatively reduce the design space under consideration before analyzing the available options [16]. However, because each of these missions are linked in a sequence within an overall exploration program, more of the design space must be considered for each individual mission destination to understand how it fits within the overall program.

Representing the system architecture design space mathematically provides a means with which to reduce the required resources to explore multiple options. Design space options are variables within a mathematical framework that can be changed without human interaction. Several frameworks exist to model complex systems and system architectures, including dynamic programming, Agent-Based Modeling and Simulation (ABMS), Discrete Event Simulation (DES), and graph theory.

Dynamic programming is both a mathematical optimization method and a computer programming method, which simplifies a complicated problem by recursively dividing it into simpler sub-problems [25]. Two key attributes must exist for a problem to be effectively solved recursively: overlapping sub-problems and optimal substructure. Overlapping sub-problems means that the same small sub-problem is performed over and over again. This could apply to system architecting where system sizing, cost estimating, propulsive burns, etc. occur multiple times within a given mission. Optimal substructure means that the optimal solution can be obtained as a combination of the solutions to the sub-problems [25],[26]. This does not necessarily apply to system architecting, where these sub-problems have interactions which, when combined, can lead to a suboptimal solution [27]. Dynamic programming does not provide a visual representation of the design space, and it makes user interaction more difficult, as many links and properties of sub-problems must be hard-coded beforehand.

ABMS is a modeling and simulation framework that predicts the behavior and interactions of multiple autonomous agents in order to assess their effect on the system as a whole. Each agent works alone subject to rules and constraints that exist within it [28]. This is analogous to the systems that exist within a system architecture, which have

certain properties unique to that type of system, but without the framework to define how those systems interact to complete the overall mission. This structure must exist external to the agents, and must be developed using a different mathematical framework within which the agents can operate.

DES is a simulation framework in which a process is presented as a series of chronological events which mark changes in the state of the systems within that process. DES is powerful for evaluating operations and manufacturing processes, typically generating performance measures such as steady state values and cycle time [29]. DES is effective at simulating the events (functions) that must occur within a system architecture, but the framework to input the design space of functions, how they are linked to locations, and how systems that perform these functions are related to each other is not intuitive for the user.


Graph theory presents a valuable mechanism for representing the flow of information or vehicles within a simple mathematical framework [30]. Graph theory has been used in developing mathematical models of many transportation and network flow problems in the past [31],[32],[33],[34], where nodes represent physical locations and edges represent the means by which information or vehicles travel between those locations. A graph theory representation of the system architecture design space enables the user to visually input the functions and locations (edges and nodes, respectively) that correspond to the mission under consideration.

Table 3 provides a summary of the mathematical frameworks considered to meet the research goals and objectives. The technical credibility and exhaustiveness goals are not discriminators at this level. Each framework could meet these goals with equal

effectiveness. Therefore, these two goals were removed from the table. While dynamic programming is useful for modeling complex processes, it does not provide an intuitive means to input the design space and quickly adapting this framework to new functions or systems provide challenges. ABMS is effective at modeling agents, or systems, with different goals within a complex architecture, but does not provide a structured framework within which those agents operate. DES provides a framework in which the functions of a system architecture can be modeled, but is not as effective at creating a visual representation of the user-defined design space where functions and locations are co-dependent. Finally, graph theory provides a structure that has been used successfully in the past at modeling transportation architectures. Nodes correspond to locations, and functions correspond to edges, which are connected to different locations based on the mission being analyzed.

Table 3: Comparison of Mathematical Frameworks

Goal	Dynamic Programming	ABMS	DES	Graph Theory
Adaptability	○	◐	●	●
Flexibility	◐	●	●	●
Intuitiveness	○	○	◐	●



● Good ◐ Average ○ Poor

Expressing the physical locations (LEO, LMO, etc.) and steady states (interplanetary trajectory) as nodes of a graph and the different means of moving between the nodes (propulsive maneuvers, entry methods, etc.) as edges formulates a mathematical representation of this design space through the creation of the adjacency and incidence matrices. The adjacency matrix defines which locations or states are

functionally connected, and the incidence matrix defines the functional connections. The selection of a given system architecture using graph theory entails defining the paths that the systems take through the space system architecture graph. A path through the graph is defined as a list of edges that are traversed, which in turn defines functions performed by the system. A structure to compactly represent this information is a matrix, called the system map, in which the column indices are associated with the systems that exist and row indices are associated with the edges, or functions, to which each system has access.

With the system map defined, the mass and cost of each system can be determined so that different system architecture options can be compared. The order of the sizing for each system as well as the information flow between system sizing and cost estimation tools are changed flexibly based on the system map. This methodology allows for a flexible system hierarchy that can be used to explore a functionally diverse system architecture design space.

To model an evolutionary mission sequence within the modeling framework using graph theory, the solution is also sequential. Each mission within the options presented in Figure 3 can be modeled as an architecture graph with its own system architecture alternatives. These architecture graphs are then solved in one or more of the possible sequences presented in Figure 3.

Figure 4 presents a notional mission sequence for an evolutionary exploration program, which sends humans to a NEO after precursor missions to Geosynchronous-Earth Orbit (GEO) and to the lunar surface. This approach is consistent with the Flexible Path approach of visiting destinations increasingly more challenging as new capabilities and technologies are developed. Each of these missions is modeled with a system

architecture graph. The paths of systems through these graphs are analyzed to determine different potential system architectures for each destination. These system architectures are in the form of system maps. While these system maps can be optimized with respect to each individual mission, they must also consider the overall mission sequence. The end result is a set of system architectures (in the form of system maps) for each mission that satisfies the evolutionary mission sequence.

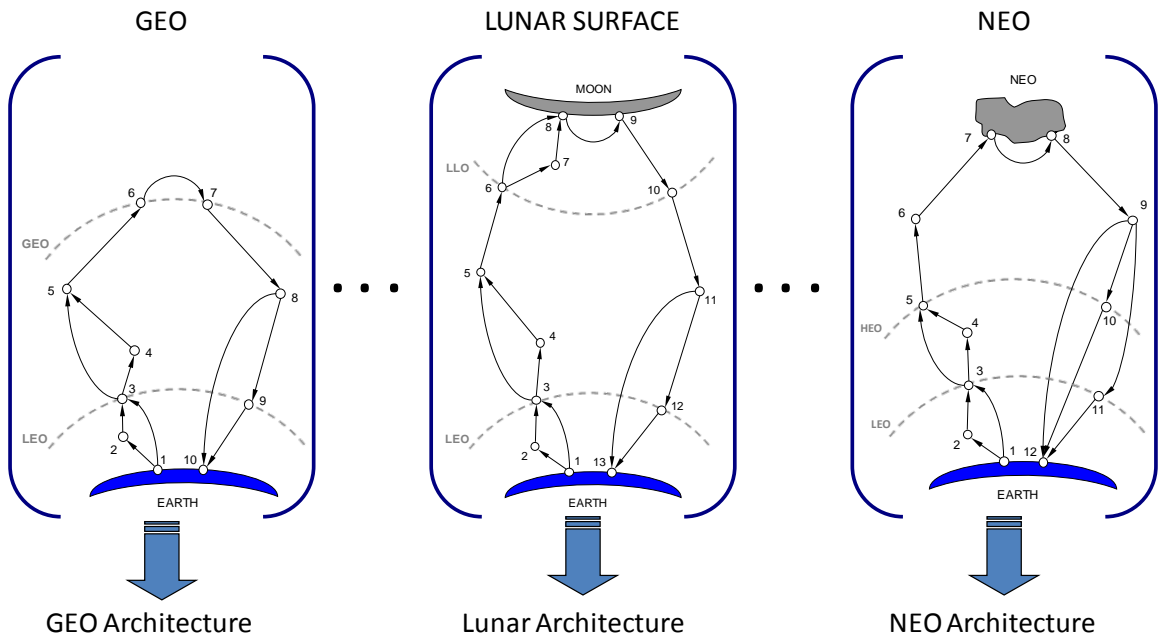


Figure 4: Solution to Evolutionary Mission Sequence Using Graph Theory Model

By developing this modeling framework, several contributions are added to the state of the art in space system architecture analysis. The framework adds the capability to rapidly explore the design space without the need to limit trade options or the need for user interaction during the exploration process. The unique mathematical representation of a system architecture, through the use of the adjacency, incidence, and system map matrices, enables automated design space exploration using stochastic optimization processes. The innovative rule-based graph traversal algorithm ensures functional feasibility of each system architecture that is analyzed, and the automatic generation of

the system hierarchy eliminates the need for the user to manually determine the relationships between systems during or before the design space exploration process. Finally, the rapid evaluation of system architectures for various mission types enables analysis of the system architecture design space for multiple destinations within an evolutionary exploration program.

1.4. Dissertation Overview

Chapter 1 introduced the reader to the purpose of this research: to improve the methods used to model space system architectures in order to enable exploration of the system architecture design space. This chapter presented the motivation for improved techniques in modeling space system architectures and provided historical perspective. Research goals and objectives were identified based on the need to improve the state of the art in system architecture modeling.

Chapter 2 presents a review of the existing literature that will be useful in formulating the modeling framework that addresses the goals and objectives of this research. This review includes an explanation of current system architecture modeling frameworks and an introduction to graph theory and its applicability to modeling a space system architecture. An overview of available methods for system sizing and estimation of cost is presented, as well as a discussion on optimization methods available for use on a highly constrained, discrete optimization problem. Finally, the chapter provides a discussion on architecture selection criteria.

Chapter 3 presents the proposed methodology that addresses the research goals and objectives expressed in Chapter 1. The application of graph theory to space system architecting will be explained in detail. This chapter will also describe the link between

the system architecture with the sizing and cost estimation of each system. This representation of the system architecture design space enables exploration and optimization based on the results of the sizing and cost estimation.

Chapter 4 uses the modeling framework to analyze a sample architecture level design space to show that it meets the goals set forth in this research. The chapter provides a description of the conversion of the system architecture design space to the representation as a graph. Then, individual system architectures are defined within this representation, and mass and cost estimates of these system architectures are compared.

Chapter 5 analyzes a sequence of missions within the flexible path exploration program using this system architecture modeling framework. The chapter presents the conversion of the system architecture design spaces into graphs and defines baseline architectures for each mission. The design space exploration identifies preferred architectures for each mission, and provides insight on the impact that architecture level decisions have on selecting architectures that are better than the baselines.

Finally, Chapter 6 provides conclusions about the modeling framework presented in this research and the implications of the design space exploration for a flexible path exploration program. This chapter discusses the recommendations on future work in this area to improve decision making, expand the system architecture design space, and provide increased fidelity and uncertainty quantification of the results.

CHAPTER 2

BACKGROUND

This chapter is a review of the existing literature that will provide an overview of the current state of the art in space system architecting. This review includes a summary of graph theory and its applicability to transportation architectures, constrained traversal through graphs, and the use of graphs to determine the order of performing system sizing. An overview of available methods for system sizing and estimation of cost is presented for use in this research. This review ends with a discussion of optimization methods available for use in the system architecture problem as well as the means with which different architecture options are evaluated.

2.1 Space System Architecture Modeling

Current space system architecture modeling frameworks utilize different methods to generate their architecture definitions and system models. The EXAMINE tool, which contains the iBAT architecture definition tool, is used to conceptually analyze end-to-end space system architectures [22]. Also, OPN was developed to define lunar and Mars system architectures to perform architecture-level trades [23]. Finally, a logistics model to optimize the space logistics associated with human exploration [21].

2.1.1. EXAMINE

EXAMINE uses the ATM to define an architecture and manage the data within the framework of the tool. ATM contains the iBAT tool, which allows the user to manually define an architecture using a graphical user interface, as shown in Figure 5. The user can provide waypoints in various physical locations such as LEO or Low Mars Orbit

(LMO), and stack vehicles at them. These stacks then move from one waypoint to another along paths that have different properties [22].

This is a manual process that requires a significant amount of user interaction to create a set of architectures to flow down into a sophisticated set of vehicle sizing algorithms. Using this tool requires extensive user interaction to define each system architecture, which is not conducive to exploring the system architecture design space. Previous attempts at incorporating architecture-level design space exploration into iBAT have been unsuccessful at creating a sufficiently flexible framework to explore multiple different types of architectures [23].

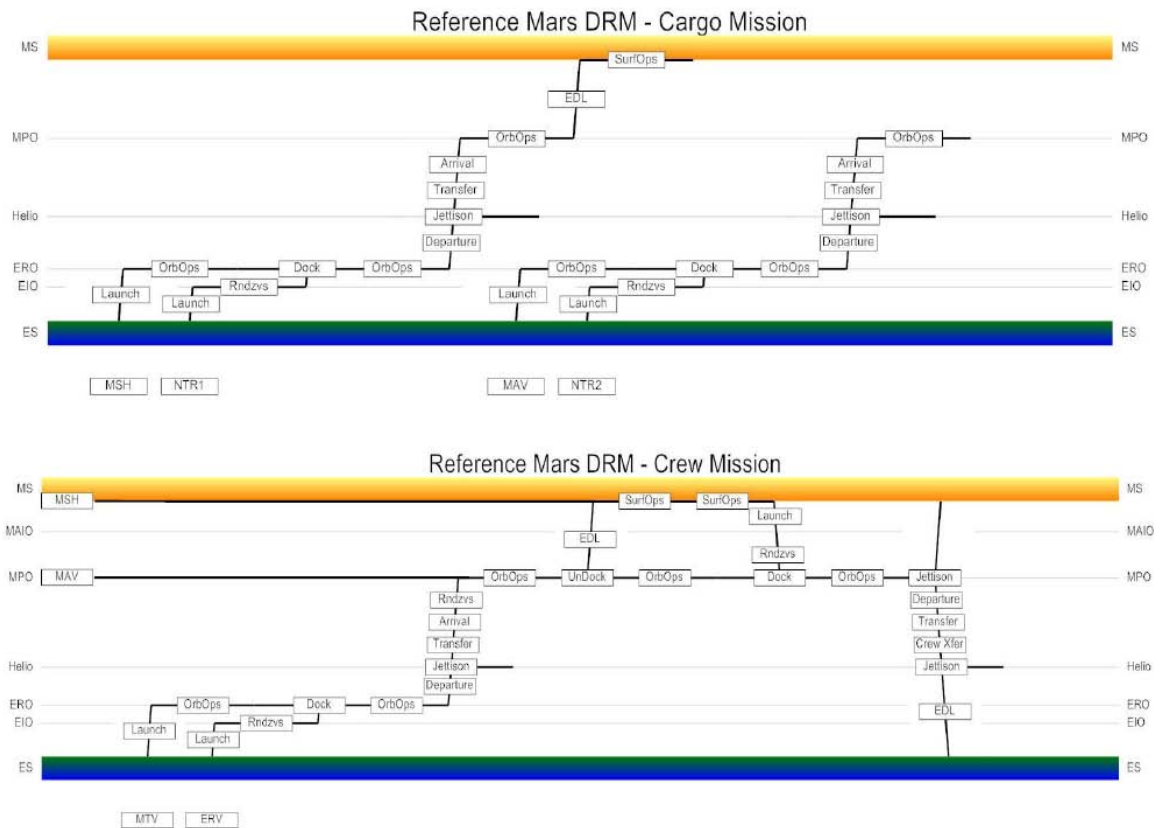


Figure 5: iBAT Graphical User Interface [22]

The primary drivers for sizing each system are the dependencies that exist between the systems. For instance, the payload defined for each of the propulsive systems is a collection of other systems. The system architecture defines which systems are included in this collection for each propulsive maneuver. This process cannot require user manipulation at each iteration if the system architecture design space is to be explored due to the many different relationships between systems that could exist.

A key component to developing a flexible space system architecture modeling framework is automatically correlating the architecture definition with the system sizing order and information flow. Solutions that have been presented previously either require too much human intervention to be automated or are not flexible enough to explore the entire architecture-level design space.

The framework that defines the interfaces between multiple systems within a space systems architecture within EXAMINE is described in the Design Structure Matrix (DSM) presented in Figure 6 [22]. EXAMINE uses the ATM to control the interfaces between the system models (named segment models in EXAMINE). While this tool is flexible and modular, enabling exploration of the entire system architecture design space, it cannot be automated in its current configuration. The ATM requires significant user interaction to define the system architecture, including the interactions between systems. Exploration of multiple architecture types would involve a lengthy human-in-the-loop process.

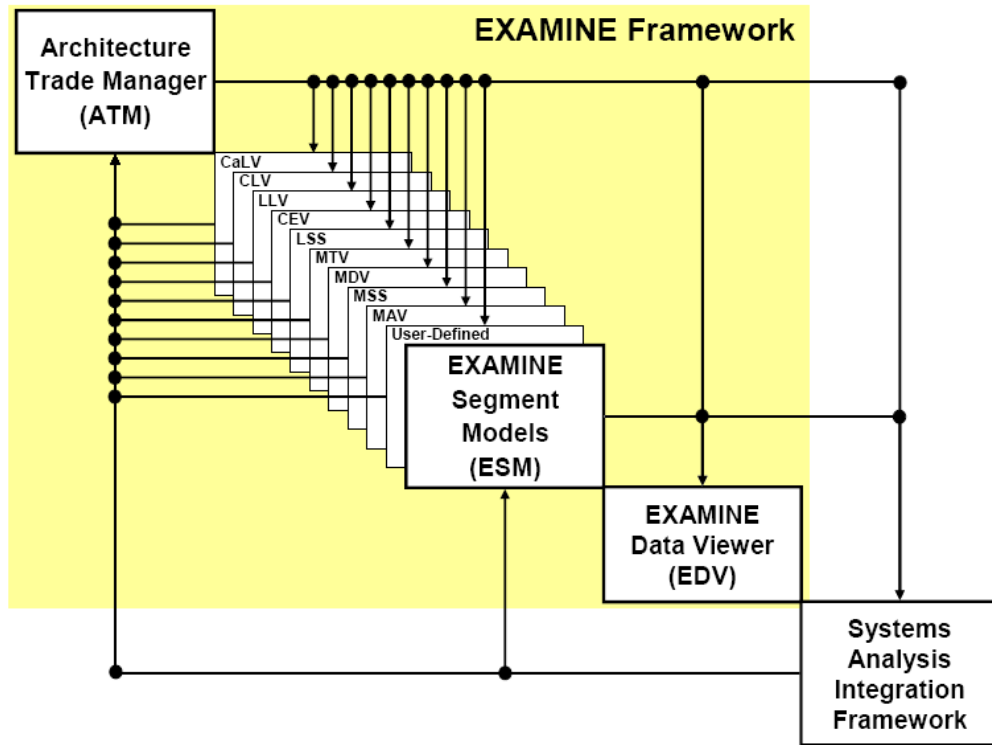


Figure 6: DSM of EXAMINE Framework from Komar et al. (2008) [22]

2.1.2. Object Process Network (OPN)

OPN is a modeling language that defines an architecture as a network of objects and processes. An example of such a network is shown in Figure 7. Tokens travel through this network going from one object to another by way of the processes. When a decision must be made, the token splits and takes all possible paths. These tokens retain the information on the path that they have taken through the network, and then feed information into a set of simple system sizing models.

Vehicles in this framework have specified functionality and travel on predetermined flights defined by their final destination (LEO, LMO, etc.) as shown in Figure 8 for a Mars architecture. This removes the possibility of multifunction elements and reduces the flexibility by fixing the paths available to each vehicle [23].

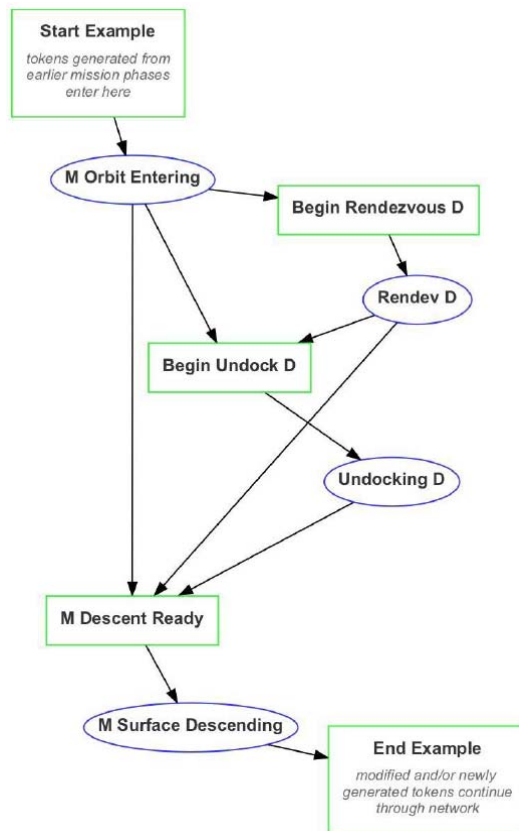


Figure 7: OPN Example [23]

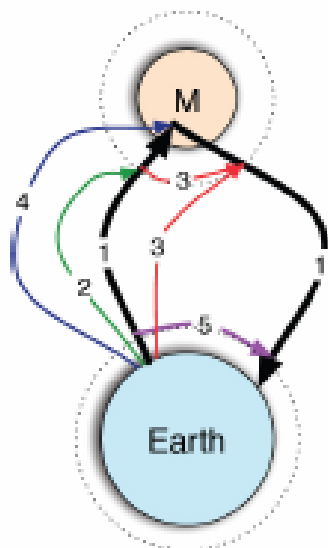


Figure 8: Specified Flights within OPN [23]

A solution that has been developed for visualizing the interactions of systems within a system architecture model using OPN is presented in *Bounova et al. (2005)* [35]. Based on a fixed system architecture definition, the sizing order of each of the systems is pre-defined in a hierarchy, presented in Figure 9 as a flow chart [35]. The benefits of this solution is the idea of expressing information flow between system sizing tools as a graph, where the nodes are the systems and the edges are information flow between those systems. This also introduces an inherent hierarchy between the systems (if no feedback loops are incorporated), which is created using directed arrows.

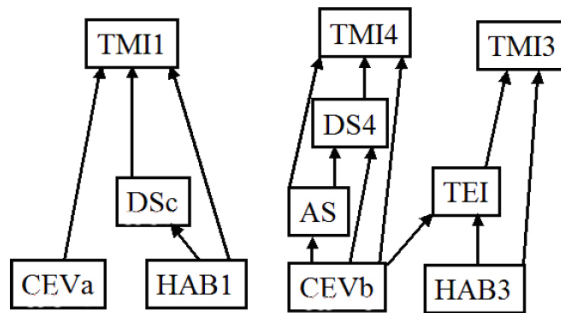


Figure 9: Fixed System Sizing Hierarchy for a System Architecture [35]

However, this graph is fixed for a given system architecture, and a general correlation between the system architecture definition and the system hierarchy has not been developed. The interrelationships between systems do not change with architecture type definition. This is a fixed method that must be manually created concurrently with the definition of the architecture-level design space. As the selected system architecture incorporates or eliminates certain systems, the mass is set to zero for that box. However, the system still exists in the framework and cannot flexibly interact with other systems that are not already connected. This solution, therefore, cannot be used to model different architectures within a broader architecture-level design space.

2.1.3. Logistics Network

The logistics network presented by Taylor (2007) [21] develops a method to optimize a logistics problem where commodities or cargo travels through a network on vehicles that are allowed to scale up and down to meet the demand. This tool uses embedded optimization to form feasible design points in each perturbation step within the global simulated annealing optimization. The emphasis of this tool is for use in developing a logistics support infrastructure for an existing human establishment [21].

Representing the supply and demand of goods using a network is a valuable analogy for developing a space system architecture, where systems must be delivered to certain locations along specified paths. However, there is a significant increase in complexity between a logistics network and a space system architecture. There are only two types of systems that travel through the logistics network: cargo and vehicles. The logistics focus of this model introduces assumptions that would not be applicable to a space transportation architecture. One such assumption is the independence between the vehicles/cargo and the paths on which they travel. This cannot be true in a space system architecture as the path that is traveled defines the operation time, propulsive requirements, and even the potential existence of certain systems. Finally, the only Figure of Merit (FOM) that is used is Initial Mass in LEO (IMLEO), although a true comparison of architectures should also include cost, risk, and other criteria [21].

Taylor identifies the limitations of this methodology to explore large design spaces as it applies to a space transportation network, and presents a solution to a restricted space logistics network [21]. Use of the logistics network to solve a transportation architecture is more practical for the air transportation problem than for space transportation due to the increased level of complexity. Therefore, a new approach

must be formulated to address the issues with this modeling framework in order to fully explore the space system architecture design space.

2.1.4. Overview

As presented earlier in Section 1.1 and reproduced here, Table 4 provides a summary of the aforementioned architecture modeling frameworks with their benefits and detriments.

While each framework is valuable to the field, each suffers from drawbacks that prohibit architecture-level design space exploration.

Table 4: Comparison Summary of Architecture Modeling Frameworks

Framework	Pros	Cons
EXAMINE	<ul style="list-style-type: none"> • Flexible (multiple mission types can be modeled) • User-defined options • Detailed, bottoms-up sizing 	<ul style="list-style-type: none"> • Manual architecture definition makes optimization difficult • Requires significant user interaction
OPN	<ul style="list-style-type: none"> • Quickly explores options • Can be used for multiple mission types 	<ul style="list-style-type: none"> • Vehicles and paths are fixed • Limited design space
Logistics Network	<ul style="list-style-type: none"> • Mathematical model for space logistics • Network enables flexibility 	<ul style="list-style-type: none"> • Logistics-related assumptions oversimplify the problem • Limited flexibility in system types • Relationships between cargo and vehicles are predefined

2.2. Graph Theory

Graph theory presents a valuable mechanism for representing the flow of information or vehicles within a simple mathematical framework [30]. Graph theory has been used in developing mathematical models of many transportation and network flow problems in the past, where nodes represent physical locations or states, and edges represent the means by which systems or information travel between those locations [30],[31],[32],[33],[34]. By expressing the physical locations and steady states within the space system architecture as nodes and the different means of moving through this design

space (LEO operations, entry methods, etc.) as edges, a mathematical representation of this design space could be developed.

A graph is a data structure that consists of two sets of data: nodes and edges. Each edge consists of an arc with two nodes as its endpoints. In directed graph theory, the order of the endpoints is important, as it defines a direction of each edge [30]. An example of a directed graph is presented in Figure 10, where the nodes (1, 2, 3, 4) are connected by directional edges (a, b, c, d, e).

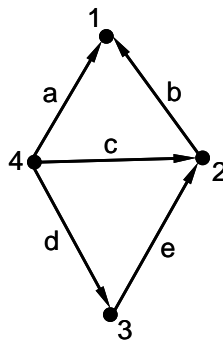


Figure 10: Example Graph

The usefulness of graph theory in system architecting is the mathematical representation it introduces. The two matrices—adjacency and incidence matrices—that fully define any given directed graph are shown in Figure 11. The adjacency matrix indicates which nodes are connected to each other. The first index in the adjacency matrix corresponds to the node from which the directed arrow is coming. The second index in the adjacency matrix corresponds to the node into which the directed arrow is going. The incidence matrix indicates which nodes the edges connect (and includes direction with the inclusion of sign). The first index in the incidence matrix corresponds to a node in the graph. The second index in the adjacency matrix corresponds to an edge

in the graph. The values of these elements in the adjacency and incidence matrices are defined in Equation (1) and Equation (2), respectively [30],[36].

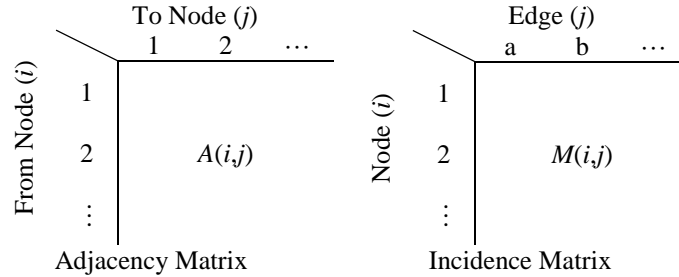


Figure 11: Adjacency and Incidence Matrix Representations

$$A(i, j) = \begin{cases} 1 & \text{if } i \rightarrow j \\ 0 & \text{if } i \nrightarrow j \end{cases} \quad (1)$$

$$M(i, j) = \begin{cases} 1 & \text{if } j \text{ leaves } i \\ -1 & \text{if } j \text{ enters } i \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

Graph theory is used in many transportation and network flow problems. Often, the nodes represent physical locations while the edges represent possible paths along which to travel between the nodes. The Königsberg bridge problem is a notable mathematical problem that laid the foundations for graph theory. In this problem, the nodes represented land masses while the edges represented bridges connecting them [30]. Similarly, the Traveling Salesman Problem (TSP) is a classical graph traversal problem, where cities to be visited are nodes and traversals between cities are edges [32].

Recent uses of graph theory involve modeling transportation and logistics networks using graph theory. The airline transportation system is often modeled as a graph with airports located at each node and flights between airports along the edges

[31]. Also, logistics networks are often modeled using the mathematical representation available within graph theory [34].

Graph theory contains several useful elements, such as the recursively defined graph and the topological sort. The recursively defined graph is defined by a set of base graphs that are connected by either “fusing” specific vertices from each base graph or by adding edges between specific vertices from each base graph [37]. This process has been used to develop the time-expanded network from a static network within the solution to the interplanetary logistics problem. In this process, the static network is duplicated for each instance in time, and the nodes are connected with new edges to accommodate motion between nodes in given amounts of time [21].

The topological sort within an acyclic directed graph creates a hierarchy of the nodes based on the information flow between them [36]. Within directed graph theory is a process known as a “topological sort,” which will prove useful in automatically linking the system architecture with the system sizing hierarchy. The nodes in the example graph presented in Figure 10 can be thought of as source or sink nodes, where the edges are leaving or entering the nodes, respectively. The source node is at the top of the hierarchy because it does not require information from any other nodes. When this node is added to the top of the hierarchy, it is removed from the graph along with all of its edges. A new graph now exists with a new set of source and sink nodes. As each node is sorted, the final node in the hierarchy would be a sink node [36]. For the graph presented in Figure 10, the topological sort yields the following order: 4, 3, 2, 1. This process can only be used on a Directed Acyclic Graph (DAG), or the sort process is invalid because there is no distinct hierarchy of the nodes.

In linking the system architecture with the system sizing, graph theory presents another useful advantage due to its underlying mathematical framework. By representing the system hierarchy as an acyclic directed graph, a mathematical representation enables rapid, automatic manipulation of a functionally diverse design space. The topological sort will be a useful tool in linking the system architecture definition with the system sizing models by automatically providing an order to the sizing of each system as the functional allocation of those systems changes.

In summary, graph theory presents a valuable mechanism for representing the flow of information or vehicles in a mathematical framework. By expressing the physical locations and steady states within the space system architecture as nodes and the different means of moving through this design space (LEO operations, entry methods, etc.) as edges, a mathematical representation of this design space could be developed. Various discrete optimization methods have also been applied to graphs, as will be necessary to explore the space system architecture design space [21],[38].

When used to solve the space system architecture design space exploration problem, graph theory does present some challenges. The adjacency and incidence matrices only contain information on which nodes are connected by which edges. The embedded information that serves as requirements for the system sizing, such as the change in velocity (ΔV), propellant usage, stay time, and time of flight, are not included in these two data sets. Therefore, the matrices must serve as pointers that call objects that contain the embedded information. Therefore, when a system requires information about an edge which it traverses, it will have to look at a separate data structure in addition to the adjacency and incidence matrices.

Also, while representing the traversal of systems through the graph is straightforward (a list of edges traversed), ensuring that these traversals create a functionally feasible system architecture must be handled outside of graph theory. Rules are defined that enforce feasibility. These rules force the existence of a system type if another system type is also present along an edge (i.e. crew and a habitat), and they force system types to traverse an edge if that system is necessary to perform the function defined by that edge. Different methods that could be used to solve this problem are presented in Section 2.5.

2.3. Performance Modeling

The level of fidelity of each system model must enable rapid evaluation of the performance and cost of the system. To achieve this, several methods are utilized to model the various systems that could be used in a given system architecture: response surface equations of higher fidelity models, regressions of data collected from existing systems or subsystems, and linear photographic scaling of similar systems.

Developing a Response Surface Equation (RSE) involves fitting a quadratic equation to a set of data via a least squares regression. The basic second order model for a RSE is

$$y = b_0 + \sum_{i=1}^k b_i x_i + \sum_{i=1}^k b_{ii} x_i^2 + \sum_{i=1}^{k-1} \sum_{j=i+1}^k b_{ij} x_i x_j + \varepsilon, \quad (3)$$

where y is the dependent variable, b_i are coefficients determined through a least squares regression, x_i are the independent variables (of which there are k), and ε is the error associated with neglecting higher order terms [39]. The data that is fit into this model is the mass or other characteristic (i.e. inert mass fraction) of a given system as calculated using a higher fidelity model, such as EXAMINE [22].

Developing a data regression on existing systems and subsystems can be used to rapidly develop bottoms-up models of complex systems. This is an accurate method for modeling systems that have been built several times before, such as tanks, propulsion systems, and power systems because the operational system includes all uncertainties, margins, and non-ideal factors. The dependent variables and the functional form of the regression are determined by the physical behavior of the system [40],[41]. Figure 12 shows a regression for the mass of liquid hydrogen tanks, which is a nearly linear function of the tank volume. The data used to develop this regression is given in Table 5 [41].

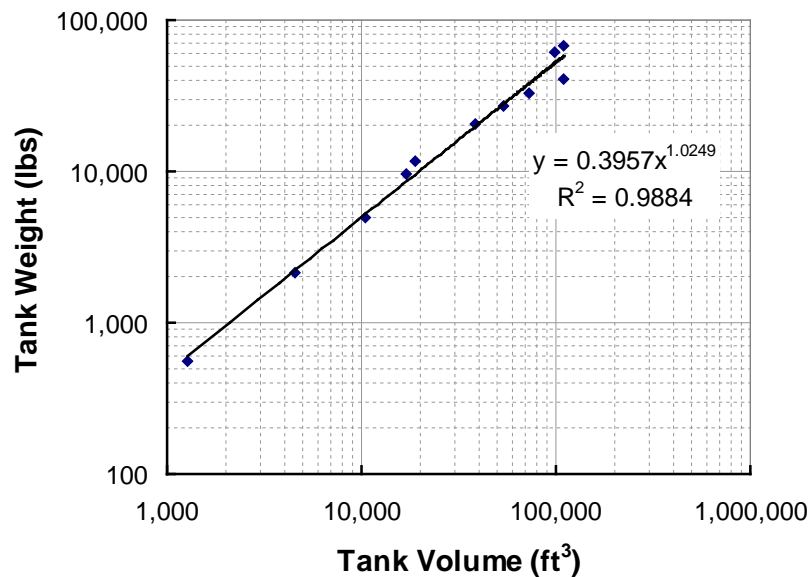


Figure 12: Regression on Existing Liquid Hydrogen Tanks [41]

Table 5: Data Used to Develop Regression for Hydrogen Tanks [41]

Vehicle	Tank Volume (ft ³)	Tank Weight (lbs)
Space Shuttle	53,646	27,088
B9U	109,799	67,478
Saturn V (S-II)	38,424	20,529
Saturn V (S-IV)	4,520	2,125
Saturn V (S-IVB)	10,524	4,987
MDC H33 Booster	72,540	32,789
Booster	98,780	61,511
MDC Orbiter	17,058	9,711
NA Orbiter	18,894	11,704
Martin TII	108,739	40,692
Centaur	1,271	560

Photographic scaling involves stretching a baseline vehicle while maintaining all characteristics (such as layout, tank pressure, and engine performance) of the baseline vehicle. This scales the vehicle up or down based on a defining characteristic. For habitation systems, subsystems linearly scale with either number of crew, stay time, or the combination of crew-days [42]. For propulsive elements, the vehicle scales based on the ratio of propellant mass in the new vehicle over the propellant mass in the baseline vehicle [40],[43]. Tank mass, as shown with the linear regression, scales almost linearly with this propellant mass ratio (which is equivalent to the propellant volume ratio for a fixed propellant type). Other subsystems, such as structure or thermal protection scale with the area ratio between the two vehicles, which is related to the volume ratio via

$$\frac{A_{new}}{A_{bl}} = \left(\frac{V_{new}}{V_{bl}} \right)^{2/3}. \quad (4)$$

To illustrate the effectiveness of these simplified sizing models, inert mass and gross mass for a propulsive system as derived from these models are compared to vehicles that have either been built (Centaur, Delta IV upper stage, and Saturn S-IVB) or designed using higher fidelity, bottoms-up analysis (from the Exploration Systems

Architecture Study (ESAS)) [7],[40],[44]. The results of this comparison are shown in Figure 13 and Figure 14, where the propulsive systems are sized to deliver a payload of 45,861 kg to various change in velocity (ΔV) requirements. Photographic scaling of the Earth Departure Stage (EDS) as presented in ESAS and a bottoms-up model using both RSE and regression estimates of subsystem mass formulate the two models used in the comparison. The simplified models match the trend of the existing systems, and the error between the estimates is insignificant. In architecture analysis, the error in the requirements, system growth, and cost estimation are typically larger than the model error presented in these figures.

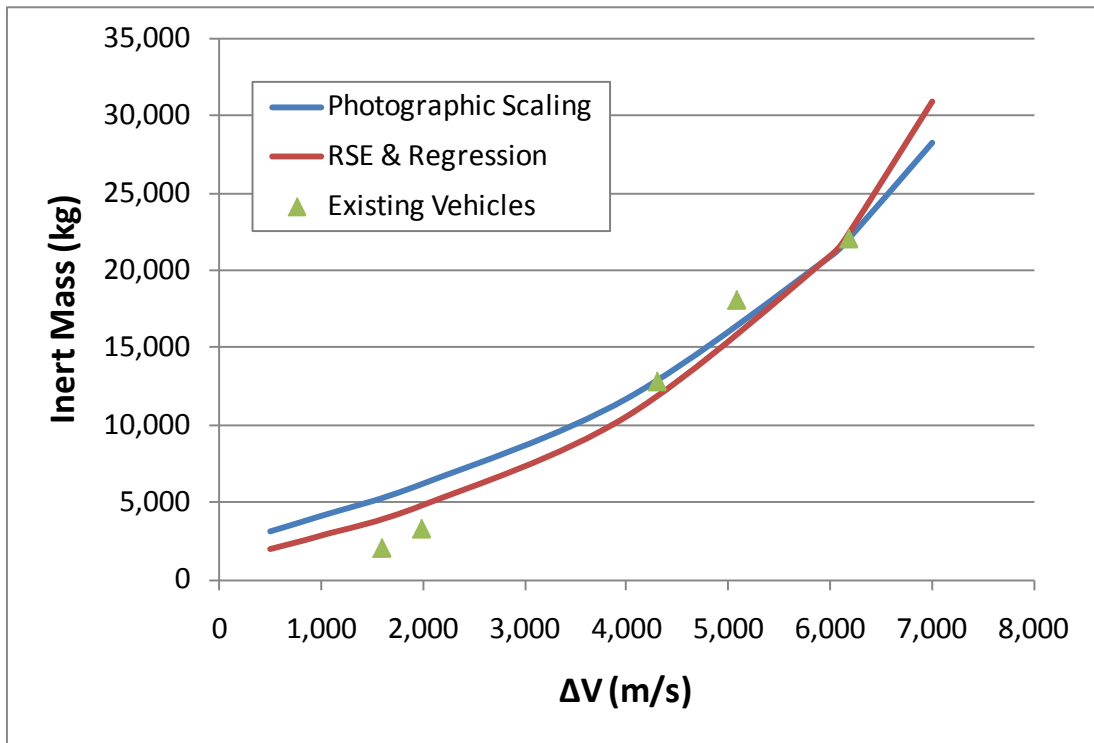


Figure 13: Inert Mass Comparison between Simplified Models and Existing Vehicles

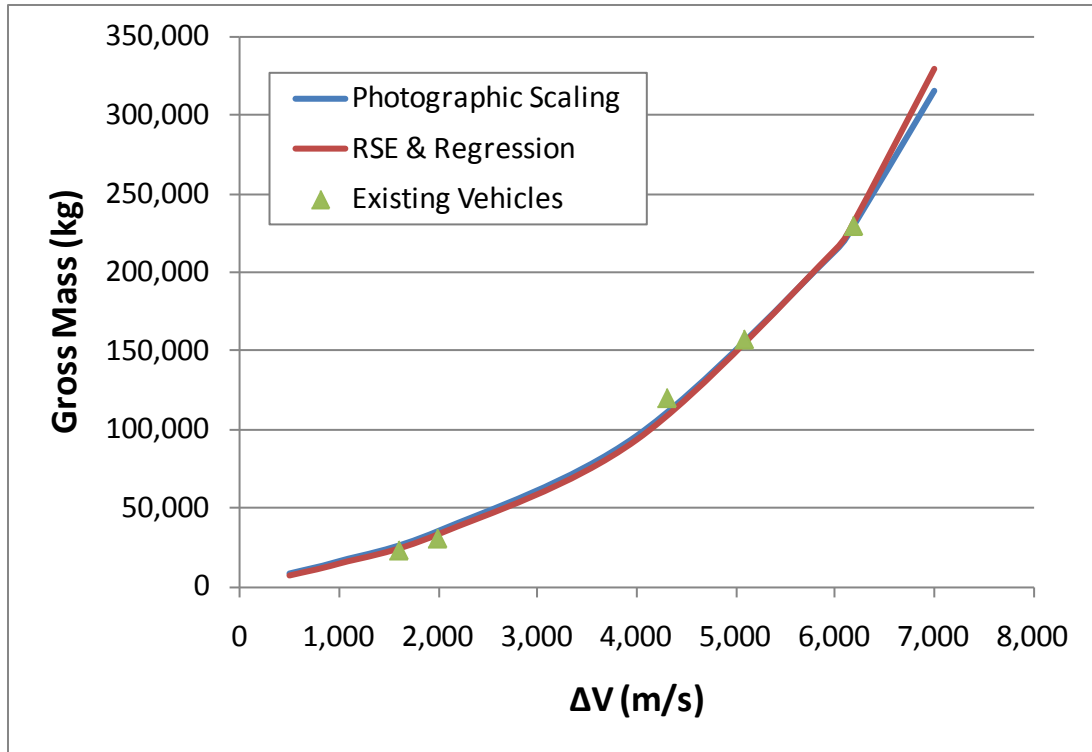


Figure 14: Gross Mass Comparison between Simplified Models and Existing Vehicles

2.4. Cost Estimation

Along with performance, cost is another metric that that determines the merit of a space system architecture. The Life Cycle Cost (LCC) of a space system is the total cost of a project across all phases, including design, development, production, operations, and disposal. The cost to develop a system from concept to a complete design that is ready for production is categorized as Design, Development, Testing, and Evaluation (DDT&E) cost. The cost to produce a system for use in the mission is the flight unit cost. Both of these costs are typically predicted during the conceptual design phase as a function of mass and system complexity [45]. The operations and disposal costs depend on drivers that are not always clearly linked to the system architecture, but rather indirect factors such as ground logistics and workforce management [46]. Finally, launch cost is

dependent upon launched mass and launch vehicle cost, which varies based on the launch vehicle used in the system architecture.

A Cost Estimating Relationship (CER) for a given subsystem is a parametric regression on the cost of analogous systems based upon the weight of the subsystem of the form

$$C = k \cdot aW^b, \quad (5)$$

where C is the subsystem cost, k is a complexity factor associated with multipliers based on certain design decisions (technology development, manufacturing methods, etc.), and a and b are constants defined by the regression on the analogous system. CER's are well suited to low-fidelity, rapid comparisons of space systems [47].

The NASA/Air Force Cost Model (NAFCOM) is a parametric cost-estimating tool that contains multiple, subsystem-level CERs based on the Resource Data Storage and Retrieval (REDSTAR) database of historical spacecraft, launch vehicles, and rocket engines [48]. Along with subsystem weight, NAFCOM uses metrics to determine the complexity factor to apply to the CER. These metrics are Manufacturing Methods, Engineering Management, New Design, Funding Availability, Test Approach, Integration Complexity, Pre-Development Study, and other subsystem-specific metrics. The total system cost is then computed as the sum of the subsystem costs plus integration and management costs [48].

Transcost is another cost estimating tool for use with launch vehicles, but uses CER's on total system mass to predict a total system DDT&E and flight unit cost [49]. This formulation is useful to estimate the cost for conceptual systems that may not have fully-defined subsystem details. The complexity factor applied to the CER is dependent

upon the system type (liquid and solid propulsive stages, crew modules, etc.) and consists of metrics such as system uniqueness, team experience, and vehicle mass fraction. An example Transcost CER for the DDT&E cost of an expendable, liquid-propulsion launch vehicle stage is shown in Figure 15 [49].

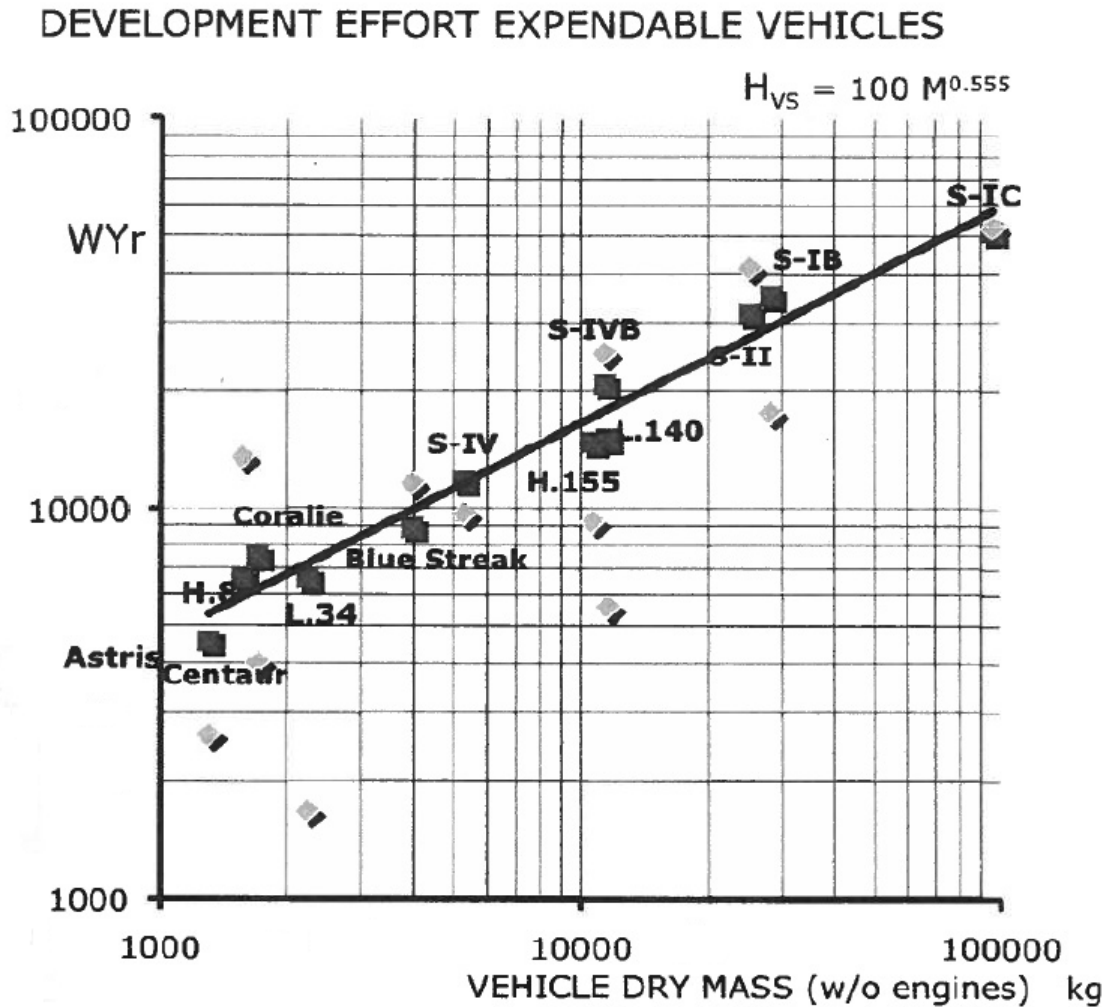


Figure 15: Transcost CER for Expendable, Liquid-Propulsion Launch Vehicle Stage [49]

The costs associated with DDT&E do not occur within a single year. Therefore, these costs are distributed over multiple years through the use of a beta distribution curve. This distribution, developed at Johnson Space Center (JSC) in the 1960s, is used to

spread parametrically derived cost estimates over the duration of the development process [45]. This percentage of total cost spent up to a certain time is defined by a Cumulative Distribution Function (CDF), which holds the form presented in Equation (6).

$$CDF = 10T^2(1 - T)^2(A + BT) + T^4(5 - 4T) \quad (6)$$

Here, T is the fraction of time of the entire DDT&E period ($0 \leq T \leq 1$), and A and B are distribution parameters such that $0 \leq A + B \leq 1$. In the case where $A = 0$ and $B = 1$, 50 percent of the cost is spent after 50 percent of the time has passed (called a 50:50 spread). In the case where $A = 0.32$ and $B = 0.68$, 60 percent of the cost is spent after 50 percent of the time has passed (called a 60:40 spread) [50]. Standard practice at NASA is to use a spread that commits more money early in the development period (e.g. 60:40 spread) for technically challenging designs and manned systems, while a 50:50 spread is adequate for systems with significant heritage or less demanding technical challenges [45].

2.5. Architecture Optimization Methods

The optimization of the space system architecture design space, when modeled using graph theory, is a discrete, non-linear optimization problem with a large number of variables and constraints [21]. Formulating the problem as a graph to represent the available paths through which systems travel (subject to a set of rules) presents unique features to this optimization problem. Solutions to similar graph traversal optimization problems, such as the Traveling Salesman Problem (TSP) [32] or the transportation network Vehicle Routing Problem (VRP) [33], have been conceived in the past. The

inclusion of a rule-based traversal of the graph by different systems renders these proposed solutions to the problem invalid [21].

For a human lunar mission, there are approximately 10^8 possible paths available to each system per flight. Many of these paths, however, are infeasible due to architectural constraints and physical rules that exist. Therefore, an algorithm must be developed to effectively traverse the graph while satisfying all rules.

Constraints are often incorporated into optimization routines through the use of penalty functions. These functions detrimentally augment the objective function when the design variables are selected such that constraints are violated. These functions have several mathematical forms, based on the application and behavior of the problem being solved [51].

In the optimization of a path through a graph, the large number of constraints introduces many penalty functions, which can undermine the effectiveness of the algorithm with a small feasible design space. A method introduced to solve this problem is embedded optimization, where a feasible solution is ensured before the objective function is actually evaluated. In this process, an optimization whose objective is to find a feasible solution regardless of its optimality is embedded within the global optimization [21]. The information flow for an optimization process using embedded optimization is presented in Figure 16.

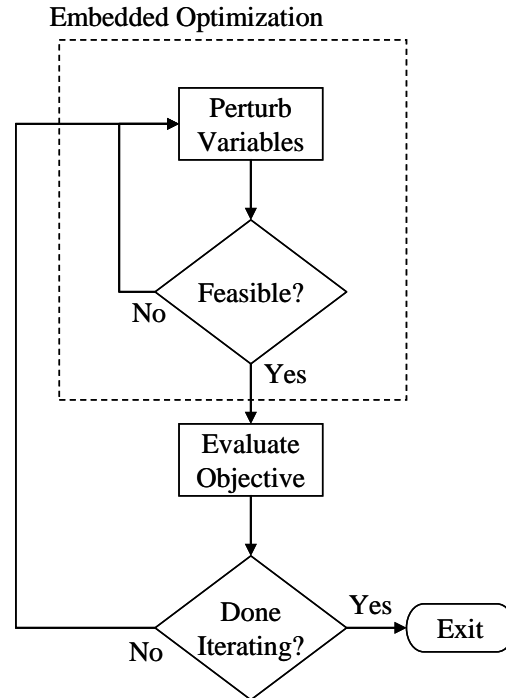


Figure 16: Information Flow for Embedded Optimization [21]

A third method is to ensure that the rules are met as the path definition is being formulated. For a system architecture design space with few feasible options relative to the total number of options, this method is efficient at preventing the evaluation of many infeasible design points. When represented as a graph, this process can eliminate the likelihood that a system will travel along an edge based on the paths and existence of other systems within the given system architecture design point.

To perform this optimization, biologically-inspired stochastic optimization methods are considered. There are several potential solutions to this type of problem, including Simulated Annealing (SA), Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), and Genetic Algorithm (GA). These solutions are not mutually exclusive, and a combination of any of these could be used within an embedded optimization process.

SA uses the principle of annealing, in which the magnitudes of random perturbations are reduced in a controlled manner. With respect to an optimization problem, this allows early perturbations to explore large expanses of the design space and avoid local minima, but reduces later perturbations to prevent the solution from moving away from the optimal solution [52]. *Taylor (2007)* uses SA in conjunction with embedded optimization to optimize an integrated logistics transportation system [21]. The random perturbations are checked at each iteration to ensure feasibility, but this process is inefficient if there are many interrelated constraints, as this problem contains.

ACO is a path-finding optimization method that was “developed from the observation of the efficient foraging behavior of ants in a colony” [38]. Multiple agents (ants) travel along various paths, and the value of the objective at the end of the path defines the amount of “pheromone” laid on each path. The paths with more pheromone at the end of the process are preferred over those with little pheromone. *Villeneuve (2007)* uses this optimization method to define paths through a graph that represents a morphological matrix of options [38]. This method is effective at defining paths through constrained graphs by eliminating options based on previous decisions.

PSO simulates the swarming behavior of a population similar to a swarm of bees. Each individual member of the population tracks its path and records the objective function. These individuals then keep track of the favorable areas of the design space and communicate with other individuals within the population [53]. This optimization method is efficient for finding global optima to discrete and continuous optimization problems. This optimization method is commonly used on multimodal mathematical problems, structural optimization problems with several discrete and continuous variables


[53], and multidisciplinary optimization problems for aerospace systems [54]. This method also has difficulty with the constrained path optimization problem because constraints are most effectively added after the perturbation steps, introducing the same inefficiency as the SA.

GA uses principles from evolution and natural selection from the field of biology. Properties from each iteration (or generation) of data points (or population) continues to the next generation if those properties yield a more optimal solution [52],[55]. A benefit of GA is the availability of commercially produced codes on various platforms (Excel, MATLAB, etc.) that can be easily implemented for a given optimization problem [56],[57]. GA is routinely used to optimize problems with discrete variables, multimodal objective functions, and/or large design spaces that cannot be fully explored with a full factorial analysis.

A comparison between these four optimization methods is presented in Table 6. GA could be a suitable option for the optimization process, but has not been used routinely within a rule-based graph traversal problem. ACO has been used extensively with path traversal, and could be used to explore the design space of the rule-based graph traversal. ACO has a higher tendency to stay at local minima, but with sufficient parameters for the evaporation and pheromone update steps, this issue can be mitigated.

Table 6: Comparison of Stochastic Optimization Methods

	SA	ACO	PSO	GA
Uses information from previous iteration	○	●	●	●
Explores large expanses of the design space (avoids local minima)	●	◐	●	●
Previously used with path optimization	●	●	◐	◐
Software packages are readily available	●	◐	◐	●
Adaptability to constrained path optimization	◐	●	○	◐



 ● Excellent ◐ Average ○ Poor

2.6. System Architecture Evaluation Criteria

Performing optimization on the space system architecture design space first requires the ability to compare two or more system architecture alternatives. According to *Donahue and Cupples (2001)*, “assessing a particular architecture as superior to another depends on how the benefits and advantages of each are valued. A significant part of the task of architecture evaluation rests on determining and prioritizing the relevant criteria [58].” The evaluation of system architectures presents several criteria, as presented in Table 7 that encompass performance, cost, and risk [58]. Multi-criteria decision making using these criteria requires qualitative assessment of several criteria and a non-unique combination into a single objective using weightings on each criterion. Solutions to both of these problems still pose issues during the decision making process.

Table 7: Set of Criteria for a Mars Architecture [58]

Performance
Minimum IMLEO
Cost
Low or Reasonable First Mission Cost
Evolution to Low Recurring Cost Missions
Risk
Acceptable Development and Operational Risk
Minimum Major Technology Development Programs
Multiple Use Technology Developments
Commonality of Architecture with other Space Activities

During the ESAS, similar criteria were developed to evaluate manned lunar system architecture alternatives, named Figures of Merit (FOMs). These FOMs, along with the sub-FOMs, are presented in Figure 17 [7]. These FOMs were combined into a single criterion using weightings that were selected by the decision makers. The value for each quantitative FOM was calculated using proxy parameters, while the qualitative FOMs were assessed using driving aspects of each alternative (number of launches, number of rendezvous, etc.). The definitions for each FOM, including the proxy parameters and drivers, are located in *Appendix 2D: ESAS FOM Definitions* of the ESAS final report [7]. While this solution considers all pertinent criteria for evaluating a system architecture, there is a significant amount of subjectivity in the FOM values and weightings, and it is difficult to quantify certain FOMs. For instance, unless Mars, NEOs, and other destinations are considered and system architectures developed, the Extensibility/Flexibility FOM cannot be accurately quantified.

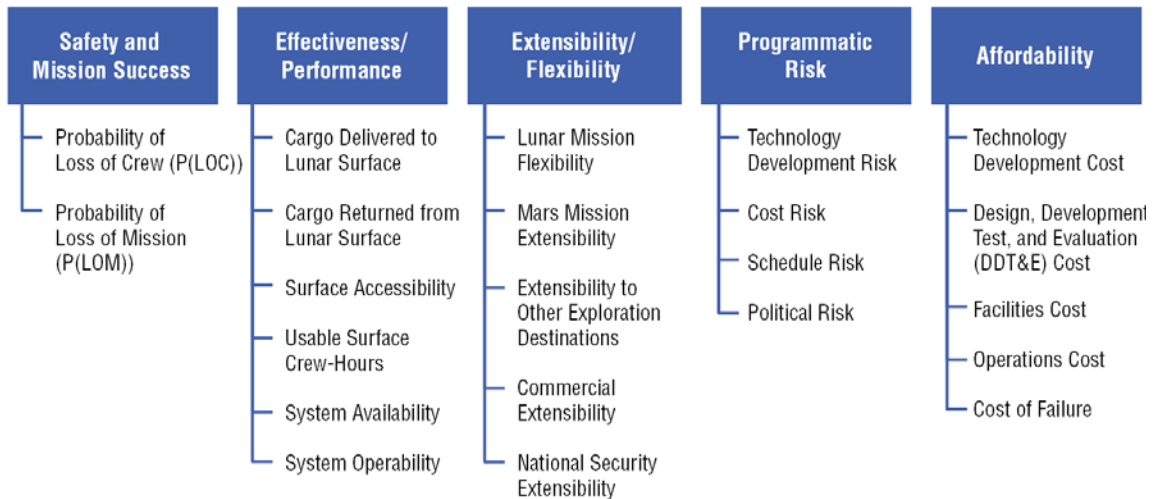


Figure 17: Figures of Merit from ESAS [7]

Similarly, the Mars DRA 5.0 used a set of multiple criteria, as shown in Figure 18. However, much of the evaluation of different alternatives was performed qualitatively. There was little quantitative analysis performed on each alternative presented in the architecture-level trade space definition. The trades were performed before the analysis in order to eliminate solutions that did not look beneficial at the outset of the study. The alternatives selected for analysis were “chosen because experience has shown that the cases...represent typical approaches, and the trends will be similar for the other branches of the trade tree” [16]. In this preliminary, qualitative analysis, quantitative data did not support the decisions that were made. Also, because the decisions were based on previous results, the less understood alternatives were ignored, even though they could be desirable.

Human Exploration Of Mars		
Long Surface Stay (Conjunction Class)	Figure of Merit	Short Surface Stay * (Opposition Class)
Similar	Total mass in Low-Earth Orbit (mt)	Similar *
45% Smaller	LEO Complexity / Size of Crew Vehicle	Larger
~3100 crew-sols	Expected Useful Crew Sols on Surface (mission return)	~80-500 crew-sols
Best	Exploration Goal Satisfaction (range, depth, frequency)	Lower
3 / 6 kg/kg	Architecture Sensitivity (gear ratios: NTR/Chem)	4 / 13 kg/kg
No Clear Advantage	Probability of Loss of Crew	Somewhat Less
Somewhat Less	Probability of Loss of Mission	No Clear Advantage
950	Total Mission Duration	650 days
500 sols	Mission Flexibility (contingency replanning)	Few sols
Less	Crew Health Risks from Radiation Exposure	More
200 / 500 / 200	Crew Exposure to Zero-G (days out / surface / back)	180 / 30 / 360
Available	Backup Lander and Surface Habitat	None
Somewhat More	Cost Through First Mission	Slight Advantage
Somewhat More	Cost Through Third Mission	Slight Advantage

Figure 18: DRA 5.0 Architecture Selection Criteria [16]

For this study, cost is used as the primary metric with which to compare architecture alternatives. Other metrics, such as risk, reliability, and schedule are difficult to quantify automatically at the level of fidelity used. Using risk as a FOM requires qualitative assessment by subject matter experts. Reliability quantification is possible, but data from NASA that would be used to develop reliability estimates is restricted. Quantifying the reliability of various system architecture options is a useful future task in this research field. Finally, quantifying schedule using launch availability and launch rate requires proprietary data from launch providers and estimates for NASA vehicles.

In system architecture analyses with a single selection criterion, IMLEO is typically used because it is indicative of the other selection criteria, such as production and launch costs, and is therefore used more often [21],[23],[59]. However, some criteria, such as technology development risk and cost, are not captured in an analysis that uses IMLEO as its sole FOM. Consider the decision to invest in an advanced

propulsion system: an improvement of 10 percent in specific impulse will lead to a smaller IMLEO. However, if the cost to develop that advanced propulsion technology is more than the savings in production and launch costs, then the technology should not be developed for the mission in question.

A means to capture this scenario is the utilization of Net Present Value (NPV) as the FOM. When evaluating a space system architecture, the current mission as well as future missions (extensibility) must be considered. NPV provides a means with which to qualitatively assess the value of a present decision and its future consequences. The development of a comprehensive system architecture modeling framework will aid in evaluating both current and future space missions with varying destinations. The use of NPV within this framework allows the system architect to then evaluate architecture-level decisions' impacts on the entire human exploration program.

Comparing alternatives that provide return on the initial investment at different times and in different ways is difficult [58],[60]. Therefore, it is useful to compare these alternatives by resolving their worth at the present time. NPV is one method in which this is accomplished. NPV is defined as the difference between the present value of benefits and the present value of costs, as shown in Equation (7). NPV analysis provides a prediction of the return that a given investment will provide in terms the present time. In economics, NPV is often used to calculate the profitability of a given investment and cash flow over time [45],[61]. A positive NPV indicates that the investment should be made, while a negative NPV indicates that it should not.

$$NPV = PV(Benefits) - PV(Costs) \quad (7)$$

The present value of money is calculated using Equation (8).

$$PV = FV(1 + i)^{-n} \quad (8)$$

The variables PV and FV are the present and future values of the money spent, respectively, and n is the number of years between the future date and present date. The nominal discount rate, i , is the sum of both the inflation rate and the real discount rate. These values are not constant with time, and are specified annually by the Office of Management and Budget (OMB) in the *OMB Circular A-94 Appendix C* [45],[62]. For Fiscal Year 2012 (FY12), the specified nominal discount and inflation rates are presented in Table 8.

Table 8: Nominal Discount and Inflation Rates for Future Years

Future Year	Nominal Discount Rate	Inflation Rate
2020	0.028	0.017
2025	0.028	0.017
2030	0.035	0.018

In engineering decision making, the LCC is the reported cost for a given project. LCC is the total cost of a project across all phases, including design, development, production, operations, and disposal. The cost term within NPV is the cost of the design, development, and production phases. The benefits of an engineering decision using NPV are not as clearly defined as cash flow. Instead, the performance benefits must be quantified using financial terms. One option for quantifying benefits of an engineering decision is to divide benefits into two categories: revenue and cost savings. Both of these are pertinent in the production, operations, and disposal phases. Revenue is a monetary return resulting from the investment. Cost savings is a reduction or elimination of an expense that would occur at some point in the future if an alternative decision were made [45]. Another option is to ensure equal performance benefit for all alternatives.

In order to calculate the NPV of a given architecture option, the benefits and costs must be translated into economic terms. For a given space system architecture, the system sizing tools determine the performance and cost of each individual system within that architecture. In space system cost estimating, many variables attribute to the actual development and production costs. Typical cost estimates in the conceptual design phase use inert mass and system complexity as the primary cost drivers within a parametric cost model [42],[47].

CHAPTER 3

METHODOLOGY

This chapter presents the methodology that will answer the research questions posed in Chapter 1. The application of graph theory to space system architecting will be explained in detail in order to convert the architecture level design space that the architect wishes to explore into a mathematical representation that is suitable for automated exploration. The procedure for converting a system architecture into a graph is described, followed by a description of the rule-based graph traversal by each system within the system architecture. Converting the graphical representation of the architecture into an objective function through the calculation of performance and cost is then discussed. This chapter concludes with an introduction to the architecture-level optimization problem that this methodology enables.

3.1. Applying Graph Theory to a Space System Architecture

In order to utilize graph theory in space system architecting, the flow of information must first be understood. Figure 19 presents a logic diagram of the procedure used to model space system architectures using graph theory. This procedure can be divided into three sections: graph generation, design space exploration, and evaluation. The graph generation section generates the mathematical representation of the system architecture design space based on the user defined inputs. The design space exploration section determines the path through which each system travels and defines the hierarchy of system sizing. The evaluation section estimates the performance and cost of each system within the specified system architecture. Finally, an optimizer

manipulates the traversal of the systems through the architecture graph in order to explore the design space.

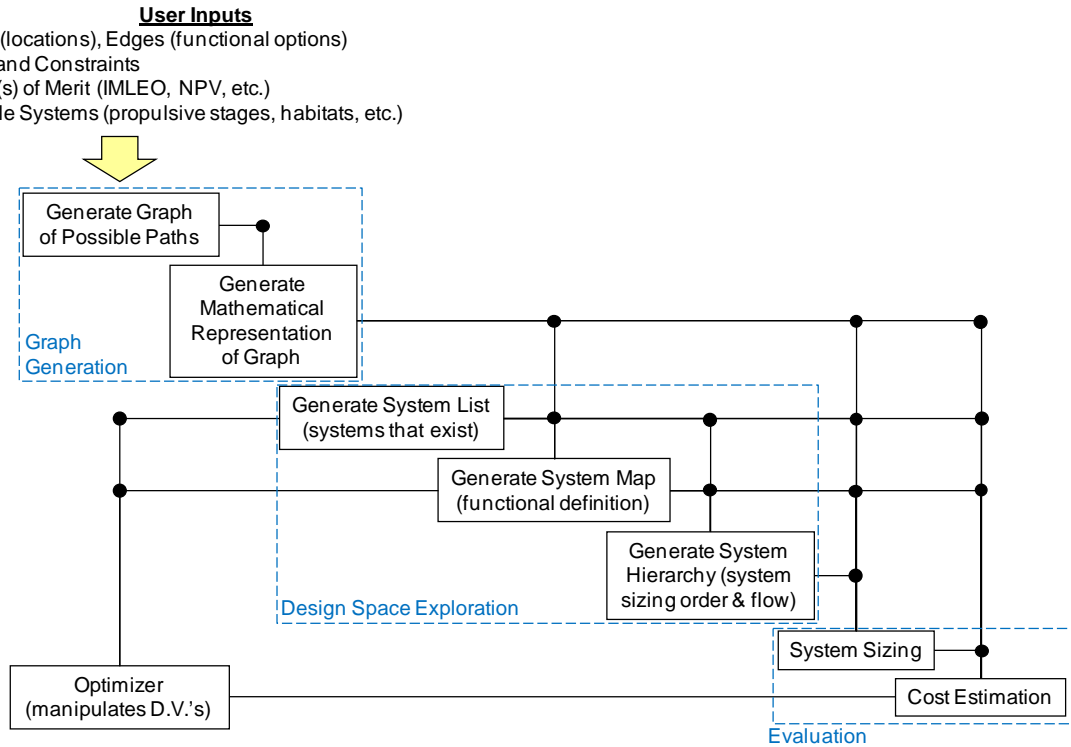


Figure 19: Procedure for Modeling a Space System Architecture Using Graph Theory

The procedure starts with the user inputting several parameters to generate a visual representation of the graph. The user specifies the nodes, edges (along with all of their options and pertinent information), and any constraints that must be met. Information can be embedded within the edges based on the type of edge that is used. The types vary depending on the function that must be performed, such as a propulsive maneuver (in-space, ascent, or descent), entry, and refueling. The information (ΔV , Thrust-to-Weight Ratio (T/W), etc.) embedded in each edge serves as the inputs and requirements for the system sizing models for all systems traveling along that edge. These mission requirements are unique for each destination and determines the information embedded within these edges.

The FOM by which the architecture options will be compared must be defined by the user. The particular FOM used is dependent upon the problem that is being posed and the decision drivers that are important to the decision maker. Finally, the possible system types, such as propulsive stages, habitats, landers, and ascent stages, and the models used to estimate their mass and cost must be defined up front by the user.

3.2. Graph Generation Model

The generation of the graphical description of the space system design space begins with a manual input of the nodes and edges by the user. These nodes and edges are defined in tabular format along with the embedded data for each, and are converted into a mathematical representation of the graph via the incidence and adjacency matrices. This representation can be duplicated to enable multiple flights (up to the total number of flights, $N_{flights}$) connected through user-specified links. This capability accommodates multiple pre-positioning flights in a given architecture.

The nodes are the first data that must be created. Figure 20 shows a candidate graph for a round-trip mission to the Moon with nodes labeled 1-14 and various edges connecting them. The definitions for the nodes in this graph are presented in Table 9, which gives physical meaning to each node: a location or steady state within the mission. The edges that connect these nodes, along with the various options that exist for each edge, are then created. Each connection between nodes is actually a set of parallel edges that contain the various functional options for traversing between the two given nodes.

The “Link Group Number” column in the node definition table allows the user to specify nodes that are repeating in the graph across all flights. These repeating nodes represent static locations in the mission where assets can be pre-positioned, such as LEO,

LLO, or the lunar surface. This definition allows the user to place any number of the nodes in groups. It is common for system architectures to have multiple flights, so it is important to have this capability available so that the model can incorporate as many (or as few) flights as necessary. The link groups, through the use of recursively defined graphs, enable multiple pre-positioning flights, as presented in Figure 21, where each flight is a complete graph shown in Figure 20 that are connected together by the nodes identified within the link groups.

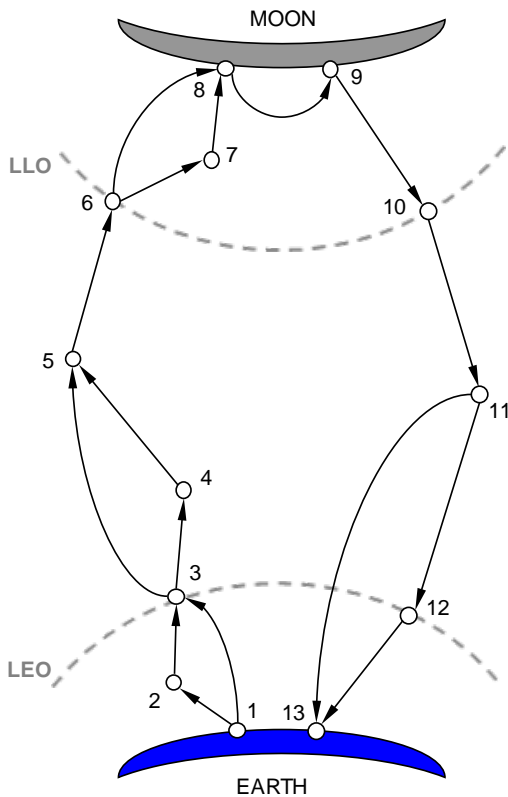


Figure 20: Lunar System Architecture Design Space as a Graph

Table 9: Node Definition for Lunar System Architecture Graph

Node No.	Node Name	Link Group No.
1	Earth Surface (Outbound)	
2	Suborbital Staging Point	
3	LEO (Outbound)	1
4	LEO Propellant Depot	
5	Trans-Lunar Trajectory (Outbound)	
6	LLO (Outbound)	2
7	Lunar Braking Point	
8	Lunar Surface (Arrival)	3
9	Lunar Surface (Departure)	3
10	LLO (Return)	2
11	Trans-Lunar Trajectory (Return)	
12	LEO (Return)	1
13	Earth Surface (Return)	

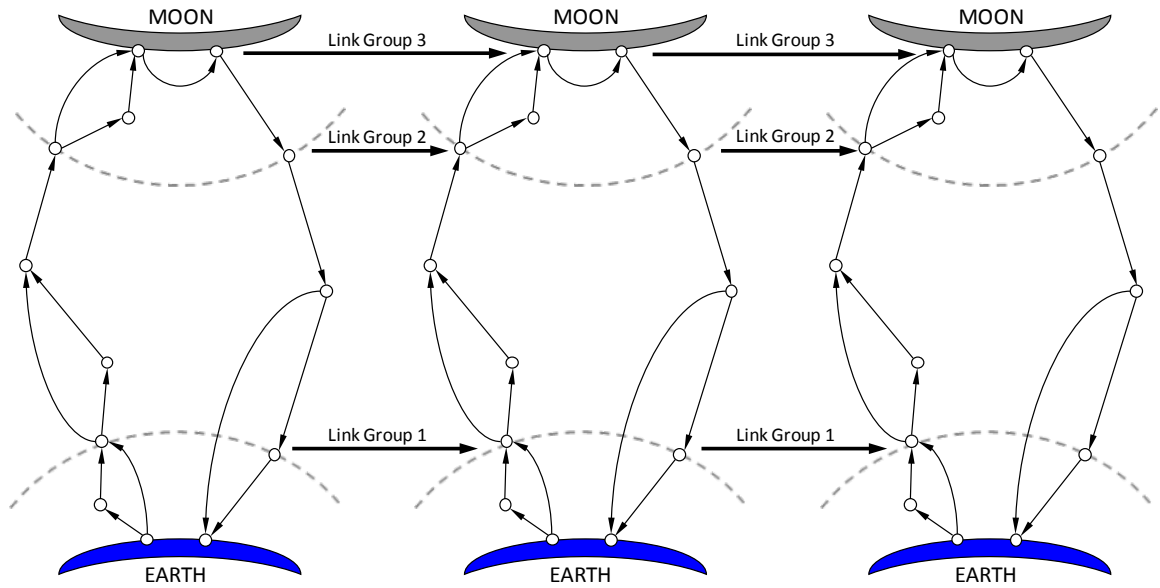


Figure 21: Recursively Defined Lunar Architecture Graph with Multiple Flights

The edges between the nodes in this graph are defined in further detail in Appendix A in Table A-2, along with the options for each edge. Each edge has a defined departure node, an arrival node, and an edge number associated with it. Information can be embedded within these edges and nodes based on the type of edge or node that is used. The edge types vary depending on the function that must be performed for a system to move from one node to another. The edge types that are available to the user are: Earth Launch; Propulsive Maneuvers (In-Space, Planetary Ascent, or Planetary Descent); In-Space Habitation; Surface Habitation; Planetary Entry, Descent, and Landing (EDL); Refuel; and Orbit Capture. The metadata that can be embedded in each type of edge (which is available to the system sizing models for systems traveling along the given edge) is summarized in Table 10.

Table 10: Metadata within Edges for Each Edge Type

Edge Type	Metadata				
Earth Launch	Scenario	Location			
Propulsive Maneuvers					
<i>Propulsive (in-space)</i>	ΔV	T/W	Engine Type	TOF	Planet
<i>Planetary Ascent</i>	ΔV	T/W	Engine Type	Planet	
<i>Planetary Descent</i>	ΔV	T/W	Engine Type	Planet	
In-Space Habitation	Scenario	t_{stay}			
Surface Habitation	Scenario	t_{stay}			
Planetary EDL	V_{entry}	L/D	ΔV	T/W	Planet
Refuel	Launch Cost				
Orbit Capture	V_{entry}	L/D			

Note: t_{stay} = Stay Time, days; TOF = Time of Flight, days; V_{entry} = Entry Velocity, m/s; and L/D = Lift-to-Drag Ratio

With this set of nodes, edges, link groups, and metadata for each edge defined, the system architecture graph presented in Figure 20 is mathematically represented using the adjacency and incidence matrices as defined in Section 2.2 using Equations (1) and (2), respectively. For the lunar system architecture design space graph presented in Figure 20, the adjacency matrix is presented in Figure 22, and a portion of the incidence matrix is presented in Figure 23. For the full, three-flight lunar system architecture graph, the 42x192 incidence matrix is too large to visibly place in this document. More detailed metadata for each edge is given in Appendix A.

3.3. Design Space Exploration

Each system travels through the graphical representation of the system architecture design space, subject to rules and user-defined constraints, to create the system map. The system map is a matrix that defines the edges along which each system traverses. The system hierarchy identifies the dependencies between systems and determines the order in which each system needs to be sized. This information is used by the system sizing tools, where the performance and cost of each individual system within the architecture are calculated.

3.3.1. System Map Overview

The system map is a representation of the traversal of each system through the graph. A path through the graph is a list of edges that are traversed. A method to compactly represent this information is the system map matrix, as shown in Figure 24, which has the system identification numbers as the column indices and the edge identification numbers as the row indices. This matrix is populated with ones and zeros as defined in Equation (9), where a one indicates that the system identified by the column index travels along the edge identified by the row index and therefore performs the function defined by that edge. This structure provides a direct relationship between the metadata within the edges and the systems to which it is pertinent.

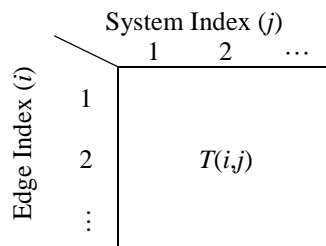


Figure 24: Matrix Representation of a Graph Traversal (System Map)

$$T(i, j) = \begin{cases} 1 & \text{if system } j \text{ traverses edge } i \\ 0 & \text{if system } j \text{ does not traverse edge } i \end{cases} \quad (9)$$

For illustration of this concept, the selected baseline architecture from the Exploration Systems Architecture Study (ESAS) is modeled. This system architecture, presented in Figure 25, is a system architecture that utilizes two different sized launch vehicles: one to deliver the cargo (named the Ares V), and one to deliver the crew (named the Ares I). The first launch delivers the Earth Departure Stage (EDS) and Lunar Surface Access Module (LSAM) to LEO using the cargo launch vehicle. The EDS also performs suborbital burning to reach LEO, where the two systems loiter until the crew arrives. The second launch delivers the crew in the Crew Exploration Vehicle (CEV) (consists of a crew capsule and service module), which rendezvous with the EDS and LSAM. The EDS then performs the Earth departure burn. The LSAM performs both the lunar arrival and descent burns, while the CEV remains in LLO unmanned. After the surface mission, the crew ascends to the CEV and discards the ascent module of the LSAM. The CEV service module then performs the Earth return burn before directly reentering Earth.

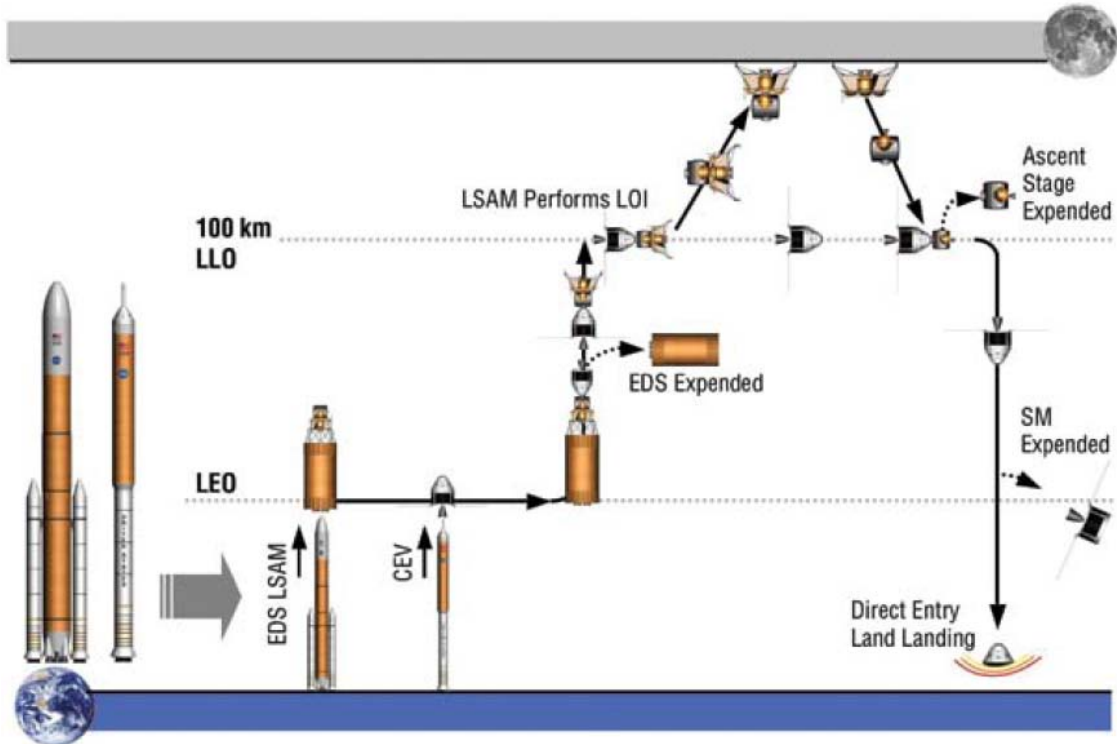


Figure 25: ESAS Baseline System Architecture Concept of Operations [7]

Table 11: System List for ESAS Baseline Architecture

System No.	System Name	System Type
1	Crew	Crew
2	CEV Crew Capsule	Crew Capsule
3	CEV Service Module	Propulsive Stage
4	Ares I	Launch Vehicle
5	Surface Habitat	Surface Habitat
6	LSAM Ascent Stage	Lunar Ascent Stage
7	LSAM Descent Stage	Lunar Descent Stage
8	Earth Departure Stage	Propulsive Stage
9	Ares V	Launch Vehicle

The system map, along with a general description of the edges that are active for the system architecture presented in Figure 25, is presented in Figure 26. The left portion of the figure presents the system map matrix, where the columns correspond to the systems, and the rows correspond to the edges. The system types for this system architecture are presented in Table 11 in the form of the system list. The information in

the right portion of the figure presents the information that describes the functional representation of each edge. More detail on the requirements embedded within these edges, such as ΔV , time of flight, and other requirements is presented in Appendix A.

	1	2	3	4	5	6	7	8	9	Edge Group Name	Edge Group Type	From Node	To Node	Edge Option Name
1	0	0	0	0	0	0	0	0	0	Earth Launch to LEO	Earth Launch	1	3	Falcon Heavy
:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
66	0	0	0	0	1	1	1	1	1	Earth Launch to Suborbital	Earth Launch	15	16	150 mt
:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
70	0	0	0	0	1	1	1	1	0	Suborbital Burn	Propulsive	16	17	LOX/LH2
:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
121	1	1	1	1	0	0	0	0	0	Earth Launch to LEO	Earth Launch	29	31	29 mt
:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
133	1	1	1	0	1	1	1	1	0	TLI from LEO	Propulsive	31	33	LOX/LH2
:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
143	1	1	1	0	1	1	1	0	0	Lunar Orbit Insertion	Propulsive	33	34	LOX/LH2
:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
147	1	0	0	0	1	1	1	0	0	Lunar Descent from LLO	Planetary Descent	34	36	LOX/LH2
:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
157	1	0	0	0	1	1	0	0	0	Lunar Surface Mission	Surface Habitation	36	37	Sortie
:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
160	1	0	0	0	1	1	0	0	0	Lunar Ascent	Planetary Ascent	37	38	LOX/CH4
:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
164	1	1	1	0	0	0	0	0	0	TEI from LLO	Propulsive	38	40	LOX/CH4
:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
172	1	1	0	0	0	0	0	0	0	Direct Entry	Planetary EDL	40	42	Capsule
:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
180	0	0	0	0	1	1	1	1	0	Link Group 1	Loiter	17	27	--
181	0	0	0	0	1	1	1	1	0	Link Group 1	Loiter	27	31	--
:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
187	0	1	1	0	0	0	0	0	0	Link Group 2	Loiter	34	38	--
:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
192	0	0	0	0	0	0	0	0	0	Link Group 3	Loiter	36	37	--

Figure 26: System Map for ESAS Baseline System Architecture

3.3.2. Rule-Based Graph Traversal

For the lunar system architecture graph presented in Figure 20, there are approximately 10^8 possible paths available to each system per flight. Many of these paths, however, are infeasible due to architectural constraints and physical rules that exist. Therefore, an algorithm is developed to effectively traverse the graph while

satisfying all rules. This rule-based traversal of the graph must exist without manual interaction to enable automated exploration of the design space.

The selection of a given system architecture involves defining the paths that each system takes through the architecture graph. The system types that exist (propulsive stage, habitat, etc.) are defined *a priori* and can be added to the system list as needed to satisfy any rules that may be violated.

The rules that must be satisfied are presented in Table 12. Two categories of rules are identified: Existence Rules and Functional Rules. The existence rule force the existence of a system type along an edge if another system type also travels along that edge. If the crew traverses an edge, then a habitat of some sort must also traverse that edge. Functional rules force system types to traverse an edge if that system is necessary to perform the function defined by that edge. For instance, an Earth Launch edge requires a Launch Vehicle system type to traverse it. These rules are only active when necessary, and only one instance of the system type is required.

Table 12: Rule-Based Traversal of Architecture Graph

Existence Rule	
Crew Instance	Surface Habitat OR In-Space Habitat OR Crew Capsule
Functional Rule	
Earth Launch	Launch Vehicle
Propulsive	Propulsive Stage OR Descent Stage OR Ascent Stage
Planetary Ascent	Descent Stage OR Ascent Stage
Planetary Descent	Descent Stage
In-Space Habitation	In-Space Habitat
Surface Habitation	Surface Habitat OR Crew Capsule
Planetary EDL	Crew Capsule
Refuel	Propellant Depot
Orbit Capture	Aerocapture System OR Crew Capsule

During the definition of the systems that exist and their path through the graph, the rules are enforced. As shown in Figure 27, the path generation algorithm cycles

through each of the edges that the systems traverse. Starting with the crew system, if all rules are not met at each traversed edge, there are two options to correct this. The first is to find an existing system that would satisfy the rule. Given that a system exists already within the architecture that can access the edge in question, this system can be used to satisfy the rule. The path generation algorithm probabilistically selects whether or not to utilize this system to satisfy the rule. If selected, the system is forced to traverse the edge with the unsatisfied rule in the system map. This is a stochastic process, and even if a system exists that could satisfy the rule, a new system could be added anyway. Alternatively, if there is not a system that could satisfy the rule already within the architecture or if the path generation algorithm probabilistically determined not to utilize the existing system, then a new system must be created to traverse the edge. This procedure is repeated until all systems and all edges have satisfied the rules. This process of manipulating the pheromone amount along edges to enforce the rules reveals the strength of Ant Colony Optimization (ACO) as a design space exploration algorithm in this application.

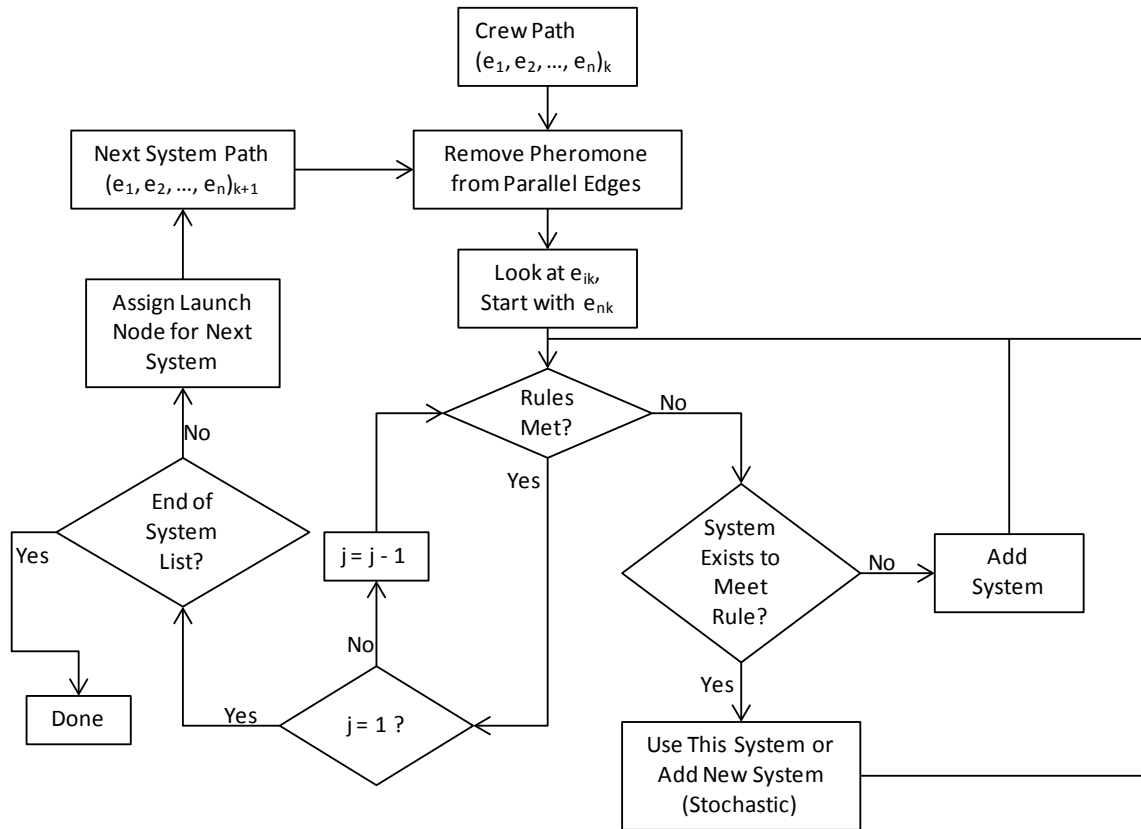


Figure 27: Algorithm for Rule-Based Graph Traversal

Also, this procedure only ensures that a system architecture is functionally feasible. The set of rules were developed to ensure that all functions were performed by appropriate systems and that the combination of systems that perform a given function is appropriate. However, the individual system architectures have not been analyzed yet, and therefore, physical feasibility has not been ensured. A system architecture that is functionally feasible could be physically infeasible if (a) the launch vehicle cannot accommodate the individual system masses, or (b) limitations in the systems prevent them from performing the required functions. For instance, a propulsive stage has a maximum possible mass ratio that can be achieved based on the inert mass fraction ($MR_{max} = 1/f_{inert}$). The required mass ratio is given by Equation (10), which is a function of the performance requirement (ΔV) and the rocket engine efficiency (I_{sp}).

$$MR = \frac{m_{initial}}{m_{final}} = e^{\Delta V / g_0 I_{sp}} \quad (10)$$

The inert mass fraction of the system is determined by the sizing tools, which are discussed in Section 3.4.2. If a given propulsive stage is allocated a required mass ratio that is too large for the system to perform, this system architecture would also be physically infeasible.

3.3.3. System Hierarchy

The next step in sizing the architecture is to determine the hierarchy of the systems being modeled. The system hierarchy is a graph that determines the order in which the systems are sized as well as the information flow between system sizing models. The hierarchy is developed by first determining the topological sort order of each system in the graph based on the length of time each system spends active in the graph. Then, the relationships are identified between systems by recording all instances where multiple systems travel along a single edge in the system map. Finally, a DAG is developed to create the mathematical representations that define the information flow between system sizing models.

Spending more time active within the graph means that the system traverses an edge with a higher identification number. Each edge is given an identification number when the architecture graph is first developed. Systems that traverse edges with higher identification numbers are then placed higher in the topological sort order. In the latter flights and in edges connecting higher numbered nodes from Figure 20, the edge identification numbers are higher. For instance, the Earth launch edges on the first flight will have low numbers, while the Earth entry edges on the final flight will have higher numbers. Therefore, if a system last travels along one of the Earth entry edges, it will be

higher on the hierarchy than a system that last travels along one of the Earth launch edges. An active system is one which traverses along a given edge. In the graphical representation, this system would have a 1 in the system map along the edge along which the system travels and a 0 in the other rows.

In the event that two or more systems travel along the same edge before becoming inactive, the topological sort algorithm must be able to select which system is to be placed higher in the hierarchy. This is decided by developing some general rules based on the type of system involved. These rules are presented in Table 13. As an example, a propellant depot system is sized after all propulsive systems so that the refueling demand and logistics are analyzed after the rest of the architecture, but before the launch vehicles that will supply these systems and propellant.

Table 13: Hierarchy of System Types

Group	System Type
1	Crew
2	In-Space Habitat
	Surface Habitat
	Crew Capsule
3	Propulsive Stage
	Ascent Stage
	Descent Stage
4	Propellant Depot
5	Launch Vehicle

The next task is to determine the links that exist between each of the system models. When two systems travel along the same edge, they can be assumed to have a relationship. Both systems are traveling through the same environment, and in the case of propulsive edges, all systems traveling along that edge will be payloads to the active propulsive system.

The complete process of developing the system hierarchy therefore has all of the required data: the topological sort and the links that exist between systems. Directed edges are created to represent each of these links. The direction of each edge is defined such that the edge always travels from the system (represented as a node in the system hierarchy graph) higher in the topological sort to the one lower. With this list of edges, the full representation of the system hierarchy graph using the adjacency and incidence matrices can be created to fully define the flow of information between system sizing tools.

The system hierarchy is a Directed Acyclic Graph (DAG) that presents the links between all of the system sizing tools. The DAG is used to ensure that the direction of information flow (derived from the topological sort) is included in this single structure. An example of a system hierarchy for the selected architecture in ESAS is shown in Figure 28 [7]. The adjacency and incidence matrices of this DAG are presented in Figure 29 and Figure 30, respectively. Each system sizing tool is a node in this graph. Edges connect each node representing information flow from one tool to another. The colored labels reveal the impact of implementing the hierarchy of system types as defined in Table 13.

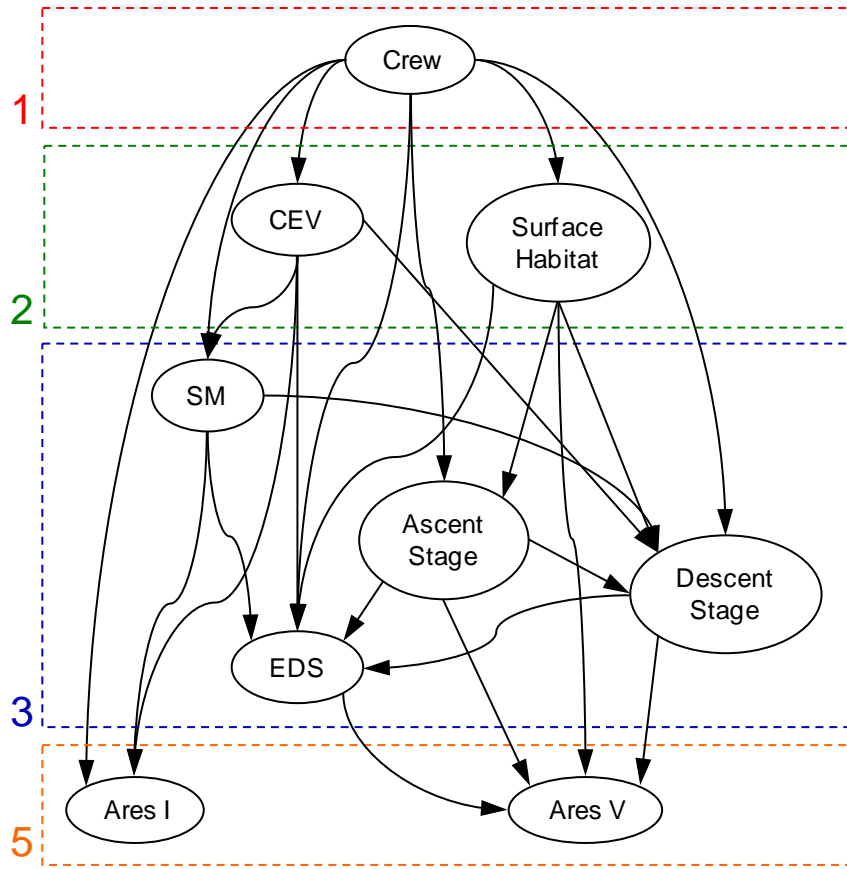


Figure 28: System Hierarchy for Lunar Architecture Selected in ESAS [7]

	Systems									
Systems	0	1	1	1	1	1	1	1	1	0
	1	0	1	1	1	1	1	1	1	0
	1	1	0	1	1	1	1	1	1	0
	1	1	1	0	0	0	0	0	0	0
	1	1	1	0	0	1	1	1	1	1
	1	1	1	0	1	0	1	1	1	1
	1	1	1	0	1	1	0	1	1	1
	1	1	1	0	1	1	1	0	1	1
	0	0	0	0	1	1	1	1	1	0

Figure 29: Adjacency Matrix for the System Hierarchy of the ESAS Baseline Architecture

FY12 dollars were created in NAFCOM for each system type. The exception is the launch vehicle system type, which used Transcost estimates that were specifically developed for launch vehicles and listed prices for existing commercial launch vehicles. System-level regressions for the DDT&E and flight unit costs were developed for a range of dry masses for each system type. The regressions are of the form of the CER in Equation (5) presented in Section 2.4, with the coefficients presented in Table 15. The CER curves for the DDT&E and flight unit costs of each system type are presented in Appendix B.

Table 15: CER Coefficients for Each System Type

System Type	DDT&E Cost		Flight Unit Cost	
	CER Coefficients		CER Coefficients	
	k·a	b	k·a	b
In-Space Habitat (4 crew)	1457.7	0.0856	46.624	0.2146
Surface Habitat (4 crew)	751.64	0.1183	124.32	0.1402
Crew Capsule	285.57	0.2667	49.923	0.2409
Propulsive Stage (Cryogenic)	29.125	0.4554	2.6147	0.4782
Propulsive Stage (Storable)	29.125	0.4554	1.8650	0.4782
Ascent Stage (Cryogenic)	405.62	0.2151	92.715	0.1606
Ascent Stage (Storable)	405.62	0.2151	66.129	0.1606
Descent Stage (Cryogenic)	168.22	0.3152	6.8608	0.4146
Descent Stage (Storable)	168.22	0.3152	4.8935	0.4146
Propellant Depot	75.492	0.3566	11.487	0.3175

The results of the Transcost calculation for Heavy Lift Launch Vehicles (HLLVs) are presented in Table 16, along with the price for the commercial launch vehicles included in the analysis. The Transcost regressions [49] were anchored to a launch vehicle similar to the Cargo Launch Vehicle, the HLLV presented in ESAS [7], [65]. To model various payload capabilities, this launch vehicle was photographically scaled, and the estimates of subsystem masses, DDT&E cost, and flight unit costs were estimated. The cost estimates are given for deliveries to both LEO and to a suborbital point, which

excludes the cost of an upper stage. This upper stage would be accounted for in the propulsive stage that performs the suborbital burn. The commercial launch vehicles assume that there is no DDT&E cost, and the flight unit cost is the price of purchasing a launch vehicle, as reported by the provider [63],[65].

Table 16: Launch Vehicle Cost Model Results Overview

Launch Vehicle	<u>Delivery to Low Earth Orbit</u>			<u>Delivery to Suborbital Point</u>		
	Payload (mt)	DDT&E Cost (FY12, \$M)	Flight Unit Cost (FY12 \$M)	Payload (mt)	DDT&E Cost (FY12, \$M)	Flight Unit Cost (FY12, \$M)
29 mt Crew LV	29	5,502	892	--	--	--
70 mt HLLV	70	13,274	1,551	136	11,004	1,295
100 mt HLLV	100	14,731	1,989	194	12,252	1,663
130 mt HLLV	130	16,746	2,796	251	14,066	2,401
150 mt HLLV	150	18,222	3,472	290	15,413	3,032
Delta IV-H	24	0	318	--	--	--
Falcon Heavy	53	0	135	201	--	135

Finally, estimating the cost of propellant delivery for architectures that utilize on-orbit refueling uses a cost-per-kilogram metric. The current price of existing commercial launch vehicles is \$14,286/kg (based on a Delta IV-H). The projected price for commercial launch vehicles in the future is \$2,358/kg (based on a Falcon Heavy). The inclusion of both of these options allows the system architect to view the difference between current capability and projected future capability.

Concluding the example of the ESAS baseline architecture, a summary of the sizing and cost estimation is provided in Figure 31 and Figure 32. For each system in this architecture, the inputs (which are requirements derived from the edge metadata and/or the other systems in the system hierarchy), the parameters used to perform intermediate calculations within each sizing tool, and the outputs of mass and cost are provided. All units are metric (kg, m, s) and millions of dollars.

INPUTS		PARAMETERS		OUTPUTS	
Crew					
Number of Crew	4			Gross Mass	368
Crew Capsule					
Number of Crew	4			Dry Mass (no Growth)	5,663
Stay Time	9			Dry Mass	6,796
				Inert Mass	7,685
				Gross Mass	7,845
				DDT&E Cost	2,862
				Flight Unit Cost	400
Service Module (Propulsive Stage)					
ΔV	1,196	Isp	353	Dry Mass (no Growth)	2,672
Payload Mass	8,213	Oxidizer Boiloff Rate	0.025	Dry Mass	3,206
Propellant Type	LOX/CH4	Fuel Boiloff Rate	0.033	Inert Mass	3,466
System T/W	0.3	Oxidizer Density	1141	Total Propellant Mass	5,080
Planet	Moon	Fuel Density	415	Usable Propellant Mass	4,820
On-Orbit Time	16	O/F Ratio	3.6	Gross Mass	8,285
				DDT&E Cost	1,059
				Flight Unit Cost	114
Surface Habitat					
Number of Crew	4			Dry Mass (no Growth)	3,221
Stay Time	7			Dry Mass	3,865
				Gross Mass	4,699
				DDT&E Cost	1,997
				Flight Unit Cost	396
Lunar Ascent Stage					
ΔV	1,968	Isp	353	Dry Mass (no Growth)	1,021
Payload Mass	5,067	Engine T/m	473	Dry Mass	1,225
Propellant Type	LOX/CH4	O/F Ratio	3.6	Inert Mass	1,476
System T/W	1.97			Total Propellant Mass	5,009
Planet	Moon			Gross Mass	6,485
				DDT&E Cost	1,800
				Flight Unit Cost	282

Figure 31: Summary of ESAS Baseline Sizing and Cost Estimation (Part 1)

INPUTS		PARAMETERS		OUTPUTS	
Lunar Descent Stage					
ΔV	924	IMF Model	Cryogenic	Dry Mass	8,558
	2,180	Isp	465	Inert Mass	10,698
Payload Mass	27,682			Total Propellant Mass	26,532
	11,552			Gross Mass	37,229
Propellant Type	LOX/LH2			DDT&E Cost	2,920
				Flight Unit Cost	293
EDS (Propulsive Stage)					
ΔV	2,442	Isp	465	Dry Mass (no Growth)	19,227
	3,247	Oxidizer Boiloff Rate	0.025	Dry Mass	23,072
Payload Mass	48,414	Fuel Boiloff Rate	0.185	Inert Mass	29,351
	64,912	Oxidizer Density	1141	Total Propellant Mass	228,538
Propellant Type	LOX/LH2	Fuel Density	71	Usable Propellant Mass	222,259
System T/W	0.8574	O/F Ratio	5.88	Gross Mass	251,610
Planet	Earth			DDT&E Cost	2,601
On-Orbit Time	15			Flight Unit Cost	292
29 mt Launch Vehicle					
Payloads	368			Number of Launches	1
	7,845			DDT&E Cost	5,502
	8,285			Flight Unit Cost	893
Staging Point	LEO				
150 mt Launch Vehicle					
Payloads	4,699			Number of Launches	1
	6,485			DDT&E Cost	15,414
	37,229			Flight Unit Cost	3,032
	251,610				
Staging Point	Suborbital				

Figure 32: Summary of ESAS Baseline Sizing and Cost Estimation (Part 2)

3.5. Exploration of the Space System Architecture Design Space

To explore the system architecture design space, an Ant Colony Optimization (ACO) algorithm is implemented that investigates the design space. Using an optimization algorithm for design space exploration evaluates design points that tend to have better figures of merit compared to a baseline. This strategy enables more rapid exploration of the beneficial regions of the design space, making this tool more useful to the system architect.

3.5.1. Optimization Method

At the start of an iteration of the ACO process, shown in Figure 33 the rule-based graph traversal algorithm creates and analyzes several system architectures within the design space. These system architectures are dependent upon the pheromone matrix that is updated at each step of the ACO process. If the matrix contains only ones, systems within the architectures traverse edges randomly. As the optimization routine continues and pheromone is deposited and evaporated, the probability that systems traverse given edges changes based on the desirability of system architectures in previous iterations.

After a set of system architectures is generated and evaluated for a given iteration, the best NPV from an architecture is compared to the best NPV from previous architectures. If, after five consecutive iterations, the best NPV is not replaced, the algorithm ends and returns the set of system architectures that have been analyzed up to that point. Otherwise, the algorithm continues with the pheromone update step until the maximum number of iterations is reached.

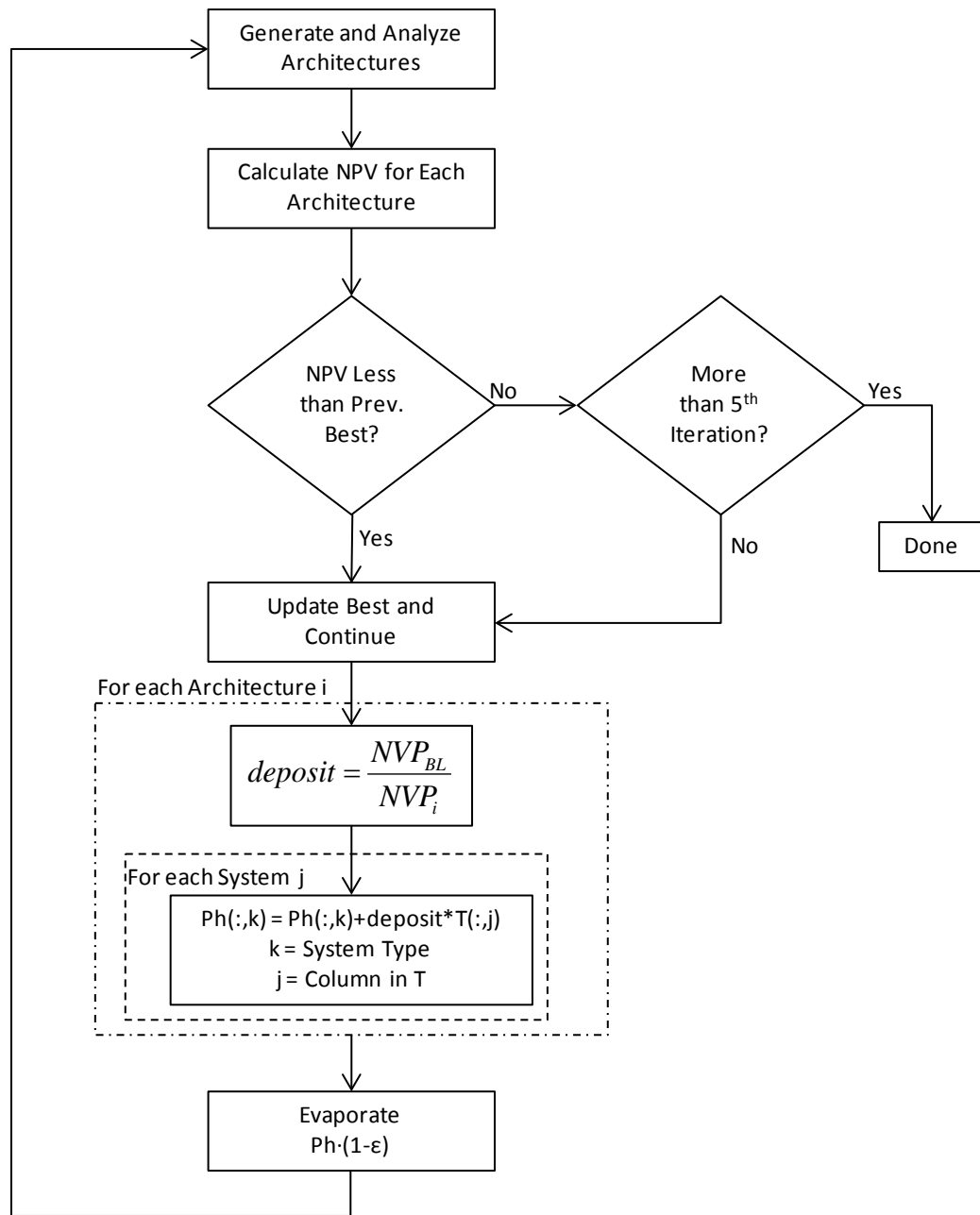


Figure 33: ACO Algorithm for System Architecture Design Space Exploration

In order to guide the rule-based graph traversal toward generating system architectures that are advantageous with respect to NPV, a pheromone update step is included in each iteration. This update step consists of two actions, pheromone deposit and pheromone evaporation. The first action, pheromone deposit, is performed for each

system within each architecture, and it increases the pheromone along edges in the graph (in effect increasing the likelihood that a system will traverse a given edge). Equation (11) provides the mechanism by which this action is done.

$$Ph(:,k) = Ph(:,k) + deposit \cdot T(:,j) \quad (11)$$

In this equation, Ph is the pheromone matrix, where each row corresponds to the edges in the graph and each column k corresponds to a system type, as defined in Table 14. The first index in parentheses is the row index, with a semicolon indicating that all rows are included, and the second index is the column index. The matrix T is the system map, as defined in Figure 24, and the deposit amount is defined in Equation (12) for each architecture i .

$$deposit = \frac{NPV_{BL}}{NPV_i} \quad (12)$$

Finally, the evaporation step removes pheromone from each edge. This process reduces the probability that the rule-based graph traversal algorithm will select architecture options that had feasible or relatively favorable options early in the design space exploration, but have since been deemed less favorable in more recent iterations. This process also keeps the pheromone matrix from continually increasing in magnitude, allowing for a more stable and scalable process. The equation to evaporate the pheromone from the graph is given in Equation (13).

$$Ph = Ph \cdot (1 - \varepsilon) \quad (13)$$

Here, Ph is the pheromone matrix, where column indices correspond to system types and row indices correspond to edges, and ε is the evaporation percentage.

The optimizer does not affect the graph definition, for that is user specified data which defines the design space. The system list is available to the optimizer to change the different systems that exist within the architecture. The system map is available to the optimizer to change the paths of the different systems by changing the ones and zeros in the system map matrix. The system hierarchy and sizing algorithms are run after the perturbation step and calculates the performance, cost, and reliability.

3.5.2. Selection Criterion

The relative investment cost for a given architecture is the difference in DDT&E and flight unit costs between the baseline architecture and the alternative under consideration. Operations, ground infrastructure, and disposal costs which are also included in the life cycle cost of a system architecture are not analyzed in the present study. Those costs are not as closely linked to system mass and cannot be used as a significant discriminator at the current level of fidelity. Doing so would require additional research to quantify these costs and relate these metrics to architecture-level decisions. The DDT&E and flight unit costs for systems and launch vehicles can be calculated with NAFCOM and Transcost, respectively. Equation (14) shows the mathematical form for the present value of cost for a given architecture relative to a baseline. The benefits of a given architecture are equivalent if the two architectures under consideration perform the same mission. Therefore, in a relative comparison, the relative present value of the benefits is zero. Equation (15) shows the mathematical form of the objective used in comparing two architectures: Relative Net Present Value

(RNPV). This definition of costs and benefits has the convention a lower RNPV denoting a preferred alternative, similar to a cost comparison.

$$PV(Costs) = PV(C_{DDT\&E}^* - C_{DDT\&E}^{bl}) + PV(C_{FlightUnit}^* - C_{FlightUnit}^{bl}) \quad (14)$$

$$RNPV = PV(Costs) \quad (15)$$

Using RNPV as an objective in an architecture-level optimization shows the impact that an architecture decision has in the long term. RNPV is able to provide the implications from a cost perspective of a given architecture alternative when considering the full human exploration program over the next several decades. With each architecture alternative, the RNPV will be calculated and compared to a baseline architecture. If the RNPV is negative (i.e. system architecture in question costs less than the baseline), then NASA would be getting more value in that architecture alternative over the baseline. If the RNPV is greater than zero, then the baseline architecture presents more value to NASA.

CHAPTER 4

VALIDATION

This chapter uses the modeling framework to analyze a sample architecture-level design space to validate that the framework meets the goals set forth in this research. The chapter provides a description of the conversion of the system architecture design space to the representation as a graph. Then, individual system architectures are defined within this representation and mass and cost estimates of these system architectures are compared.

4.1. Lunar Mission Design Space

In January 2004, President George W. Bush, through the issuance of the Vision for Space Exploration, provided a goal for NASA to return humans to the Moon by 2020. In response to this direction, the Exploration Systems Architecture Study (ESAS) explored several options for achieving this goal, and using the FOMs presented in Figure 17 in section 2.6, selected an architecture that met the constraints and scored highly with respect to the FOMs [7].

ESAS identified the various mission modes that existed in the design space, as shown in Figure 34. The modes were split into the taxonomy based on whether or not there was a rendezvous in Earth orbit, lunar orbit, both, or none. The Apollo mission utilized a Lunar Orbit Rendezvous (LOR) without Earth Orbit Rendezvous (EOR). Other combinations include EOR only (also known as EOR-Direct), LOR only, and EOR-LOR, which ESAS deemed the best solution [7].

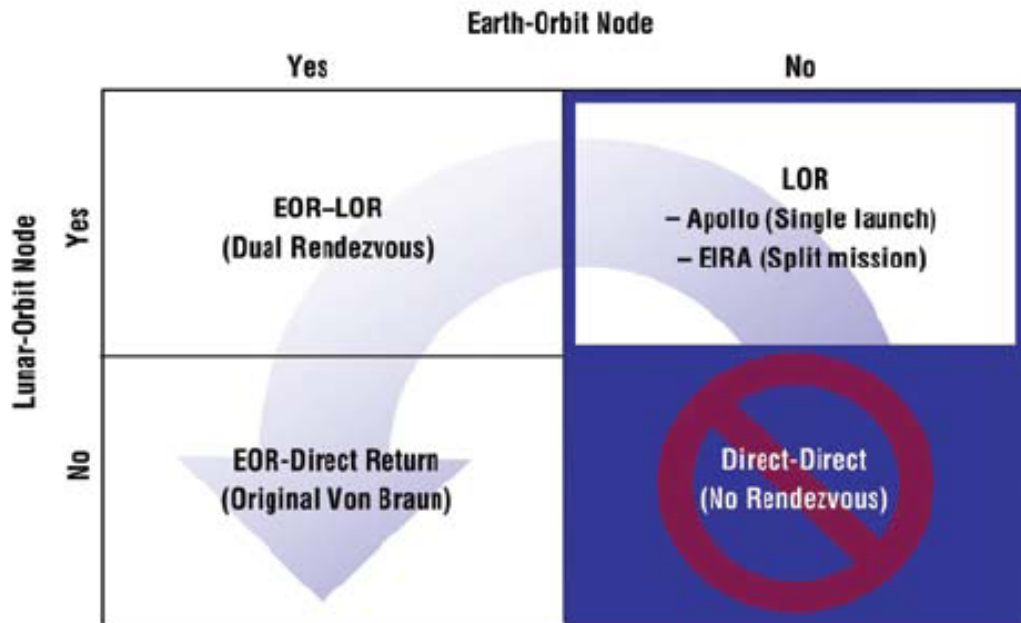


Figure 34: ESAS Lunar Mission Mode Taxonomy [7]

These options fit within the lunar system architecture design space graph presented in Figure 20 and Table 9 with the edges defined in Appendix A. To explore the same modes for the lunar design space defined by the graph, the system architect defines different paths for systems to take within that graph. The various options defined by ESAS in Figure 34 are enumerated below, along with an option that is dissimilar to the ESAS system architectures, which will demonstrate the flexibility of the modeling framework.

4.2. Architecture Definition

Throughout the analysis of this system architecture design space, several systems are given acronyms to denote their primary function. Table 17 presents a summary of these systems, including the full name, acronym, system type (defining the sizing tool used), and potential functions that these systems perform in the various system architecture alternatives. The Earth Departure Stage (EDS) is a propulsive stage that

performs large propulsive burns, such as Trans-Lunar Injection (TLI) or Lunar Orbit Insertion (LOI), and has a large propellant capacity. The Lunar Surface Access Module (LSAM) is divided into three systems that provide the three functions that are required for lunar surface access: planetary descent, surface habitation, and planetary ascent. There is flexibility in the functionality of these stages as the descent stage can also perform LOI, the ascent stage can also perform Trans-Earth Injection (TEI), and the surface habitat can be removed in lieu of a crew capsule. The Crew Exploration Vehicle (CEV) consists of two systems: a Command Module (CM), which is the crew capsule that provides habitation in space or on the surface to replace the surface habitat, and provides Earth entry capability; and a Service Module (SM) which is a propulsive stage that performs the TEI burn. While functionally equivalent, the physical differences between the EDS and SM are presented in Figure 35. The SM is a small stage that has multiple tanks positioned radially while the EDS has two large tanks positioned axially. This differentiation is automatically made within the sizing tool based on propellant load. Also, while these system names are used throughout in different architectures, the size and propellant usage are typically not equal for each instance.

Table 17: Overview of Systems Used in ESAS Mission Modes Comparison

System Name	Acronym	System Type	Potential Function(s)
Earth Departure Stage	EDS	Propulsive Stage	TLI, LOI
Lunar Surface Access Module	LSAM	Lunar Descent Stage	LOI, Planetary Descent, Planetary Ascent
		Lunar Ascent Stage	Planetary Ascent, TEI
		Surface Habitat	Crew Habitation (surface)
Crew Exploration Vehicle Command Module	CEV CM	Crew Capsule	Crew Habitation (in-space or surface), Earth Entry
Crew Exploration Vehicle Service Module	CEV SM	Propulsive Stage	TEI

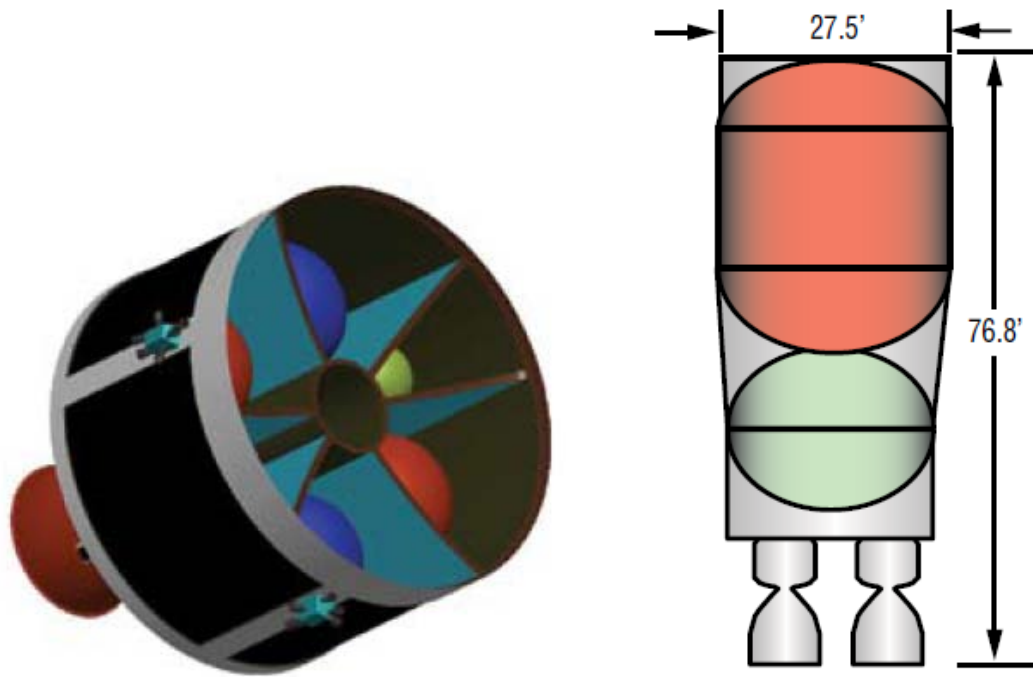


Figure 35: Comparison of CEV SM (left) and EDS (right) Configuration (not to scale) [7]

The first system architecture, presented in Figure 36, is the LOR-LOR system architecture. The first launch delivers an Earth Departure Stage and a two-stage LSAM to LEO using a Heavy Lift Launch Vehicle (HLLV). The EDS performs the TLI and LOI burns. The LSAM (which consists of a surface habitat, an ascent stage, and a descent stage) loiters in LLO. The next launch delivers an EDS and the crew in the CEV command module and the service module. Again, the EDS performs the TLI and LOI burns. The crew rendezvous in LLO with the LSAM, and descends to the surface while the CEV remains in LLO unmanned. After the surface mission, the crew ascends to the CEV and discards the ascent module of the LSAM. The CEV SM then performs the TEI burn to return directly to Earth. In this and all subsequent architectures, the standard EDS and LSAM descent module use Liquid Oxygen/Liquid Hydrogen (LOX/LH₂) propellant, and the LSAM ascent module and CEV service module use Liquid

Oxygen/Liquid Methane (LOX/CH₄) propellant. Within the present framework, this implementation can be easily changed to explore more of the design space.

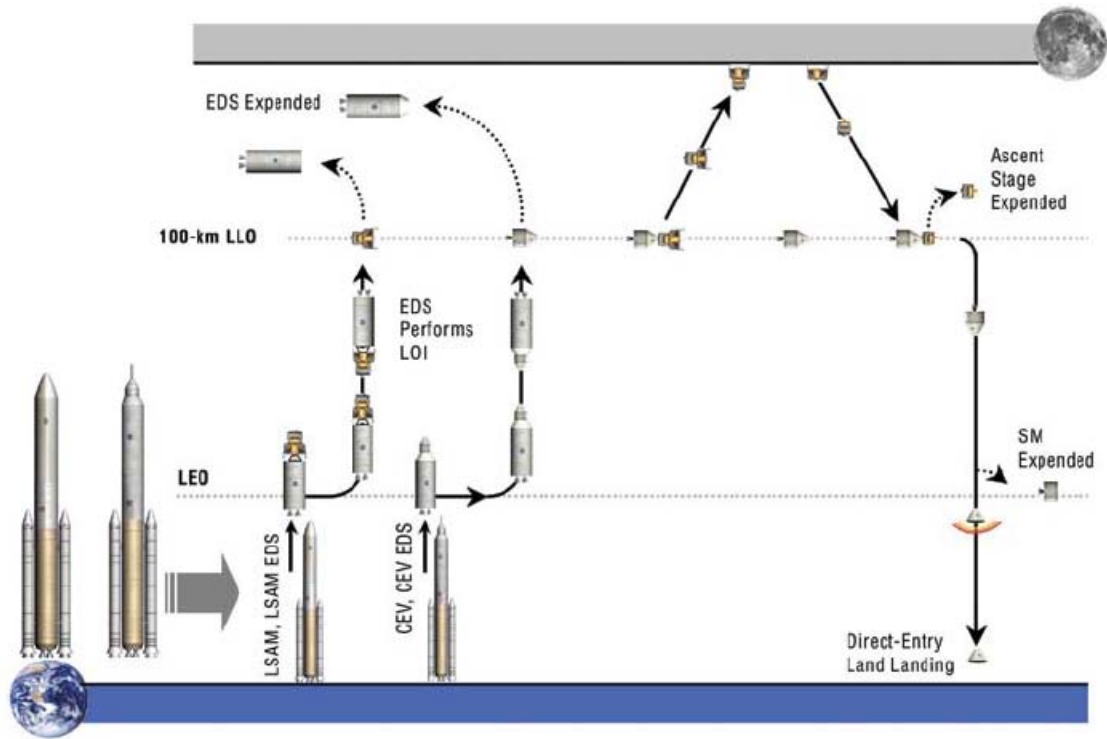


Figure 36: LOR-LOR Lunar System Architecture Option [7]

The second system architecture, presented in Figure 37, is an EOR-LOR system architecture that utilizes two different sized launch vehicles: one to deliver the cargo, and one to deliver the crew. The first launch delivers the EDS and LSAM to LEO using the cargo launch vehicle. The EDS also performs suborbital burning to reach LEO, where the two systems loiter until the crew arrives. The second launch delivers the crew in the CEV, which rendezvous with the EDS and LSAM. The EDS then performs the TLI burn. The LSAM performs both the LOI and descent burns, while the CEV remains in LLO unmanned. After the surface mission, the crew ascends to the CEV and discards the ascent module of the LSAM. The CEV service module then performs the TEI burn to return directly to Earth.

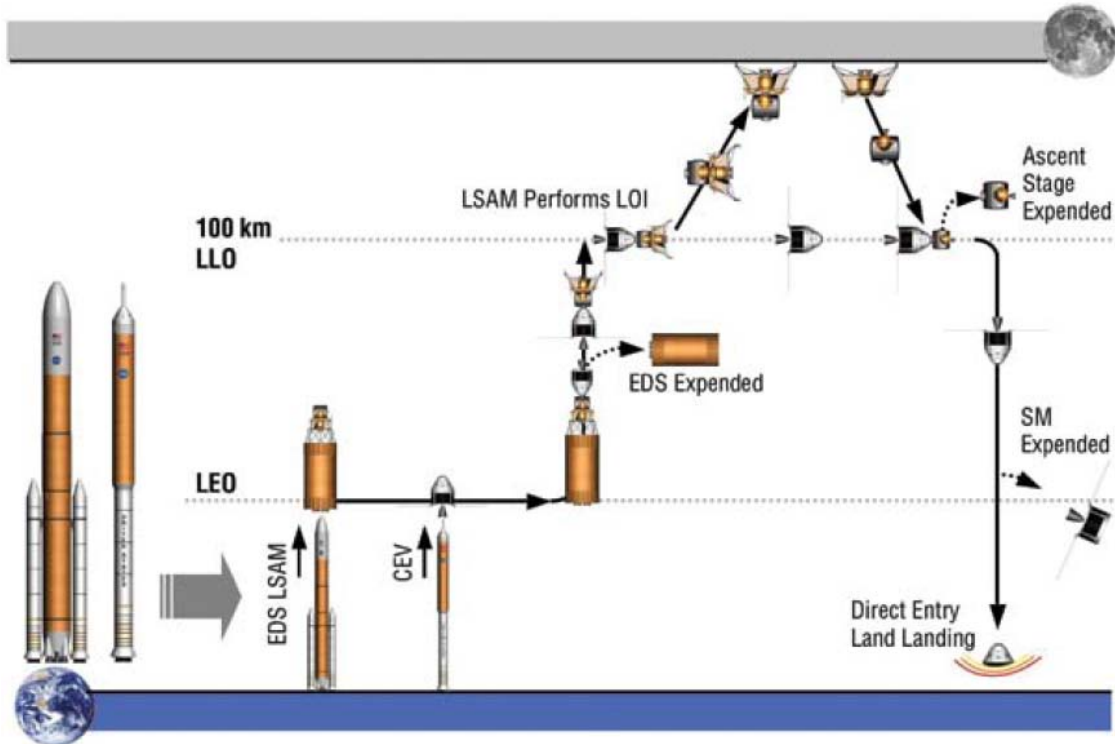


Figure 37: 1.5-Launch EOR-LOR System Architecture Option [7]

The third system architecture, presented in Figure 38, is an EOR-Direct system architecture that does not rendezvous in LLO at any time. The first two launches deliver two EDSs to LEO. The third launch delivers the crew in the CEV and a descent stage, which rendezvous with the two EDSs. In this architecture, the CEV will serve as the surface habitat and ascent stage. The EDSs then combine to perform the TLI and LOI burns. The descent stage performs the descent burn, and the crew lives in the CEV during the surface mission. After the surface mission, the crew ascends, using the CEV SM to perform both the ascent and TEI burns to return directly to Earth.

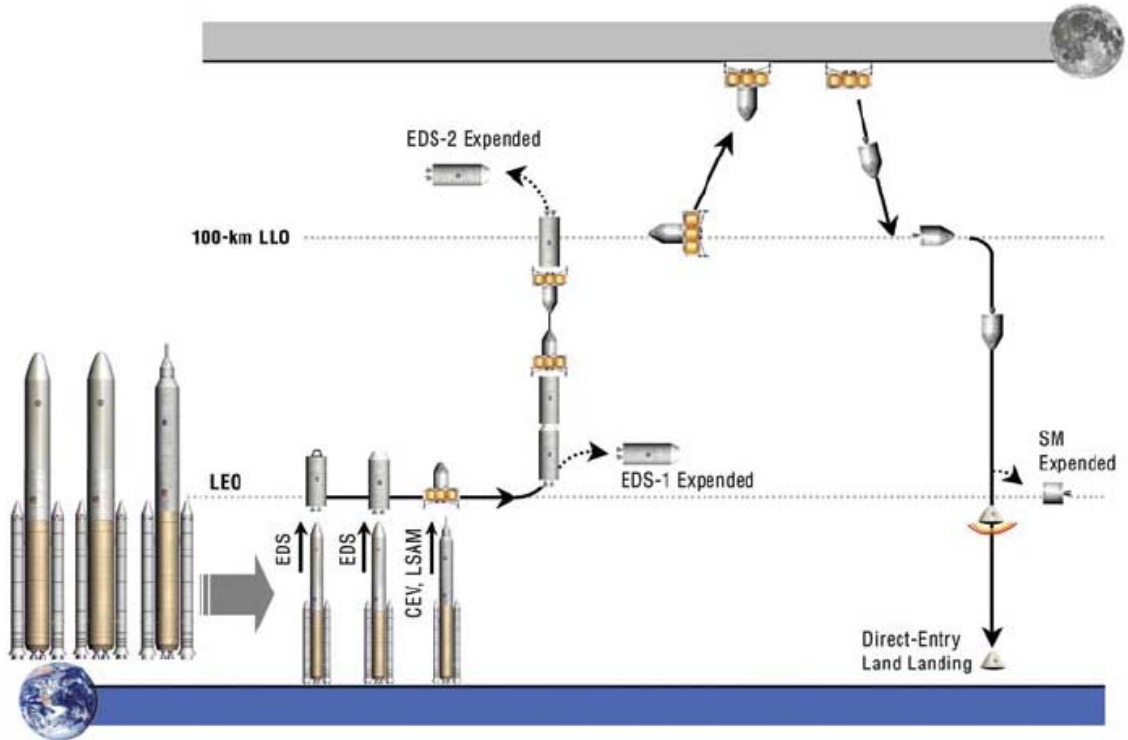


Figure 38: EOR-Direct System Architecture Option [7]

The fourth and final system architecture presented in ESAS, presented in Figure 39, is a modified EOR-Direct system architecture that leaves the CEV service module in LLO while crew still uses the capsule as the surface habitation. The first two launches deliver two EDSs to LEO. The third launch delivers the crew in the CEV, a descent stage, and an ascent stage, which rendezvous with the two EDSs in LEO. The EDSs again combine to perform the TLI and LOI burns. The descent stage performs the descent burn, and the crew lives in the CEV during the surface mission. In this system architecture, the CEV service module remains in LLO. After the surface mission, the crew ascends using the ascent stage, rendezvous with the CEV service module. The CEV service module then performs the TEI burn and the crew returns directly to Earth.

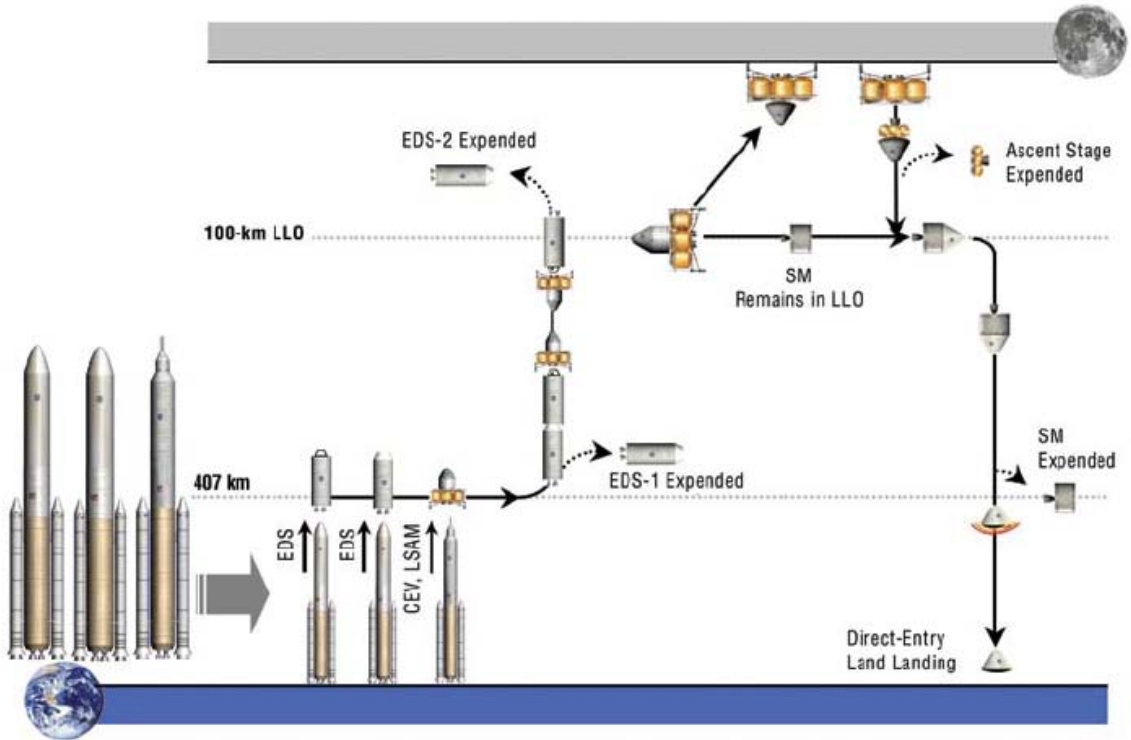


Figure 39: Modified EOR-Direct (SM Remains in LLO) System Architecture Option [7]

To supplement these system architectures and to test the flexibility of the modeling framework, another system architecture type was added to the trade space. The system architecture, presented in Figure 40, is an EOR-LOR mission that utilizes commercial launch vehicles and on-orbit refueling [66]. This architecture type is significantly different than the architectures presented in ESAS which utilize HLLVs and do not include on-orbit refueling.

In this system architecture, a commercial launch vehicle (in this case, the Falcon Heavy under development by Space Exploration Technologies (SpaceX)) delivers a propellant depot to LEO. Then, propellant is transferred into the depot using subsequent commercial launches. Once the propellant depot is filled, an EDS is delivered to LEO, which receives all the propellant that was stored in the propellant depot. The next launch delivers the CEV and a two-stage LSAM to LEO. In this system architecture, the crew

can be launched in the CEV or utilize a commercial crew launch capability and transfer into the CEV on orbit. From LEO, the EDS performs the TLI and LOI burns. The CEV remains in LLO while the crew performs the surface mission in the LSAM. After the surface mission, the crew ascends to the CEV and discards the ascent module of the LSAM. The CEV service module then performs the TEI burn to return directly to Earth.

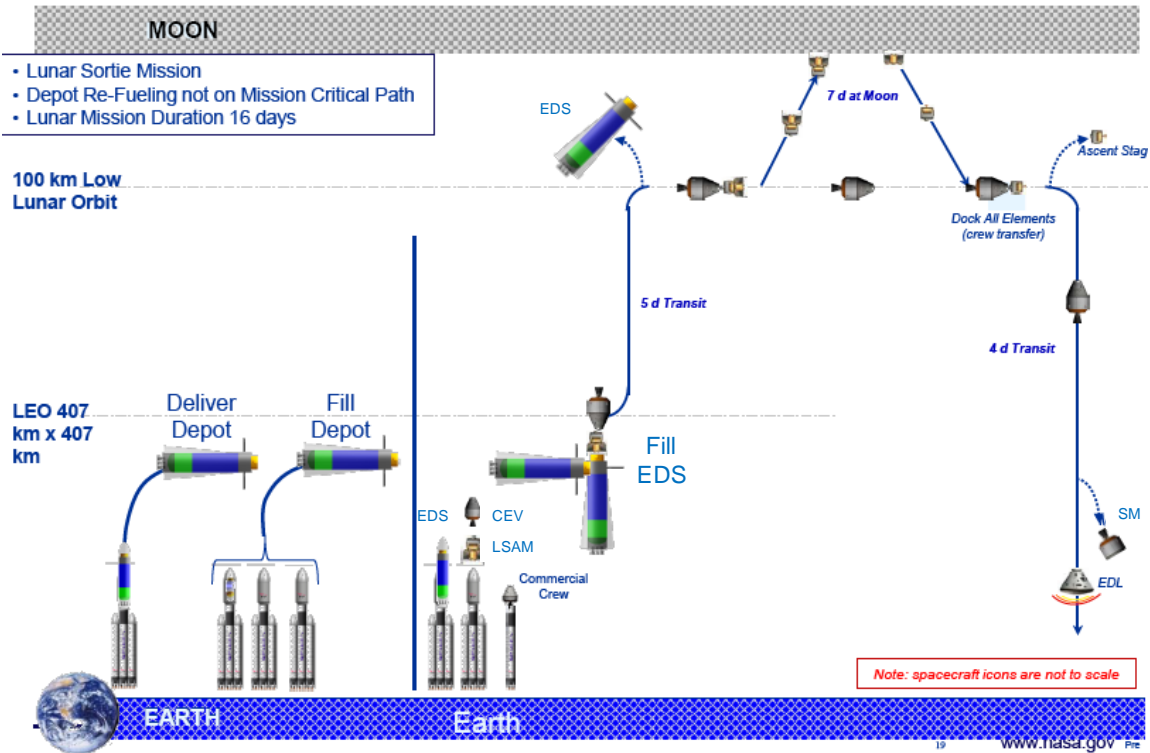


Figure 40: EOR-LOR System Architecture with On-Orbit Refueling and Commercial Launch Vehicles [66]

4.3. Analysis Results and Validation

The lunar system architectures above were analyzed using the modeling framework, providing estimates of mass, cost, and NPV. Beyond the architecture options, propellant type was also varied. ESAS selected LOX/CH₄ propellant for the CEV and LSAM to promote commonality with Mars missions with in-situ resource utilization. The present framework also permits the exploration of multiple propellant

types, such as Nitrogen Tetroxide (NTO)/Monomethylhydrazine (MMH). The complete design space is presented in Table 18. The results and a discussion on the validity of the analysis results are presented in the following section.

Table 18: Overview of Architecture Options (Bold Text Indicates Baseline)

No.	Mode	EDS Propellant	CEV SM Propellant	Ascent Stage Propellant
1	LOR-LOR	LOX/LH2	LOX/CH4	LOX/CH4
2	LOR-LOR	LOX/LH2	NTO/MMH	NTO/MMH
3	1.5-Launch EOR-LOR	LOX/LH2	LOX/CH4	LOX/CH4
4	1.5-Launch EOR-LOR	LOX/LH2	NTO/MMH	NTO/MMH
5	1.5-Launch EOR-LOR	LOX/CH4	LOX/CH4	LOX/CH4
6	EOR-Direct	LOX/LH2	LOX/CH4	LOX/CH4
7	EOR-Direct	LOX/LH2	NTO/MMH	NTO/MMH
8	EOR-Direct (SM in LLO)	LOX/LH2	LOX/CH4	LOX/CH4
9	EOR-Direct (SM in LLO)	LOX/LH2	NTO/MMH	NTO/MMH
10	Commercial with Depots	LOX/LH2	LOX/CH4	LOX/CH4
11	Commercial with Depots	LOX/LH2	NTO/MMH	NTO/MMH

4.3.1. Analysis Results

The resulting estimates of cost and mass are presented in Table 19. While system architecture number 5 is functionally feasible as defined above, it is physically infeasible due to the low specific impulse of LOX/CH4 stages. The larger propulsive stages resulting in the use of LOX/CH4 propellant do not fit into any launch vehicle option included in the graph. Note that the launch vehicles used in this analysis have a fixed LEO payload delivery capability, while the propulsive stages were sized to meet the functional requirements of the system architecture. Also, a single launch vehicle system can represent multiple launch vehicles if the payloads must be divided onto multiple flights.

Table 19: Results from ESAS Mode Analysis

System No.	System	Qty	DDT&E Cost (FY12, \$M)	Flight Unit Cost (FY12, \$M)	Inert Mass (kg)	Gross Mass (kg)
Architecture 1						
1	Crew	4	--	--	--	--
2	Crew Capsule (CEV CM)	1	2,862	400	7,845	7,845
3	Propulsive Stage (CEV SM)	1	1,059	114	3,466	8,285
4	Propulsive Stage (TEI Stage 2)	1	1,269	138	5,740	39,007
5	Launch Vehicle (130 mt HLLV)	1	16,746	2,796	--	--
6	Surface Habitat	1	1,997	396	4,699	4,699
7	Ascent Stage	1	1,800	282	1,476	6,485
8	Descent Stage	1	2,609	253	7,487	19,680
9	Propulsive Stage (TEI Stage 1)	1	1,587	174	9,418	69,678
10	Launch Vehicle (130 mt HLLV)	1	--	2,796	--	--
Architecture 2						
1	Crew	4	--	--	--	--
2	Crew Capsule (CEV CM)	1	2,862	400	7,845	7,845
3	Propulsive Stage (CEV SM)	1	963	74	2,820	8,016
4	Propulsive Stage (TEI Stage 2)	1	1,262	137	5,666	38,419
5	Launch Vehicle (130 mt HLLV)	1	16,746	2,796	--	--
6	Surface Habitat	1	1,997	396	4,699	4,699
7	Ascent Stage	1	1,683	191	1,158	6,422
8	Descent Stage	1	2,608	253	7,475	19,618
9	Propulsive Stage (TEI Stage 1)	1	1,584	174	9,388	69,416
10	Launch Vehicle (130 mt HLLV)	1	--	2,796	--	--
Architecture 3 (Baseline)						
1	Crew	4	--	--	--	--
2	Crew Capsule (CEV CM)	1	2,862	400	7,845	7,845
3	Propulsive Stage (CEV SM)	1	1,059	114	3,466	8,285
4	Launch Vehicle (Crew, 29 mt)	1	5,502	893	--	--
5	Surface Habitat	1	1,997	396	4,699	4,699
6	Ascent Stage	1	1,800	282	1,476	6,485
7	Descent Stage	1	2,920	293	10,698	37,229
8	Propulsive Stage (EDS)	1	2,601	292	29,351	251,610
9	Launch Vehicle (150 mt HLLV, Suborbital)	1	15,414	3,032	--	--
Architecture 4						
1	Crew	4	--	--	--	--
2	Crew Capsule (CEV CM)	1	2,862	400	7,845	7,845
3	Propulsive Stage (CEV SM)	1	963	74	2,820	8,016
4	Launch Vehicle (Crew, 29 mt)	1	5,502	893	--	--
5	Surface Habitat	1	1,997	396	4,699	4,699
6	Ascent Stage	1	1,709	194	1,245	6,917
7	Descent Stage	1	2,928	294	10,792	37,800
8	Propulsive Stage (EDS)	1	2,613	294	29,660	254,697
9	Launch Vehicle (150 mt HLLV, Suborbital)	1	15,414	3,032	--	--
Architecture 5						
PHYSICALLY INFEASIBLE						

Continued on next page

System No.	System	Qty	DDT&E Cost (FY12, \$M)	Flight Unit Cost (FY12, \$M)	Inert Mass (kg)	Gross Mass (kg)
Architecture 6						
1	Crew	4	--	--	--	--
2	Crew Capsule (CEV CM)	1	2,862	400	7,845	7,845
3	Launch Vehicle (100 mt HLLV)	1	14,731	1,990	--	--
4	Ascent Stage (also SM)	1	2,415	351	5,851	26,866
5	Descent Stage	1	2,972	300	11,317	41,028
6	Propulsive Stage (LOI Stage)	1	1,014	109	3,790	21,736
7	Propulsive Stage (TLI Stage)	1	2,051	228	16,778	135,772
8	Launch Vehicle (100 mt HLLV)	2	--	3,979	--	--
Architecture 7						
1	Crew	4	--	--	--	--
2	Crew Capsule (CEV CM)	1	2,862	400	7,845	7,845
3	Launch Vehicle (100 mt HLLV)	1	14,731	1,990	--	--
4	Ascent Stage (also SM)	1	2,294	241	4,964	28,725
5	Descent Stage	1	2,993	303	11,574	42,640
6	Propulsive Stage (LOI Stage)	1	1,030	111	3,928	22,685
7	Propulsive Stage (TLI Stage)	1	2,085	232	17,396	141,621
8	Launch Vehicle (100 mt HLLV)	2	--	3,979	--	--
Architecture 8						
1	Crew	4	--	--	--	--
2	Crew Capsule (CEV CM)	1	2,862	400	7,845	7,845
3	Launch Vehicle (100 mt HLLV)	1	14,731	1,990	--	--
4	Ascent Stage	1	1,988	304	2,348	10,433
5	Descent Stage	1	2,745	270	8,789	26,357
6	Propulsive Stage (LOI Stage)	1	1,506	165	7,655	21,344
7	Propulsive Stage (TLI Stage)	1	1,857	205	13,452	104,897
8	Launch Vehicle (100 mt HLLV)	2	--	3,979	--	--
9	Propulsive Stage (CEV SM)	1	1,059	114	3,466	8,285
Architecture 9						
1	Crew	4	--	--	--	--
2	Crew Capsule (CEV CM)	1	2,862	400	7,845	7,845
3	Launch Vehicle (100 mt HLLV)	1	14,731	1,990	--	--
4	Ascent Stage	1	1,887	208	1,982	11,144
5	Descent Stage	1	2,756	272	8,910	27,012
6	Propulsive Stage (LOI Stage)	1	1,515	166	7,757	21,715
7	Propulsive Stage (TLI Stage)	1	1,871	207	13,667	106,858
8	Launch Vehicle (100 mt HLLV)	2	--	3,979	--	--
9	Propulsive Stage (CEV SM)	1	963	74	2,820	8,016

Continued on next page

System No.	System	Qty	DDT&E Cost (FY12, \$M)	Flight Unit Cost (FY12, \$M)	Inert Mass (kg)	Gross Mass (kg)
Architecture 11						
1	Crew	4	--	--	--	--
2	Crew Capsule (CEV CM)	1	2,862	400	7,845	7,845
3	Propulsive Stage (CEV SM)	1	1,059	114	3,466	8,285
4	Ascent Stage	1	1,800	282	1,476	6,485
5	Descent Stage	1	2,609	253	7,487	19,680
6	Launch Vehicle (Falcon Heavy)	2	--	270	--	--
7	Propulsive Stage (TLI/LOI Stage)	1	1,851	205	13,249	103,920
8	Propellant Depot	1	2,150	512	32,532	135,582
9	Launch Vehicle (Falcon Heavy)	1	--	135	--	--
10	Surface Habitat	1	1,997	396	4,699	4,699
Architecture 12						
1	Crew	4	--	--	--	--
2	Crew Capsule (CEV CM)	1	2,862	400	7,845	7,845
3	Propulsive Stage (CEV SM)	1	963	74	2,820	8,016
4	Ascent Stage	1	1,709	194	1,245	6,917
5	Descent Stage	1	2,619	254	7,572	20,095
6	Launch Vehicle (Falcon Heavy)	2	--	270	--	--
7	Propulsive Stage (TLI/LOI Stage)	1	1,859	206	13,378	105,108
8	Propellant Depot	1	2,158	516	32,880	137,126
9	Launch Vehicle (Falcon Heavy)	1	--	135	--	--
10	Surface Habitat	1	1,997	396	4,699	4,699

The modeling framework does not include the DDT&E cost of a launch vehicle system if a similarly sized launch vehicle is already developed for a given system architecture. The two LOR-LOR architectures (numbers 1 and 2) only have two launches, but the launch vehicle is the 130 mt HLLV. The four EOR-Direct architectures (numbers 7, 8, 9, and 10) use three launches of the smaller 100 mt HLLV. The 1.5-launch architectures (numbers 3, 4, and 5) utilize the 150 mt HLLV (which is larger than the three launch vehicles considered for the design space exploration of Chapter 5, but can be modeled within the framework), which stages at a suborbital point. The EDS must perform the rest of the ascent ΔV in addition to its in-space burns. Note that the 150 mt classification indicates the launch vehicle's payload capability to LEO. The suborbital staging mass for the configuration used in this analysis is on the order of twice

the LEO payload capability. This ratio can vary depending on the ΔV splits and thrust-to-weight ratio.

Figure 41 presents a comparison of the Initial Mass in Low Earth Orbit (IMLEO) and RNPV for each system architecture. The IMLEO is the total mass of the systems including the refueled propellant, but not including the launch vehicles and suborbital propellant. Note that there is no clear trend between IMLEO and RNPV. The refueling architectures that use commercial launch vehicles have the lowest RNPV, but also have the highest IMLEO. Alternatively, the LOR-LOR architectures have the lowest IMLEO, but have higher RNPV than many system architectures with higher IMLEO. Overall, this plot shows distinct levels of RNPV for each launch vehicle type—Falcon Heavy (10 and 11), 100 mt HLLV (6, 7, 8, and 9), 130 mt HLLV (1 and 2), and 150 mt HLLV with 29 mt crew launch vehicle (3 and 4). The larger HLLV DDT&E and flight unit costs increase the RNPV while the commercial launch vehicles provide significant cost savings due to their low flight unit costs and no DDT&E cost. Alternatively, system architecture decisions such as propellant type have a relatively small impact on RNPV.

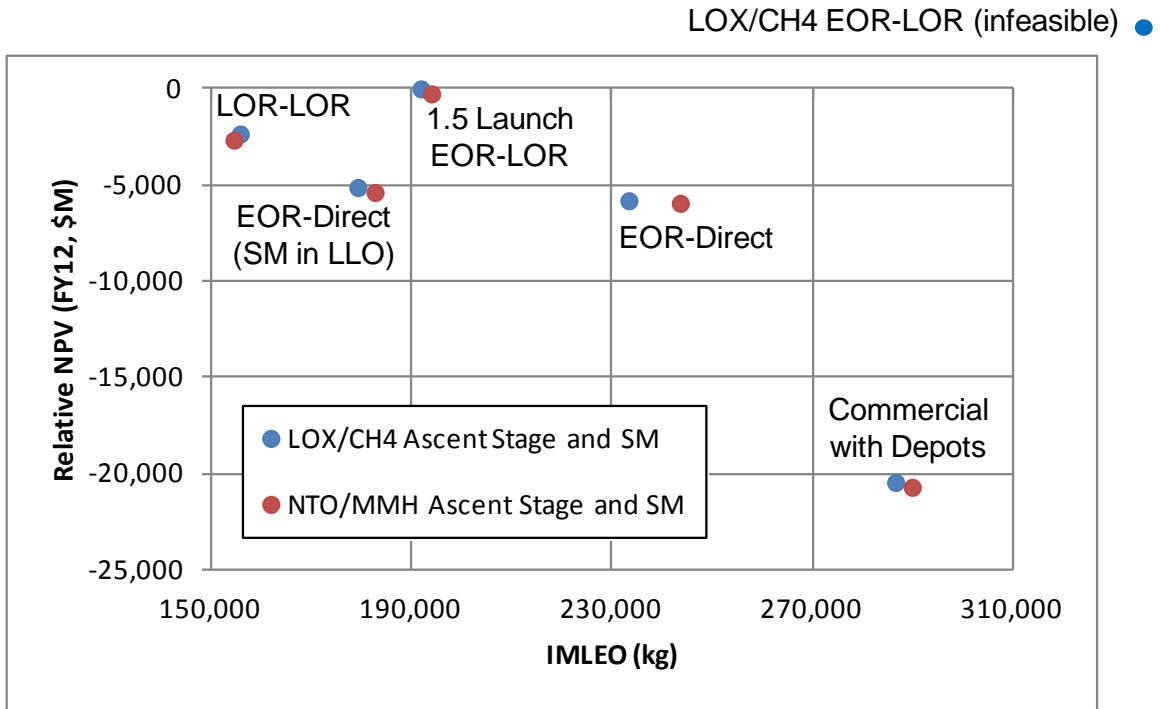


Figure 41: Mass and RNPV Summary Plot for ESAS Mode Analysis

The difference between the IMLEO in system architectures 6 and 7 is larger because changes in the ascent/TEI combo stage has a very high impact on all of the systems lower in the system hierarchy compared to the architectures that use single-use systems (separate ascent and TEI stages). In stages that perform smaller ΔV s, the Inert Mass Fraction (IMF) is increasingly more impactful on gross mass. Therefore, switching propellants on the ascent stage and CEV service module propellants to NTO/MMH, which has lower I_{sp} and higher IMF, increases the gross mass of the larger stage more. This increase in system mass also increases the payload mass for the stages lower in the system hierarchy. The larger change in the ascent/TEI combo stage mass in architecture 7 over architecture 6 (EOR-Direct architectures), therefore, produces a larger increase in IMLEO as compared to the other system architecture pairs.

Figure 42 separates the overall RNPV into its components of DDT&E RNPV and flight unit RNPV. The pairs of points indicate the different architecture modes presented

above, and the difference between the two points within a given pair represents the difference in cost for changing the ascent stage and CEV service module from LOX/CH4 to NTO/MMH. It is noteworthy that the data plotted does not include flight rate and learning effects on mission cost. These effects exist over multiple missions to a given destination and for missions that use multiples of a given system.

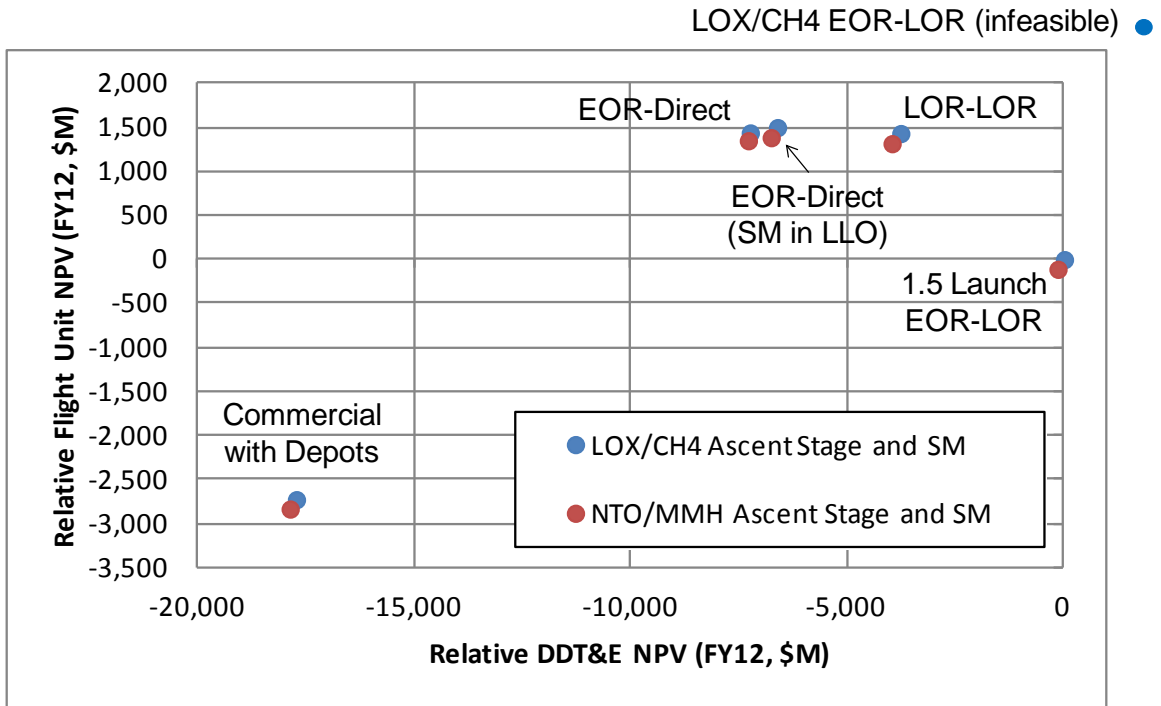


Figure 42: Relative DDT&E NPV and Flight Unit NPV Summary Plot for ESAS Mode Analysis

In every case, NTO/MMH propellant provides a savings in RNPV, although this change is very small compared to the changes between the system architecture modes. The RNPV only considers DDT&E and flight unit costs; however, the toxicity of NTO/MMH would have a significant impact on the operations cost of an architecture. This impact would need to be quantified before this architecture decision was made. The baseline EOR-LOR system architectures (3 and 4) have the lowest flight unit RNPV of any of the ESAS architectures that utilize an HLLV. However, these architectures also

have the highest DDT&E RNPV of any architecture. Using the smaller HLLV reduces the DDT&E RNPV, but the flight unit RNPV is very similar for the system architectures that use HLLVs. The baseline architectures use a combination of HLLV and a crew launch vehicle, which reduces the flight unit RNPV. Finally, the commercial launch architecture with propellant refueling provides significant savings in both DDT&E and flight unit RNPV.

The results in Figure 42 show that the LOR-LOR and both EOR-Direct architectures have higher flight unit RNPV and lower DDT&E RNPV than the baseline. The DDT&E cost for the LOR-LOR architectures is approximately \$3-4B more than the EOR-Direct architectures because of the number of systems and the launch vehicle selection. The increased DDT&E cost of a surface habitat (approximately \$2.0B) and TEI stage/CEV service module (approximately \$1.2B), as well as an extra \$2B in DDT&E cost for the 130 mt HLLV over a 100 mt HLLV, contribute to the total cost increase. These increases are offset by a smaller ascent stage, descent stage, and TLI/LOI stages for the LOR-LOR architectures. Finally, the EOR-Direct architectures that leave the CEV service module in LLO have a slightly higher DDT&E cost because of the additional system (ascent stage and TEI stage/CEV service module are separated). While the two systems are smaller, the development of two smaller systems is more expensive than one larger system.

All of the LOR-LOR and EOR-Direct architectures (1, 2, 6, 7, 8, and 9) have a similar flight unit RNPV. All are approximately \$1.5B more than the baseline architecture in flight unit RNPV. Two launches of the 130 mt HLLV (LOR-LOR architectures) is approximately \$5.6B, while three launches of the 100 mt HLLV (EOR-

Direct architectures) is approximately \$6.0B. The elimination of systems in the EOR-Direct architectures reduces the flight unit cost to equalize these two system architecture modes. Not coincidentally, the additional cost for the LOR-LOR and EOR-Direct architectures is primarily driven by the difference between the cost of the 29 mt crew launch vehicle and an HLLV. The launch cost for the EOR-LOR architectures is approximately \$3.9B.

More significant than the savings from any of the HLLV-based architectures is the savings realized by using commercial launch vehicles (10 and 11). The elimination of approximately \$15-21B of launch vehicle DDT&E cost by using commercial launch vehicles is clearly seen in Figure 42. Also, the flight unit cost for system architectures with commercial launch vehicles is reduced from \$4-6B to hundreds of millions of dollars. It must also be acknowledged that the actual RNPV savings of the system architectures with commercial launch vehicles and refueling is not as large as the launch vehicle savings alone. This additional cost is due to the infrastructure that must be developed for on-orbit refueling (namely, a propellant depot).

Again, changes in the other system architecture decisions, such as rendezvous location and propellant type, have an order-of-magnitude smaller impact on the overall NPV as compared to the launch vehicle selection. This is clearly shown in Figure 42 by comparing the magnitude of the RNPV difference between the propellant types for each system architecture pair. In every case, replacing LOX/CH₄ on the CEV service module and ascent stage with NTO/MMH improves the RNPV. However, the difference is extremely small for a given system architecture, and the complications associated with the toxicity of that propellant must be considered before that decision is made.

LOX/CH₄ has a higher specific impulse than NTO/MMH, but a worse IMF (due to lower bulk density and cryogenic thermal control). For the smaller systems, these impacts offset, and the gross mass for the systems is similar. Therefore, the impact on the systems lower in the hierarchy is also minimal. The exception to this is when a propellant with a lower Isp is used for a large stage. This situation occurs in system architecture number 5, where LOX/LH₂ is replaced by LOX/CH₄, resulting in an infeasible solution. Also, as previously discussed, changing the ascent/TEI stage to NTO/MMH has a more significant impact than it does in other system architectures.

4.3.2. Validation

Based on the analysis performed using the modeling framework, the ESAS baseline system architecture does not have the best RNPV of the options analyzed in the ESAS trade space. The EOR-Direct architecture with the NTO/MMH ascent/TEI stage had the lowest RNPV of the options presented in ESAS. However, the FOMs used to select the ESAS baseline architecture are presented in Figure 17 and include factors beyond affordability. These were safety & mission success, effectiveness & performance, extensibility/flexibility, and programmatic risk [7].

As presented in the ESAS report and reproduced here in Figure 43, the selected EOR-LOR baseline system architecture has the lowest probability of Loss of Crew (LOC) of the analyzed options, making it the best option of the ESAS modes with respect to the safety & mission success FOM [7]. The graphical representation of the system architecture design space enforces that the same mission objectives (surface payload, crew size, surface duration, etc.) are accomplished by each architecture. Therefore, the effectiveness & performance FOM is not a discriminator between architectures.

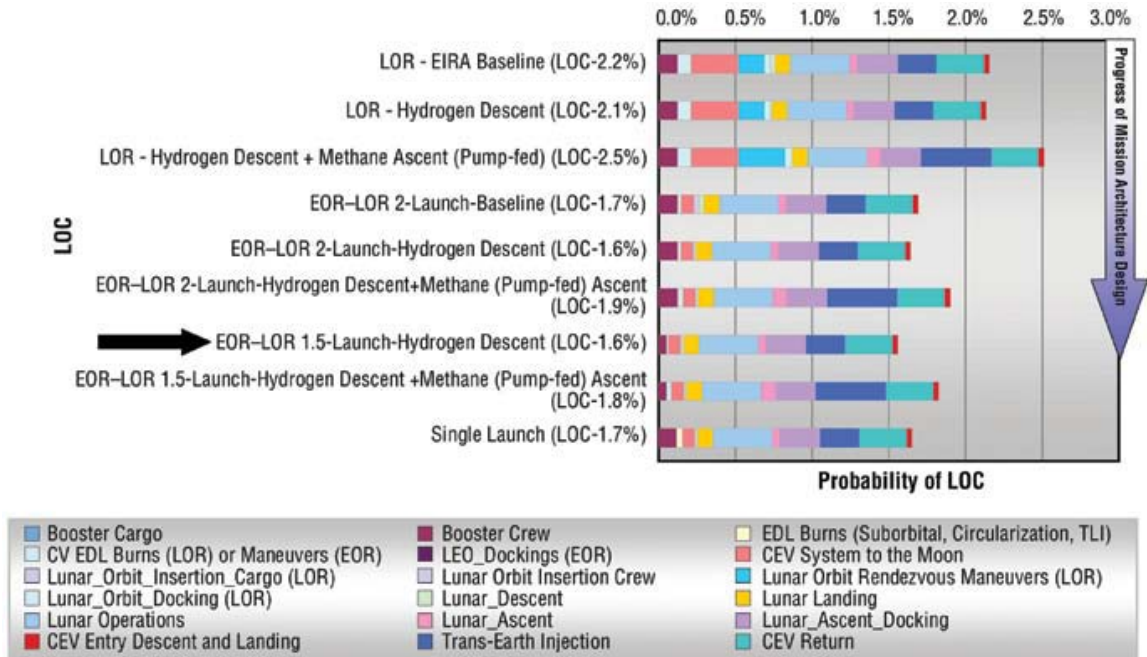


Figure 43: Loss of Crew (LOC) FOM Comparison from ESAS Mission Modes [7]

The selection of LOX/CH4 propellant usage in the ascent stage and the CEV SM relate directly to the extensibility/flexibility FOM. The use of In-Situ Resource Utilization (ISRU) at Mars commonly produces oxygen and methane for consumables and propellant. Also, developing a large launch vehicle would be useful to deliver the required payloads for a human Mars mission. The baseline architecture develops the largest of the launch vehicles.

Also, one of the requirements during ESAS was to deliver crew to the International Space Station (ISS) as quickly as possible to accommodate the retirement of the Space Shuttle. Therefore, the near-term development of a small crew launch vehicle (later renamed Ares I) met that requirement, and it improved the programmatic risk by using Shuttle-derived hardware to create an initial capability that was still useful for human exploration while the HLLV was under development [7].

Finally, the results of the analysis performed using the modeling framework estimates the baseline EOR-LOR as the highest RNPV. This is primarily driven by the

DDT&E RNPV, which is the highest of any of the system architectures analyzed. Alternatively, the baseline architecture has the lowest flight unit RNPV of the ESAS architecture options, which is beneficial for a continued campaign of lunar missions. The analysis presented in ESAS concludes that the estimated cost of all of the architectures is of a similar order of magnitude. The analysis performed with the modeling framework is consistent with this conclusion, for the system architecture that utilizes commercial launch vehicles has an order of magnitude lower RNPV than all of the system architectures that utilize HLLVs.

CHAPTER 5

FLEXIBLE PATH DESIGN SPACE EXPLORATION

This chapter uses the modeling framework to explore the system architecture design space for the three missions within the flexible path evolutionary exploration program (GEO, lunar, and NEO). The chapter provides a description of the system architecture design space graphs for each mission, and provides baseline architectures against which alternatives are compared. Then, each system architecture design space is explored using ACO, and the system architecture with the lowest RNPV for each mission is presented. Finally, from the design space exploration, implications are derived concerning launch vehicle selection, the use of propellant depots and on-orbit refueling, various aggregation strategies, and the value of using RNPV as a selection criterion.

5.1. Flexible Path Design Space

The system architecture design space consists of options such as launch vehicle selection, propellant type, staging location, and aggregation strategy. Alternatives for these various options are included in the graphical representation of the design space and presented in Table 20. The Falcon Heavy and Delta IV-H commercial launch vehicles are included with the 70 mt, 100 mt, and 130 mt HLLVs, which were the launch vehicle configurations used in the HEFT analysis [11]. The four propellant types considered are LOX/LH₂, LOX/RP-1, LOX/CH₄, and NTO/MMH. Various staging locations are included to divide the ΔV requirements among systems, and different aggregation locations are included in each architecture design space. Finally, different refueling costs were used to represent both current and potentially reduced launch costs.

Table 20: System Architecture Design Space Options with Alternatives

Option	Alternatives				
Launch Vehicles	Falcon Heavy	Delta IV-H	70 mt HLLV	100 mt HLLV	130 mt HLLV
Propellant Types	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	
Staging Locations	Suborbital	LEO	HEO	LLO	Braking Stage
Aggregation Strategy	LEO	GEO	LLO	Lunar Surface	HEO
Refuel Cost	N/A	Current (i.e. Delta)	Reduced (i.e. Falcon)		

Note: LOX/RP-1 = Liquid Oxygen/Rocket Propellant-1, HEO = High Earth Orbit

5.1.1. System Architecture Design Space Representation

Similar to the graphical representation of the system architecture design space presented in Figure 20, the GEO and NEO mission design spaces are also represented as a graph. The GEO mission assumes a 9-day stay at GEO. The lunar mission assumes a 7-day sortie at a polar location. Finally, the NEO mission assumes an easy-NEO class mission (such as 2000SG344). All architectures assume a crew of four is delivered to the destination, and the scientific or exploration merit of such a mission is not considered in the design space. The edge metadata for the system architecture design space graphs presented in Appendix A provide the required ΔV , time of flight, stay time, T/W, and other requirements for each mission. To explore the impacts of changes in these mission requirements, more analysis would be required.

The graph representation of the GEO system architecture design space is presented in Figure 44 and Table 21. The edge definition for this graph is presented in Appendix A in Table A-1. This graph enables LEO and/or GEO aggregation of systems (through the use of the link groups) with up to two flights, refueling in LEO, and can return directly or stop in LEO before reentry. Launch to LEO can be performed by either

staging suborbitally or ascending directly to LEO and the propellant types available for all burns are presented in Table 20.

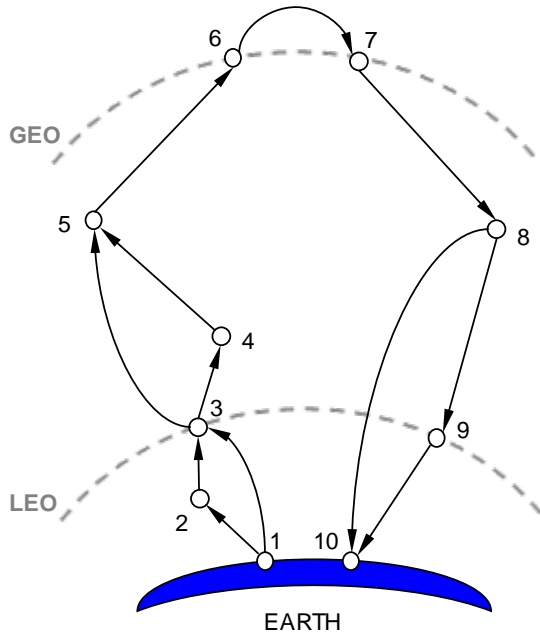


Figure 44: GEO System Architecture Design Space as a Graph

Table 21: Node Definition for GEO System Architecture Graph

Node No.	Node Name	Link Group No.
1	Earth Surface (Outbound)	
2	Suborbital Staging Point	
3	LEO (Outbound)	1
4	LEO Propellant Depot	
5	Geosynchronous Transfer (Outbound)	
6	GEO (Arrival)	2
7	GEO (Departure)	2
8	Geosynchronous Transfer (Return)	
9	LEO (Return)	1
10	Earth Surface (Return)	

The graph representation of the lunar system architecture design space is presented in Figure 20 and Table 9 in Chapter 3. The edge definition for this graph, which includes ΔV , T/W, times of flight, and other mission requirements is presented in Appendix A in Table A-2. The lunar architecture graph enables LEO, LLO, and lunar surface aggregation. LEO refueling and the option to return to LEO before reentry are also included in this graph. Finally, systems can be deployed in up to three flights.

In the NEO system architecture design space, systems can depart from LEO or High-Earth Orbit (HEO). These two options are split into two separate graphs to simplify the rule-based graph traversal algorithm and allow each case to be run simultaneously, decreasing the amount of run time needed to explore the design space. The first graph departs from HEO, while the second departs from LEO. All other options, such as

refueling, aggregation, and return options, remain consistent across the graphs. Once the analysis is performed, the results from the two design space explorations will be combined into a single set of data.

The graph representation of the NEO system architecture design space with a HEO departure is presented in Figure 45 and Table 22. The edge definition for this graph is presented in Appendix A in Table A-3. This graph enables aggregation in LEO or HEO. Because of the short departure windows and extremely long synodic periods for NEOs, aggregation at a NEO is impractical in general. For specific cases, NEO aggregation is feasible if the NEO in question has two closely-spaced departure windows. However, this cannot be incorporated into the design space until a NEO or set of NEOs is/are selected that have this property. Also, LEO refueling and the option to return to either HEO or LEO before reentry are included in the graph. Finally, systems can be deployed in up to three flights.

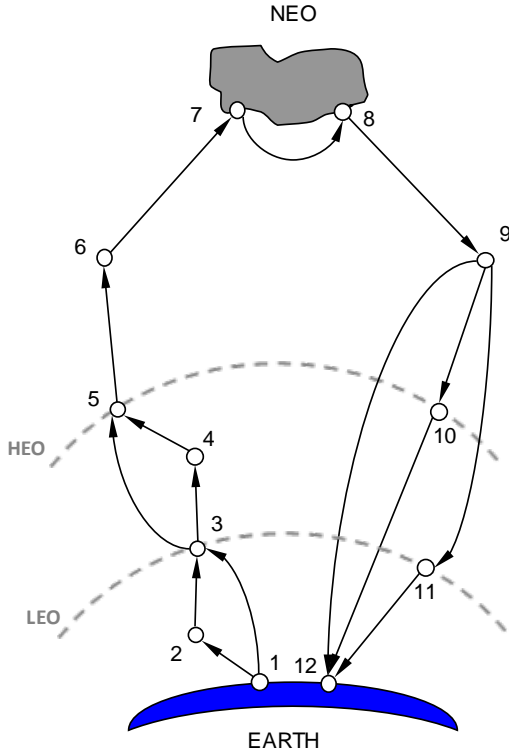


Figure 45: NEO System Architecture Design Space with HEO Departure as a Graph

Table 22: Node Definition for NEO System Architecture Graph with HEO Departure

Node No.	Node Name	Link Group No.
1	Earth Surface (Outbound)	
2	Suborbital Staging Point	
3	LEO (Outbound)	1
4	LEO Propellant Depot	
5	HEO (Outbound)	2
6	Trans-NEO Trajectory (Outbound)	
7	NEO (Arrival)	
8	NEO (Departure)	
9	Trans-NEO Trajectory (Return)	
10	HEO (Return)	2
11	LEO (Return)	1
12	Earth Surface (Return)	

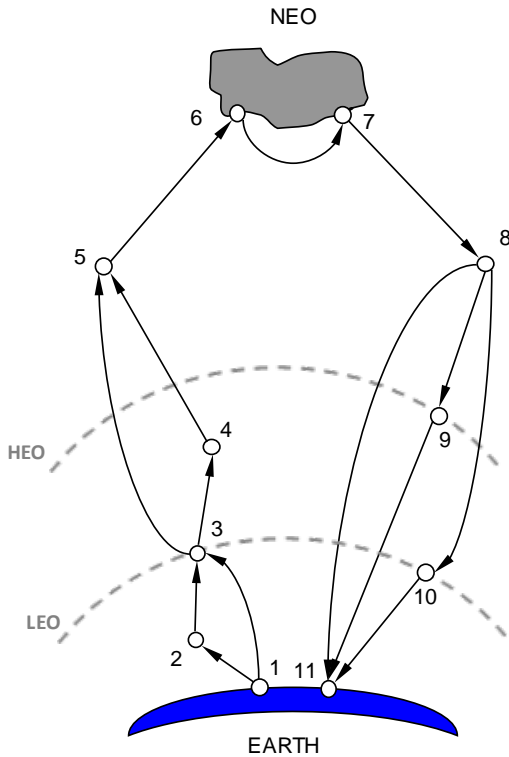


Figure 46: NEO System Architecture Design Space with LEO Departure as a Graph

Table 23: Node Definition for NEO System Architecture Graph with LEO Departure

Node No.	Node Name	Link Group No.
1	Earth Surface (Outbound)	
2	Suborbital Staging Point	
3	LEO (Outbound)	1
4	LEO Propellant Depot	
5	Trans-NEO Trajectory (Outbound)	
6	NEO (Arrival)	
7	NEO (Departure)	
8	Trans-NEO Trajectory (Return)	
9	HEO (Return)	
10	LEO (Return)	1
11	Earth Surface (Return)	

The graph representation of the NEO system architecture design space with a LEO departure is presented in Figure 46 and Table 23. The edge definition for this graph is presented in Appendix A in Table A-4. This graph enables LEO aggregation, LEO refueling, and the option to return to LEO or HEO before reentry. Again, systems can be deployed in up to three flights.

5.1.2. Baseline System Architectures

For each of the system architecture design spaces, a baseline system architecture is compared to each of the alternatives. The baseline system architectures for each of the missions are representative of architectures that utilize HLLVs and attempt to minimize the number of launches and events (for improved mission reliability).

The GEO baseline system architecture, presented in Figure 47, delivers the crew in the CEV and an EDS on a single HLLV. The EDS, which has LOX/LH2 propellant, performs the LEO departure and GEO arrival burns. The crew then performs the mission at GEO in the CEV. After the mission is complete, the CEV SM, which contains LOX/CH4 propellant, performs the GEO departure burn, and the crew returns to the Earth in the CEV. A summary of the cost and mass estimates, as calculated by the modeling framework, for this system architecture is presented in Table 24.

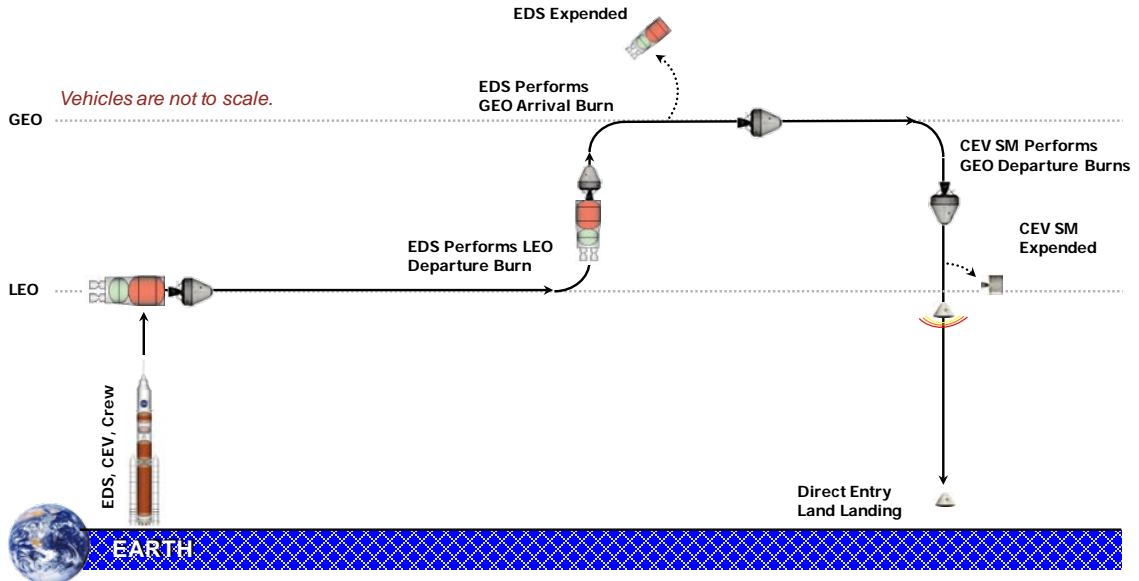


Figure 47: Baseline GEO System Architecture Concept of Operations

Table 24: Cost and Mass Estimates for Baseline GEO System Architecture

System No.	System	DDT&E Cost (FY12, \$M)	Flight Unit Cost (FY12, \$M)	Inert Mass (kg)	Gross Mass (kg)
1	Crew	--	--	--	--
2	Crew Capsule (CEV CM)	2,862.0	400.3	7,845	7,845
3	Propulsive Stage (CEV SM)	1,265.2	137.2	5,058	13,947
4	Propulsive Stage (EDS)	1,415.3	154.3	7,307	51,775
5	Launch Vehicle (100 mt HLLV)	14,731.0	1,989.6	--	--

The lunar baseline system architecture, presented in Figure 48, is the baseline architecture from ESAS. This architecture utilizes two different sized launch vehicles: one to deliver the cargo, and one to deliver the crew. The first launch delivers the EDS and LSAM to LEO using the cargo launch vehicle. The EDS, which contains LOX/LH2 propellant, also performs suborbital burning to reach LEO, where the two systems loiter until the crew arrives. The second launch delivers the crew in the CEV, which then rendezvous with the EDS and LSAM in LEO. The EDS then performs the TLI burn. The descent stage of the LSAM, which contains LOX/LH2 propellant, performs both the LOI and descent burns, while the CEV remains in LLO unmanned. After the surface mission, the crew ascends to the CEV and discards the ascent module of the LSAM,

which contains LOX/CH4 propellant. The CEV service module, which also contains LOX/CH4 propellant, then performs the TEI burn to return directly to Earth. A summary of the cost and mass estimates, as calculated by the modeling framework, for this system architecture is presented in Table 25.

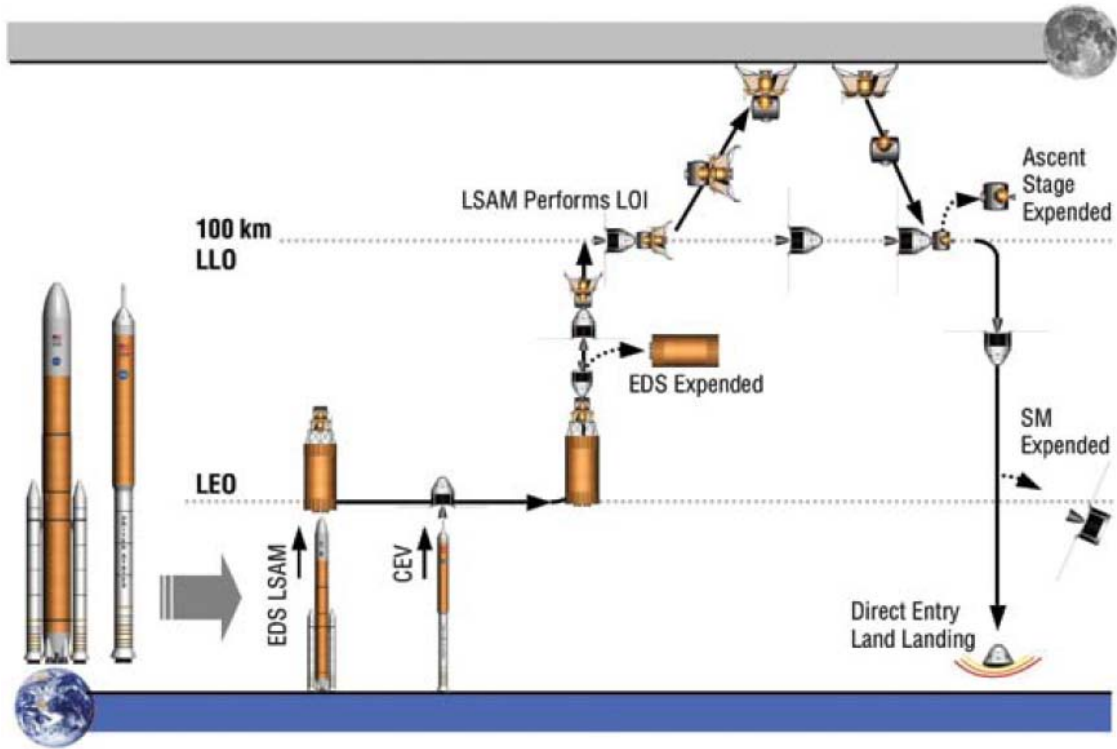


Figure 48: Baseline Lunar System Architecture Concept of Operations [7]

Table 25: Cost and Mass Estimates for Baseline Lunar System Architecture

System No.	System	DDT&E Cost (FY12, \$M)	Flight Unit Cost (FY12, \$M)	Inert Mass (kg)	Gross Mass (kg)
1	Crew	--	--	--	--
2	Crew Capsule (CEV CM)	2,862.0	400.3	7,845	7,845
3	Propulsive Stage (CEV SM)	1,058.8	113.8	3,466	8,285
4	Launch Vehicle (Crew, 29 mt)	5,502.1	892.7	--	--
5	Surface Habitat	1,997.0	395.8	4,699	4,699
6	Ascent Stage	1,800.4	282.1	1,476	6,485
7	Descent Stage	2,919.9	292.9	10,698	37,229
8	Propulsive Stage (EDS)	2,601.0	292.4	29,351	251,610
9	Launch Vehicle (150 mt HLLV, Suborbital)	15,413.9	3,031.9	--	--

Finally, the NEO baseline system architecture, presented in Figure 49, requires two HLLV launches, which rendezvous in HEO. The first launch delivers an in-space habitat, in which the crew will live during the transfers and destination mission, using a LOX/LH2 EDS to move the habitat from LEO to HEO. The second launch delivers the crew in the CEV, also using a LOX/LH2 EDS for the propulsive burns. This second EDS is also used to perform the Trans-NEO Injection (TNI) burn. At the destination, the CEV service module, which contains LOX/CH4 propellant, performs the NEO arrival burn, and after the destination mission is complete, it also performs the NEO departure burn. The in-space habitat and SM are discarded before the crew re-enters in the CEV. A summary of the cost and mass estimates, as calculated by the modeling framework, for this system architecture is presented in Table 26.

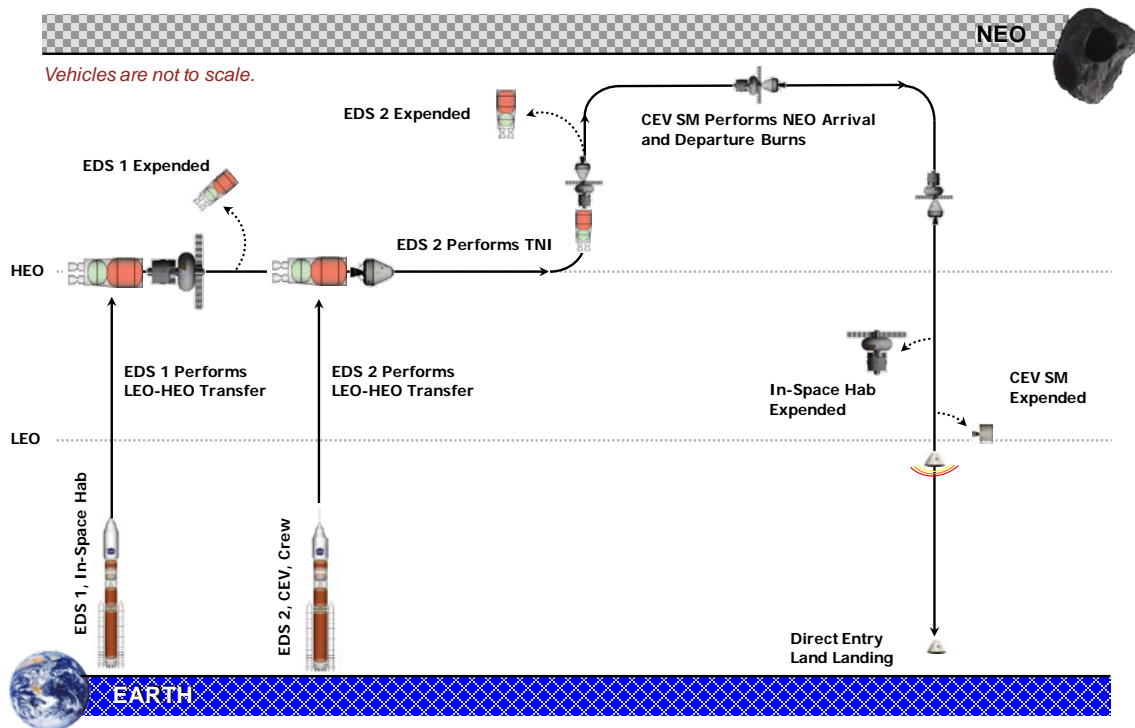


Figure 49: Baseline NEO System Architecture Concept of Operations

Table 26: Cost and Mass Estimates for Baseline NEO System Architecture

System No.	System	DDT&E Cost (FY12, \$M)	Flight Unit Cost (FY12, \$M)	Inert Mass (kg)	Gross Mass (kg)
1	Crew	--	--	--	--
2	Crew Capsule (CEV CM)	2,862.0	400.3	7,845	7,845
3	Propulsive Stage (CEV SM)	836.2	88.8	2,130	4,114
4	Propulsive Stage	1,032.9	110.9	3,655	22,829
5	In-Space Habitat	3,369.9	381.1	27,263	27,263
6	Propulsive Stage	1,233.7	133.6	5,444	36,232
7	Launch Vehicle (100 mt HLLV)	14,731.0	1,989.6	--	--
8	Launch Vehicle (100 mt HLLV)	--	1,989.6	--	--

Table 27 presents the DDT&E and the flight unit costs for each baseline system architecture as well as the NPV. The results in this table do not consider savings for systems that would have already been developed to accomplish a previous mission, such as the CEV, a launch vehicle, or a propulsive stage. It is also noteworthy that, while the cost of the NEO mission is significantly more than the cost of the GEO mission, the NPV is similar due to the time value of money. Finally, the IMLEO for each baseline is presented.

Table 27: Summary of Cost, NPV, and IMLEO for Baseline System Architectures

Baseline Mission	DDT&E Cost (FY12 \$M)	Flight Unit Cost (FY12 \$M)	NPV (FY12 \$M)	IMLEO (kg)
GEO	20,273	2,681	21,590	73,935
Lunar	34,155	5,702	35,498	191,754
NEO	24,066	5,094	22,279	98,651

5.2. Design Space Exploration Results

The system architectures that are produced and defined by a system map ensure functional feasibility. A system architecture that is functionally feasible is one in which all the functions defined within the graph are mapped to a valid system in the architecture. For instance, all propulsive burns must have a propulsive stage, and all instances of the crew must also contain a habitat (i.e. crew capsule, in-space habitat, or surface habitat).

However, not all of the functionally feasible system architectures are physically feasible. For instance, if the systems within the architecture are too massive for the launch vehicle to deliver, the system architecture as defined is physically infeasible. Also, propulsive stages have a maximum achievable mass ratio for a given inert mass fraction, which would force the system architecture to be physically infeasible. This situation would happen if a given propulsive stage was assigned to too many propulsive burns or has a very large payload.

The points shown on the design space exploration plots are the system architectures analyzed during the design space exploration that are both functionally and physically feasible. The baseline system architectures are located at the origin in each plot, and the RNPV for each system architecture is plotted. Both the DDT&E and the flight unit RNPV are presented, and the points to the lower left of the plot correspond to lower total RNPV.

5.2.1. GEO System Architecture Results

Figure 50 presents the results from the GEO system architecture design space exploration. This plot contains the results from the analysis of 353 feasible system architectures, and the modeling framework took an average of 10 seconds to analyze each system architecture on a Dell XPS 15 laptop with a 2nd Generation Intel® Core™ i7-2640M processor and 8 GB of RAM. The GEO design space exploration discovered a system architecture that improved DDT&E cost by nearly \$15 billion and flight unit cost by approximately \$2 billion. A feature of note on this plot, which also exists on the plots of each subsequent design space exploration plot, is the correlation between DDT&E RNPV and minimum flight unit RNPV. As DDT&E increases, the minimum flight unit

cost also increases, but does not do so linearly. This phenomenon comes from the linkage of the CER through inert mass as presented in Section 2.4. Both DDT&E and flight unit costs are estimated using CERs based on inert mass. The powers of the CER, however, are not equal, and therefore, the cost increases at different rates as the mass increases.

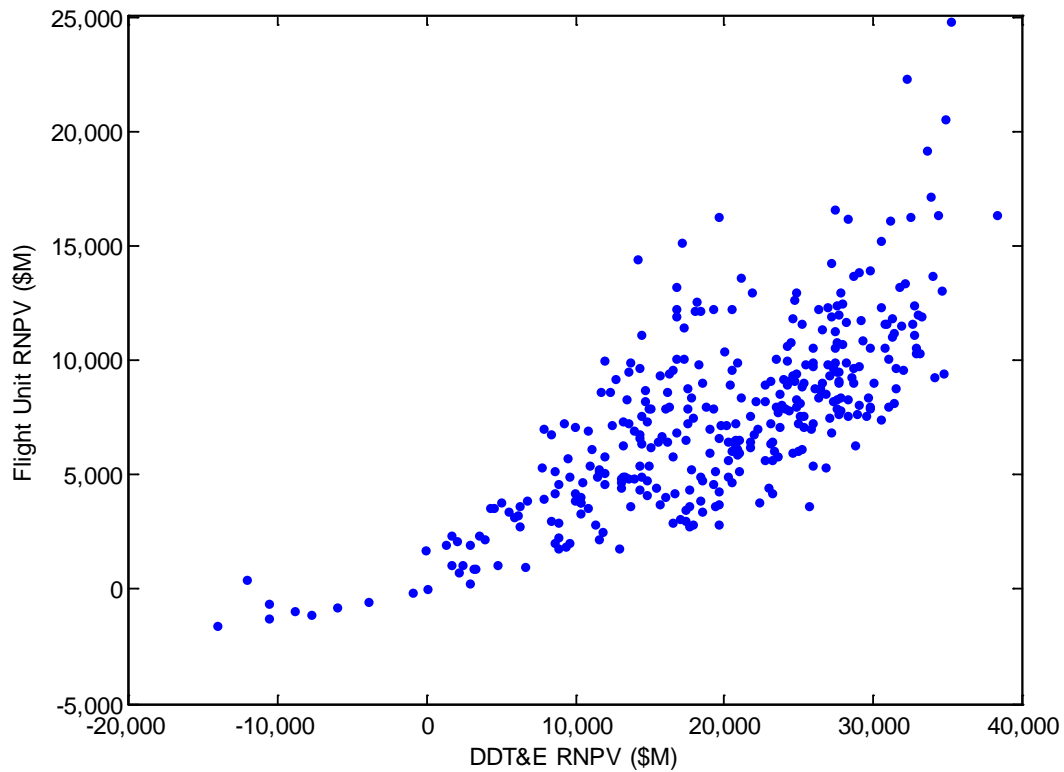


Figure 50: Results from GEO System Architecture Design Space Exploration

To identify the architectures that are attractive to the system architect, Figure 51 zooms into the region that represents an improvement in RNPV over the baseline, with each system architecture labeled using a unique identifying number. The colors of the individual points indicate the type of launch vehicle used in the system architecture. Table 28 provides a description of the main system architecture options used in each of these system architectures. The two red points identify the baseline HLLV architecture

and an alternative HLLV architecture (694) that utilizes suborbital burning, LOX/LH2, and LOX/CH4 propellants. Changing these architecture options provides an approximate \$1.25B savings in DDT&E RNPV and \$100M in flight unit RNPV. With only one HLLV system architecture identified that performs better than the baseline indicates that the baseline system architecture is nearly as good as possible while utilizing a HLLV. Changes in propellant usage and in staging location do not significantly affect the overall RNPV of the system architecture when compared to the launch vehicle cost.

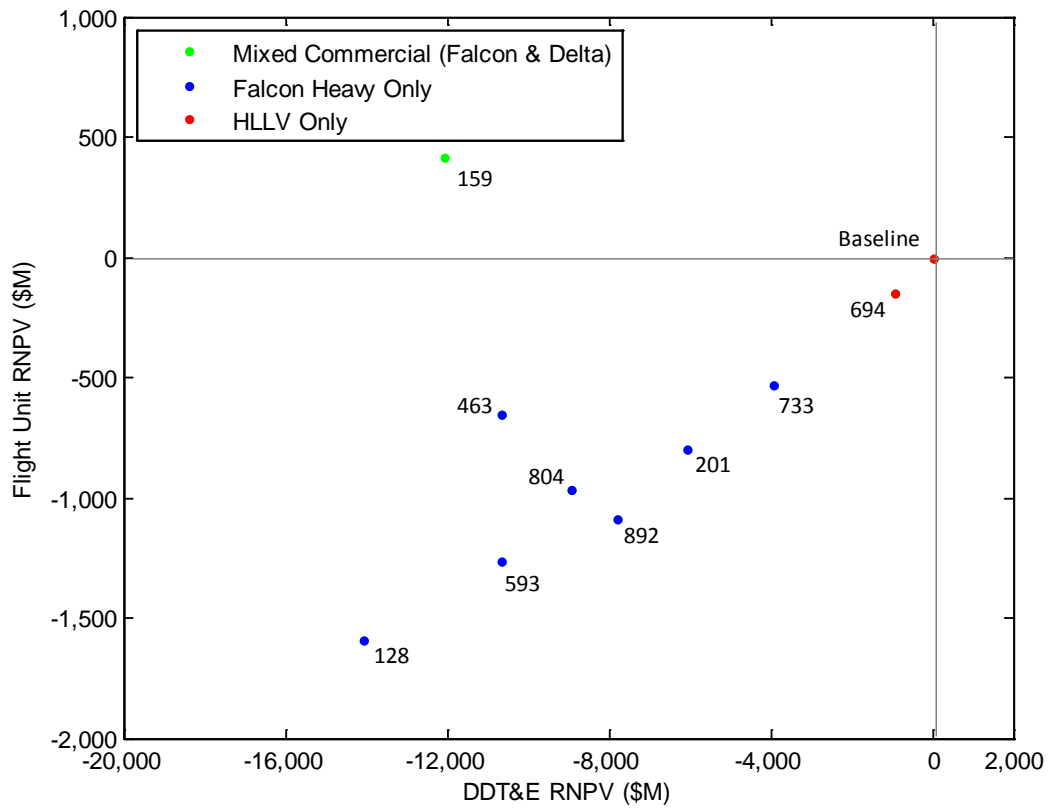


Figure 51: GEO System Architecture Design Points that Improve RNPV over the Baseline

Table 28: Description of Improved System Architectures from GEO Design Space

No.	LV Type	LEO or Suborbital	Pre-Deploy	Departure Propellant	Arrival Propellant	Return Propellant	Destination Habitation	Depot?
128	Falcon Heavy	LEO	LEO/Direct	LOX/LH2	LOX/CH4	LOX/LH2	Capsule	No
159	Falcon + Delta	Both	LEO/Direct	LOX/LH2	LOX/LH2	LOX/RP-1	Capsule	Yes
201	Falcon Heavy	Suborbital	LEO/Direct	LOX/LH2	LOX/LH2	LOX/LH2	Habitat	No
463	Falcon Heavy	LEO	LEO/Direct	LOX/RP-1	LOX/RP-1	LOX/CH4	Habitat	No
593	Falcon Heavy	Suborbital	LEO/Direct	LOX/LH2	LOX/CH4	NTO/MMH	Capsule	No
694	100 mt HLLV	Suborbital	LEO/Direct	LOX/LH2	LOX/CH4	LOX/LH2	Capsule	No
733	Falcon Heavy	Suborbital	LEO/Direct	LOX/RP-1	LOX/CH4	LOX/RP-1	Habitat	No
804	Falcon Heavy	Suborbital	LEO/Direct	NTO/MMH	LOX/RP-1	LOX/LH2	Habitat	No
892	Falcon Heavy	Suborbital	LEO/Direct	LOX/LH2	LOX/RP-1	NTO/MMH	Habitat	No

However, every other design point that has a better RNPV than the baseline utilizes a commercial launch vehicle. All but one of these commercial launch architectures uses a Falcon Heavy exclusively. The single, mixed-fleet system architecture (159) uses Falcon Heavy launch vehicles and Delta IV-H launch vehicles, which have less LEO payload capability and a higher launch cost per kilogram. Enabling the use of this smaller launch vehicle is the inclusion of a propellant depot, which enables the delivery of high capacity, empty propulsive stages that are refueled on-orbit. Also of note is that the DDT&E RNPV for this architecture is lower than all but one of the Falcon Heavy architectures, but the flight unit RNPV is significantly higher. This is due to the high cost per kilogram of delivered payload on a Delta IV-H. Using a Falcon Heavy exclusively in this architecture could reduce the flight unit RNPV significantly. Also, because the Falcon Heavy is capable of launching fully-fueled propulsive stages for the GEO mission, a propellant depot potentially increases the number of launches and

systems to be developed, therefore increasing the RNPV of the architecture relative to one without a propellant depot.

Finally, these system architectures use every option for propellant type in their architectures, indicating that it is not a significant discriminator. Also, the habitation options available for the GEO mission are either the CEV crew capsule or a dedicated in-space habitat. Both are included in these points, indicating that it is also not a significant discriminator. Alternatively, every point in this set of system architectures does not pre-deploy assets in GEO, and every point bypasses LEO and directly reenters from GEO. This would indicate that the EOR (or direct) aggregation strategy is preferred and that it is not desirable to return to LEO before reentry.

Figure 52 describes the system architecture that has the lowest RNPV as a result of the GEO design space exploration. Similar to the baseline architecture, a single EDS and CEV are launched to LEO, the LOX/LH2 EDS performs the LEO departure burn and GEO arrival burns, and the CEV service module, which contains LOX/CH4 propellant, performs the TEI burn. Unlike the baseline architecture, however, these elements are launched using two commercially-provided Falcon Heavy launch vehicles. The use of a commercial launch vehicle eliminates the DDT&E cost for the launch vehicle, which is the most expensive element to develop, and reduces the launch cost. This reduction in DDT&E and launch costs is the primary source of savings over the baseline architecture. As discovered during the ESAS mission mode analysis, the propellant type has a small relative effect on the RNPV of a given system architecture.

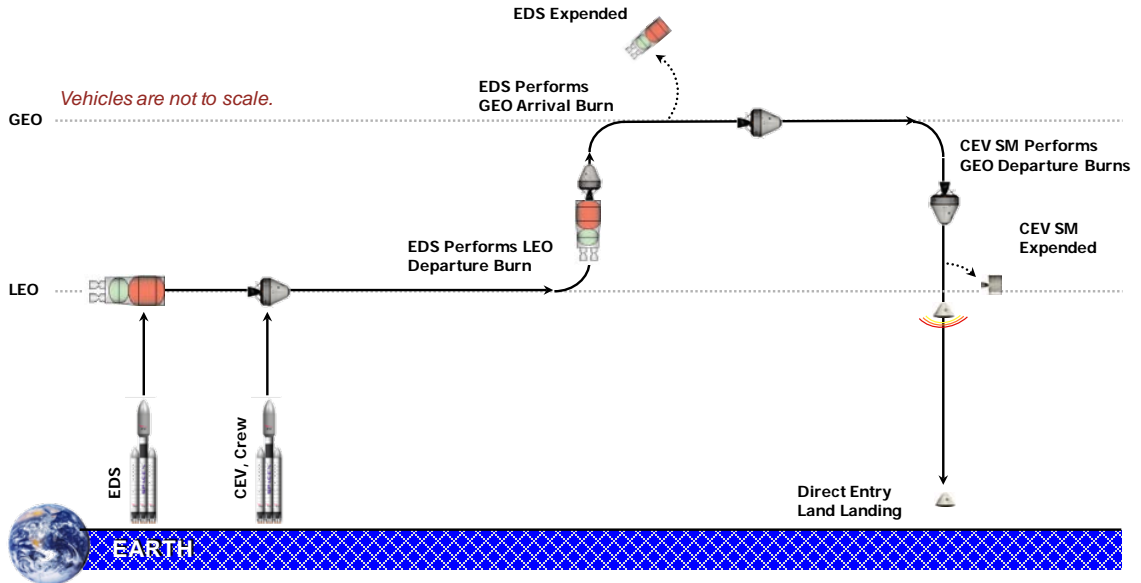


Figure 52: Best GEO System Architecture Concept of Operations

5.2.2. Lunar System Architecture Results

Figure 53 presents the results from the lunar system architecture design space exploration. This plot contains the results from the analysis of 97 feasible system architectures, and the modeling framework took an average of 16 seconds to analyze each system architecture. The lunar design space exploration discovered a system architecture that improved DDT&E cost by nearly \$20 billion and flight unit cost by over \$3 billion.

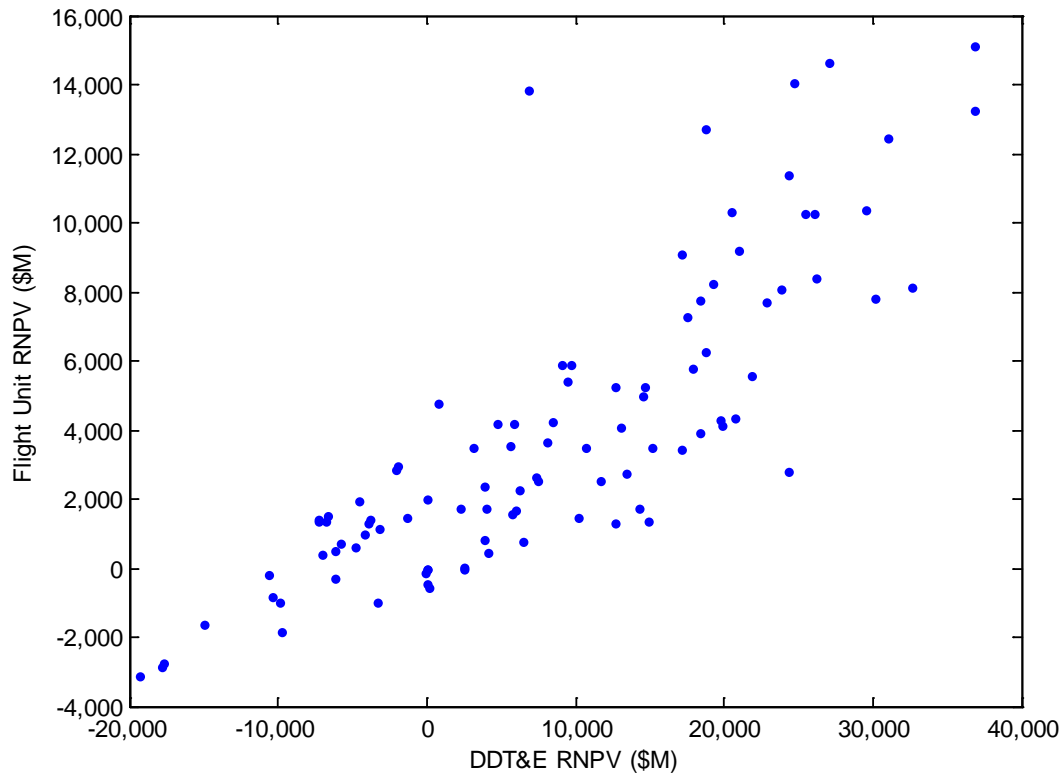


Figure 53: Results from Lunar System Architecture Design Space Exploration

To identify the architectures that are attractive to the system architect, Figure 54 zooms into the region that represents an improvement in RNPV over the baseline, with each system architecture labeled using a unique identifying number. The colors of the individual points indicate the type of launch vehicle used in the system architecture. Table 29 provides a description of the main system architecture options used in each of these system architectures.

Overall, the lunar mission is more demanding on the launch vehicles than GEO or NEO missions due to the requirement for surface access (descent and ascent functionality). Therefore, this design space shows an increase in the number of system architectures that use an HLLV and have a lower RNPV than the baseline system architecture. Aggregation in LEO is present in each architecture, and some architectures

include additional pre-deployed assets in LLO or on the lunar surface. Also, Earth return assets (propulsive stage to perform TEI, crew capsule, etc.) are left in LLO during the surface mission for all architectures that are better than the baseline, indicating that this option is desirable for feasible and affordable system architecture.

LOX/LH2 is the propellant of choice for departure (due to its high specific impulse), but LOX/CH4 and LOX/RP-1 are also feasible. No architectures use NTO/MMH for departure, which indicates that this option tends to produce physically infeasible system architectures. The use of a braking stage during lunar descent is used in only one system architecture that is improved over the baseline. The limited scope of the design space in performing braking during lunar descent makes it difficult to produce a conclusion on the use of this strategy. A more comprehensive set of ΔV splits must be examined to determine its usefulness.

Finally, this set of data points includes the use of a dedicated surface habitat and the use of the crew capsule on the surface, indicating that, again, this is not a primary driver. The use of a propellant depot in the system architectures that have lower RNPV than the baseline is more frequent in the lunar design space than the GEO design space. Due to the challenging set of requirements for a lunar mission, the ability for a propellant depot to alleviate the demand on the launch vehicle is useful for both HLLV and commercial architectures.

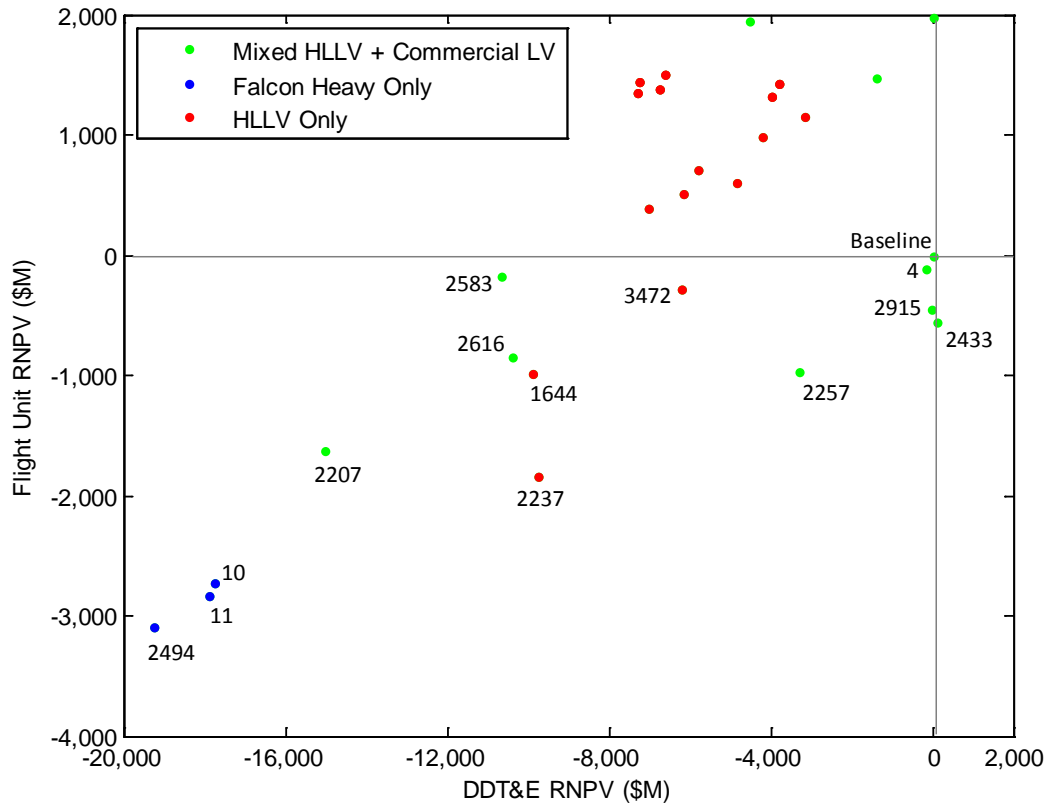


Figure 54: Lunar System Architecture Design Points that Improve RNPV over the Baseline

Table 29: Description of Improved System Architectures from Lunar Design Space

No.	Pre-Deploy	Departure Propellant	Braking?	Return Assets	Destination Habitation	Depot?
4	LEO	LOX/LH2	No	LLO	Habitat	No
10	LEO	LOX/LH2	No	LLO	Habitat	Yes
11	LEO	LOX/LH2	No	LLO	Habitat	Yes
1644	LEO	LOX/CH4	No	LLO	Capsule	No
2207	LEO + Surface	LOX/LH2	No	LLO	Habitat	Yes
2237	LEO	LOX/LH2	No	LLO	Capsule	No
2257	LEO	LOX/CH4	Yes	LLO	Habitat	No
2433	LEO + Surface	LOX/LH2	No	LLO	Habitat	Yes
2494	LLO	LOX/LH2	No	LLO	Habitat	No
2583	LEO + LLO	LOX/LH2	No	LLO	Habitat	Yes
2616	LEO	LOX/RP-1	No	LLO	Capsule	No
2915	LEO + Surface	LOX/CH4	No	LLO	Habitat	No
3472	LEO	LOX/RP-1	No	LLO	Capsule	No

Figure 55 describes the system architecture that has the lowest RNPV as a result of the lunar design space exploration. The propulsive stages of the LSAM (with a LOX/LH2 descent stage, a LOX/RP-1 ascent stage), the CEV (which contains the CM and a LOX/RP-1 SM), and the surface habitat rendezvous in LLO before performing the surface mission (LOR mission mode). The launch vehicle used in this architecture is the Falcon Heavy. The first pair of launches deploys the propulsive stages of the LSAM to LLO using an EDS that performs suborbital burning. The EDS replaces the Falcon Heavy upper stage to perform the suborbital burning. After rendezvous in LEO with the LSAM propulsive elements, this EDS performs the TLI burn, and the LSAM descent stage performs the LOI burn. These elements loiter in LLO until the crew arrives. The next flight requires two launches of the Falcon Heavy to deliver an EDS plus the CEV and surface habitat. These elements rendezvous in LEO before the EDS performs TLI and LOI. The surface habitat is transferred to the LSAM, and the crew descends to perform the surface mission. Again, using a commercial launch vehicle instead of an HLLV provides significant DDT&E and unit cost savings. This system architecture, however, introduces the complexity that dividing payloads into smaller launch vehicles can have, revealing a potential issue with reliability due to complex on-orbit operations.

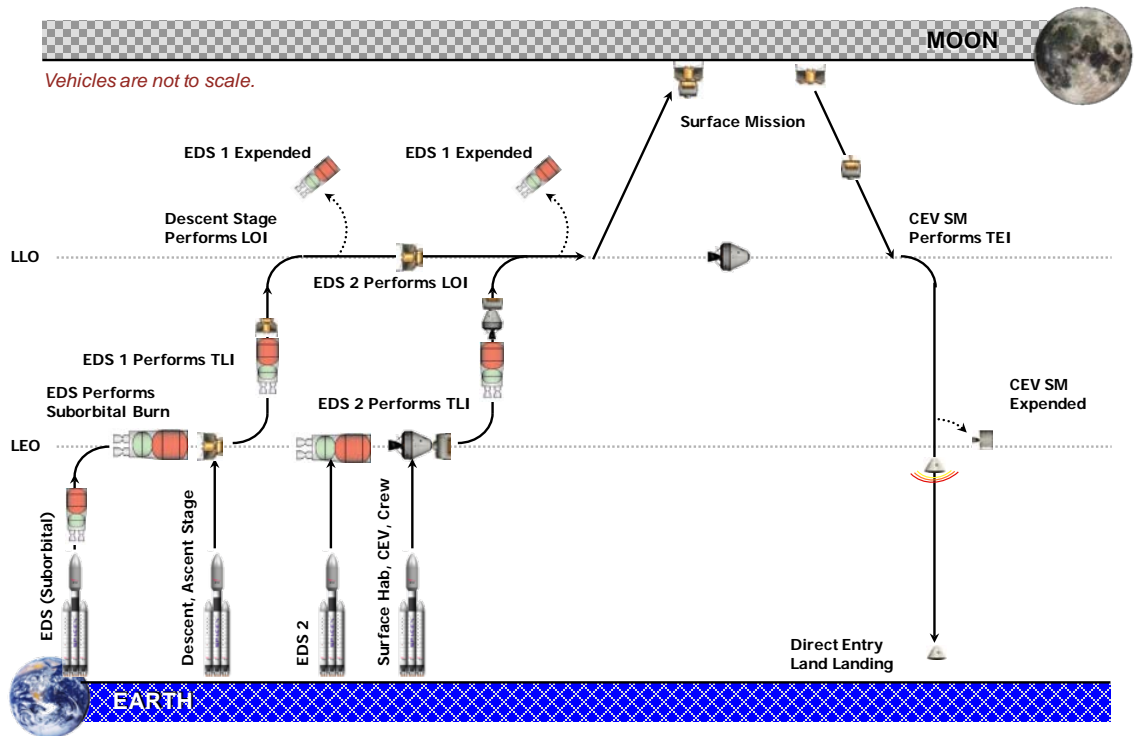


Figure 55: Best Lunar System Architecture Concept of Operations

5.2.3. NEO System Architecture Results

Finally, Figure 56 presents the results from the two (LEO aggregation and HEO aggregation) NEO system architecture design space explorations. This plot contains the results from the analysis of 1,434 feasible system architectures, and the modeling framework took an average of 14 seconds to analyze each system architecture. The NEO design space explorations discovered a system architecture that improved DDT&E cost by over \$10 billion and flight unit cost by approximately \$3 billion.

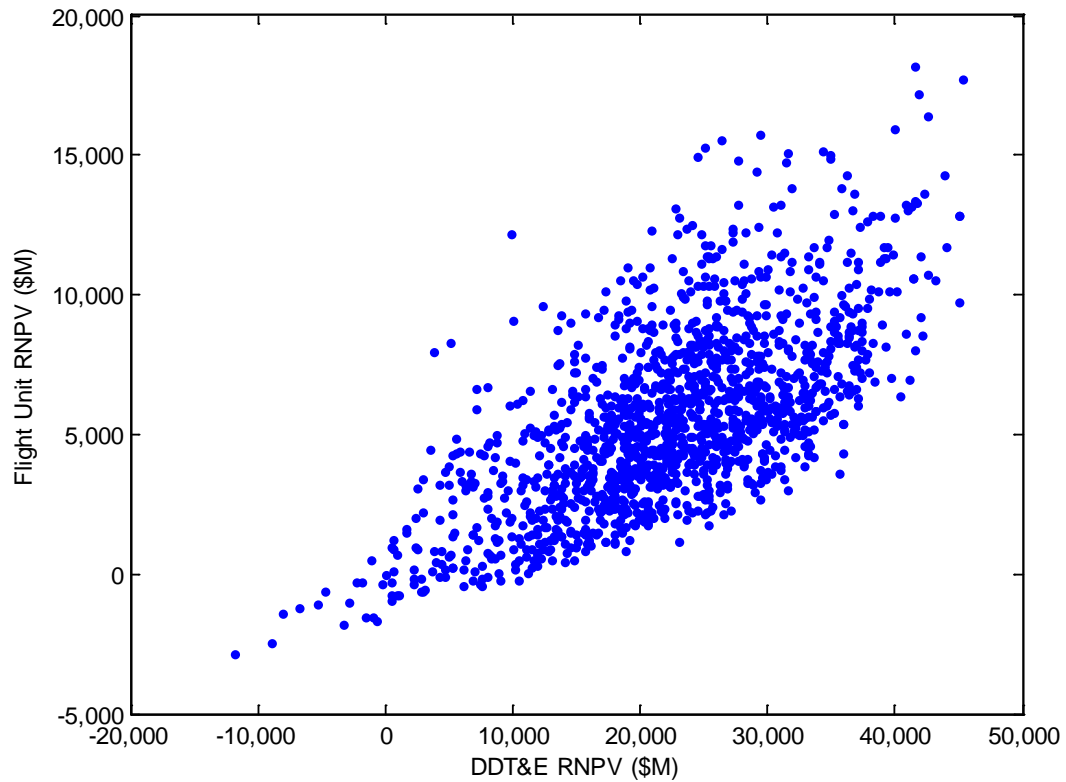


Figure 56: Results from NEO System Architecture Design Space Exploration

To identify the architectures that are attractive to the system architect, Figure 57 zooms into the region that represents an improvement in RNPV over the baseline, with each system architecture labeled using a unique identifying number. The colors of the individual points indicate the type of launch vehicle used in the system architecture. Table 30 provides a description of the main system architecture options used in each of these system architectures.

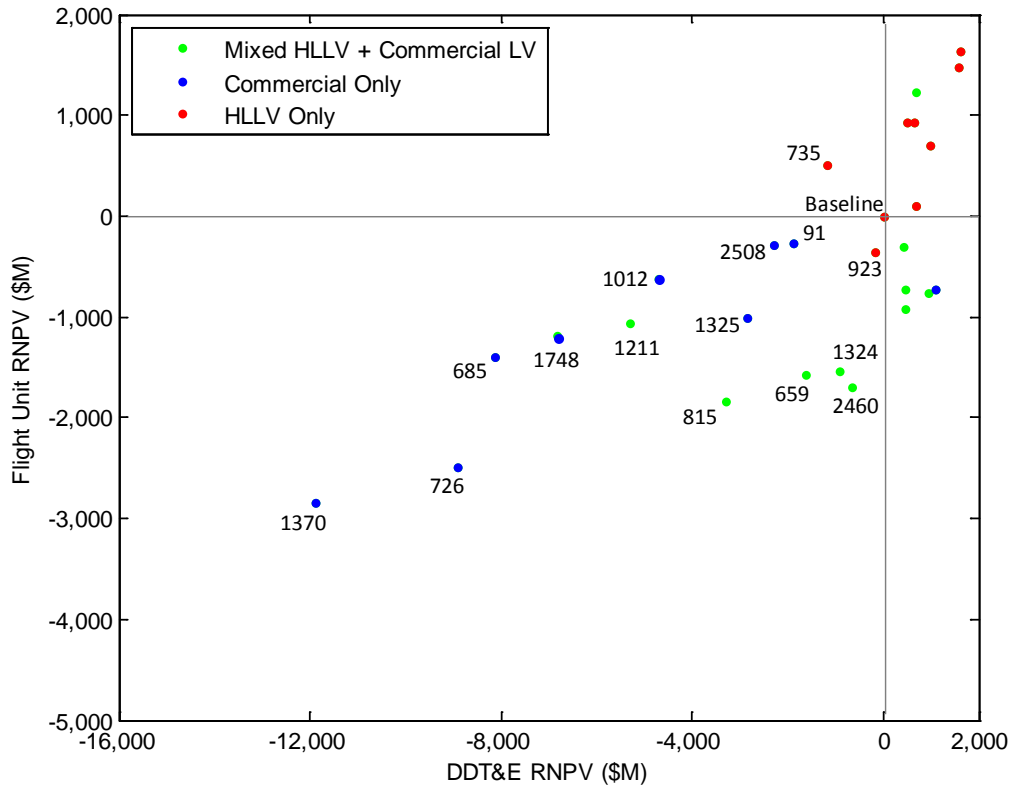


Figure 57: NEO System Architecture Design Points that Improve RNPV over the Baseline

In this architecture design space, one HLLV architecture is better than the baseline. This system architecture also uses a 100 mt HLLV, but this architecture departs from LEO instead of HEO. Within the design space, assets are pre-deployed to both LEO and HEO, independently of the orbit from which the mission departs. All propellants are represented, again indicating that this is not a significant driver in the system architecture RNPV. Alternatively, propellant selection has an impact on the physical feasibility of the system architectures. For the propulsive edges with high ΔV or with large payloads (i.e. Earth departure burns), the LOX/LH2 propellant combination is frequently selected due to its high specific impulse.

Finally, a propellant depot is used much more frequently in this design space than the previous two. The use of a propellant depot enables an all-commercial mixed fleet

architecture, where both Falcon Heavy and Delta IV-H launch vehicles are used. While this strategy increases the flight unit RNPV (due to the higher Delta IV-H cost, this strategy would promote competition, improve launch availability and reliability with redundancy, and decrease the required flight rate of a single provider). There are also many system architectures that use both HLLVs and commercial launch vehicle. These architectures have reduced flight unit RNPV due to the reduced cost to launch payload to LEO, but the DDT&E RNPV savings is less significant than the all-commercial options.

Table 30: Description of Improved System Architectures from NEO Design Space

No.	LV Type	Pre-Deploy	Departure Location	LEO Departure Propellant	HEO Departure Propellant	Arrival Propellant	Return Propellant	Depot?
91	Falcon Heavy	LEO	HEO	LOX/CH4	LOX/RP-1	NTO/MMH	LOX/CH4	Yes
659	100 mt + Falcon	LEO	LEO	NTO/MMH	--	LOX/CH4	LOX/RP-1	No
685	Falcon Heavy	HEO	HEO	LOX/RP-1	LOX/RP-1	LOX/CH4	LOX/RP-1	Yes
726	Falcon Heavy	LEO	LEO	LOX/CH4	--	LOX/RP-1	LOX/CH4	No
735	70 mt HLLV	HEO	HEO	LOX/RP-1	NTO/MMH	LOX/RP-1	LOX/LH2	No
815	100 mt + Falcon	LEO	HEO	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	Yes
923	100 mt HLLV	LEO	LEO	NTO/MMH	--	LOX/LH2	LOX/RP-1	No
1012	Falcon + Delta	HEO	HEO	LOX/RP-1	NTO/MMH	LOX/CH4	LOX/CH4	Yes
1211	130 mt + Falcon	HEO	HEO	LOX/LH2	NTO/MMH	LOX/CH4	LOX/LH2	No
1324	70 mt + Falcon	HEO	HEO	LOX/CH4	LOX/RP-1	LOX/LH2	LOX/CH4	Yes
1325	Falcon Heavy	LEO	HEO	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	Yes
1370	Falcon Heavy	LEO	LEO	LOX/LH2	--	LOX/CH4	LOX/CH4	No
1748	Falcon + Delta	LEO	HEO	LOX/CH4	LOX/RP-1	NTO/MMH	LOX/CH4	Yes
2460	70 mt + Falcon	LEO	HEO	LOX/RP-1	NTO/MMH	NTO/MMH	LOX/LH2	No
2508	Falcon Heavy	LEO	HEO	LOX/RP-1	LOX/RP-1	LOX/LH2	LOX/RP-1	Yes

Figure 58 describes the system architecture that has the lowest RNPV as a result of the NEO design space exploration. This LEO aggregation mission that utilizes Falcon Heavy launch vehicles first launches an EDS that must perform suborbital burning. The next flight delivers the in-space habitat, CEV, and crew to LEO. After rendezvous in LEO, the EDS performs the Earth departure burn. The CEV SM performs the NEO arrival and departure burns. During the one-year mission, the crew lives in the in-space habitat. Just before re-entry, the in-space habitat is expended and the crew transfers into the crew capsule.

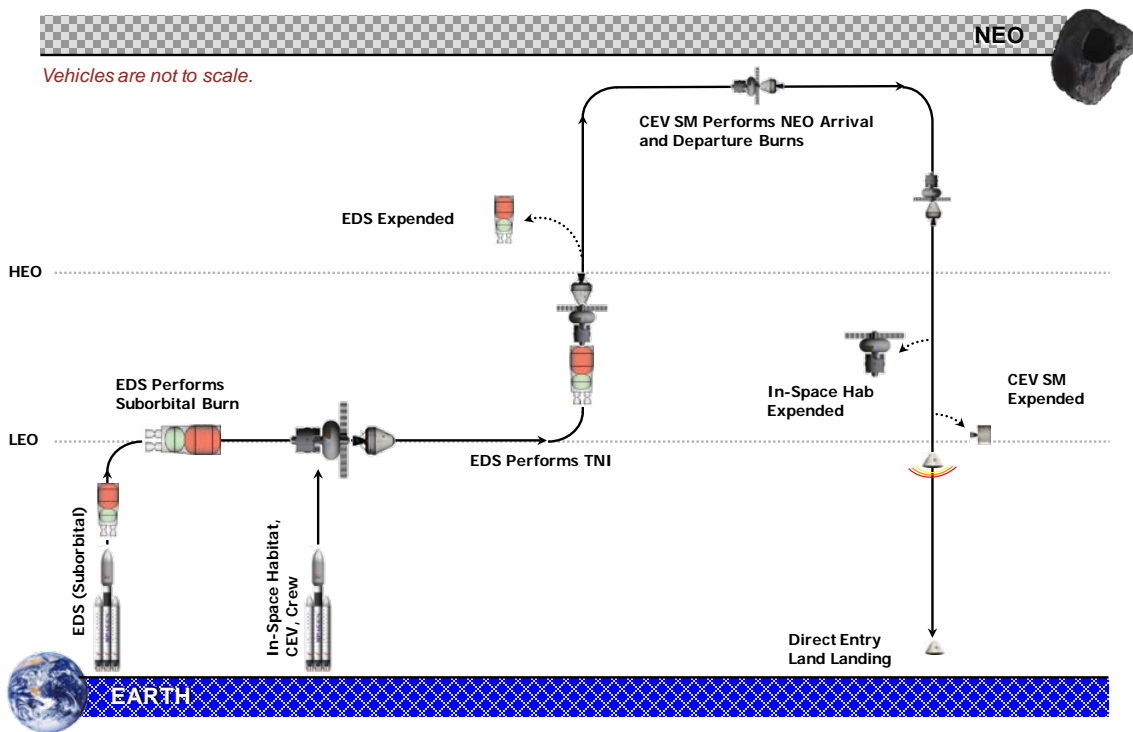


Figure 58: Best NEO System Architecture Concept of Operations

5.2.4. Evolutionary Exploration Program

When the alternative system architectures for each mission are combined in an evolutionary exploration program, certain systems can be used across multiple architectures, reducing the need for redundant development projects. For instance, if a

propulsive stage has been developed for the GEO mission that can also be used for the lunar and/or NEO mission (perhaps with offloaded propellant), then there is no DDT&E cost for that system in the subsequent missions. This is the impetus behind the flexible path option of capability and technology development to explore more challenging destinations over time.

Therefore, Figure 59 presents an evolutionary capability development that enables systems to be used across multiple missions. The set of initial capabilities, as defined by the GEO system architecture, is the Falcon Heavy launch vehicle, a LOX/LH2 EDS with a 42 mt propellant capacity (the Block 1 EDS), a crew capsule capable of accommodating a crew of four for 9 days, and a LOX/CH4 service module propulsive stage with an 8.9 mt propellant capacity. These systems are sized to perform the GEO mission, but are also capable of performing functions in the lunar and NEO system architectures. The Block 1 EDS and CEV SM are both sized by the GEO mission requirements. Because the crew performs the mission in the crew capsule, no destination-specific capabilities are required to perform the GEO mission.

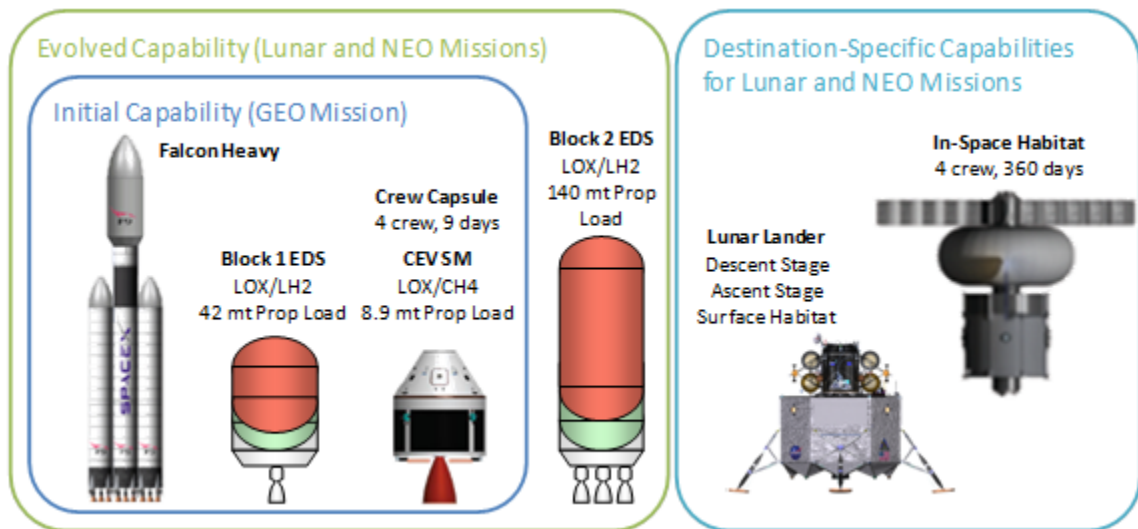


Figure 59: Evolutionary Exploration Program Capability Development

After the GEO mission capabilities have been developed, one additional transportation capability is required for the lunar and NEO missions: an evolved version of the Block 1 EDS which has a 140 mt propellant capacity (Block 2). The Block 2 EDS replaces the Falcon Heavy upper stage in the system architectures that utilize suborbital burning and in-space propulsion. When the Block 2 EDS is combined with the systems already developed for the GEO mission, only destination-specific systems must be developed for the lunar and NEO missions. For the lunar mission, a lunar lander, which consists of a descent stage, ascent stage, and surface habitat, must be developed. For the NEO mission, an in-space habitat that is capable of accommodating a crew of four for 360 days must be developed.

Evolving the system architecture in this fashion creates significant savings in the lunar and NEO missions. The DDT&E cost of the existing systems, which is included when the individual missions are analyzed independently, is eliminated when considered as part of an evolutionary exploration program. For the lunar system architecture, \$5,210M in DDT&E cost is eliminated due to the Block 1 EDS, crew capsule, and SM; and for the NEO system architecture, \$5,803M is eliminated due to the Block 2 EDS, crew capsule, and SM. Recall that the Block 1 EDS and SM were sized to perform the GEO mission. Therefore, the systems that are used in the lunar and NEO missions are oversized for the given function. Therefore, the flight unit cost is higher than it would be if a system was developed to exactly perform that function. However, this increase in flight unit cost is insignificant compared to the elimination of the DDT&E cost.

5.3. Design Space Implications

The analysis of the ESAS mission modes revealed that propellant selection has a smaller effect on the NPV of a given system architecture relative to the launch vehicle selection. The exploration of the three design spaces enables an examination of other effects on the RNPV of the system architecture, such as the use of a propellant depot and various aggregation strategies.

5.3.1. Launch Vehicle Selection

Figure 60 through Figure 64 present the results plots for the GEO system architecture design space exploration. Each plot identifies the system architectures that utilize a specific launch vehicle. These architectures may or may not use the identified launch vehicle exclusively, as some of the feasible system architectures use multiple launch vehicle types.

The Falcon Heavy launch vehicle, as shown in Figure 60, is used in all seven of the architectures with the lowest RNPV. Also, the system architectures that use the Falcon Heavy do not exceed approximately \$20B above the baseline architecture in DDT&E RNPV. Many others, which use HLLVs, extend to approximately \$39B above the baseline. Figure 61 presents architectures that use the Delta IV-H launch vehicle. While these architectures also have a relatively low RNPV, there are far fewer architectures that use a Delta IV-H than the Falcon Heavy. This is due to its smaller LEO payload capability of the Delta IV-H, which eliminates many potential system architectures because they are not physically feasible with a low payload capability.

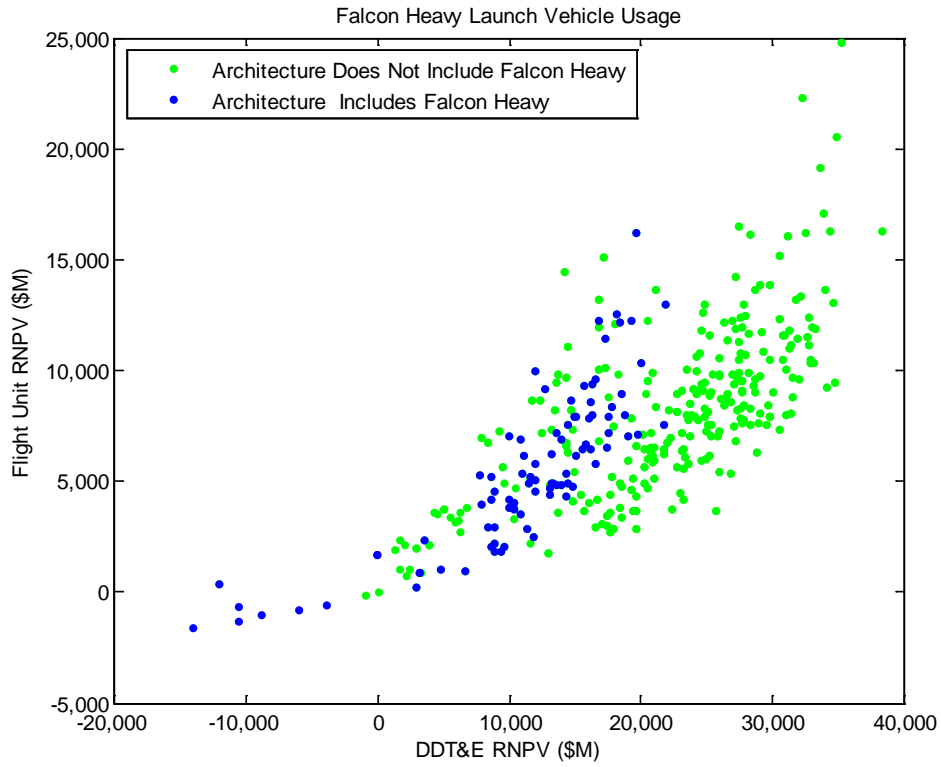


Figure 60: RNPV of GEO System Architectures that Utilize Falcon Heavy Launch Vehicles

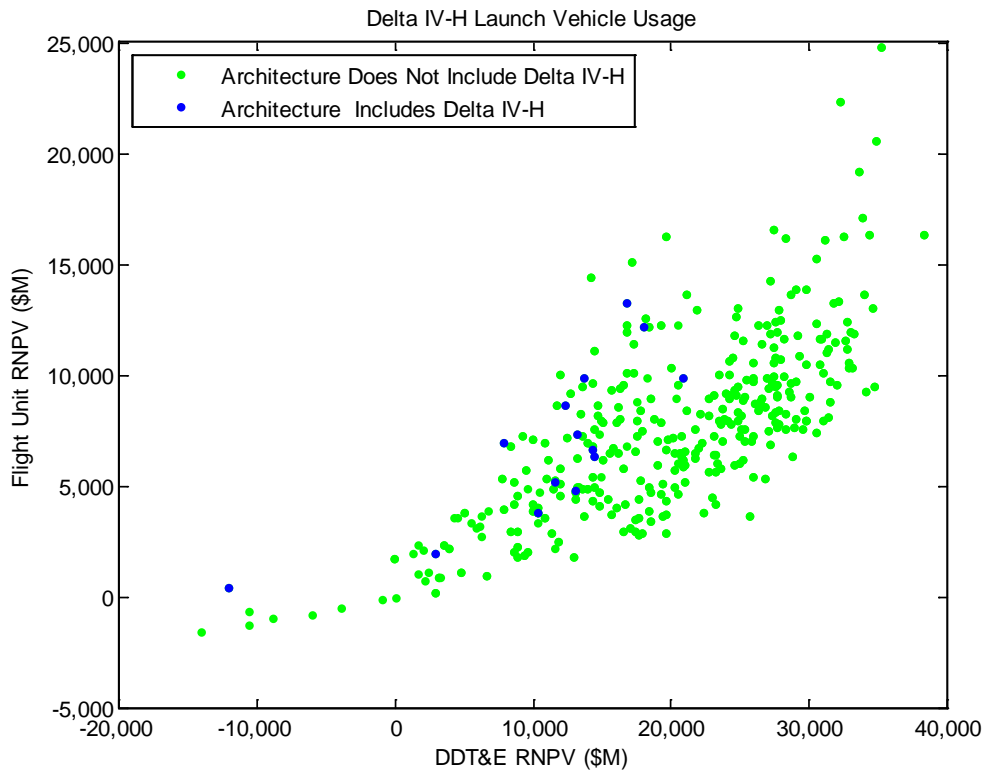


Figure 61: RNPV of GEO System Architectures that Utilize Delta IV-H Launch Vehicles

Figure 62, Figure 63, and Figure 64 present the architectures that include a 70 mt, 100 mt, and 130 mt HLLV, respectively. These architectures extend higher in DDT&E RNPV than do the commercial launch vehicles. Also, in general, there are a significantly higher number of feasible system architectures that use any of these launch vehicles than there are that use commercial launch vehicles. Because the launch vehicle LEO payload capability is a driving factor in the physical feasibility of a system architecture, the HLLVs enable more physically feasible architectures to be analyzed.

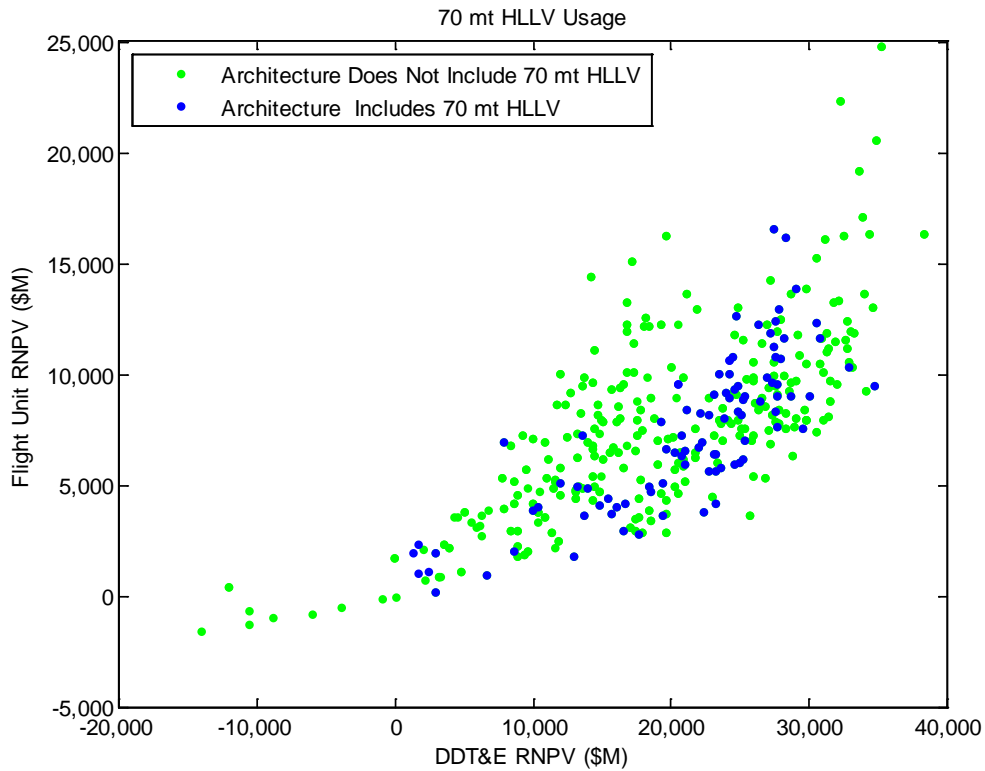


Figure 62: RNPV of GEO System Architectures that Utilize 70 mt HLLVs

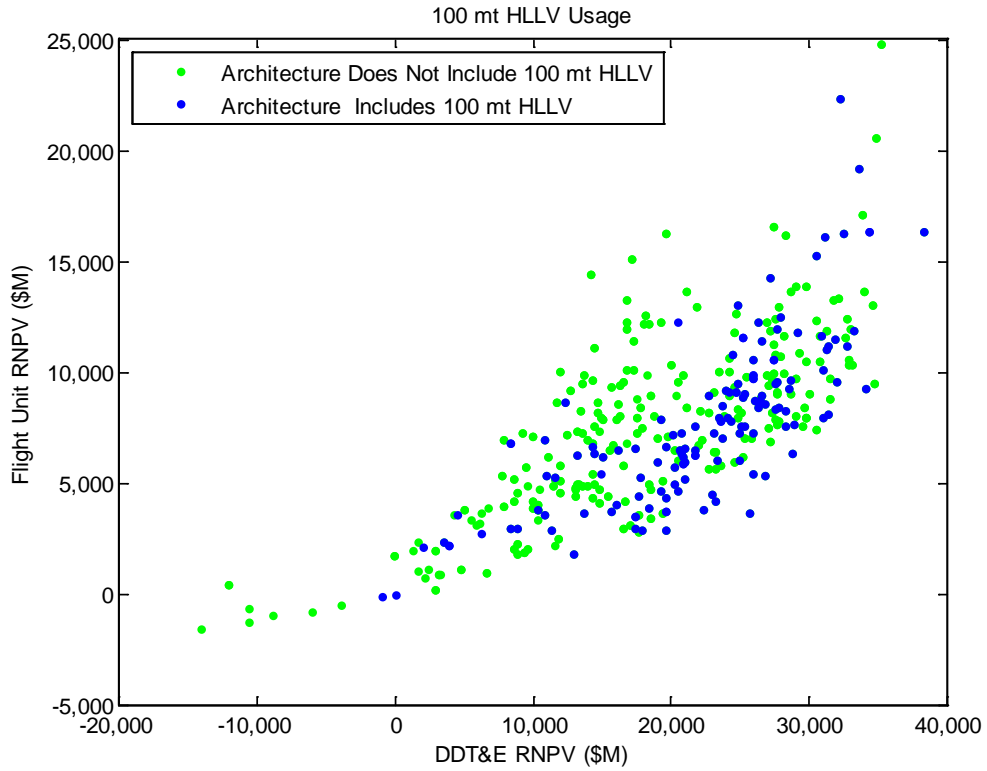


Figure 63: RNPV of GEO System Architectures that Utilize 100 mt HLLVs

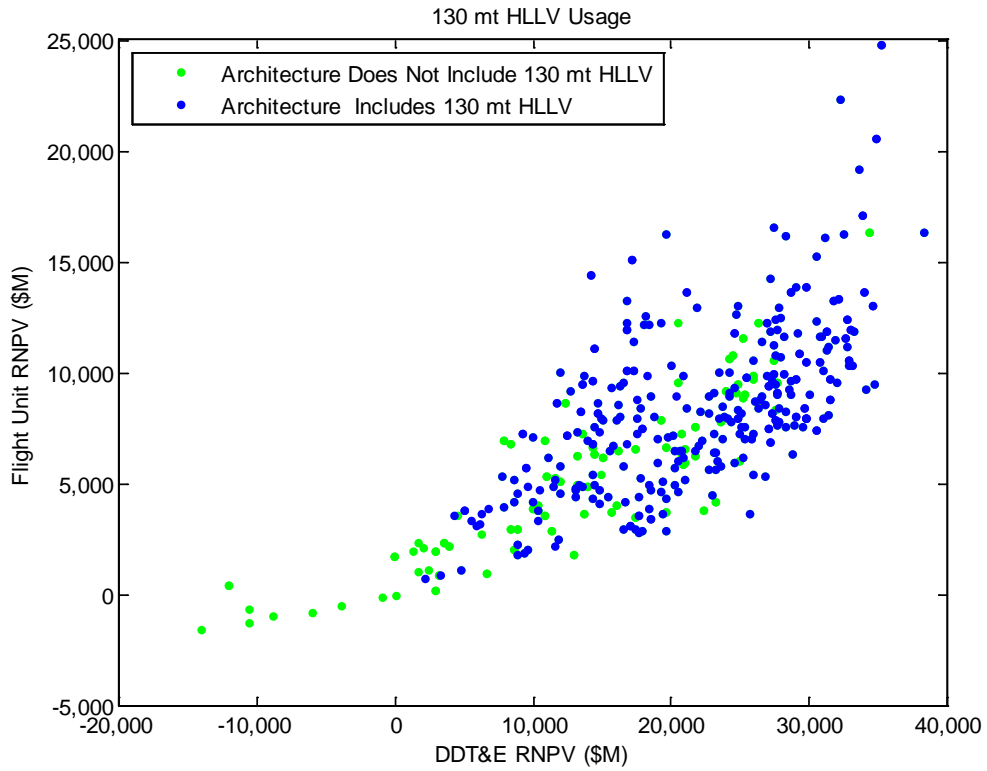


Figure 64: RNPV of GEO System Architectures that Utilize 130 mt HLLVs

Finally, Figure 65 presents a comparison between the RNPV of the use of various launch vehicles for the GEO system architecture design space exploration. A box and whisker plot is useful in displaying the distribution of large sets of data. The red line within the box represents the median of the data. The upper and lower bounds of the box represent the 75th and 25th quartiles, respectively. Finally the whiskers extend to the extremes of the data set up to a certain maximum length. The maximum length of the whisker above the box is defined as three times the difference between the 75th quartile and the median, and the maximum length of the whisker below the box is defined as three times the difference between the median and the 25th quartile. Any data points outside of this range are considered outliers and are plotted as points. The box and whisker plot presents the RNPV of system architectures that exclusively use a given launch vehicle. Box and whisker plots for the DDT&E and flight unit RNPV is presented in Appendix C. Mixed fleet architectures have been filtered out of this analysis to view the effect of a given launch vehicle alone.

Of note on Figure 65 is that there are no system architectures that use a Delta IV-H exclusively. This does not imply that it is impossible to perform a GEO mission with a Delta IV-H, but that the optimizer did not analyze any feasible system architectures that do so. This could be due to a small physically feasible design space (resulting from the low LEO payload capability of the Delta IV-H) that the ACO algorithm did not explore. The physically feasible design space of the other four launch vehicle types is significantly larger, and therefore, the ACO algorithm generated many feasible design points with which to compare.

More prominently, however, the figure shows a clear distinction between the use of the Falcon Heavy and the three HLLVs. Nearly the entire set of Falcon Heavy architectures has a lower RNPV than the entire set of HLLVs. While the HLLV architectures show gradually increasing RNPV as the LEO payload capability increases, the distinction is not as large as that of the Falcon Heavy. The conclusion that can be made from this architecture design space, therefore, is that the use of a Falcon Heavy as opposed to an HLLV significantly decreases the RNPV of GEO system architectures.

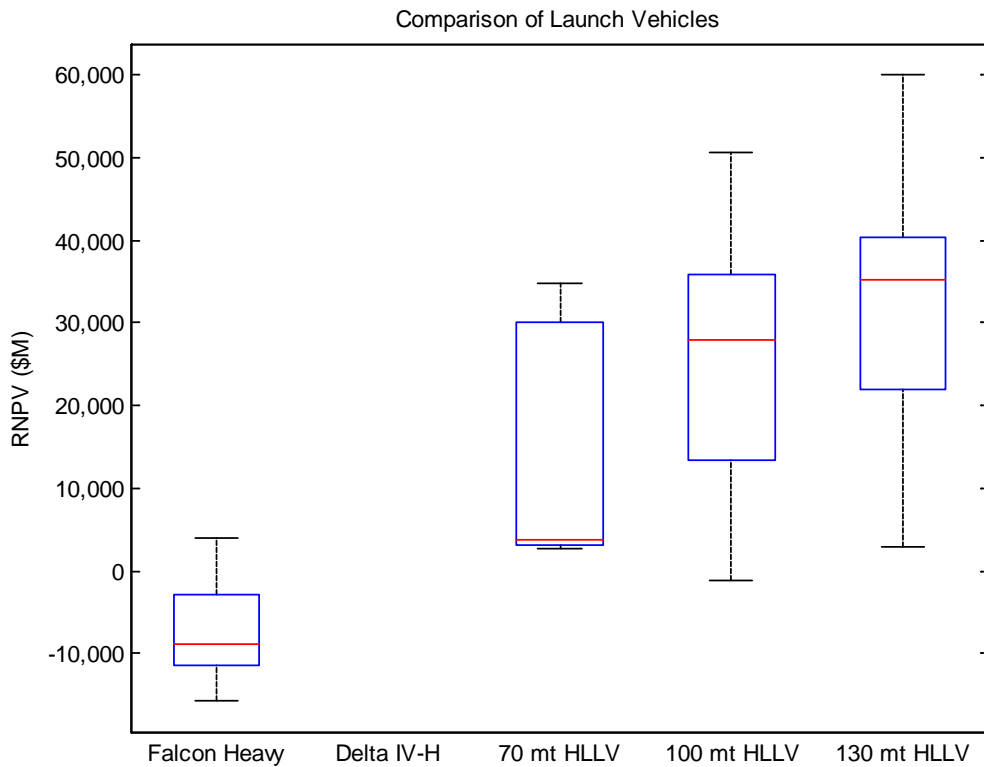


Figure 65: Box and Whisker Plot of RNPV for GEO Architectures that Exclusively Use a Given Launch Vehicle

The results plots for the lunar system architecture design space exploration that identify the usage of specific launch vehicles are located in Appendix C. For the lunar system architecture, the ACO algorithm analyzed significantly fewer design points that are both functionally and physically feasible. Therefore, clear trends are more difficult to

discern than in the GEO and NEO results plots. The Falcon Heavy launch vehicle is used in the four system architectures with the lowest RNPV.

Figure 66 presents a comparison between the RNPV of the use of the launch vehicle types for the lunar system architecture design space exploration. The box and whisker plot presents the RNPV of system architectures that use either commercial launch vehicles or HLLVs exclusively. Because the results are relatively sparse, creating a box plot of each individual launch vehicle type has very few architectures to compare. Box and whisker plots for the DDT&E and flight unit RNPV of both the individual launch vehicles and the launch vehicle categories is presented in Appendix C.

Again, the figure shows a clear distinction between the use of commercially available launch vehicles and the use of HLLVs. Regardless of other system architecture decisions that exist within the data points, the launch vehicle proves to be dividing the cost into two groups based on what type of launch vehicle is used.

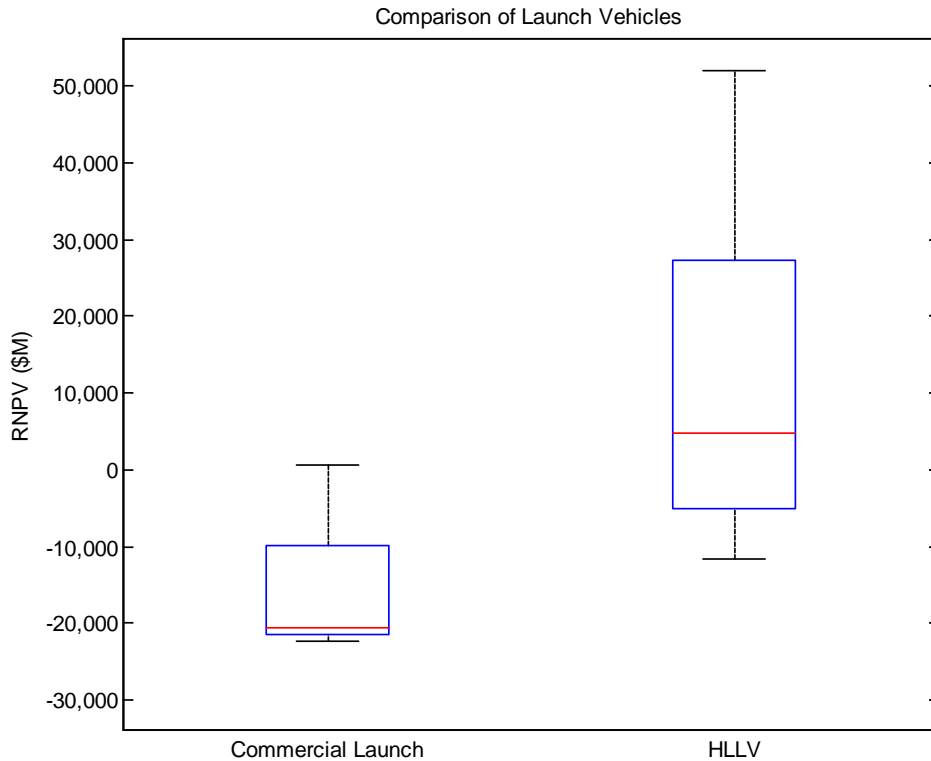


Figure 66: Box and Whisker Plot of RNPV for Lunar System Architectures by Launch Vehicle Type

Appendix C contains the results plots for the NEO system architecture design space exploration, identifying the system architectures that utilize a specific launch vehicle. In concurrence with the GEO and lunar system architecture design space exploration, the commercial launch vehicles correspond to the architecture options with lower RNPV. Also, there are significantly more system architectures that utilize HLLVs than commercial launch vehicles due to the large LEO payload capability of the HLLVs relative to the commercial launch vehicles.

Finally, Figure 67 presents a comparison between the RNPV of the use of various launch vehicles for the NEO system architecture design space exploration. The box and whisker plot presents the RNPV of system architectures that exclusively use a given launch vehicle. Box and whisker plots for the DDT&E and flight unit RNPV is presented

in Appendix C. The trends in the NEO design space concur with those of the GEO and lunar system architectures. The system architectures that exclusively use Falcon Heavy launch vehicles have a lower RNPV than the system architectures that use HLLVs.

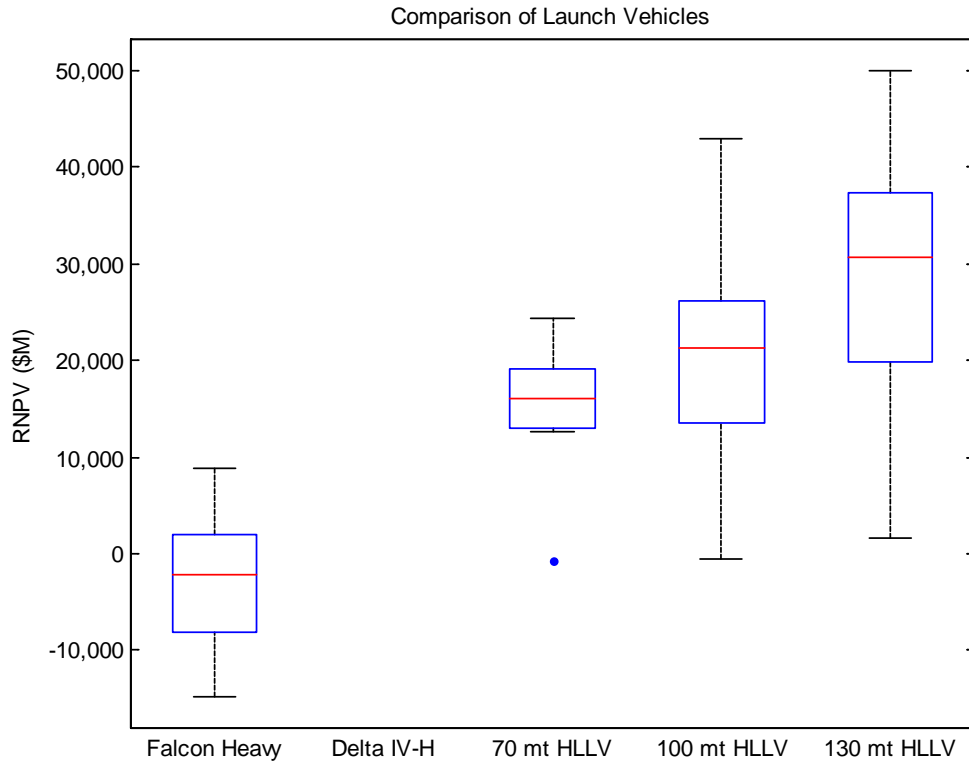


Figure 67: Box and Whisker Plot of RNPV for NEO System Architectures that Exclusively Use a Certain Launch Vehicle

The exploration of the system architecture design spaces reveals a significant reduction in RNPV by selecting commercially available launch vehicles. The time value of money used in the RNPV formulation encourages saving money in the near term. Using commercially provided launch vehicles with the performance capability to accomplish the mission and delaying the development of an HLLV until it is required are preferred to reduce RNPV. The nearly eliminated DDT&E cost and reduced flight unit cost for the commercial launch vehicles reduces RNPV dramatically compared to HLLVs

that NASA must develop and operate. The sufficient performance and low predicted flight unit cost of the Falcon Heavy lead to architectures that utilize this launch vehicle.

However, the detriment to using commercial launch vehicles with lower LEO payload capabilities is the increased number of mission-critical flight hardware launches. When all of the commercial launches must be successful to achieve mission success, adding more launches increases the probability of loss of mission dramatically. Solutions to this issue include the use of HLLVs to reduce the number of launches or the use of a propellant depot to reduce the number of mission-critical flight hardware launches. The risk can be further mitigated in the latter scenario by utilizing redundant commercial launch providers to deliver propellant to the depot. The use of a HLLV in a system architecture significantly increases the RNPV over the best system architectures that utilize commercial launch vehicles. The use of propellant depots, therefore, should be considered to determine the effect of utilizing on-orbit refueling on the RNPV of the system architecture.

5.3.2. Propellant Depots and On-Orbit Refueling

Figure 68 presents the results from the GEO system architecture design space exploration, where system architectures that utilize a propellant depot are identified. There are 228 design points that utilize propellant depots and 126 design points that do not. The increased number of design points can be related to the ability for the architecture decision to overcome system mass growth and inefficient architecture design. Therefore, the inclusion of a propellant depot enables the system architectures to overcome these issues. Delivering the propellant separately enables smaller launch vehicles to deliver empty propulsive stages (which typically weight 10-20 percent of the

gross weight of the stage). Only one of the system architecture options that are better than the baseline architecture utilizes a propellant depot. This option uses both Falcon Heavy and Delta IV-H launch vehicles, while the other options that are better than the baseline use a Falcon Heavy launch vehicle exclusively (without a propellant depot).

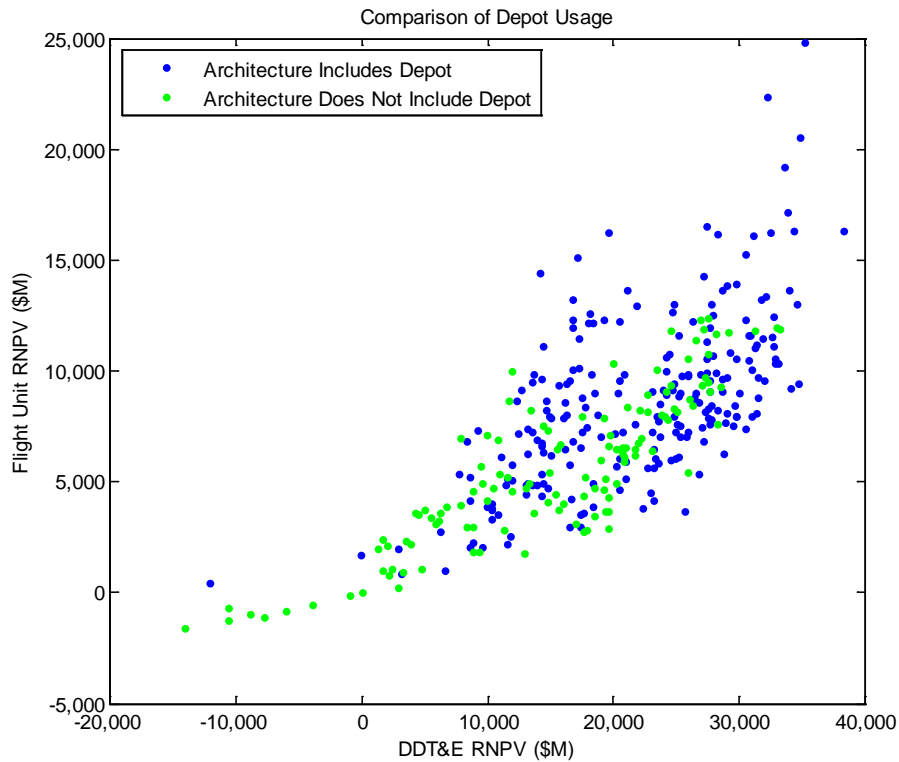


Figure 68: RNPV of GEO System Architectures that Use Propellant Depots

Figure 69 presents a box and whisker plot that compares the results from the GEO design space exploration that include propellant depots and those that do not. The design space does not reveal an obvious difference between the two options like the launch vehicle selection comparison. The RNPV values for the two sets of data overlap, and at the lower RNPV values, which are most interesting to the system architect, there is little difference between the two options. Although the RNPV for the lowest options that utilizes a propellant depot is higher than the lowest option without a depot, the increase is not significantly high enough to eliminate it from consideration.

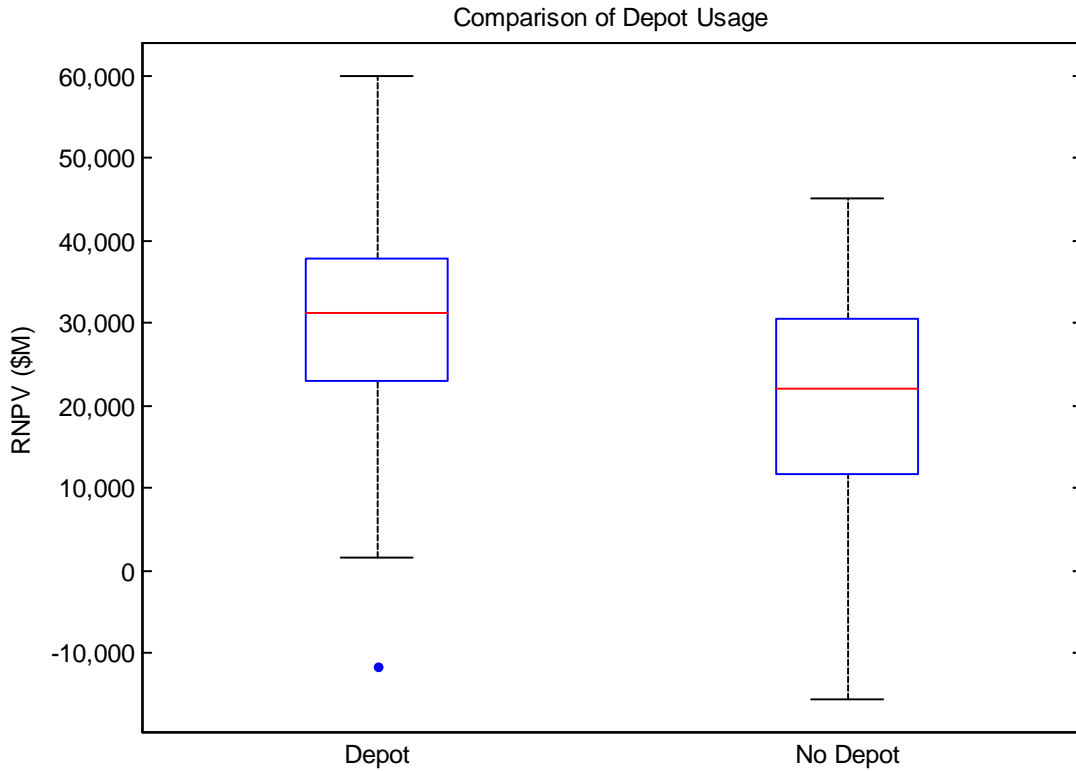


Figure 69: Box and Whisker Plot of RNPV for GEO System Architectures that Use a Propellant Depot

Figure 70 presents the results from the lunar system architecture design space exploration, where system architectures that utilize a propellant depot are identified. There are 57 design points that utilize propellant depots and 40 design points that do not. The option that has the minimum RNPV does not utilize a propellant depot. However, the best option that utilizes a propellant depot is only a slight increase in RNPV relative to the savings that both system architecture options provide over the baseline. Both of these options use the Falcon Heavy launch vehicle exclusively.

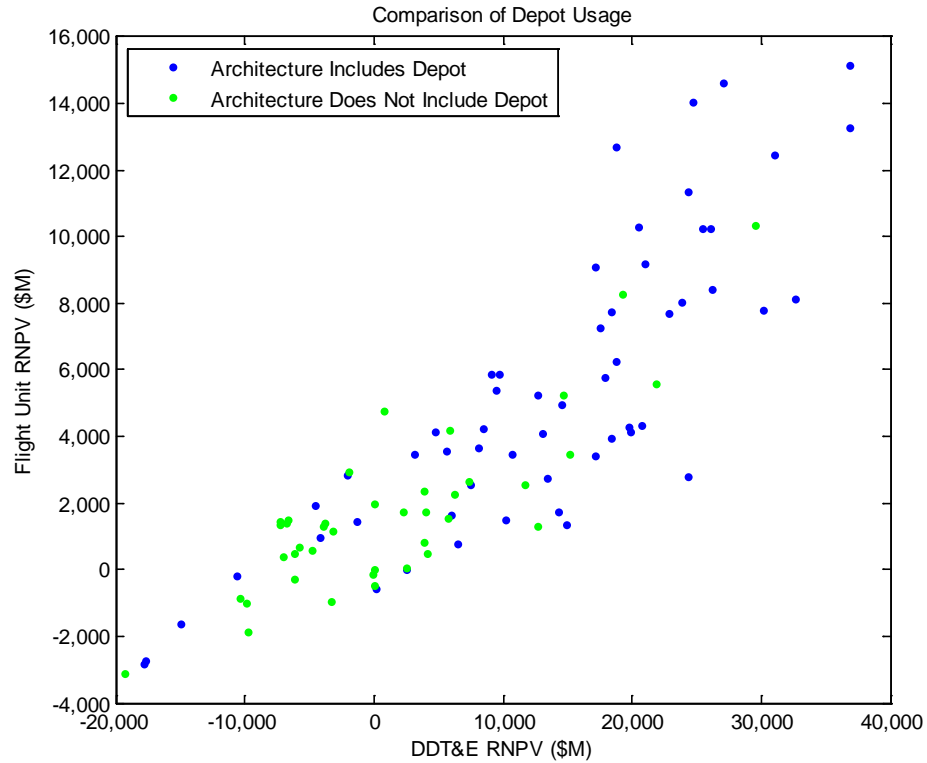


Figure 70: RNPV of Lunar System Architectures that Use Propellant Depots

Figure 71 presents a box and whisker plot that compares the results from the lunar design space exploration that include propellant depots and those that do not. Again, the design space does not reveal a significant difference between the two options. At the lower RNPV values, there is little difference between the two options, which again indicates that the decision to include a propellant depot should not be eliminated from consideration.

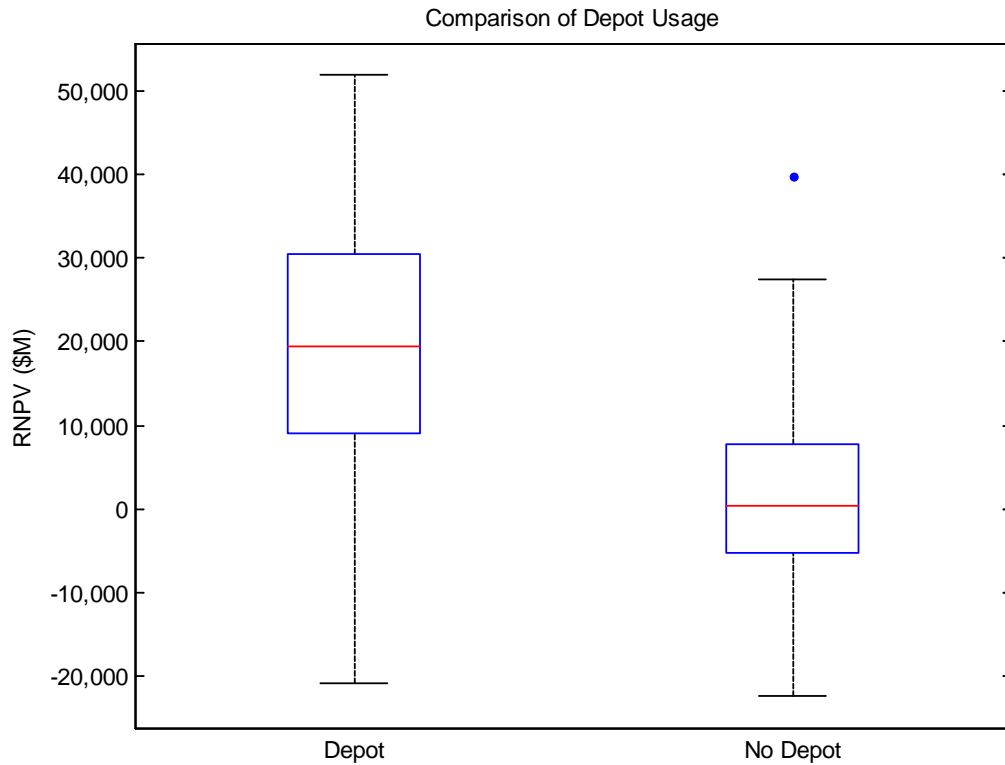


Figure 71: Box and Whisker Plot of RNPV for Lunar System Architectures that Use a Propellant Depot

Figure 72 presents the results from the NEO system architecture design space exploration, where system architectures that utilize a propellant depot are identified. There are 1,235 design points that utilize propellant depots and 200 design points that do not. Again, this significant difference in the number of feasible design points is due to the ability for system architectures that include propellant depots to overcome system mass growth and architecture inefficiencies.

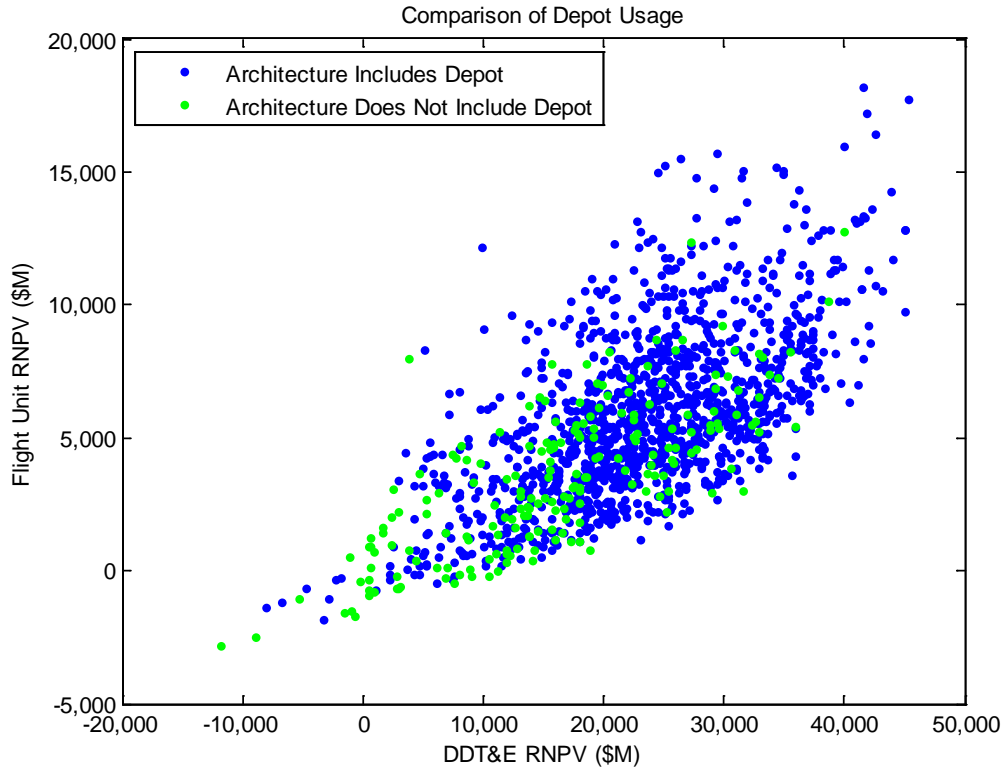


Figure 72: RNPV of NEO System Architectures that Use Propellant Depots

Figure 73 presents a box and whisker plot that compares the results from the NEO design space exploration that include propellant depots and those that do not. The results from this design space concur with the lunar and NEO design spaces. The RNPV values for the two sets of data overlap, and there is little difference between the two options at the lower RNPV values.

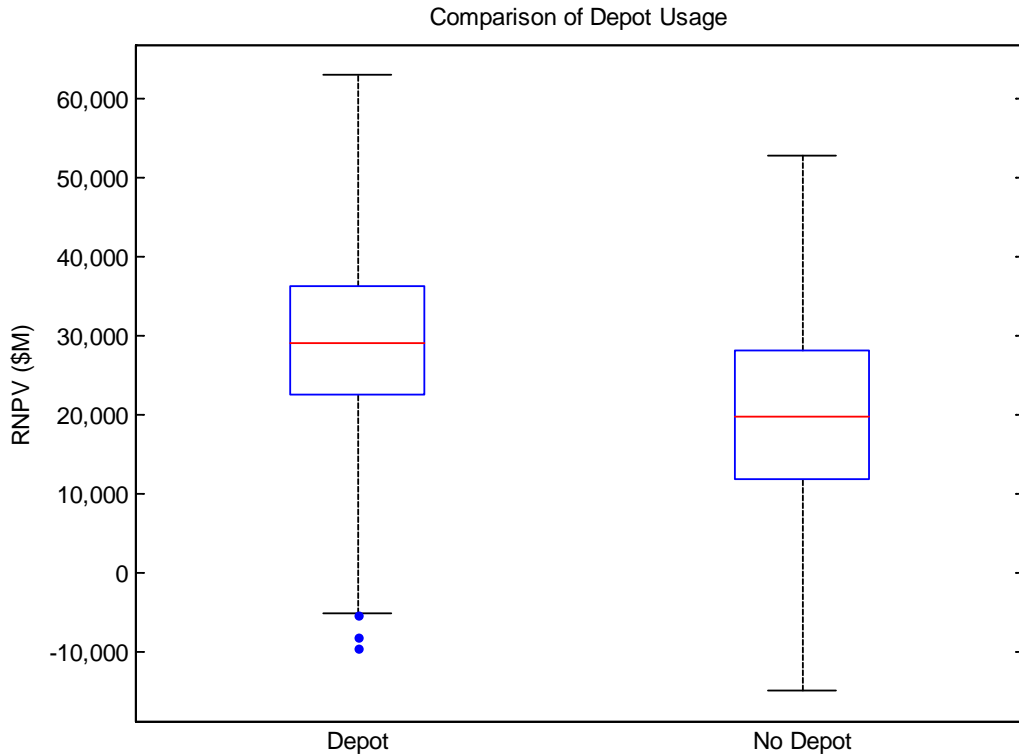


Figure 73: Box and Whisker Plot of RNPV for Lunar System Architectures that Use a Propellant Depot

Overall, the system architectures that include a propellant depot have similar RNPV to the system architectures that do not. The best system architectures that include a propellant depot have higher RNPV than the best system architectures that do not include propellant depots. However, the difference is not significant enough to exclude the use of propellant depots as a means to reduce the number of mission-critical flight hardware launches.

Also, there are a larger number of physically feasible system architectures that include a propellant depot than those that do not. This reveals the robustness of system architectures that include depots, which have the ability to mitigate system map growth and inefficient architectures. Finally, the use of propellant depots with commercial launch vehicles provides a commercial market for payload delivery to LEO. The

increased demand will promote competition between launch providers, reducing launch cost, and the increased flight rate will improve launch vehicle reliability over time.

One option that is not considered in this analysis is the use of on-orbit refueling without propellant depots, where the propellant delivery flights fuel the propulsive elements directly without going through a propellant depot. While this reduces the cost of the system architecture by eliminating the DDT&E and flight unit cost of a propellant depot, it adds the operational complexity and increased risk of numerous dockings with flight hardware, multiple launches in the critical path (which can be mitigated using redundant launch vehicle providers), and long loiter durations in LEO before the crewed mission begins.

5.3.3. Aggregation Strategy

The ability to pre-deploy assets to locations such as LEO, GEO, HEO, LLO, and the lunar surface is a key functionality included in the graph theory architecture modeling framework. This enables the delivery of payloads to a destination in smaller increments, enables the more efficient division of system functionality (i.e. leave TEI propulsive stage in LLO during the surface mission), and enables the full utilization of the launch vehicle capability by placing smaller payloads into higher energy orbits for later use.

Within the GEO system architecture design space, assets can be pre-deployed in LEO, pre-deployed in GEO, or go directly to GEO without any rendezvous. Using only the system map and system list, one cannot discern between an Earth Orbit Rendezvous (EOR) strategy, where assets are pre-deployed to LEO, and one where assets directly travel to GEO on a single launch. Because a single Earth launch edge can contain

multiple launches (if the payload must be divided into multiple launches), each design point must be analyzed individually to determine if it uses EOR or a single launch.

Figure 74 presents a comparison of the RNPV of EOR/Direct system architectures with Geosynchronous Orbit Rendezvous (GOR) system architectures, and Figure 75 presents this information in a box and whisker plot. Box and whisker plots for the DDT&E and flight unit RNPV are located in Appendix C. The best system architectures with respect to RNPV use an EOR/Direct strategy.

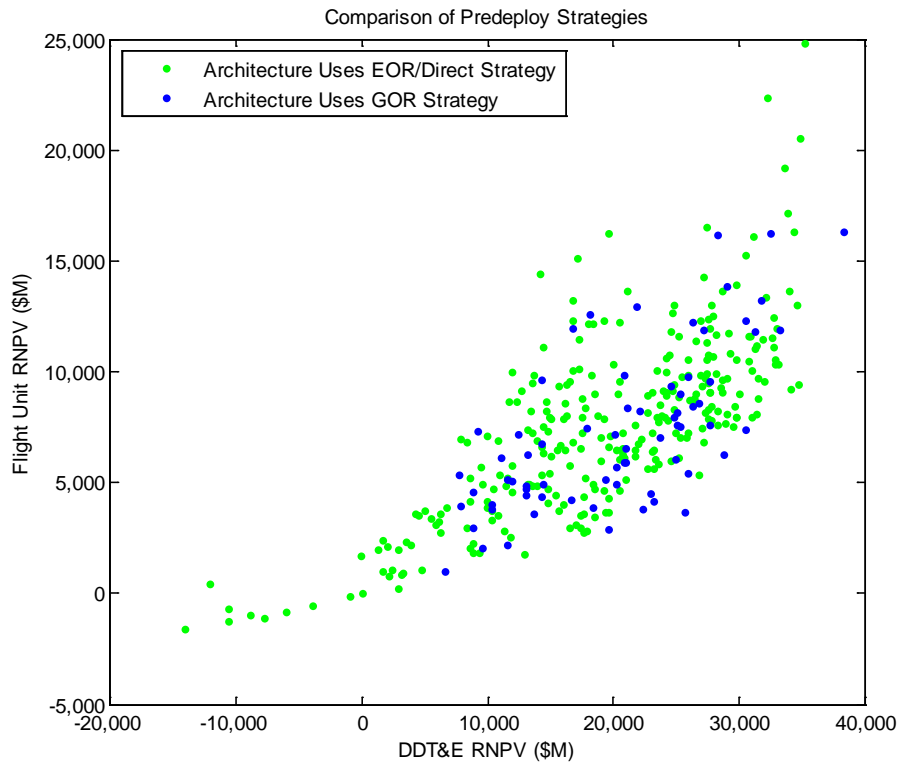


Figure 74: RNPV of GEO System Architectures for Different Pre-Deploy Strategies

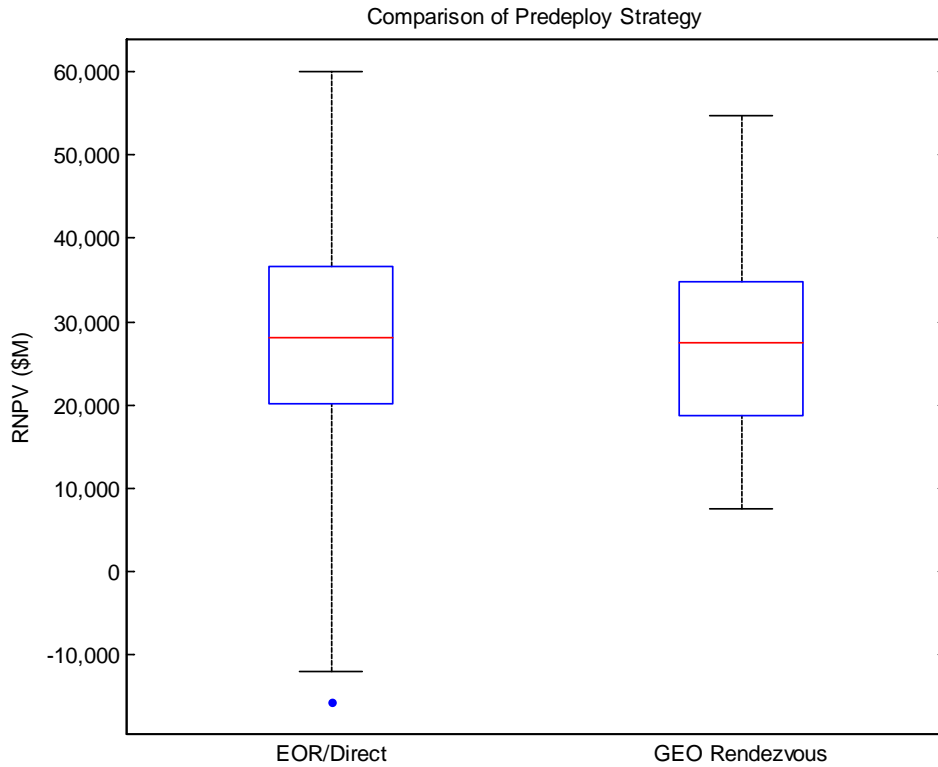


Figure 75: Box and Whisker Plot of GEO System Architectures for Different Pre-Deploy Strategies

Within the lunar system architecture design space, assets can be pre-deployed in LEO, pre-deployed in LLO, or pre-deployed to the lunar surface. This provides a significantly more interesting design space than either the GEO or NEO system architecture design spaces. Figure 76 presents a box and whisker plot of the RNPV for multiple pre-deploy strategies in a lunar mission. The plots for DDT&E and flight unit RNPV separately are located in Appendix C. The EOR/Direct, just as in the GEO mission delivers all assets to LEO before placing all assets at once on the trans-lunar trajectory. The LOR places assets in LLO, while Surface Rendezvous places assets on the lunar surface. This plot also compares combinations of pre-deploy strategies, such as EOR and LOR, EOR and surface rendezvous, LOR and surface rendezvous, and rendezvous at all three.

The system architecture design space exploration did not analyze any system architectures that utilized surface rendezvous exclusively or that utilized LOR and surface rendezvous. Again, this does not imply that system architectures that use these combinations of pre-deploy strategies are infeasible, but that the ACO algorithm did not analyze any physically feasible system architectures that used this strategy. The trends show that the EOR and LOR only system architectures provide a benefit with respect to RNPV over other pre-deploy strategies. The EOR, when combined with either LOR or surface rendezvous is also feasible, and can provide improvement in RNPV over the baseline, but not as significant as the EOR and LOR pre-deploy strategies. Finally, using all three pre-deploy strategies do not tend to provide improvement in RNPV over the baseline potentially due to the significant complexity of on-orbit operations, the number of systems required, and increased number of launches to deploy assets to these locations.

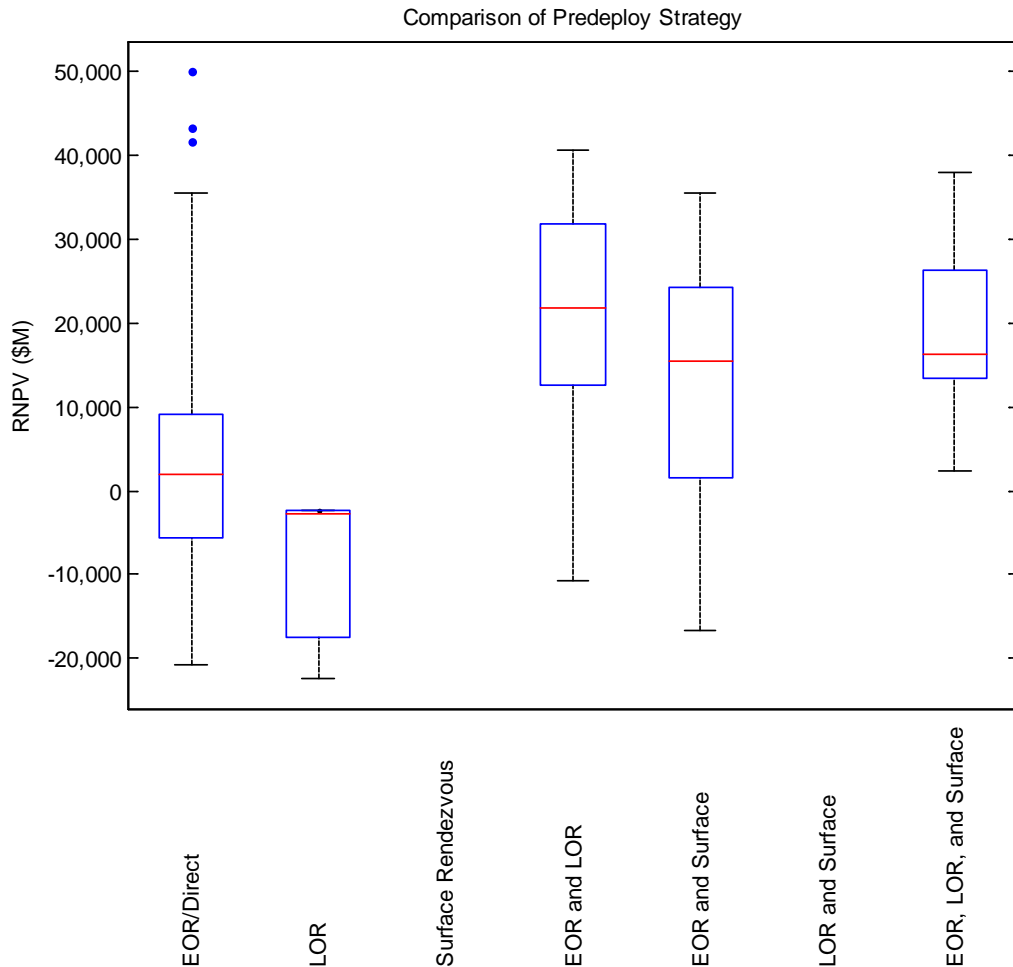


Figure 76: Box and Whisker Plot for Lunar System Architecture for Different Pre-Deploy Strategies

Also of interest is the ability to leave assets in LLO during the surface mission similar to the Apollo system architecture, where the Command Module and Service Module remained in LLO while the lunar lander performed the surface mission. Figure 77 presents a box and whisker plot for RNPV of system architectures that either take all systems to the surface, or leave some of the systems in LLO during the surface mission. Again, additional plots for DDT&E and flight unit RNPV can be found in Appendix C. The figure shows a clear advantage to leaving some assets in LLO during the surface mission. Of note is that there are far more physically feasible system architectures that

leave assets in LLO during the surface mission, indicating that this strategy significantly reduces the sensitivities to mass growth and system architecture inefficiency. Due to the impact that taking elements to and from the lunar surface has on the mass of the rest of the systems within the architecture, inefficient system architecting that takes all systems to the lunar surface can result in physically infeasible architectures. The baseline lunar system architecture uses EOR and leaves assets in LLO during the surface mission, while the best system architecture from the design space exploration uses LOR and leaves assets in LLO during the surface mission.

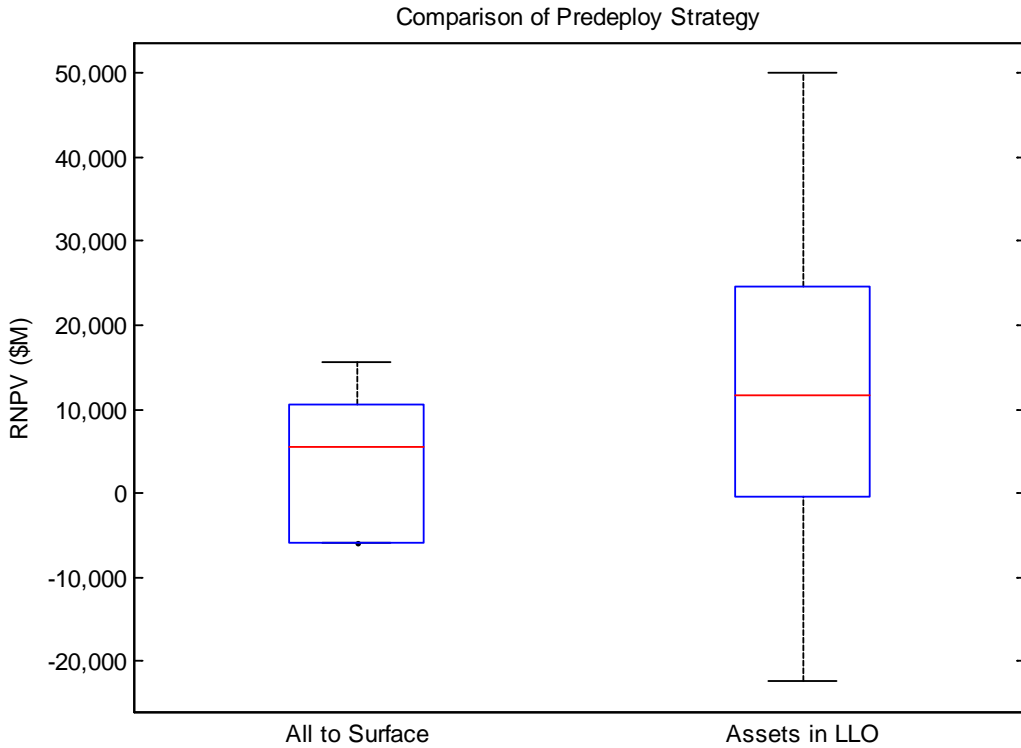


Figure 77: Box and Whisker Plot of Lunar System Architectures for Location of Assets during a Surface Mission

Within the NEO system architecture design space, assets can be pre-deployed in LEO or HEO. Again, due to the long synodic period and relatively small departure windows for NEOs in general, pre-deployed at the NEO is not included in the NEO

system architecture design space. Figure 78 presents a box and whisker plot of RNPV for the two pre-deploy strategies, and plots of the components of RNPV are located in Appendix C. The figure does not present a conclusive argument for either strategy, but the LEO rendezvous strategy is used in the system architecture with the lowest RNPV. A system architecture can still have significant cost savings over the baseline (which uses HEO rendezvous) using both strategies, with launch vehicle selection being the driving factor in the reduction of RNPV over the baseline system architecture.

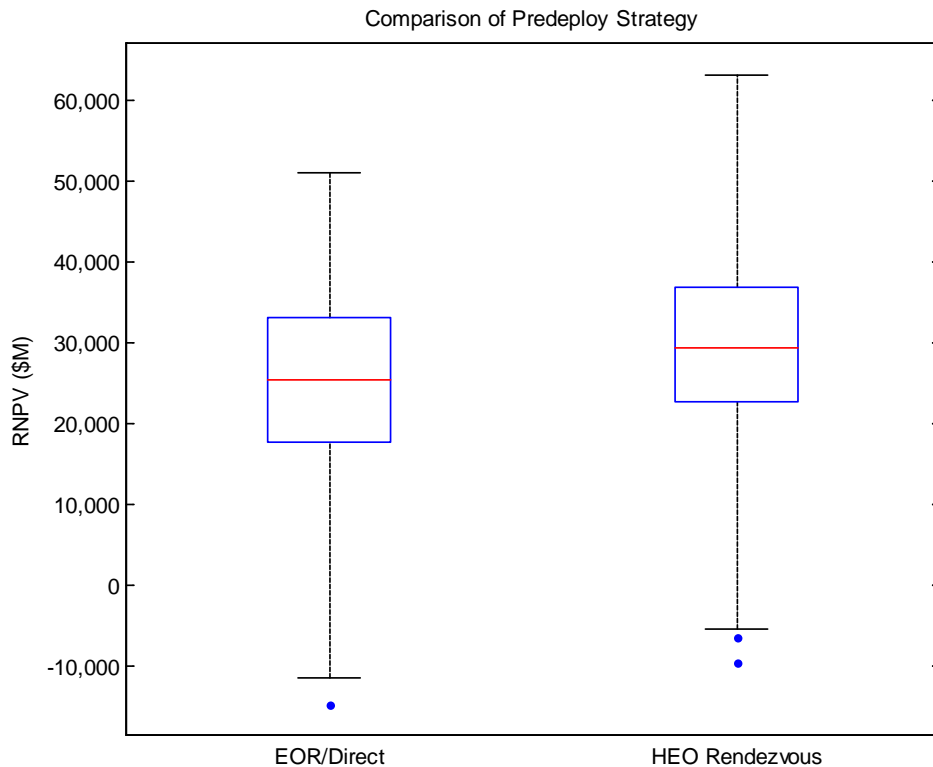


Figure 78: Box and Whisker Plot of NEO System Architectures for Different Pre-Deploy Strategies

5.3.4. Comparison between *IMLEO* and *RNPV*

Because *IMLEO* is a common metric used as a selection criterion, it is valuable to understand the relationship between it and *RNPV*. The results from the design space exploration for each mission are presented in Figure 79 – Figure 81, which compare the

IMLEO and RNPV of the analyzed system architectures. Figure 79 presents the results of the GEO mission design space exploration, with the launch vehicle selection identified. The figure does not show a clear trend between IMLEO and RNPV across the entire design space. For instance, design points that utilize a 130 mt HLLV with an IMLEO of approximately 100 mt have a significantly higher RNPV than those with an IMLEO near 300 mt but utilize a Falcon Heavy launch vehicle. Therefore, optimizing a system architecture based solely on IMLEO may not select an affordable system architecture.

However, given a specific launch vehicle, there is a positive correlation between IMLEO and RNPV. For instance, if the use of a Falcon Heavy launch vehicle is predetermined, the system architecture alternative with the lower IMLEO would also tend to have a lower RNPV. This phenomenon is primarily due to the increased number of required launches to place the required mass in LEO.

Figure 80 presents the comparison for the lunar design space, and Figure 81 presents the comparison for the NEO design space. Similar to the GEO mission design space, there is not a significant correlation between the IMLEO and RNPV in general because the launch vehicle cost drives the overall RNPV of the architecture. For both of these design spaces, the system architecture that has the lowest RNPV does not have the lowest IMLEO. The system architectures that utilize a Falcon Heavy, for a given IMLEO, have a lower RNPV than system architectures that use HLLVs. The disparity in launch vehicle cost per kilogram between commercial vehicles and HLLVs is the primary driver of the cost difference, not the difference in LEO payload requirements.

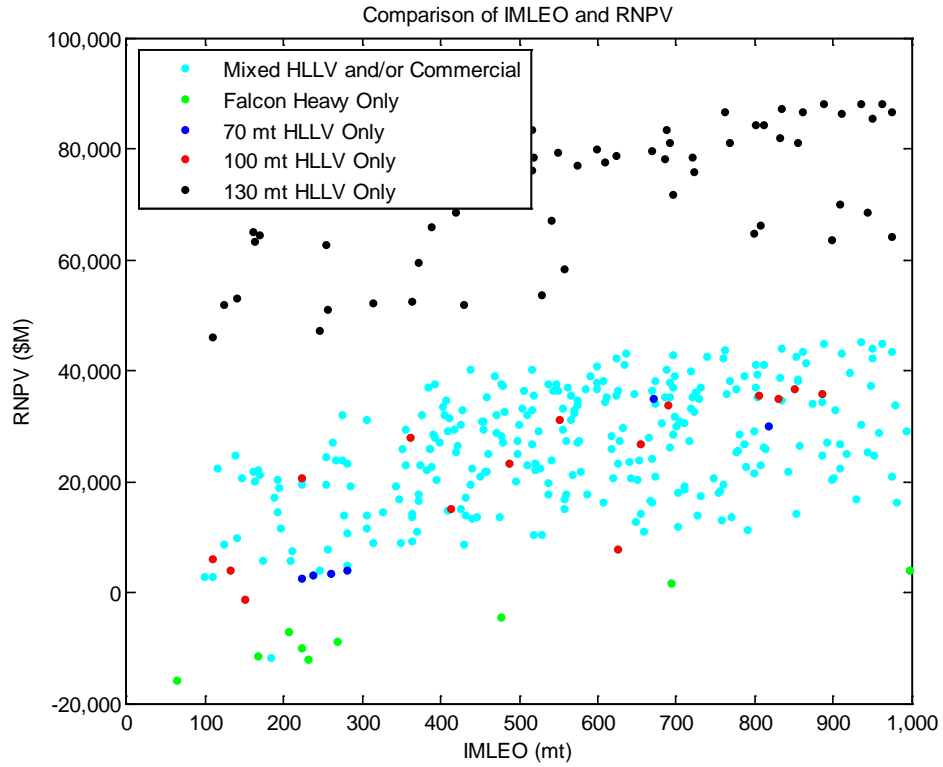


Figure 79: Comparison of IMLEO and RNPV for the GEO System Architecture Design Space

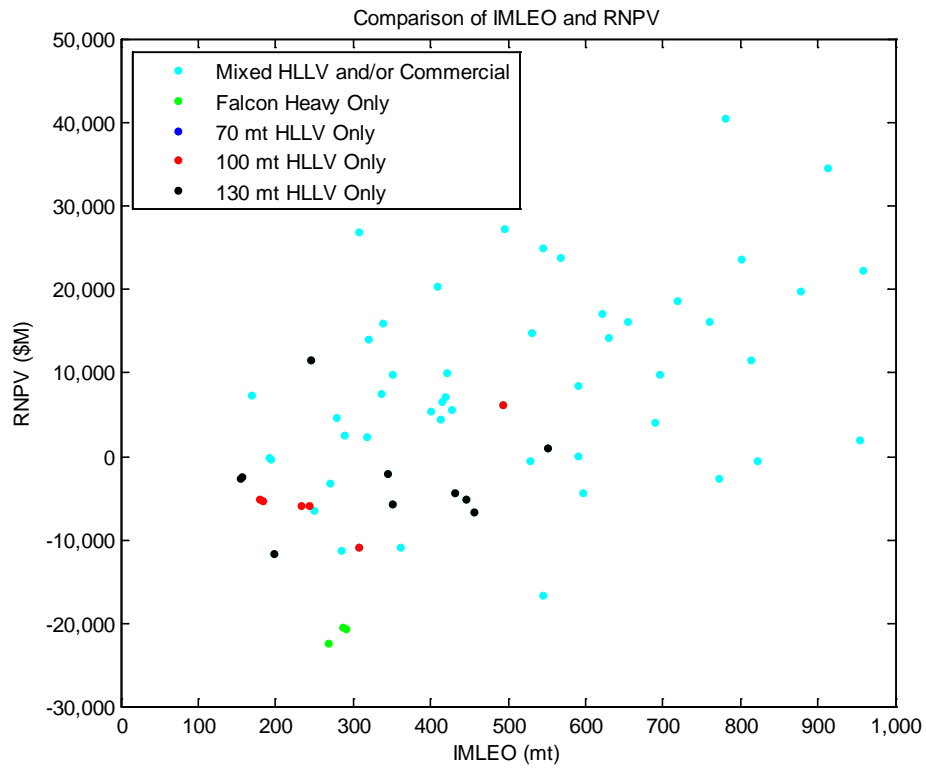


Figure 80: Comparison of IMLEO and RNPV for the Lunar System Architecture Design Space

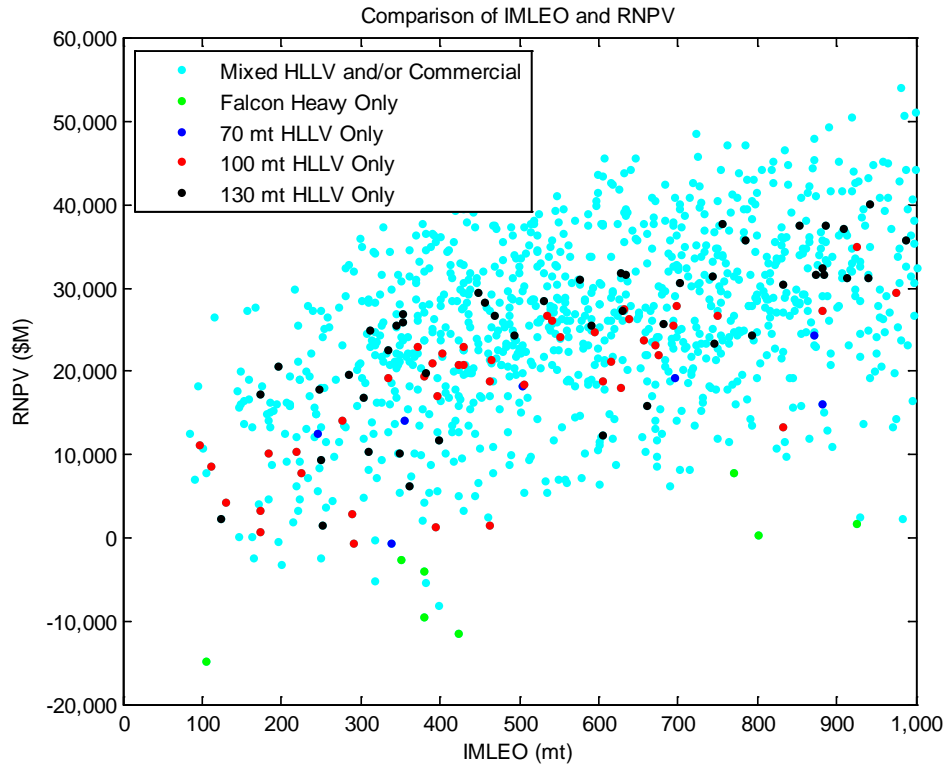


Figure 81: Comparison of IMLEO and RNPV for the NEO System Architecture Design Space

5.3.5. Summary

The exploration of the three system architecture design spaces reveals that launch vehicle selection is the primary driver of RNPV for a system architecture. In each system architecture design space, the selection of the launch vehicle, regardless of other architecture decisions, drastically alters the RNPV. The use of commercial launch vehicles, such as the Falcon Heavy, provides the best RNPV over the use of HLLVs. The DDT&E cost is zero, the flight unit cost is significantly less than HLLVs, and if the LEO payload capability is sufficient to perform the mission, a commercial launch vehicle is the preferred option.

Changes in other system architecture options have a lesser impact on the overall RNPV as compared to the use of a less expensive launch vehicle. The use of a propellant

depot does not significantly change RNPV when looking at the overall design space. At the best system architectures, the system architecture(s) that include a propellant depot show a slight increase in RNPV over the best system architecture without a propellant depot. However, this analysis does not take into account other factors, such as reliability, development risk mitigation, reusability, and launch availability. These factors must be considered before making a system architecture decision.

The aggregation strategies that are included in the system architecture design space exploration also reveal decisions that can reduce the RNPV. In the GEO mission, Earth orbit rendezvous or direct (if feasible) contains system architectures that have lower RNPV than GEO rendezvous. The lunar design space exploration revealed that EOR or LOR alone provide the lowest RNPV. Also, EOR with either LOR or lunar surface rendezvous can provide improvement over the baseline, but this effect is not as significant as launch vehicle selection. Also, leaving assets in LLO during the lunar surface mission reduces RNPV and decreases sensitivity to potential mass growth risk. Finally, aggregation of assets in LEO or HEO for a NEO mission does not show a distinct difference between the two options. Through the design space exploration, architectures with an EOR/direct strategy have better RNPV than HEO aggregation architectures, but this difference is small, and not as significant as the launch vehicle selection.

Finally, IMLEO is not necessarily correlated to overall architecture cost because the launch vehicle cost drives the overall RNPV of the architecture. For a given launch vehicle, the IMLEO and RNPV are correlated, but not in general. Therefore, using IMLEO as a selection criterion across the entire design space may not result in the system architecture with the lowest RNPV.

CHAPTER 6

CONCLUSIONS AND FUTURE WORK

This chapter provides conclusions about the modeling framework presented in this research and the implications of the design space exploration for a flexible path exploration program. It also presents recommendations on future work in this area to improve decision making, expand the system architecture design space, and provide increased fidelity and uncertainty quantification of the results.

6.1. Conclusions

The primary goal of the research presented in this dissertation is to improve upon space system architecture modeling in order to enable exploration of the architecture-level design space. The research presents a methodology to model the space system architecture design space using graph theory, creating a mathematical framework for design space exploration. The framework must meet five goals: technical credibility, adaptability, flexibility, intuitiveness, and exhaustiveness. The ability to model multiple aggregation strategies, staging locations, and system implementations (i.e. propellant type) throughout the design space creates a credible estimate of performance and cost for each system architecture within the design space. Comparing the results to previous system architecture studies validates the ability of the modeling framework to explore and analyze the system architecture design space. The use of graph theory enables the user to adapt the framework to any function or location within a given design space. This dissertation has demonstrated the flexibility of the modeling framework to analyze system architectures to multiple destinations (GEO, lunar, and NEO). Graph theory

creates a visual representation of the system architecture design space by using nodes to represent physical locations and steady states and using edges to represent the means to travel between those nodes (i.e. functions). A graphical user interface can be integrated with this tool to create a more intuitive experience for the system architect in both generating the system architecture design space and visualizing the results. Finally, the mathematical framework is able to analyze multiple options for aggregation, staging, system implementation, and launch strategy. Many system architecture studies in the past have allocated thousands of man-hours to produce few architecture alternatives, while this framework is capable of producing thousands of architecture alternatives without constant user interaction.

The goal of improved system architecture modeling is met through this research due to the accomplishment of several research objectives, as first posed in Section 1.2. Graph theory is capable of developing a mathematical representation of the space system architecture design space applicable to multiple mission types. Constraints, requirements, and interrelationships between systems are enforced through manipulation of the amount of pheromone along each edge. The pheromone matrix, as defined in ant colony optimization, defines the probability that a given system will traverse an edge. If the traversal of an edge would result in an infeasible architecture, the pheromone amount along that edge is set to zero. The system architecture definition, or system map, is flexibly linked to the system sizing tools through the use of topological sort, which develops a hierarchy of the systems to be sized and defines the information flow between system sizing tools. Finally, RNPV is used as a selection criterion to capture decision drivers across the evolutionary exploration program. Although this metric does not

necessarily capture all decision drivers, such as development risk, launch availability, and political risk, it is capable of providing the system architect insight into decisions that would lead to an affordable system architecture unlike a selection criterion based solely on mass.

By developing this modeling framework, several contributions have been added to the state of the art in space system architecture analysis. The framework adds the capability to rapidly explore the design space without the need to limit trade options or the need for user interaction during the exploration process. The unique mathematical representation of a system architecture, through the use of the adjacency, incidence, and system map matrices, enables automated design space exploration using stochastic optimization processes. The innovative rule-based graph traversal algorithm ensures functional feasibility of each system architecture that is analyzed, and the automatic generation of the system hierarchy eliminates the need for the user to manually determine the relationships between systems during or before the design space exploration process. Finally, the rapid evaluation of system architectures for various mission types enables analysis of the system architecture design space for multiple destinations within an evolutionary exploration program.

To demonstrate the functionality of this modeling framework, this dissertation presents the system architecture design space exploration of three missions within an evolutionary exploration program (GEO, lunar, and NEO). Each system architecture design space is represented as a graph, and is explored through the use of ant colony optimization. Alternative system architectures, which have significant reductions in cost over the baseline architectures, are produced for each mission, and a gradual capability

development strategy is presented that reduces cost over the evolutionary exploration program.

The results of the design space exploration reveal that the launch vehicle selection is the primary driver in the RNPV of a given system architecture. Other considerations, such as propellant type, staging location, and aggregation strategy provide less impact on the NPV of a given architecture. The use of commercial launch vehicles almost eliminates the DDT&E cost for the launch vehicle and reduces the cost per kilogram delivered to LEO. The RNPV formulation prefers to save money in the near term, when it has its greatest value (due to discounting and inflation). Therefore, when feasible, delaying the production of a HLLV provides greater value.

The detriment for using commercial launch vehicles is the increased number of flights required to deliver the in-space hardware. This increase in number of flights reduces the probability of mission success due to the increased operational complexity and increased launch failure risk. One solution to this issue is to develop an HLLV to reduce the number of required flights. This will, however, increase the overall cost of the system architecture by an order of magnitude. Alternatively, a propellant depot could reduce the number of critical launches that carry flight hardware and still use commercial launch vehicles to reduce the overall RNPV.

6.2. Future Work

Although this modeling framework has been effectively used to analyze an evolutionary exploration program, there are several areas where future work would improve decision making, enable exploration of new areas of the design space, and increase the model fidelity and uncertainty quantification. A notional block diagram of

the current capability of the modeling framework is presented in Figure 82. Additional capability can be added to this framework to improve its user interface, increase the fidelity of the analysis performed, and add more value to the decision-making process of the system architect. A notional block diagram of the framework with these additional capabilities is presented in Figure 83.

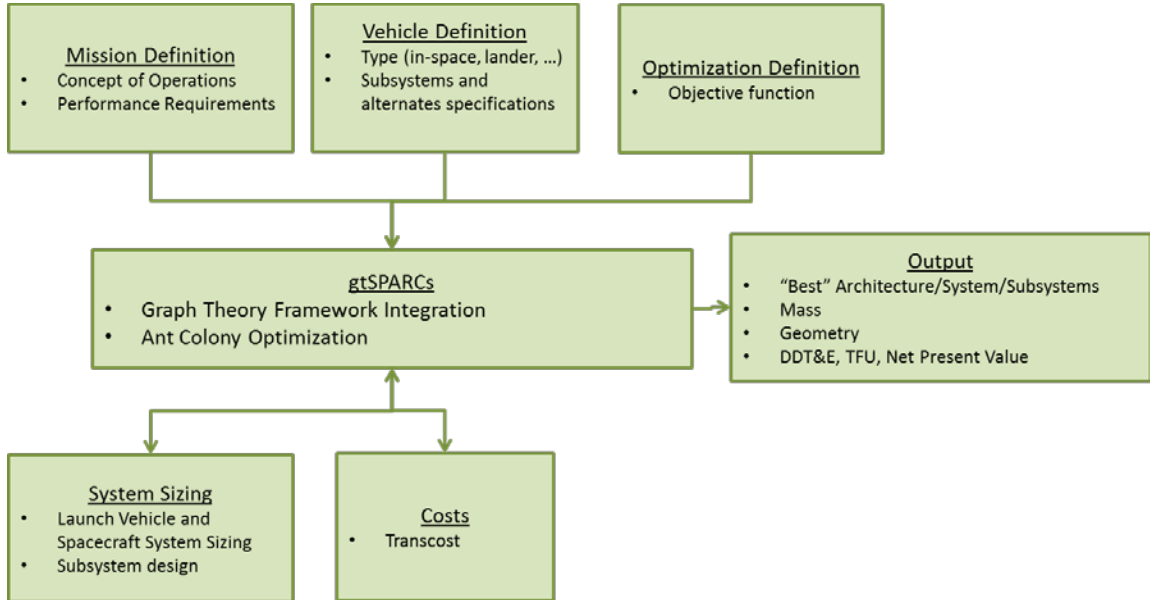


Figure 82: Notional Block Diagram of Current Modeling Framework Capability

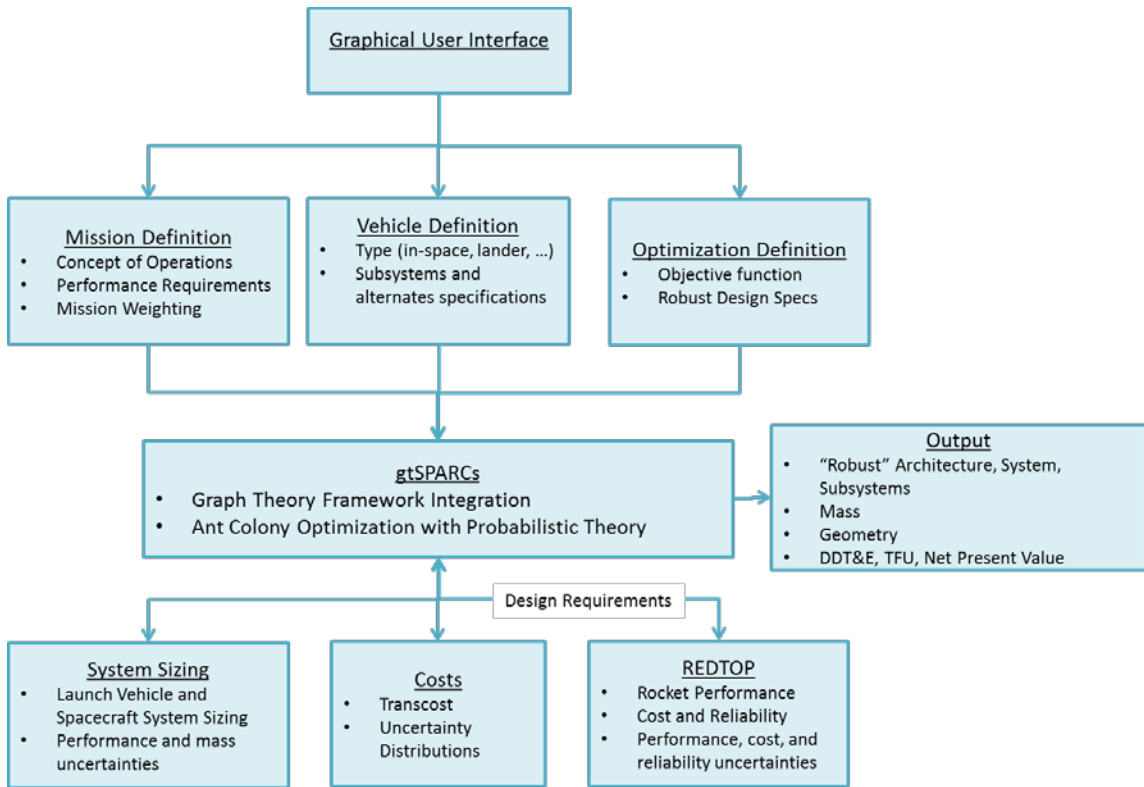


Figure 83: Notional Block Diagram of Future Modeling Framework Capability

Just as IMLEO is unable to capture all decision drivers, RNPV cannot be the only selection criterion used to make an architecture decision. Other costs that were not included in the calculation of DDT&E and flight unit cost, such as operations cost, disposal cost, and the fixed cost of operating a launch vehicle must be included. Risk and reliability of the system architecture must be estimated to determine a relative probability of mission success between two architecture alternatives. The event-based nature of representing a system architecture as a graph works well with the correlation of a function/event with a probability of failure.

Improved fidelity of the individual system sizing and cost estimation models will improve the value of the modeling framework to the system architect. As important as improved fidelity is also the understanding of uncertainty. Uncertainty in the inputs and in the models can change what the system architect would consider the optimal

architecture. The capability to capture these uncertainties in the edge definition and system sizing and cost estimation is needed to make informed decisions.

Beyond improved analysis capability, improvements in speed and model flexibility will provide more value to the system architect. The current framework consists of MATLAB classes and functions and Excel workbooks. The information flow between these two programs is slow and can be cumbersome. Transition to a consistent code, such as C#, would reduce run times by orders of magnitude. Also, enabling feedback between the systems and the edges (currently, the information flow is only from the edges to the systems) would increase the usefulness of the framework. This would improve the ability to analyze refueling options, suborbital burning, and drop stage performance, among others. Finally, the ability for the system architect to override the automated system hierarchy to force a certain system to perform a function would improve the adaptability.

Finally, the three missions analyzed in this dissertation are not all-inclusive by any stretch. Even within each of the design spaces, the impact of changing mission requirements for a given destination has significant impacts on the system architectures. These impacts should be understood in order to make an informed decision. Also, other destinations, such as cis-lunar locations, different NEO classes, the moons of Mars, and the Mars surface are examples of a rich set of system architecture design spaces that are still yet to be explored.

APPENDIX A

This appendix contains the definitions of each edge for the three system architecture design space graphs presented in this document. Table A-1 contains the edge definitions for the GEO mission graph, Table A-2 contains the edge definitions for the lunar mission graph, and Table A-3 and Table A-4 contains the edge data for the NEO mission graph with HEO aggregation and LEO aggregation, respectively.

Table A-1: GEO System Architecture Design Space Graph Definition

Edge No.	Edge Group Name (Type)	Metadata	Edge Options				From Node	To Node	
1	Earth Launch to LEO (Earth Launch)	Name	Falcon Heavy	Delta IV-H	70 mt	100 mt	130 mt	1	3
		Scenario	Falcon Heavy	Delta IV-H	70 mt	100 mt	130 mt		
		StagePt	LEO	LEO	LEO	LEO	LEO		
2	Earth Launch to Suborbital (Earth Launch)	Name	Falcon Heavy	70 mt	100 mt	130 mt	1	2	
		Scenario	Falcon Heavy	70 mt	100 mt	130 mt			
		StagePt	Suborbital	Suborbital	Suborbital	Suborbital			
3	Suborbital Burn (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	2	3	
		deltaV	2442	2442	2442	2442			
		SystemTW	0.8574	0.8574	0.8574	0.8574			
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH			
		TOF	0.1	0.1	0.1	0.1			
		Planet	Earth	Earth	Earth	Earth			
4	GTO Departure (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	3	5	
		deltaV	2420	2420	2420	2420			
		SystemTW	0.3	0.3	0.3	0.3			
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH			
		TOF	0.22	0.22	0.22	0.22			
		Planet	Earth	Earth	Earth	Earth			
5	LEO Refueling (Refuel)	Name	Delta IV-H	Falcon			3	4	
		LaunchCost	14286	2358					
6	GTO Departure from Depot (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	4	5	
		deltaV	2420	2420	2420	2420			
		SystemTW	0.3	0.3	0.3	0.3			
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH			
		TOF	0.22	0.22	0.22	0.22			
		Planet	Earth	Earth	Earth	Earth			
7	GEO Insertion (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	5	6	
		deltaV	1775	1775	1775	1775			
		SystemTW	0.3	0.3	0.3	0.3			
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH			
		TOF	0.1	0.1	0.1	0.1			
		Planet	Earth	Earth	Earth	Earth			
8	GEO Mission (In-Space Habitation)	Name	Standard				6	7	
		Scenario	Standard						
		Stay Time	9						

Continued on next page

Edge No.	Edge Group Name (Type)	Metadata	Edge Options				From Node	To Node
9	GEO Departure to GTO (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	7	8
		deltaV	1775	1775	1775	1775		
		SystemTW	0.3	0.3	0.3	0.3		
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH		
		TOF	0.22	0.22	0.22	0.22		
		Planet	Earth	Earth	Earth	Earth		
10	LEO Insertion Burn (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	8	9
		deltaV	2420	2420	2420	2420		
		SystemTW	0.3	0.3	0.3	0.3		
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH		
		TOF	0.5	0.5	0.5	0.5		
		Planet	Earth	Earth	Earth	Earth		
11	Direct Earth Entry from GTO (Planetary EDL)	Name	Capsule				8	10
		Ventry	9.4					
		LoD	0.1					
		deltaV	0					
		SystemTW	0					
		Planet	Earth					
12	Entry from LEO (Planetary EDL)	Name	Capsule				9	10
		Ventry	8.2					
		LoD	0.1					
		deltaV	0					
		SystemTW	0					
		Planet	Earth					

Table A-2: Lunar System Architecture Design Space Graph Definition

Edge No.	Edge Group Name (Type)	Metadata	Edge Options				From Node	To Node	
1	Earth Launch to LEO (Earth Launch)	Name	Falcon Heavy	Delta IV-H	70 mt	100 mt	130 mt	1	3
		Scenario	Falcon Heavy	Delta IV-H	70 mt	100 mt	130 mt		
		StagePt	LEO	LEO	LEO	LEO	LEO		
2	Earth Launch to Suborbital (Earth Launch)	Name	Falcon Heavy	70 mt	100 mt	130 mt	1	2	
		Scenario	Falcon Heavy	70 mt	100 mt	130 mt			
		StagePt	Suborbital	Suborbital	Suborbital	Suborbital			
3	Suborbital Burn (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	2	3	
		deltaV	2442	2442	2442	2442			
		SystemTW	0.8574	0.8574	0.8574	0.8574			
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH			
		TOF	0.1	0.1	0.1	0.1			
		Planet	Earth	Earth	Earth	Earth			
4	TLI from LEO (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	3	5	
		deltaV	3274	3274	3274	3274			
		SystemTW	0.3	0.3	0.3	0.3			
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH			
		TOF	5	5	5	5			
		Planet	Earth	Earth	Earth	Earth			
5	Refuel at LEO Depot (Refuel)	Name	Delta IV-H	Falcon			3	4	
		LaunchCost	14286	2358					
6	TLI from LEO Depot (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	4	5	
		deltaV	3274	3274	3274	3274			
		SystemTW	0.3	0.3	0.3	0.3			
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH			
		TOF	5	5	5	5			
		Planet	Earth	Earth	Earth	Earth			

Continued on next page

Edge No.	Edge Group Name (Type)	Metadata	Edge Options				From Node	To Node
7	LOI (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	5	6
		deltaV	924	924	924	924		
		SystemTW	0.3	0.3	0.3	0.3		
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH		
		TOF	1	1	1	1		
Planet	Moon	Moon	Moon	Moon				
8	Lunar Descent from LLO (Planetary Descent)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	6	8
		deltaV	2203	2203	2203	2203		
		SystemTW	1.66	1.66	1.66	1.66		
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH		
		Planet	Moon	Moon	Moon	Moon		
9	Braking (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	6	7
		deltaV	1762	1762	1762	1762		
		SystemTW	0.3	0.3	0.3	0.3		
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH		
		TOF	0.1	0.1	0.1	0.1		
Planet	Moon	Moon	Moon	Moon				
10	Lunar Descent from Braking (Planetary Descent)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	7	8
		deltaV	441	441	441	441		
		SystemTW	1.66	1.66	1.66	1.66		
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH		
		Planet	Moon	Moon	Moon	Moon		
11	Lunar Surface Mission (Surface Habitation)	Name	Sortie				8	9
		Scenario	Sortie					
		StayTime	7					
12	Lunar Ascent (Planetary Ascent)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	9	10
		deltaV	1968	1968	1968	1968		
		SystemTW	1.97	1.97	1.97	1.97		
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH		
		Planet	Moon	Moon	Moon	Moon		
13	TEI from LLO (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	10	11
		deltaV	1196	1196	1196	1196		
		SystemTW	0.3	0.3	0.3	0.3		
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH		
		TOF	4	4	4	4		
Planet	Moon	Moon	Moon	Moon				
14	Direct Entry (Planetary EDL)	Name	Capsule				11	13
		Ventry	11.1					
		LoD	0.1					
		deltaV	0					
		SystemTW	0					
Planet	Earth							
15	EOI (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	11	12
		deltaV	3359	3359	3359	3359		
		SystemTW	0.3	0.3	0.3	0.3		
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH		
		TOF	1	1	1	1		
Planet	Earth	Earth	Earth	Earth				
16	Entry from LEO (Planetary EDL)	Name	Capsule				12	13
		Ventry	8.2					
		LoD	0.1					
		deltaV	0					
		SystemTW	0					
Planet	Earth							

Table A-3: NEO System Architecture (HEO Aggregation) Design Space Graph Definition

Edge No.	Edge Group Name (Type)	Metadata	Edge Options				From Node	To Node	
1	Earth Launch to LEO (Earth Launch)	Name	Falcon Heavy	Delta IV-H	70 mt	100 mt	130 mt	1	3
		Scenario	Falcon Heavy	Delta IV-H	70 mt	100 mt	130 mt		
		StagePt	LEO	LEO	LEO	LEO	LEO		
2	Suborbital (Earth Launch)	Name	Falcon Heavy	70 mt	100 mt	130 mt	1	2	
		Scenario	Falcon Heavy	70 mt	100 mt	130 mt			
		StagePt	Suborbital	Suborbital	Suborbital	Suborbital			
3	Suborbital Burn (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	2	3	
		deltaV	2442	2442	2442	2442			
		SystemTW	0.8574	0.8574	0.8574	0.8574			
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH			
		TOF	0.1	0.1	0.1	0.1			
4	LEO-HEO Burn (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	3	5	
		deltaV	3025	3025	3025	3025			
		SystemTW	0.3	0.3	0.3	0.3			
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH			
		TOF	2	2	2	2			
5	Refuel at LEO Depot (Refuel)	Name	Delta IV-H	Falcon			3	4	
		LaunchCost	14286	2358					
6	Depot-HEO Burn (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	4	5	
		deltaV	3025	3025	3025	3025			
		SystemTW	0.3	0.3	0.3	0.3			
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH			
		TOF	2	2	2	2			
7	TNI from HEO (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	5	6	
		deltaV	219	219	219	219			
		SystemTW	0.3	0.3	0.3	0.3			
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH			
		TOF	191	191	191	191			
8	NEO Arrival Burn (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	6	7	
		deltaV	142	142	142	142			
		SystemTW	0.3	0.3	0.3	0.3			
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH			
		TOF	1	1	1	1			
9	NEO Destination Mission (In-Space Habitation)	Name	Standard				7	8	
		Senario	Standard						
		Stay Time	14						
10	TEI Burn (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	8	9	
		deltaV	178	178	178	178			
		SystemTW	0.3	0.3	0.3	0.3			
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH			
		TOF	153	153	153	153			
11	Direct Entry (Planetary EDL)	Name	11.15				9	12	
		Ventry	0.1						
		LoD	0						
		deltaV	0						
		SystemTW	Earth						

Continued on next page

Edge No.	Edge Group Name (Type)	Metadata	Edge Options				From Node	To Node
12	HEO Arrival Burn (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	9	10
		deltaV	3148	3148	3148	3148		
		SystemTW	0.3	0.3	0.3	0.3		
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH		
		TOF	2	2	2	2		
		Planet	Earth	Earth	Earth	Earth		
13	EDL from HEO (Planetary EDL)	Name	Capsule				10	12
		Ventry	10.7					
		LoD	0.1					
		deltaV	0					
		SystemTW	0					
		Planet	Earth					
14	LEO Arrival Burn (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	9	11
		deltaV	3148	3148	3148	3148		
		SystemTW	0.3	0.3	0.3	0.3		
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH		
		TOF	0.1	0.1	0.1	0.1		
		Planet	Earth	Earth	Earth	Earth		
15	EDL from LEO (Planetary EDL)	Name	Capsule				11	12
		Ventry	8.2					
		LoD	0.1					
		deltaV	0					
		SystemTW	0					
		Planet	Earth					

Table A-4: NEO System Architecture (LEO Aggregation) Design Space Graph Definition

Edge No.	Edge Group Name (Type)	Metadata	Edge Options				From Node	To Node	
1	Earth Launch to LEO (Earth Launch)	Name	Falcon Heavy	Delta IV-H	70 mt	100 mt	130 mt	1	3
		Scenario	Falcon Heavy	Delta IV-H	70 mt	100 mt	130 mt		
		StagePt	LEO	LEO	LEO	LEO	LEO		
2	Earth Launch to Suborbital (Earth Launch)	Name	Falcon Heavy	70 mt	100 mt	130 mt	1	2	
		Scenario	Falcon Heavy	70 mt	100 mt	130 mt			
		StagePt	Suborbital	Suborbital	Suborbital	Suborbital			
3	Suborbital Burn (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	2	3	
		deltaV	2442	2442	2442	2442			
		SystemTW	0.8574	0.8574	0.8574	0.8574			
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH			
		TOF	0.1	0.1	0.1	0.1			
		Planet	Earth	Earth	Earth	Earth			
4	Trans-NEO Injection from LEO (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	3	5	
		deltaV	3244	3244	3244	3244			
		SystemTW	0.3	0.3	0.3	0.3			
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH			
		TOF	191	191	191	191			
		Planet	Earth	Earth	Earth	Earth			
5	Refuel at LEO Depot (Refuel)	Name	Delta IV-H	Falcon			3	4	
		LaunchCost	14286	2358					
6	TNI from Depot (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	4	5	
		deltaV	3244	3244	3244	3244			
		SystemTW	0.3	0.3	0.3	0.3			
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH			
		TOF	191	191	191	191			
		Planet	Earth	Earth	Earth	Earth			

Continued on next page

Edge No.	Edge Group Name (Type)	Metadata	Edge Options				From Node	To Node
7	NEO Arrival Burn (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	5	6
		deltaV	142	142	142	142		
		SystemTW	0.3	0.3	0.3	0.3		
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH		
		TOF	1	1	1	1		
		Planet	NEO	NEO	NEO	NEO		
8	NEO Destination Mission (In-Space Habitation)	Name	Standard				6	7
		Scenario	Standard					
		Stay Time	14					
9	TEI Burn (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	7	8
		deltaV	178	178	178	178		
		SystemTW	0.3	0.3	0.3	0.3		
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH		
		TOF	153	153	153	153		
		Planet	NEO	NEO	NEO	NEO		
10	Direct Entry (Planetary EDL)	Name	Capsule				8	11
		Ventry	11.15					
		LoD	0.1					
		deltaV	0					
		SystemTW	0					
11	HEO Arrival Burn (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	8	9
		deltaV	3148	3148	3148	3148		
		SystemTW	0.3	0.3	0.3	0.3		
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH		
		TOF	2	2	2	2		
		Planet	Earth	Earth	Earth	Earth		
12	EDL from HEO (Planetary EDL)	Name	Capsule				9	11
		Ventry	10.7					
		LoD	0.1					
		deltaV	0					
		SystemTW	0					
13	LEO Arrival Burn (Propulsive)	Name	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH	8	10
		deltaV	3148	3148	3148	3148		
		SystemTW	0.3	0.3	0.3	0.3		
		EngineType	LOX/LH2	LOX/RP-1	LOX/CH4	NTO/MMH		
		TOF	0.1	0.1	0.1	0.1		
		Planet	Earth	Earth	Earth	Earth		
14	EDL from LEO (Planetary EDL)	Name	Capsule				10	11
		Ventry	8.2					
		LoD	0.1					
		deltaV	0					
		SystemTW	0					
		Planet	Earth					

APPENDIX B

This appendix contains the CER curves to estimate the DDT&E and flight unit costs of each system within a given system architecture. The CERs for the system types identified in Table 14 appear in Figure B-1 through Figure B-7 below.

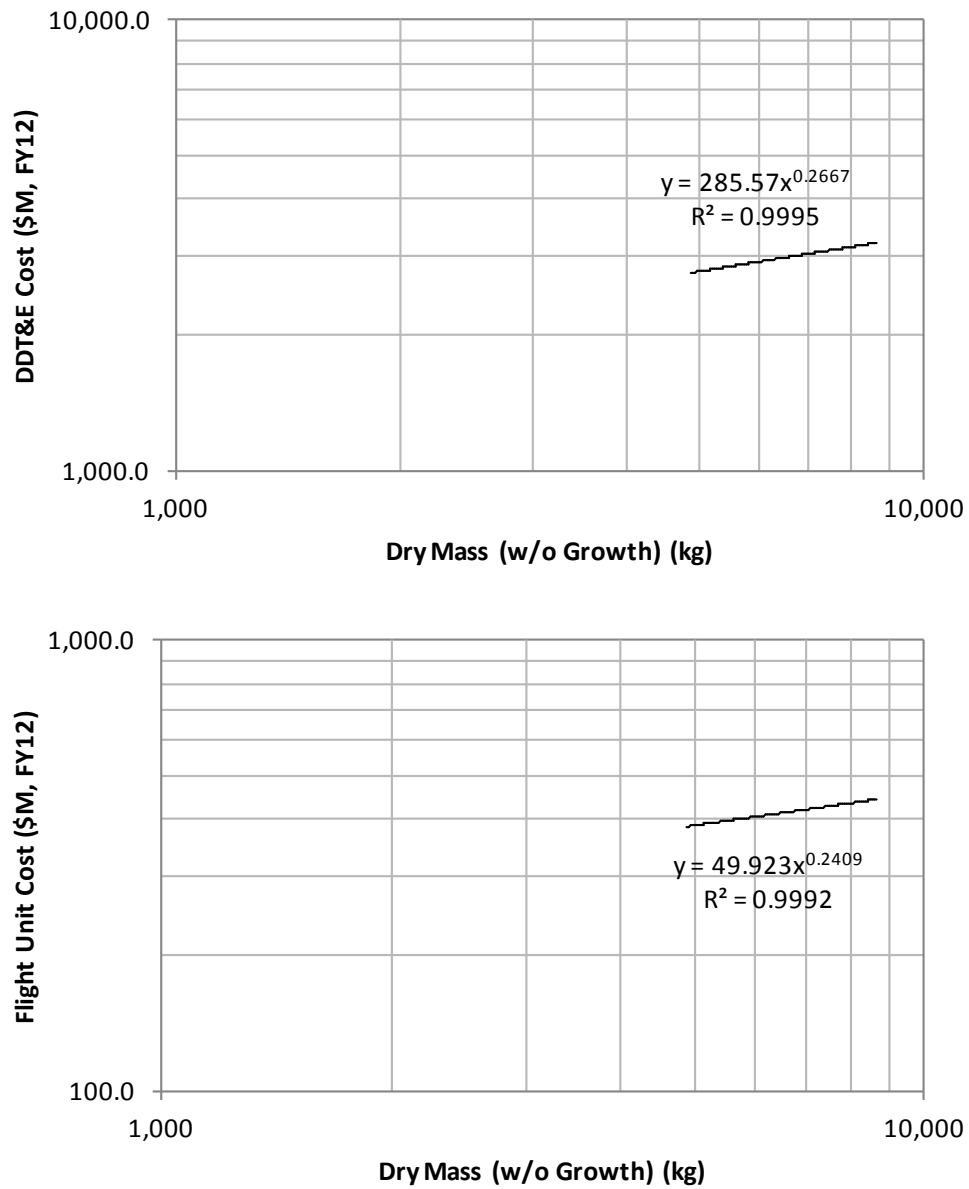


Figure B-1: Crew Capsule CER for DDT&E Cost (Top) and Flight Unit Cost (Bottom)

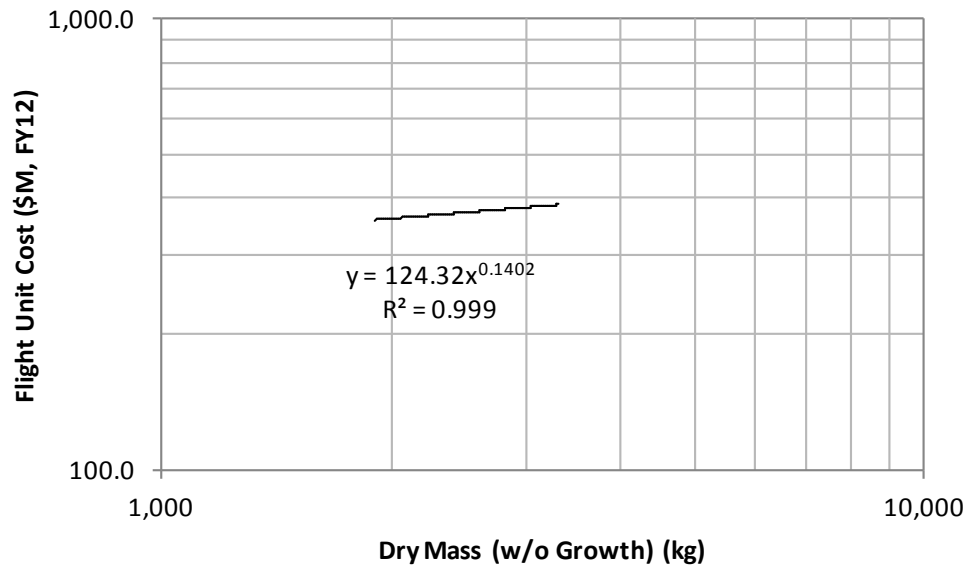
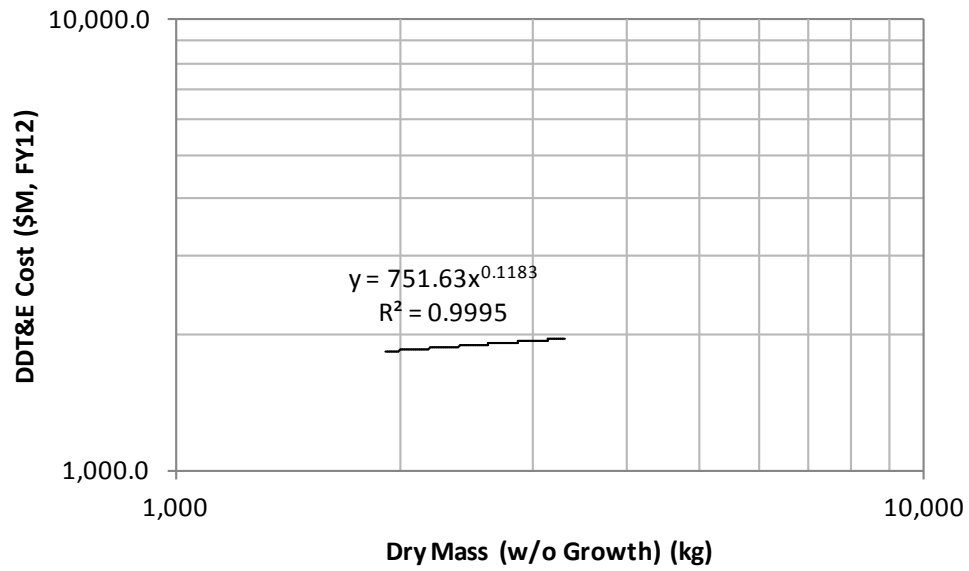


Figure B-2: Surface Habitat CER for DDT&E Cost (Top) and Flight Unit Cost (Bottom)

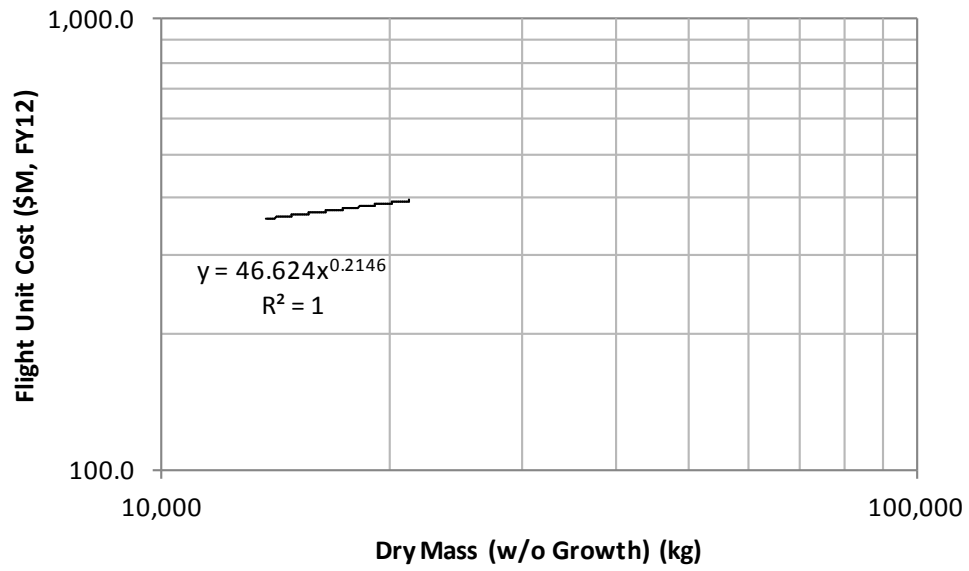
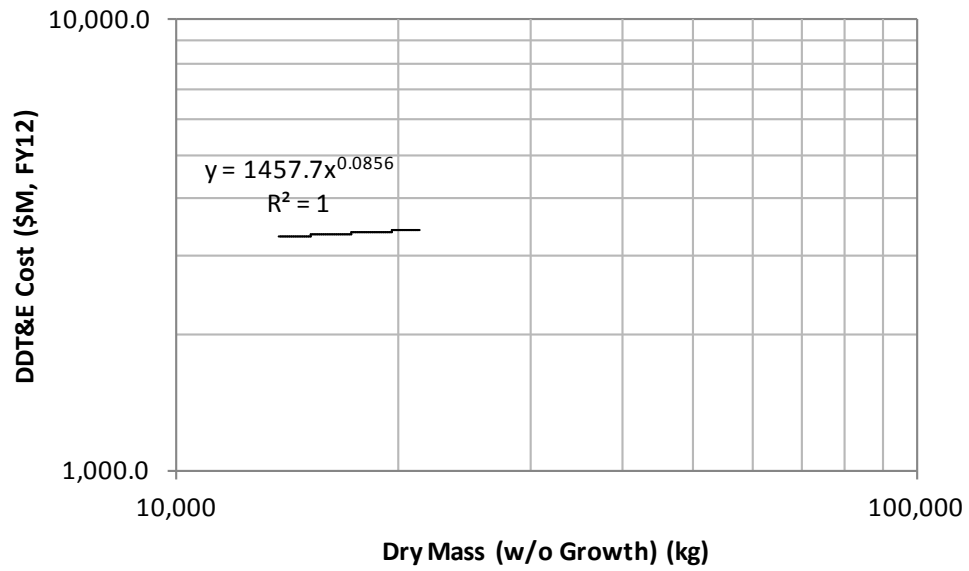


Figure B-3: In-Space Habitat CER for DDT&E Cost (Top) and Flight Unit Cost (Bottom)

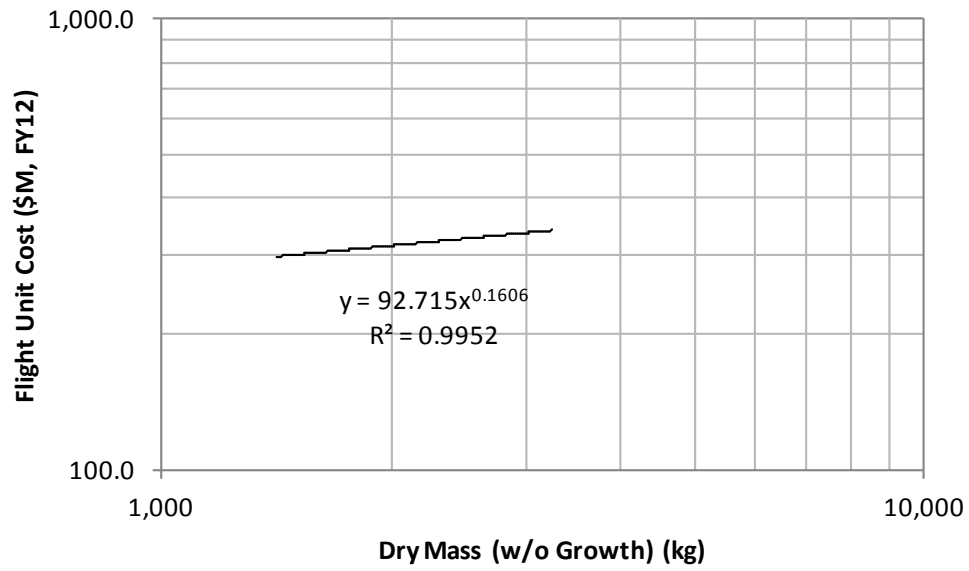
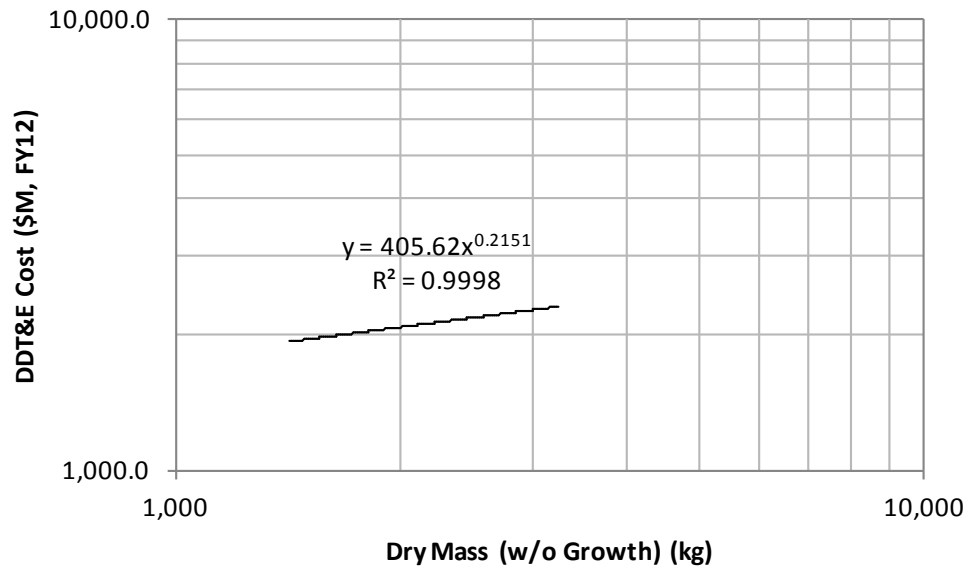


Figure B-4: Cryogenic Lunar Ascent Stage CER for DDT&E Cost (Top) and Flight Unit Cost (Bottom)

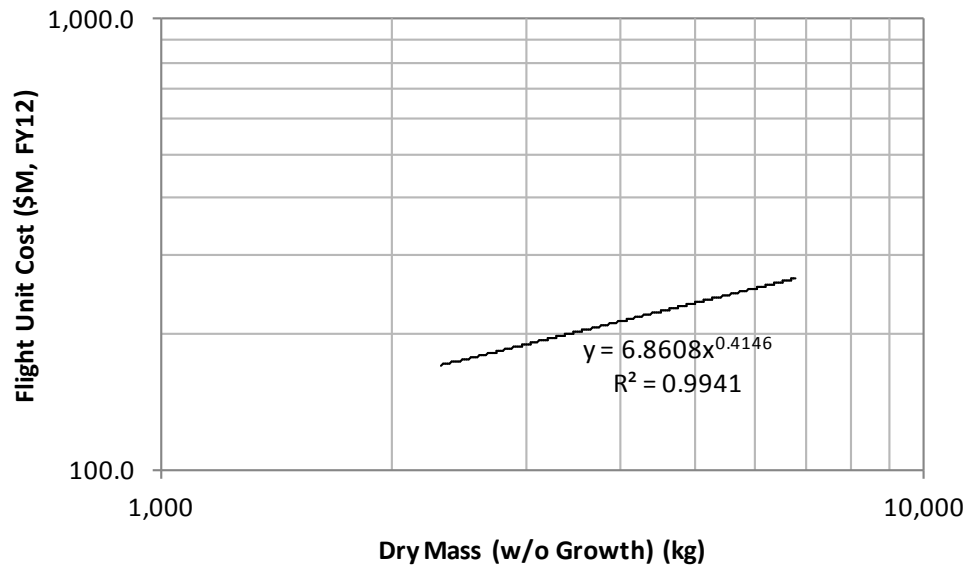
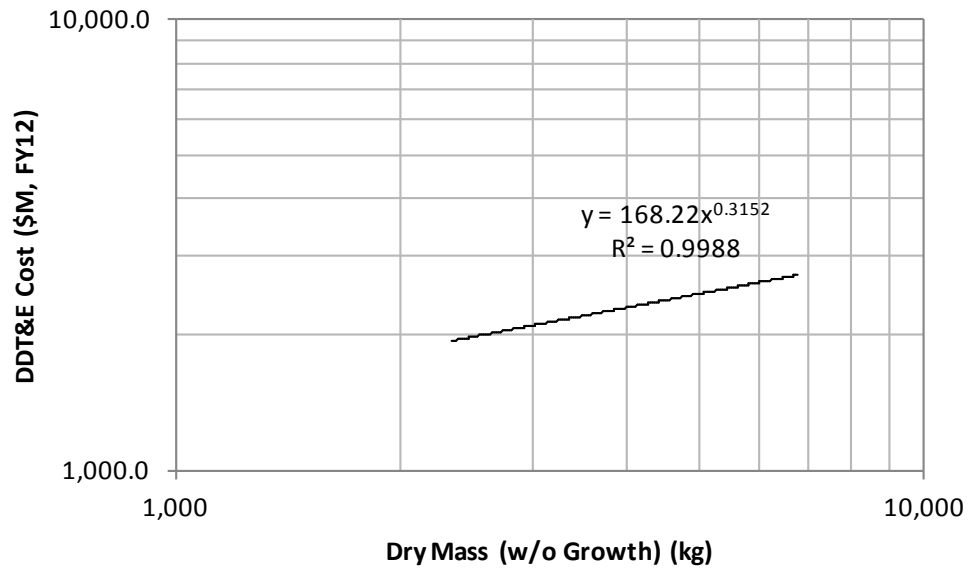


Figure B-5: Cryogenic Lunar Descent Stage CER for DDT&E Cost (Top) and Flight Unit Cost (Bottom)

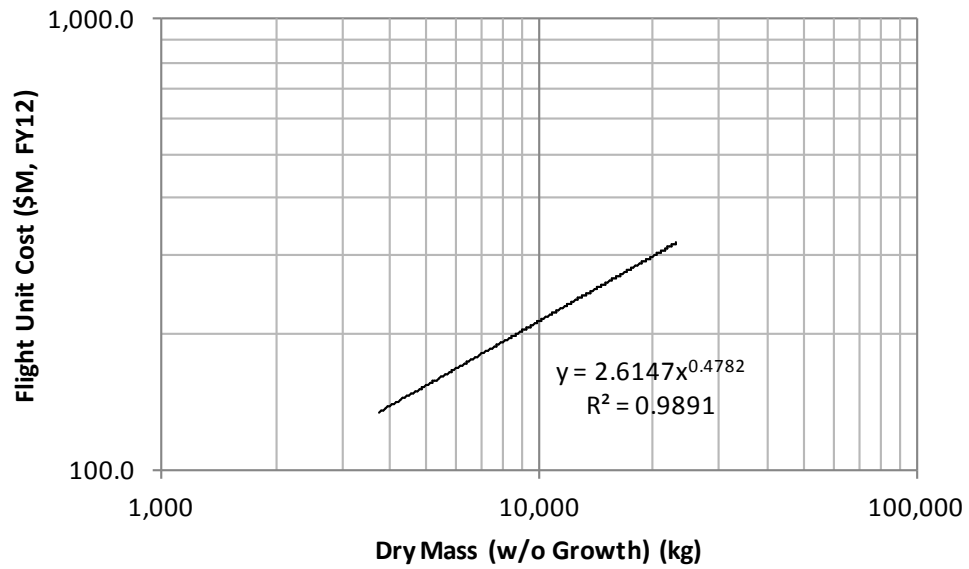
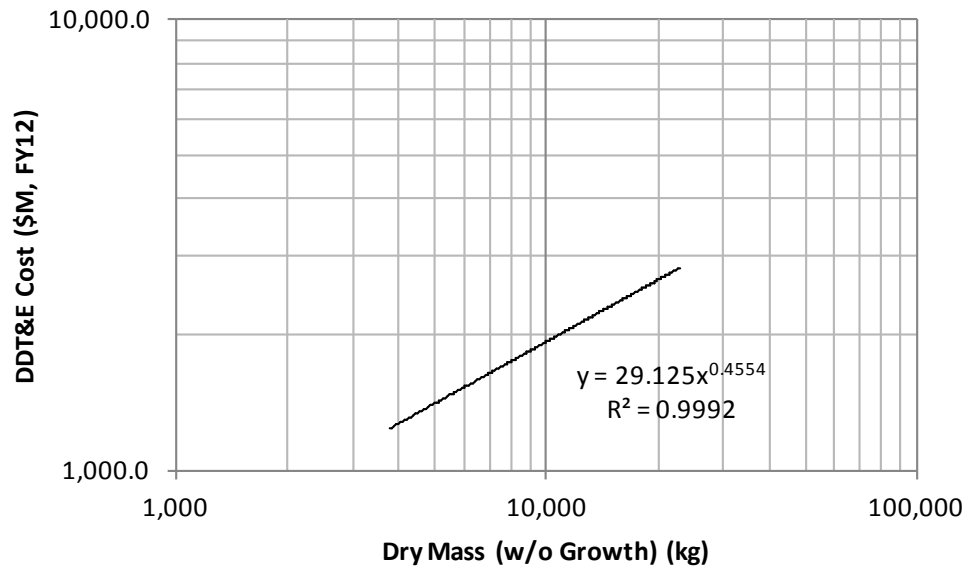


Figure B-6: Cryogenic Propulsive Stage CER for DDT&E Cost (Top) and Flight Unit Cost (Bottom)

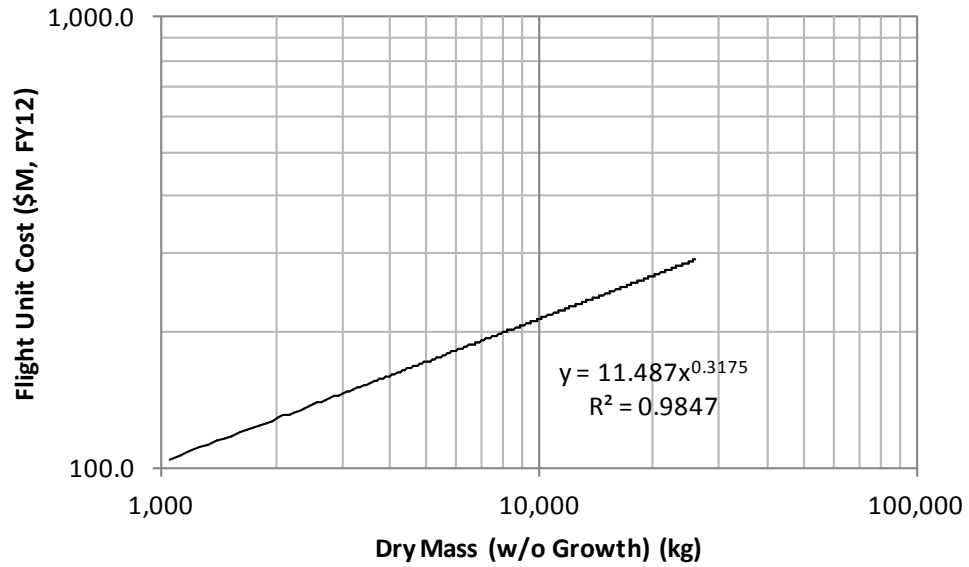
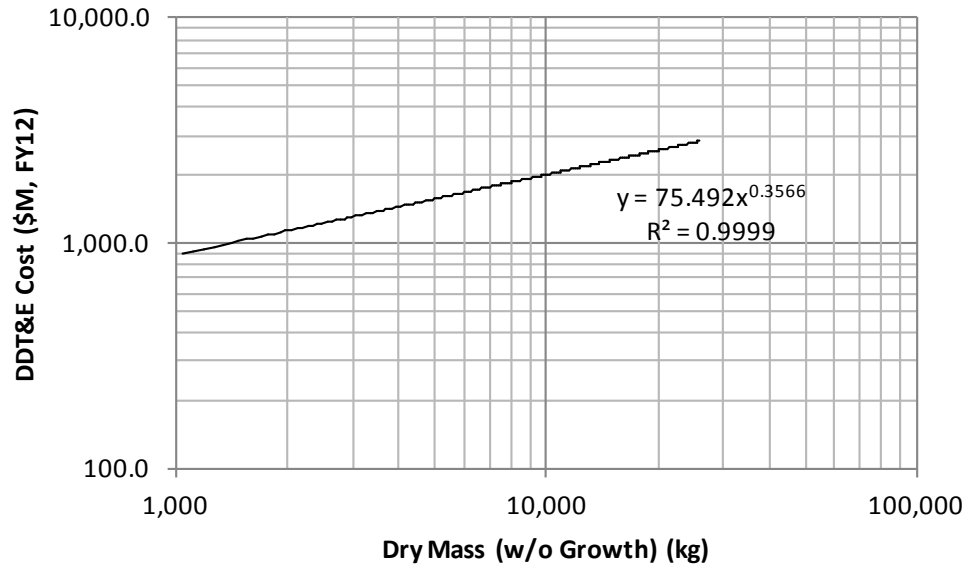


Figure B-7: Propellant Depot CER for DDT&E Cost (Top) and Flight Unit Cost (Bottom)

APPENDIX C

This appendix contains more information on the results presented in Chapter 5.

C.1. Launch Vehicles

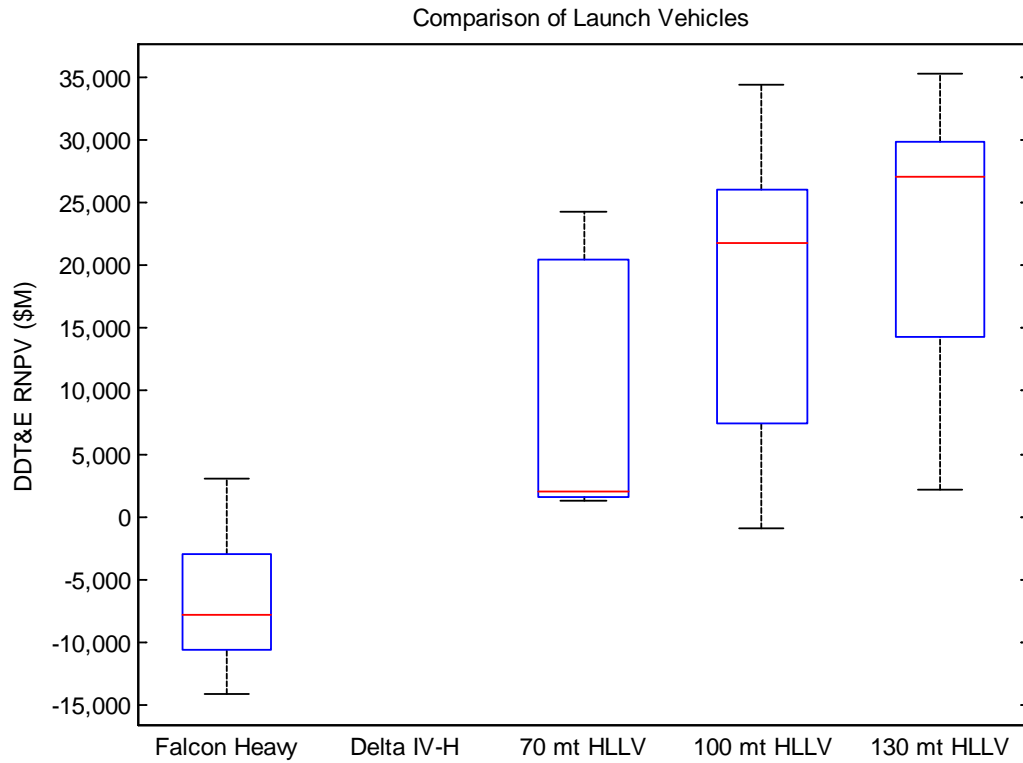


Figure C-1: Box and Whisker Plot of DDT&E RNPV for GEO System Architectures that Exclusively Use a Certain Launch Vehicle

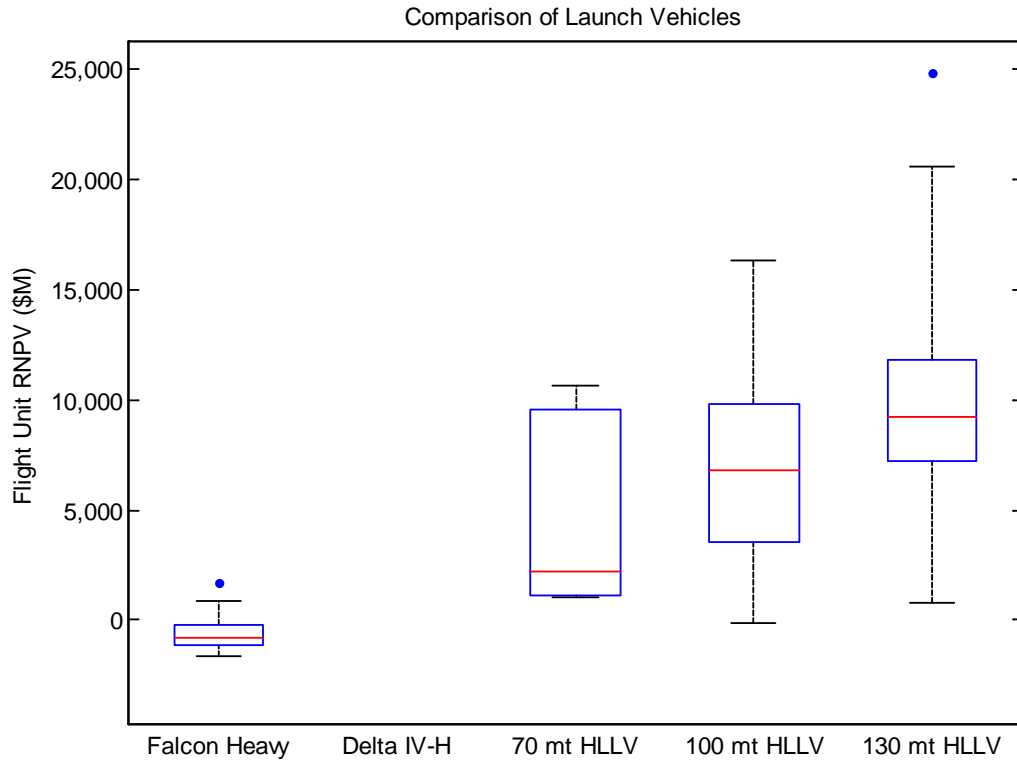


Figure C-2: Box and Whisker Plot of Flight Unit RNPV for GEO System Architectures that Exclusively Use a Certain Launch Vehicle

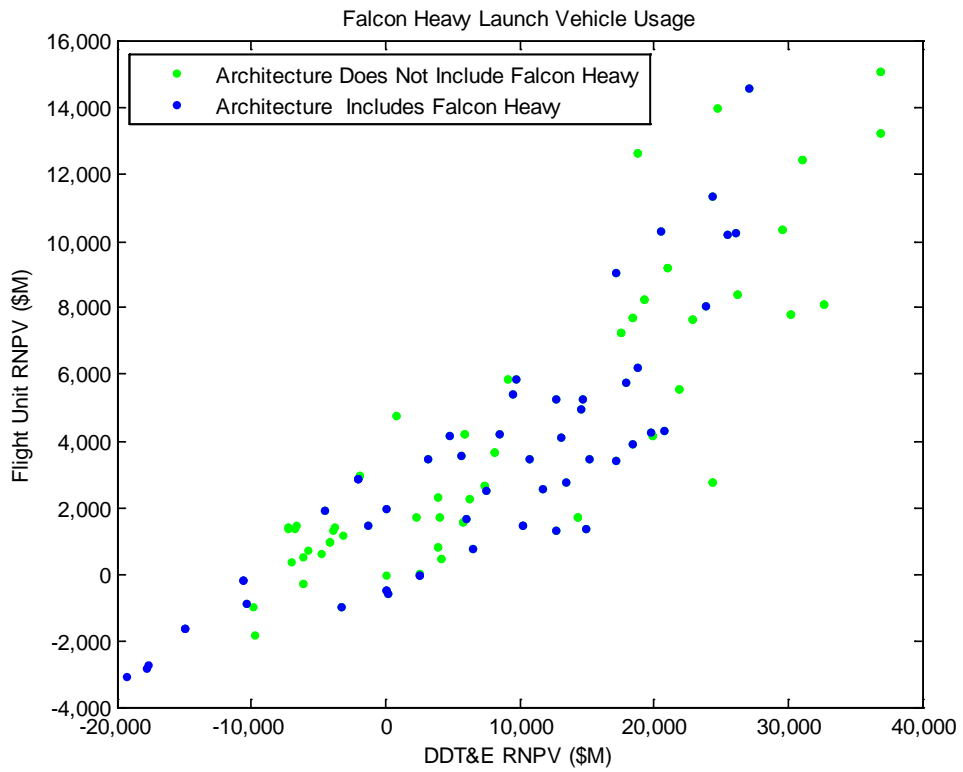


Figure C-3: RNPV of Lunar System Architectures that Utilize a Certain Launch Vehicle

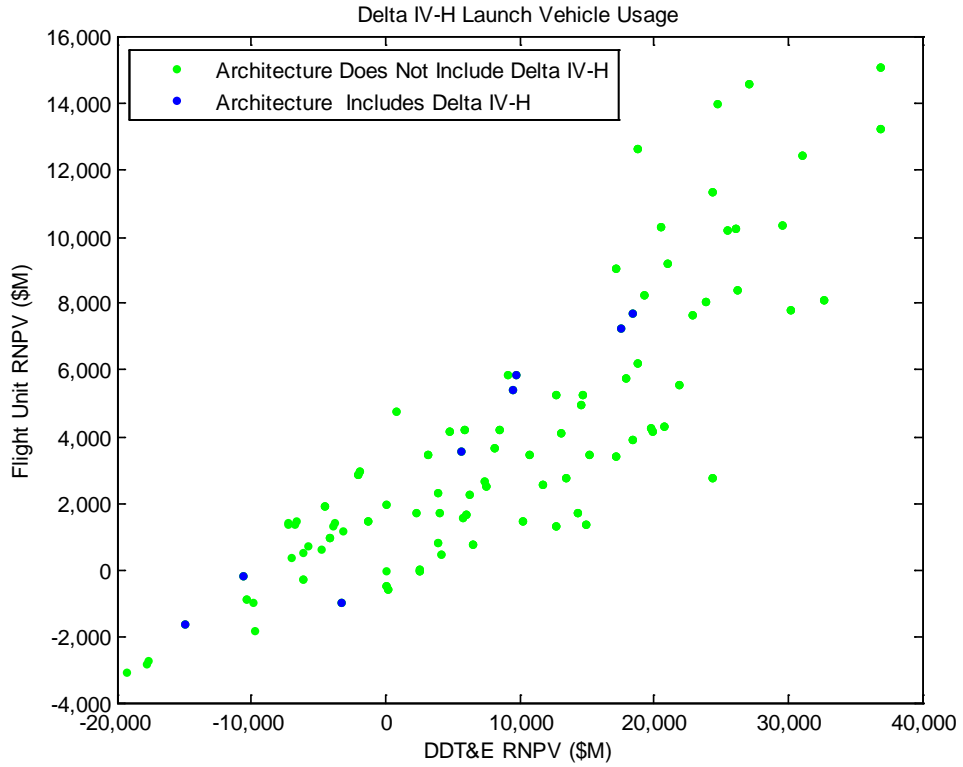


Figure C-4: RNPV of Lunar System Architectures that Utilize a Certain Launch Vehicle

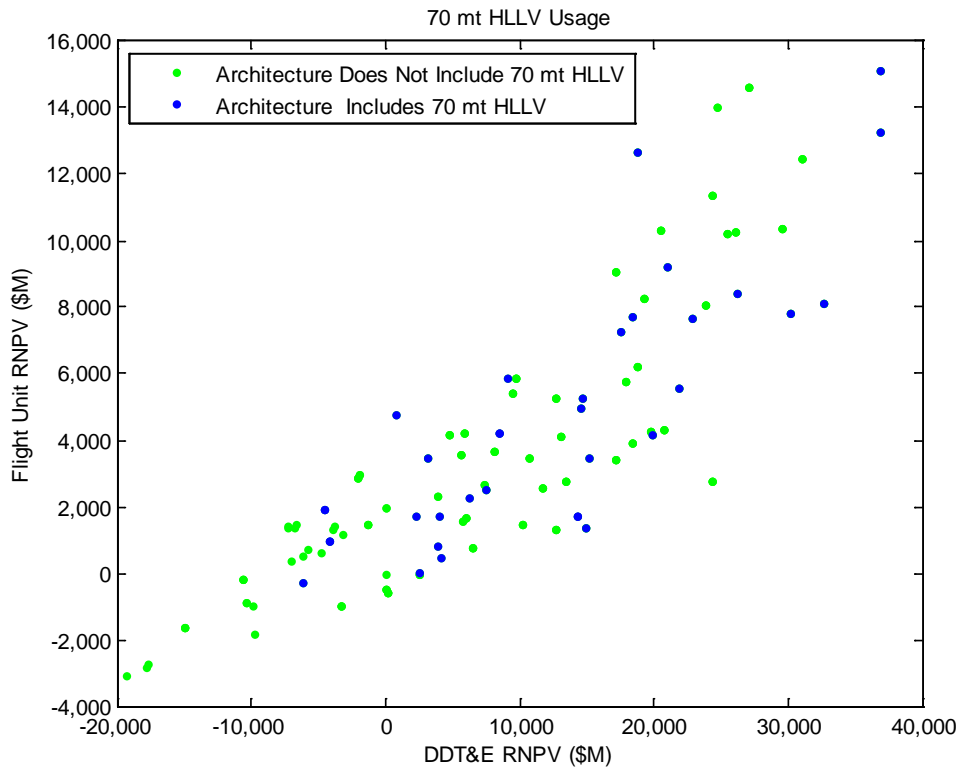


Figure C-5: RNPV of Lunar System Architectures that Utilize a Certain Launch Vehicle

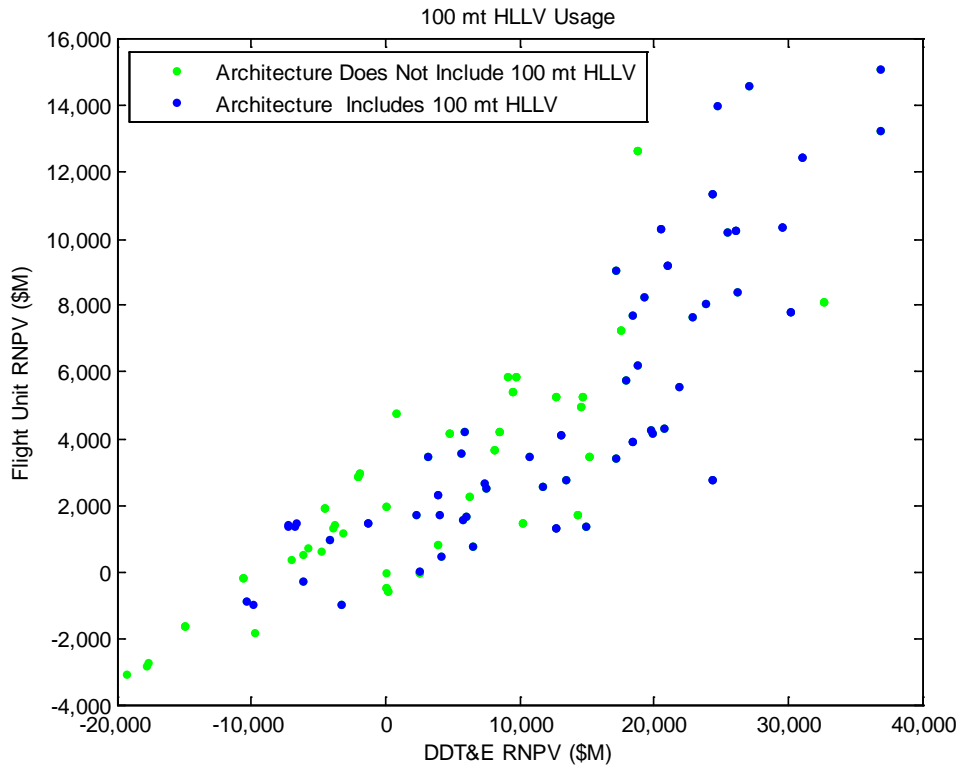


Figure C-6: RNPV of Lunar System Architectures that Utilize a Certain Launch Vehicle

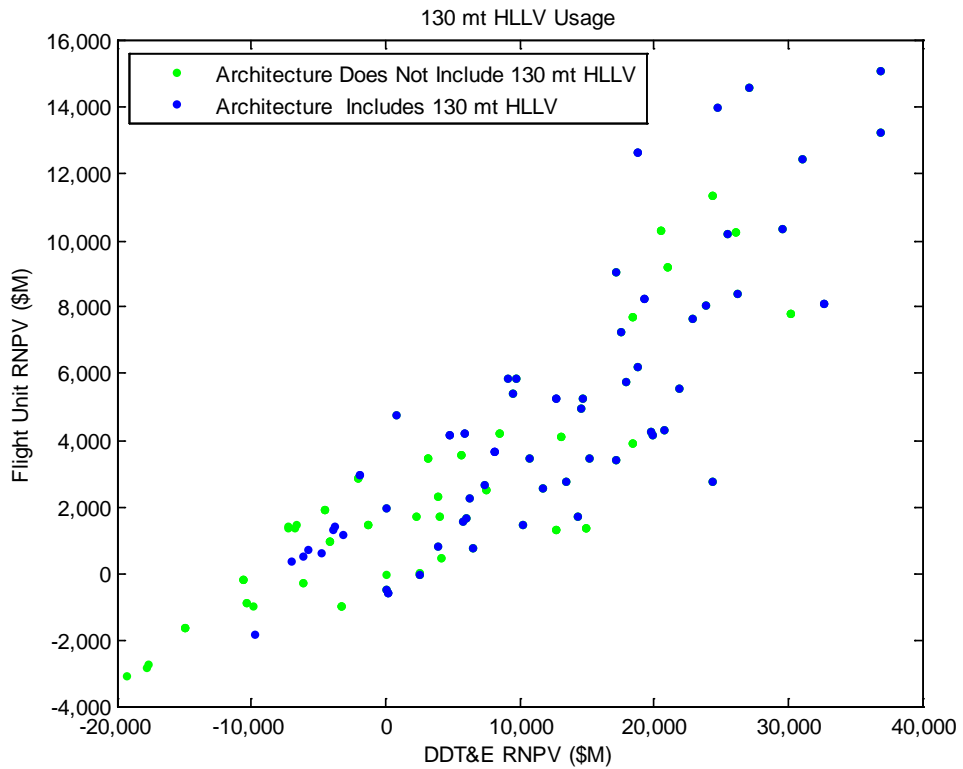


Figure C-7: RNPV of Lunar System Architectures that Utilize a Certain Launch Vehicle

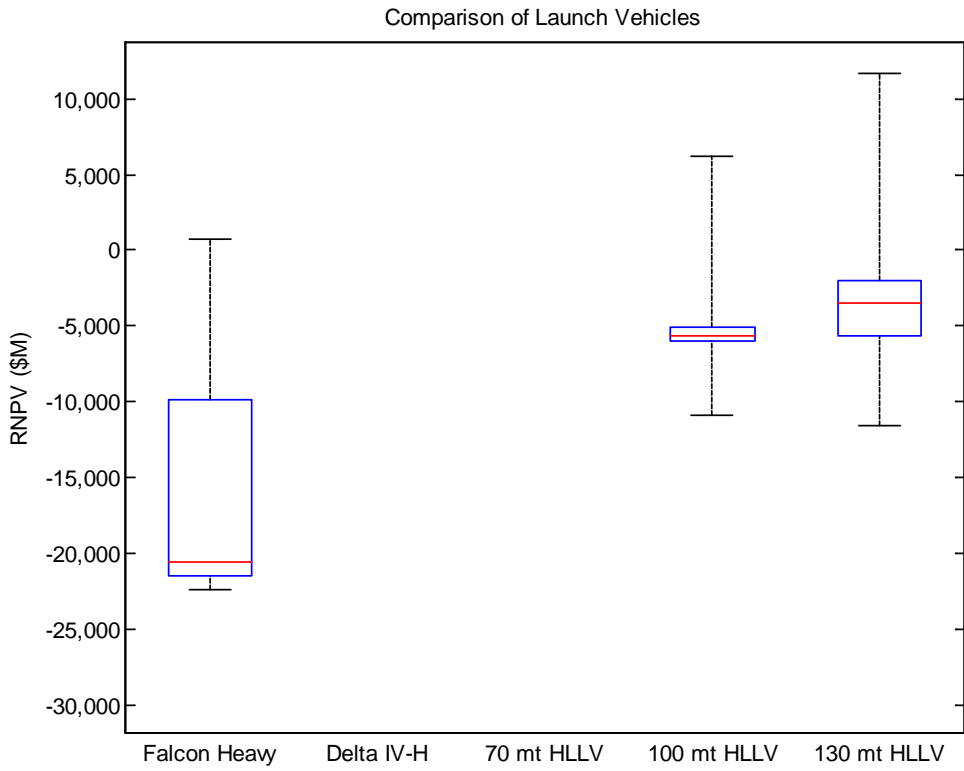


Figure C-8: Box and Whisker Plot of RNPV for Lunar System Architectures that Exclusively Use a Certain Launch Vehicle

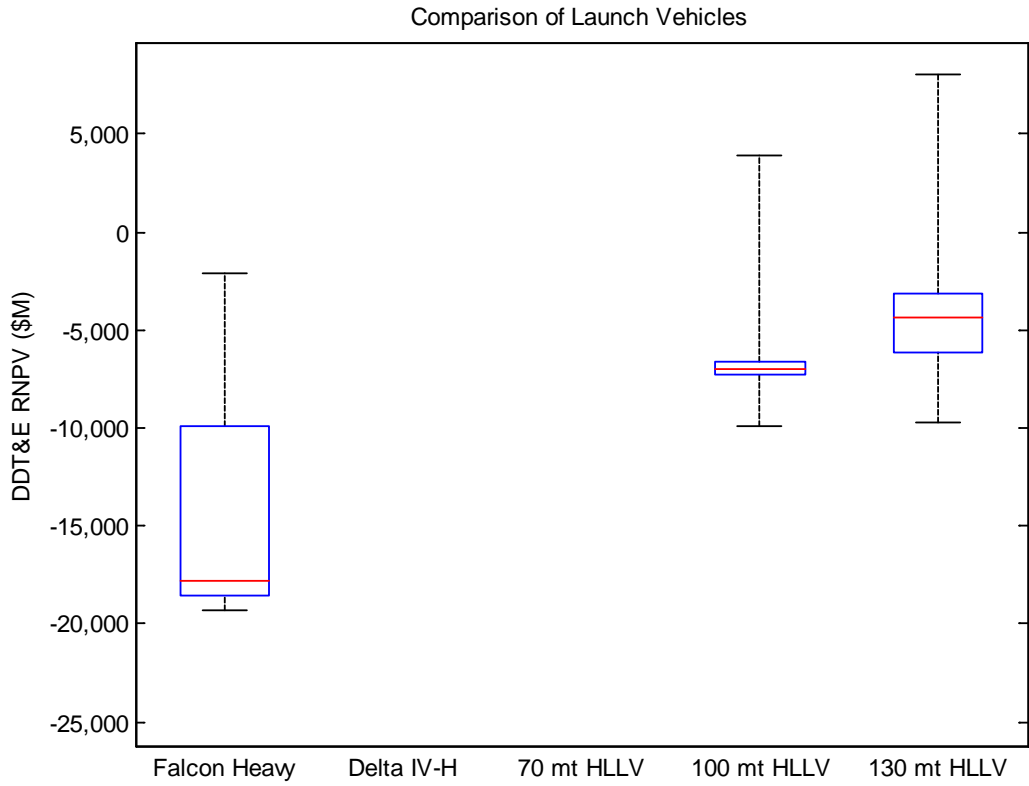


Figure C-9: Box and Whisker Plot of DDT&E RNPV for GEO System Architectures that Exclusively Use a Certain Launch Vehicle

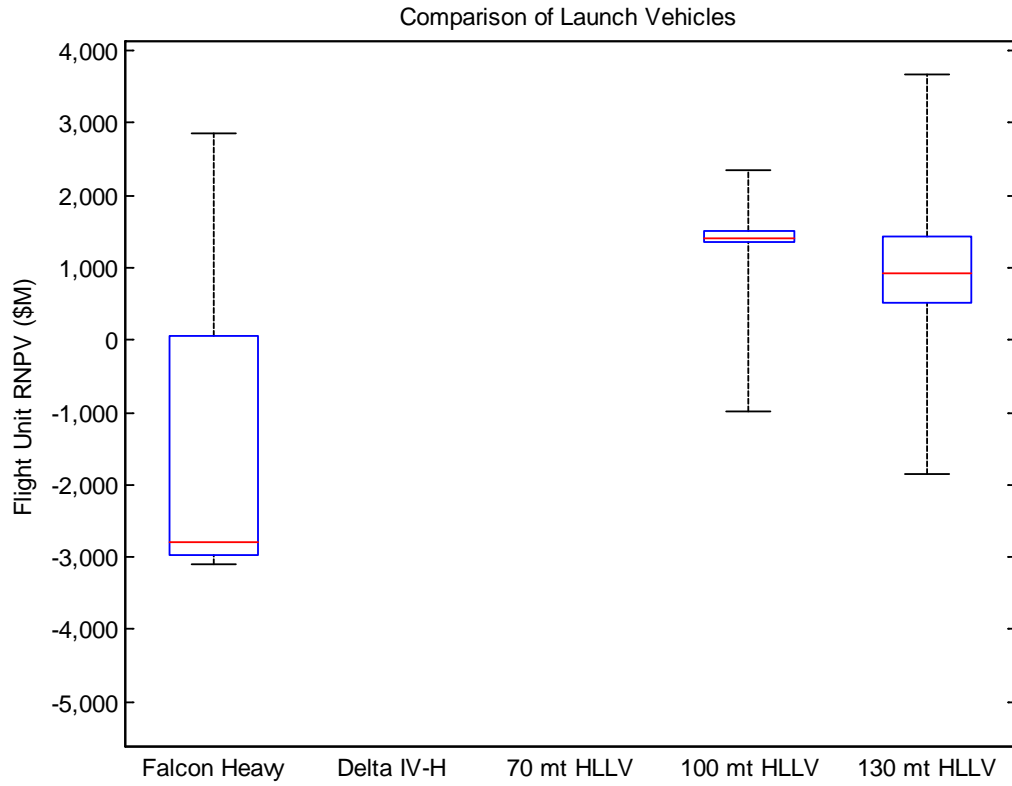


Figure C-10: Box and Whisker Plot of Flight Unit RNPV for GEO System Architectures that Exclusively Use a Certain Launch Vehicle

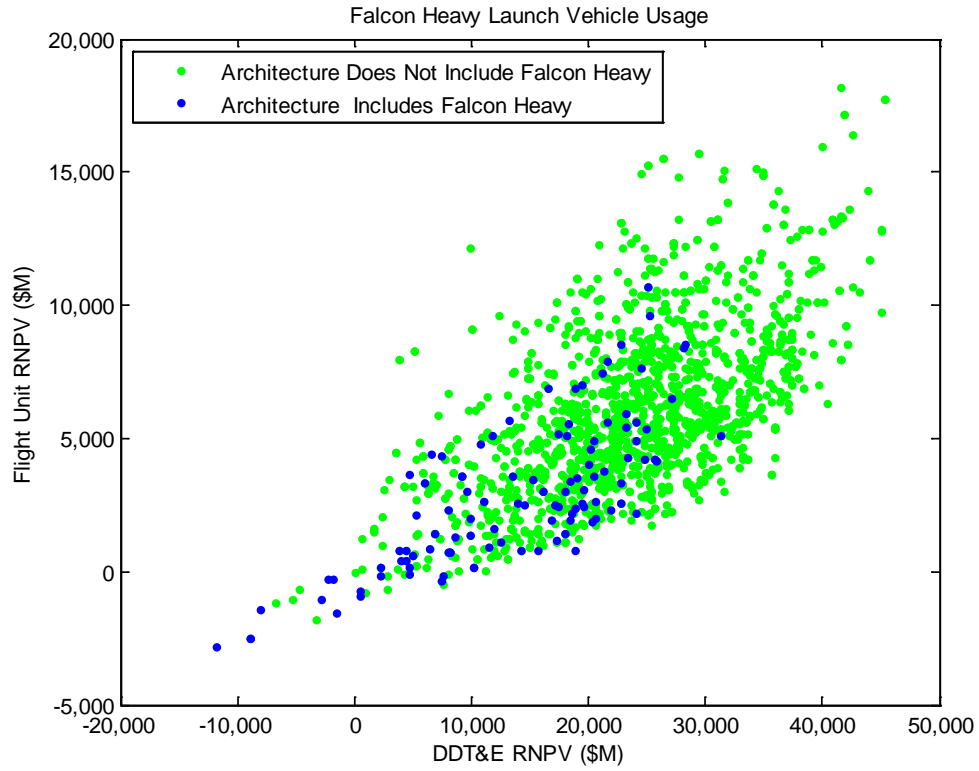


Figure C-11: RNPV of NEO System Architectures that Utilize a Certain Launch Vehicle

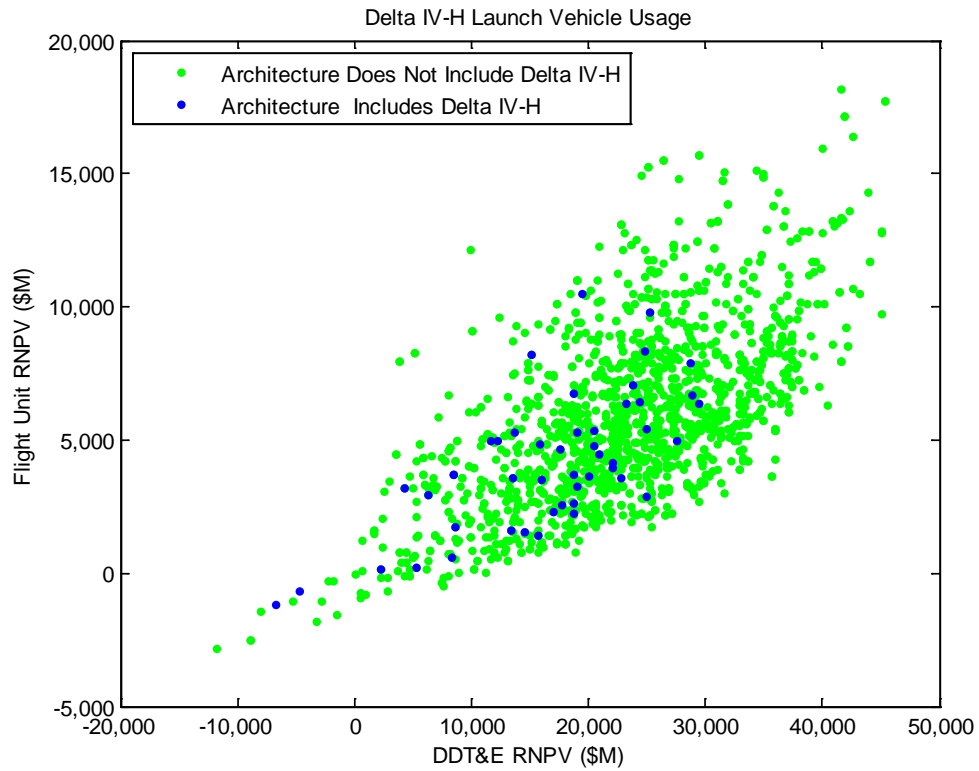


Figure C-12: RNPV of NEO System Architectures that Utilize a Certain Launch Vehicle

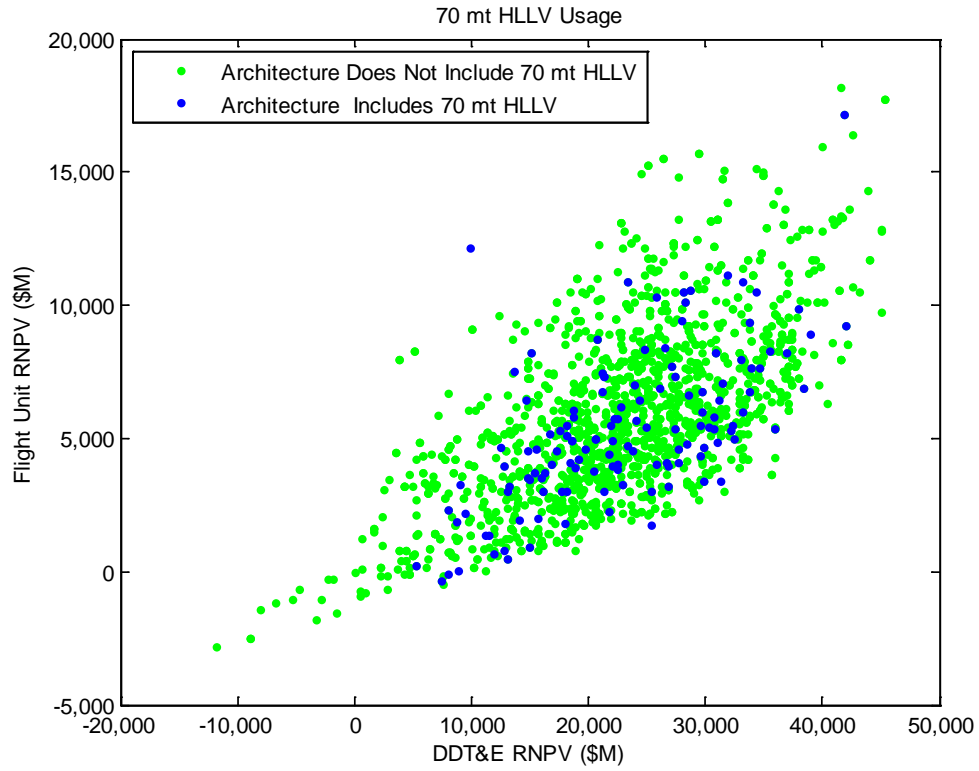


Figure C-13: RNPV of NEO System Architectures that Utilize a Certain Launch Vehicle

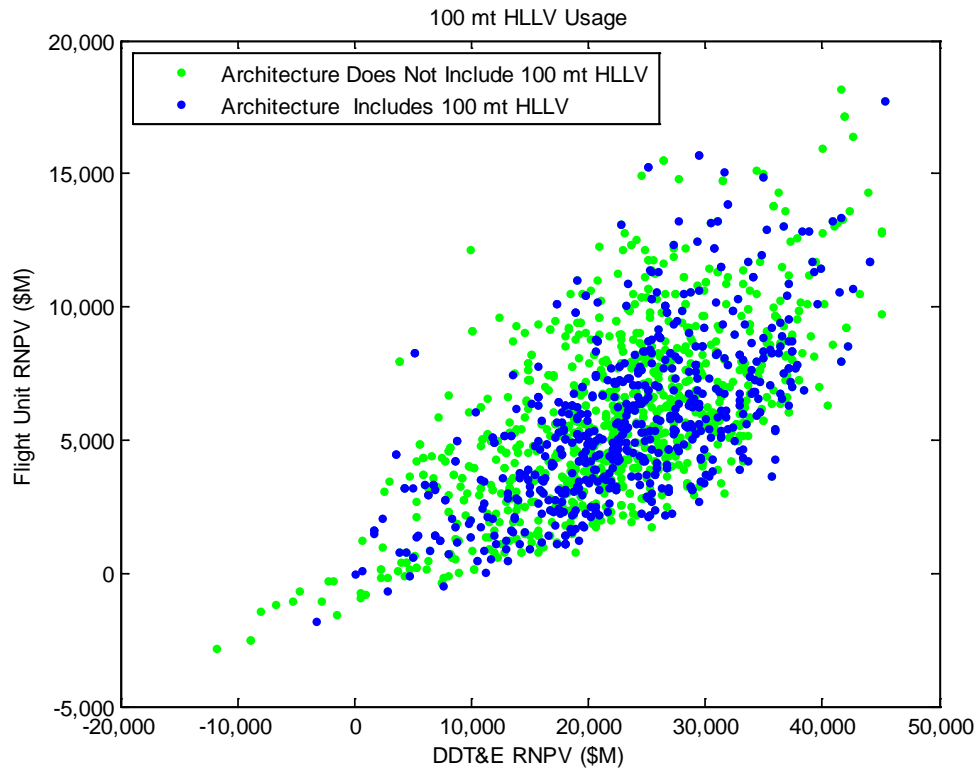


Figure C-14: RNPV of NEO System Architectures that Utilize a Certain Launch Vehicle

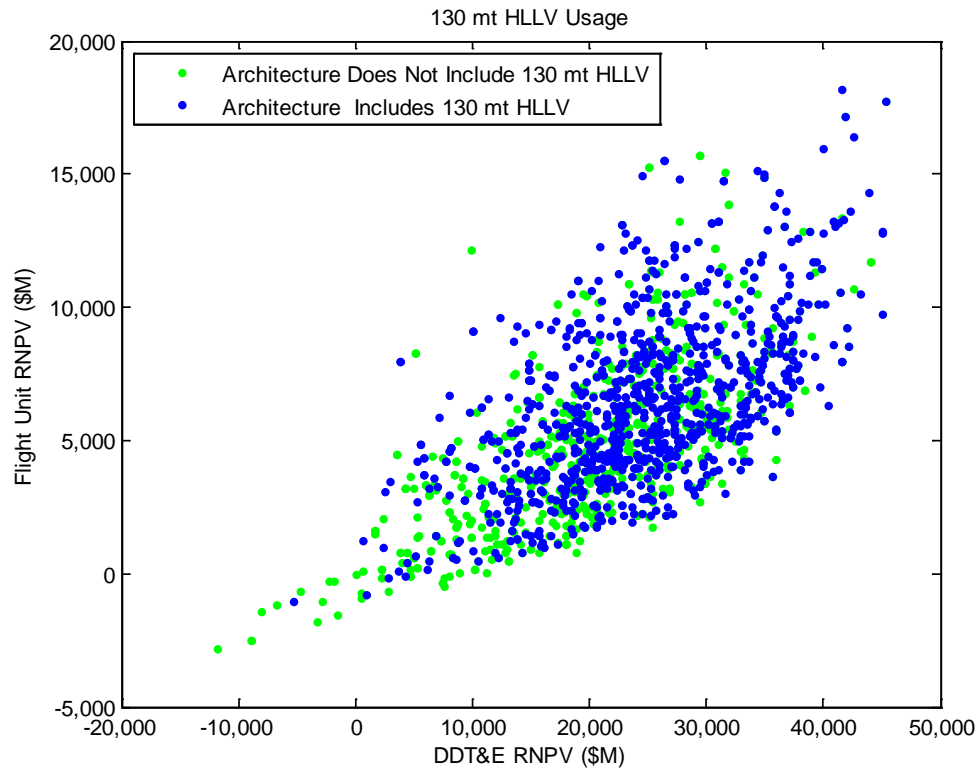


Figure C-15: RNPV of NEO System Architectures that Utilize a Certain Launch Vehicle

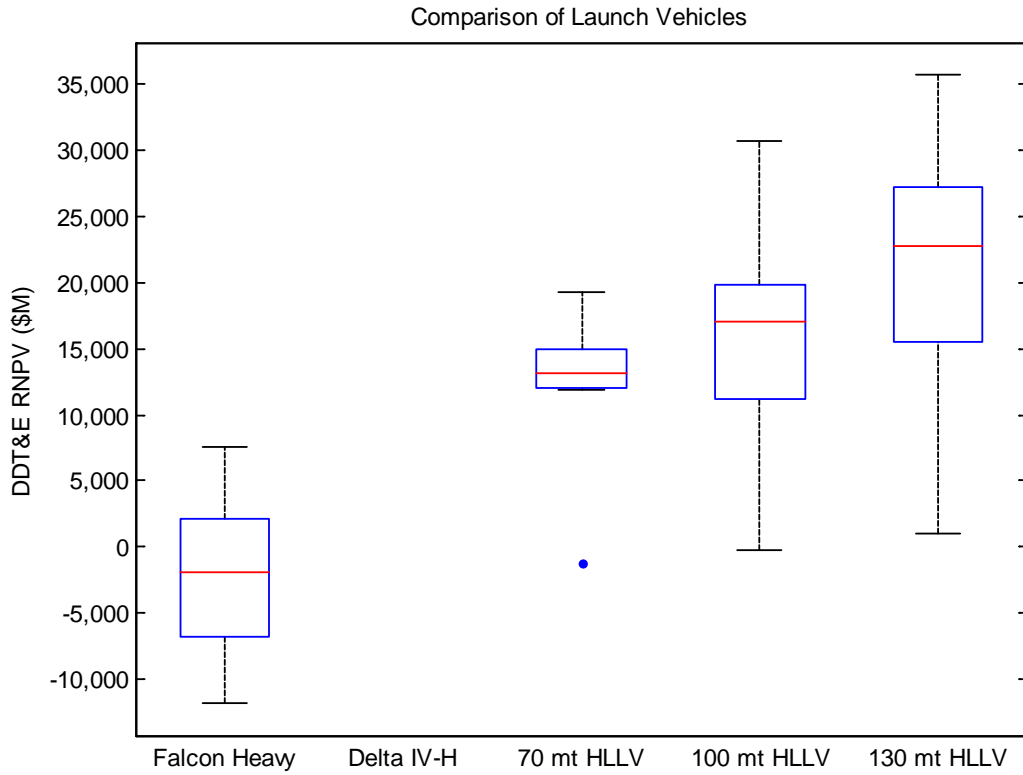


Figure C- 16: Box and Whisker Plot of DDT&E RNPV for NEO System Architectures that Exclusively Use a Certain Launch Vehicle

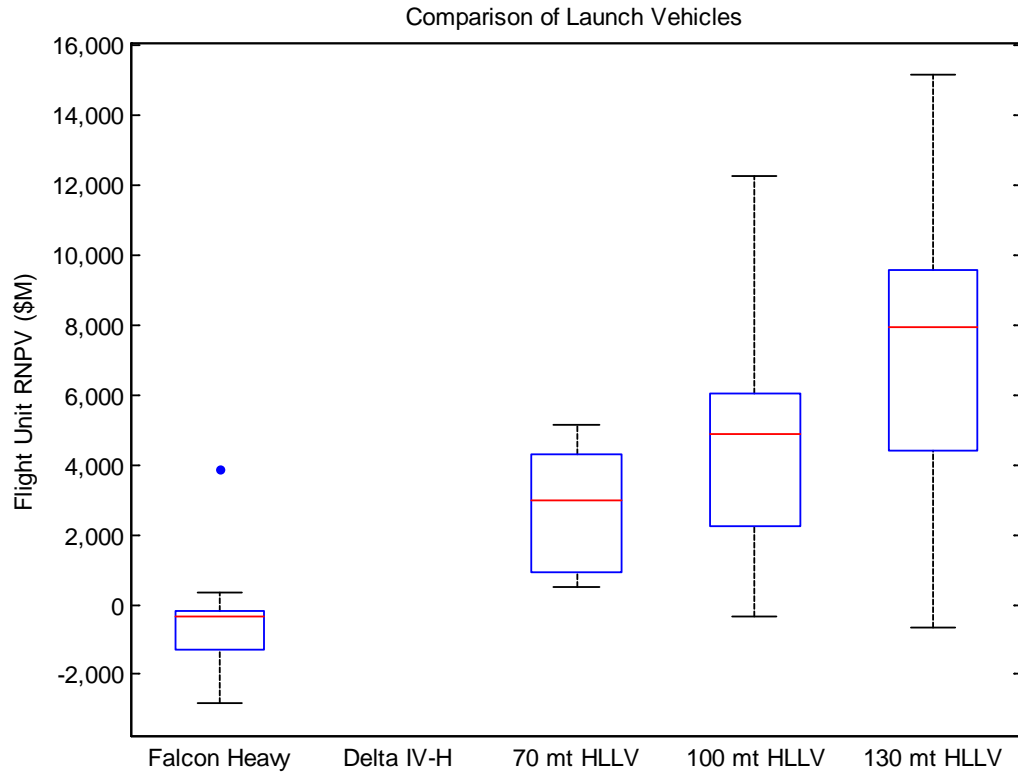


Figure C-17: Box and Whisker Plot of DDT&E RNPV for NEO System Architectures that Exclusively Use a Certain Launch Vehicle

C.2. Propellant Depot

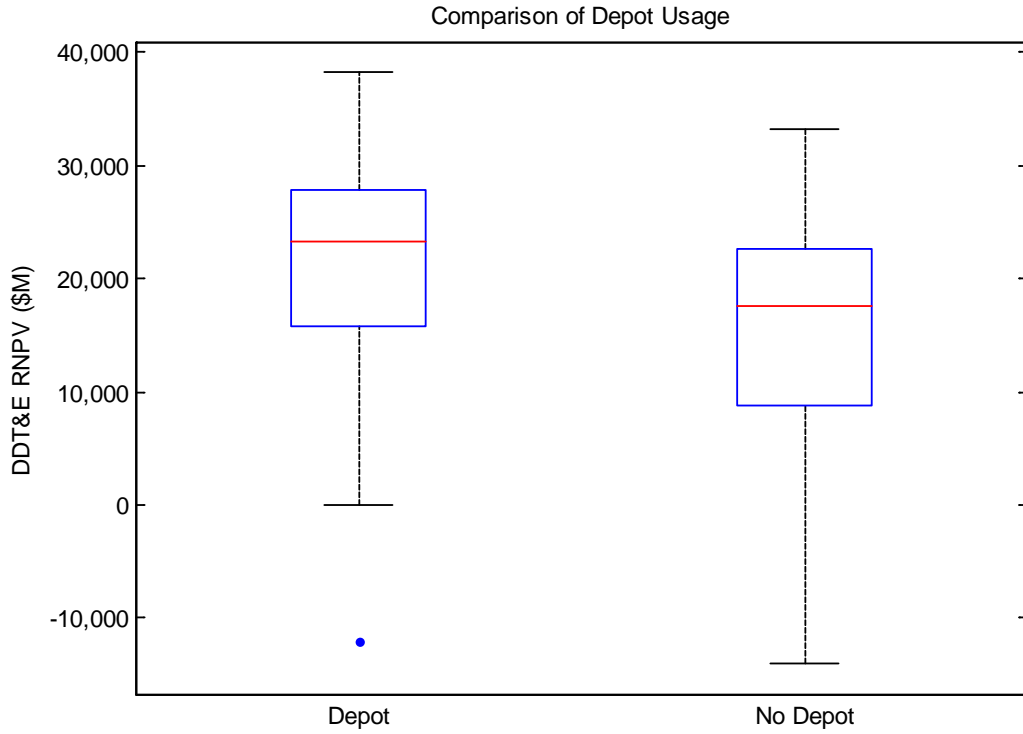


Figure C-18: Box and Whisker Plot of DDT&E RNPV for GEO System Architectures that Use a Propellant Depot

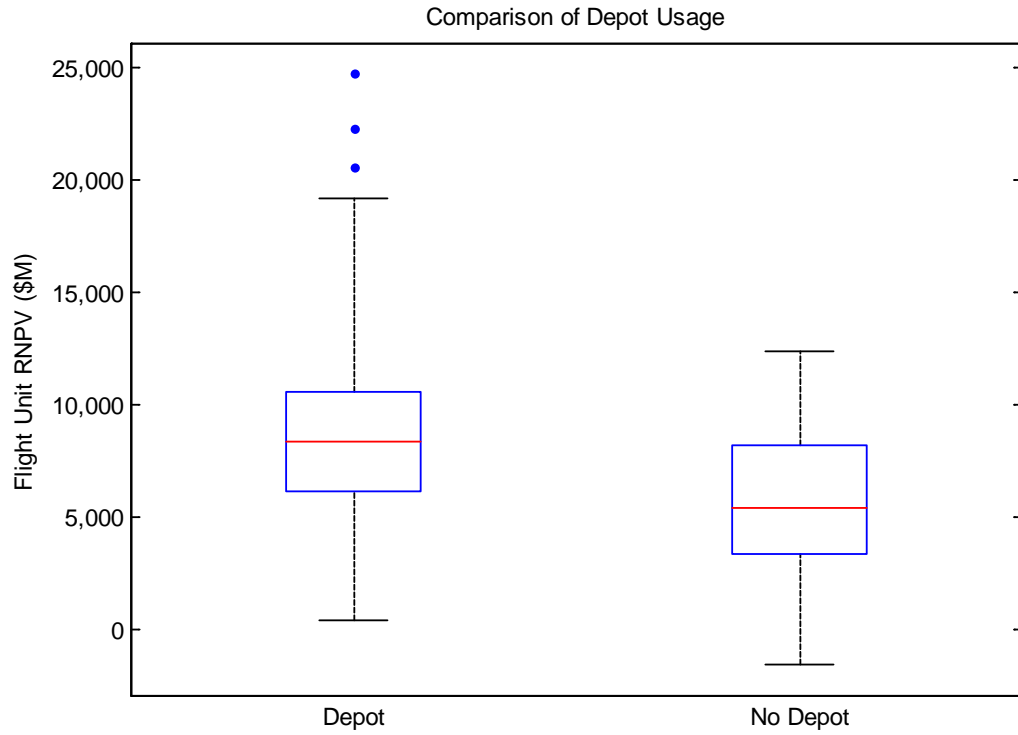


Figure C-19: Box and Whisker Plot of Flight Unit RNPV for GEO System Architectures that Use a Propellant Depot

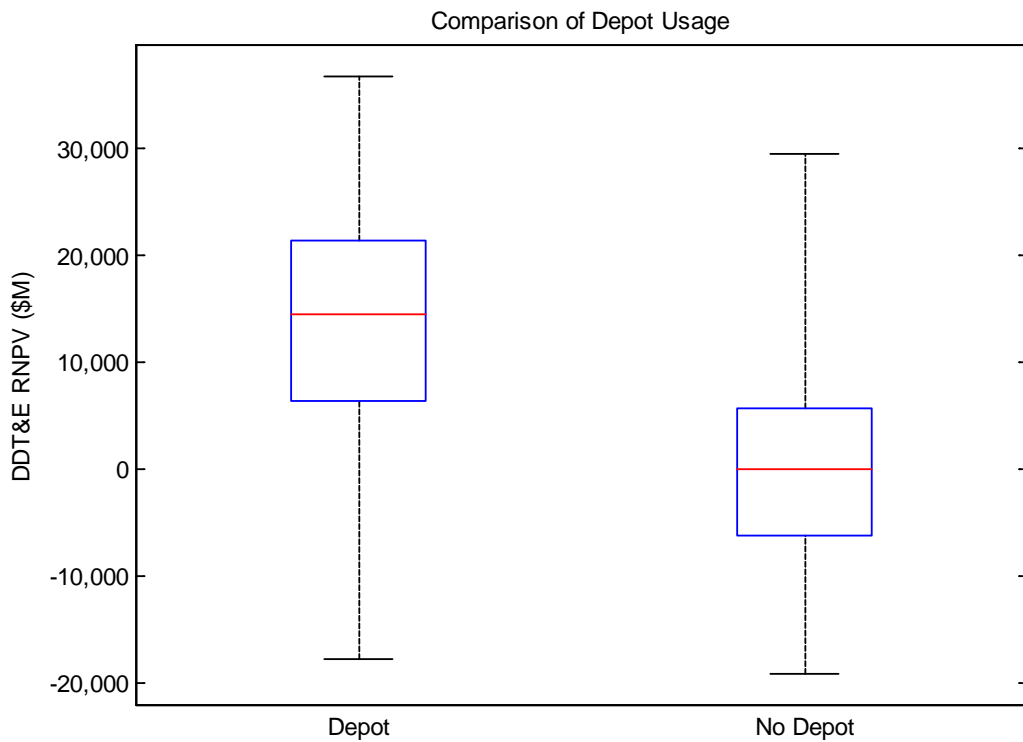


Figure C-20: Box and Whisker Plot of DDT&E RNPV for Lunar System Architectures that Use a Propellant Depot

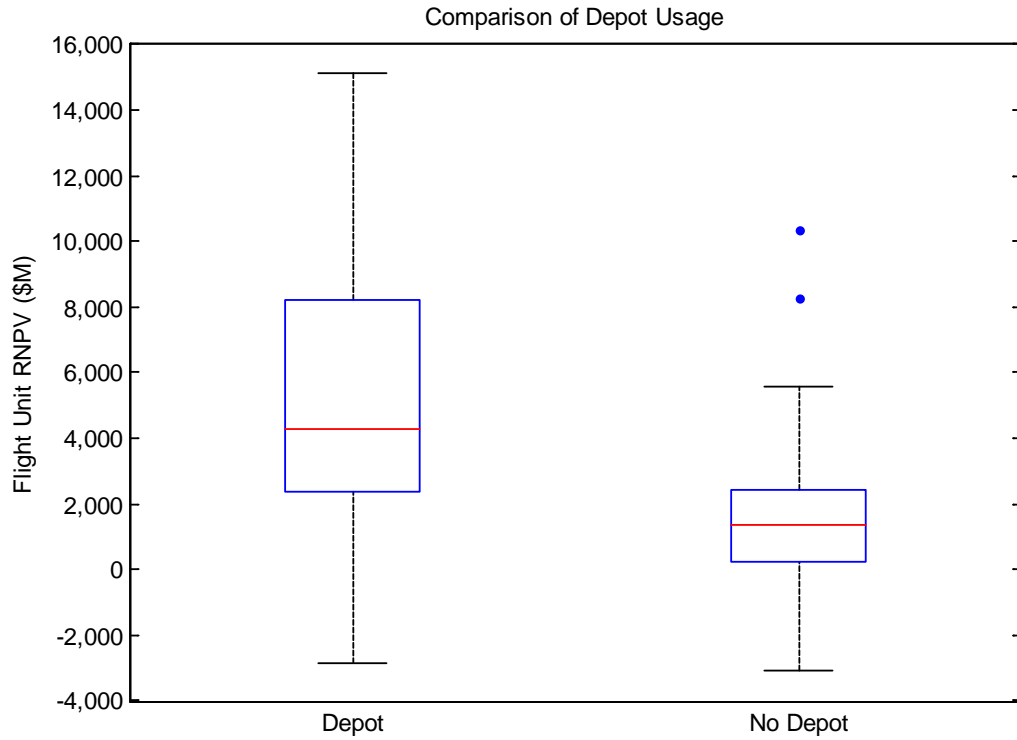


Figure C-21: Box and Whisker Plot of Flight Unit RNPV for Lunar System Architectures that Use a Propellant Depot

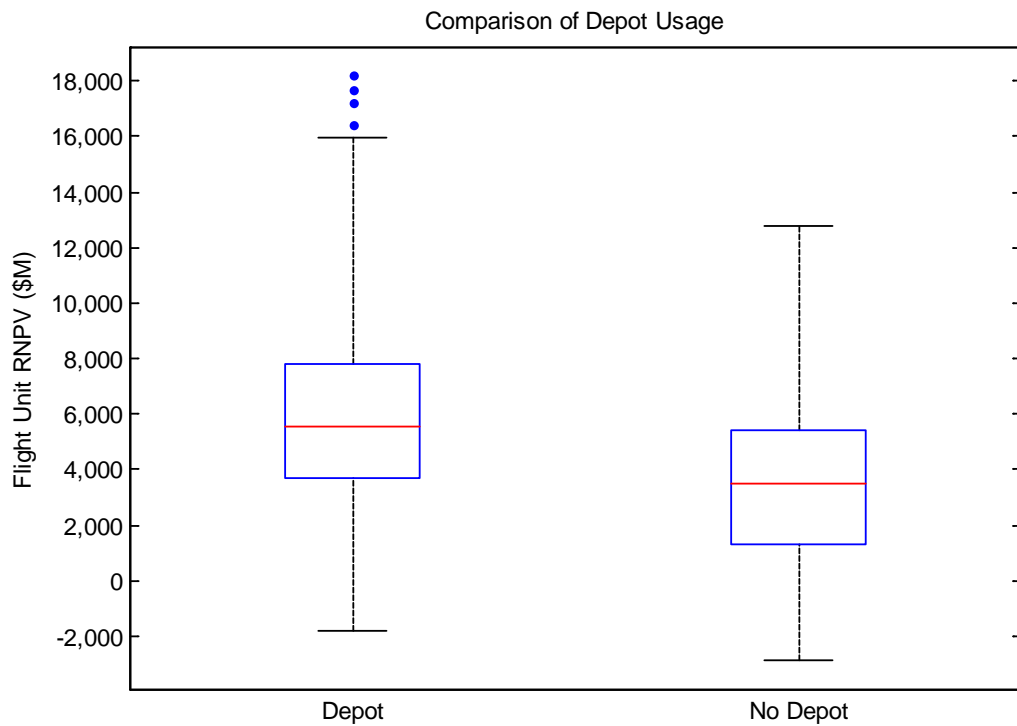


Figure C-22: Box and Whisker Plot of Flight Unit RNPV for NEO System Architectures that Use a Propellant Depot

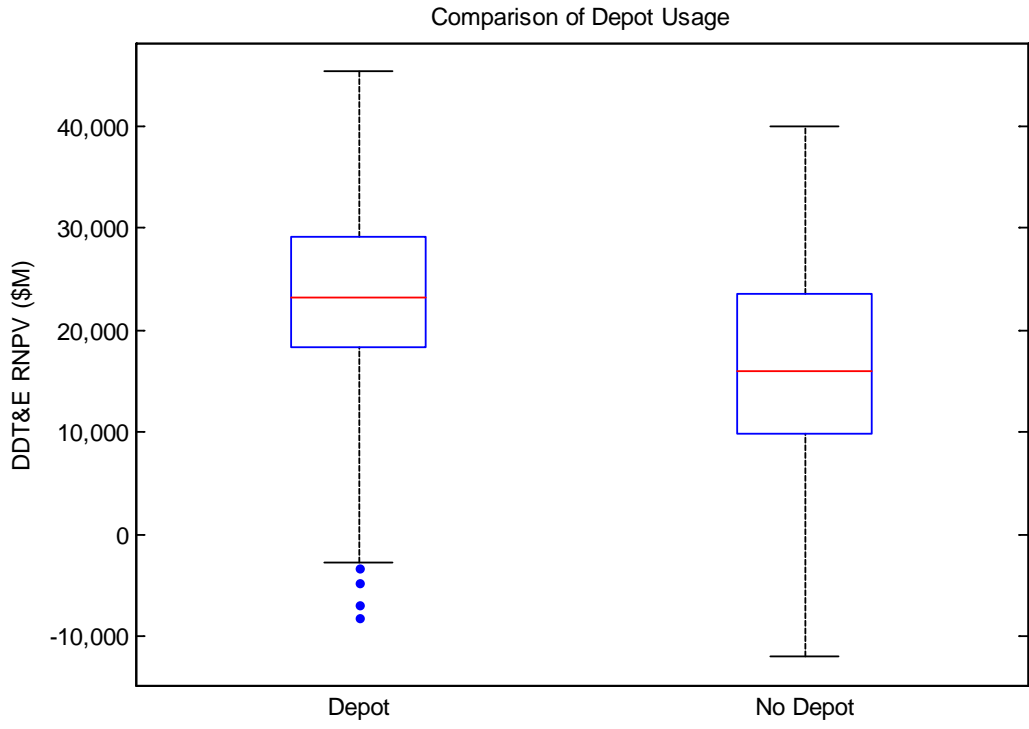


Figure C-23: Box and Whisker Plot of DDT&E RNPV for NEO System Architectures that Use a Propellant Depot

C.3. Aggregation Strategy

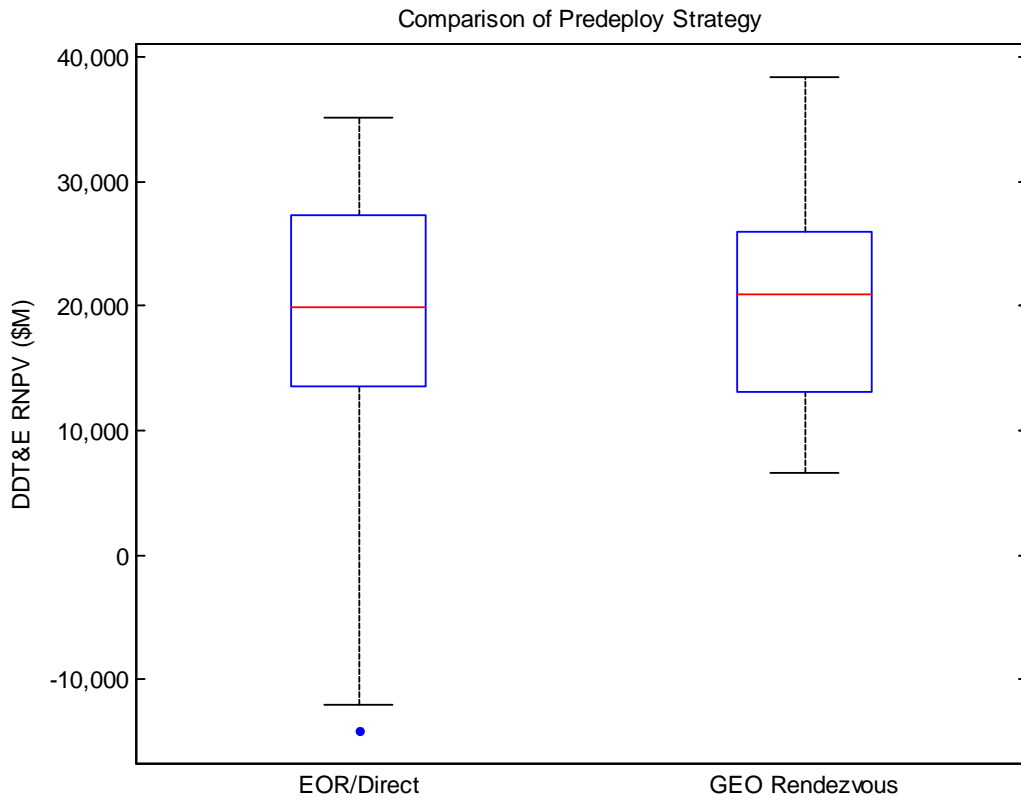


Figure C-24: Box and Whisker Plot of DDT&E RNPV for GEO System Architectures that Use Different Aggregation Strategies

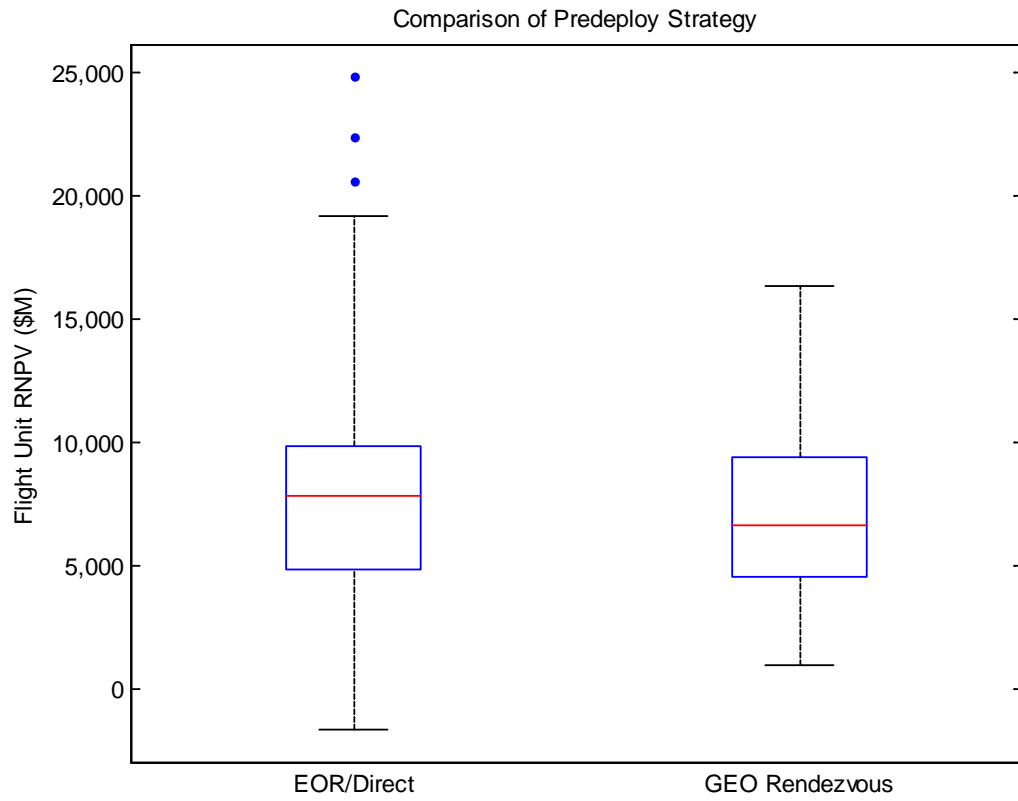


Figure C-25: Box and Whisker Plot of Flight Unit RNPV for GEO System Architectures that Use Different Aggregation Strategies

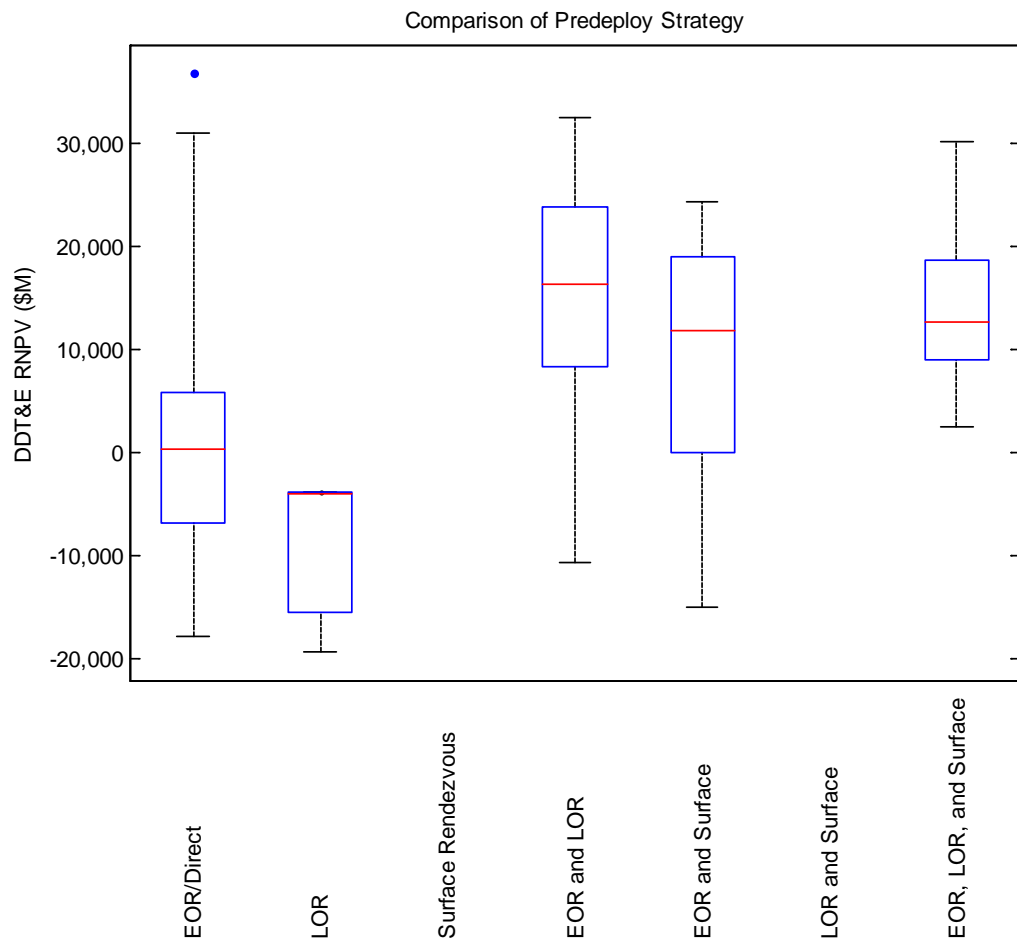


Figure C- 26: Box and Whisker Plot of DDT&E RNPV for Lunar System Architectures that Use Different Aggregation Strategies

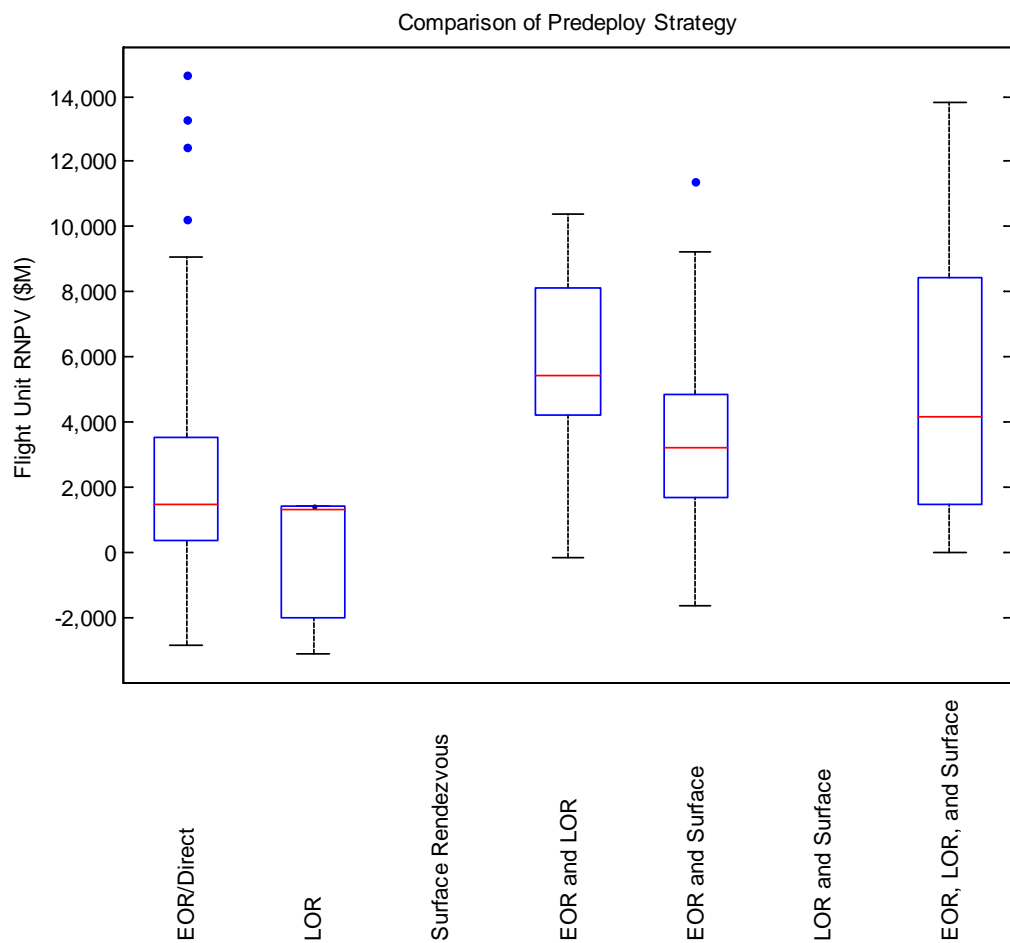


Figure C-27: Box and Whisker Plot of Flight Unit RNPV for Lunar System Architectures that Use Different Aggregation Strategies

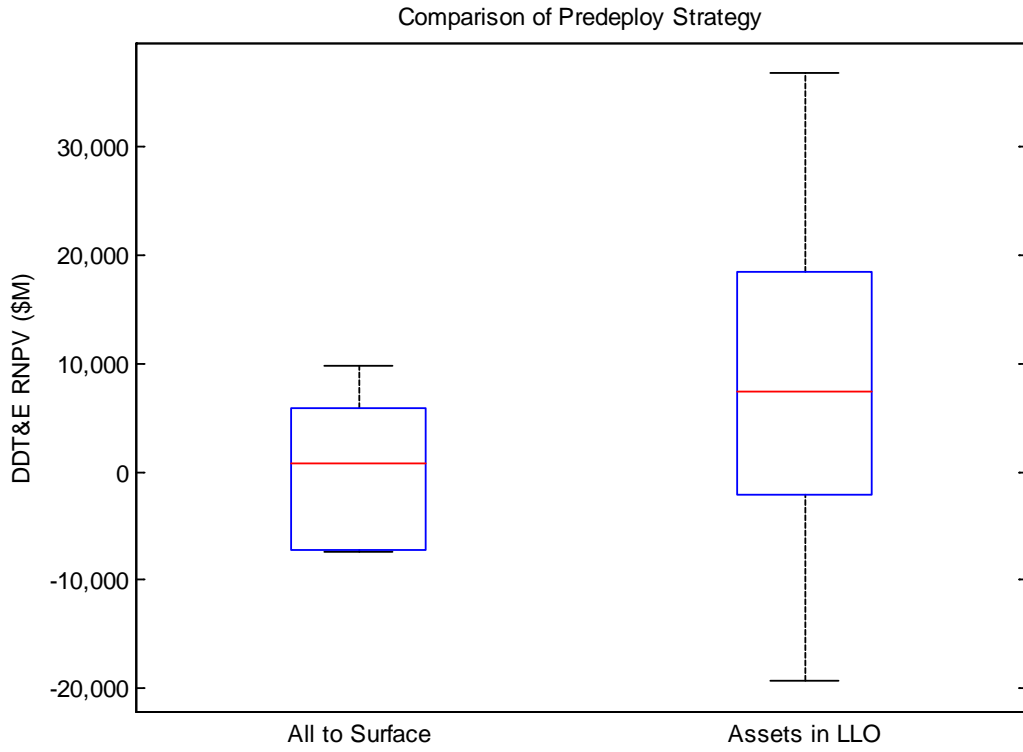


Figure C-28: Box and Whisker Plot of DDT&E RNPV for Lunar System Architectures for Location of Assets during Surface Mission

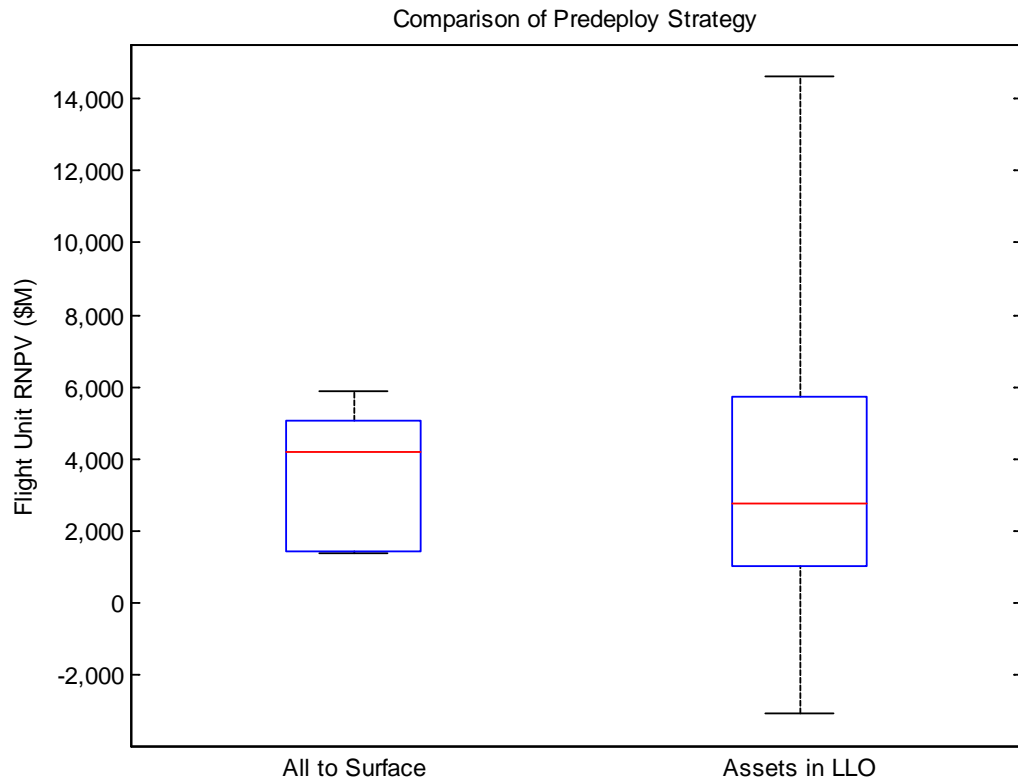


Figure C-29: Box and Whisker Plot of Flight Unit RNPV for Lunar System Architectures for Location of Assets during Surface Mission

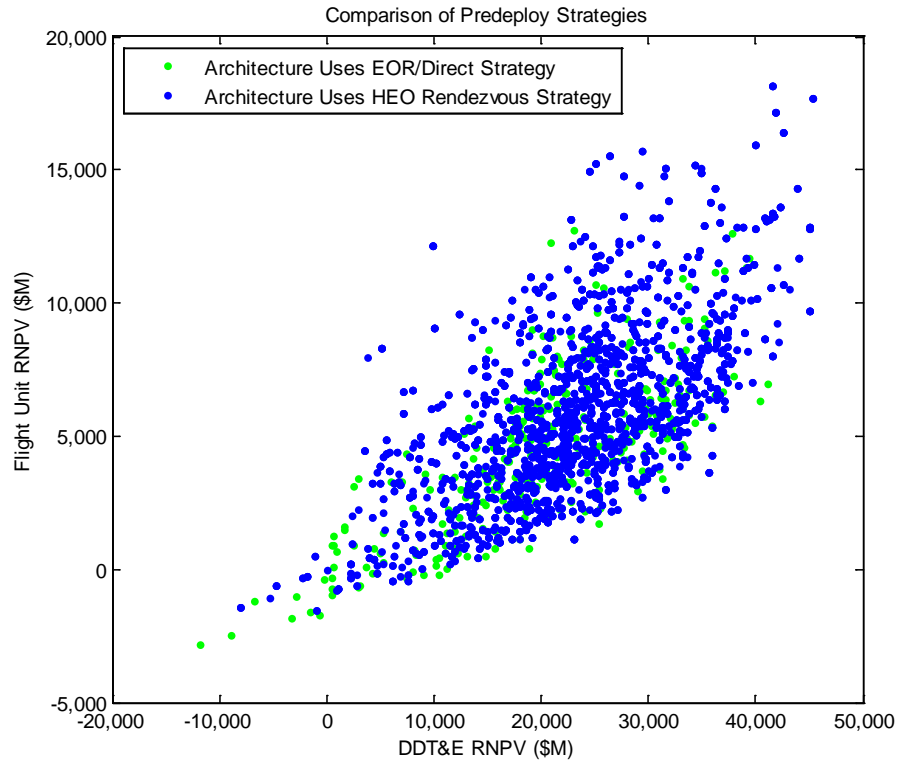


Figure C-30: RNPV of NEO System Architectures for Different Aggregation Strategies

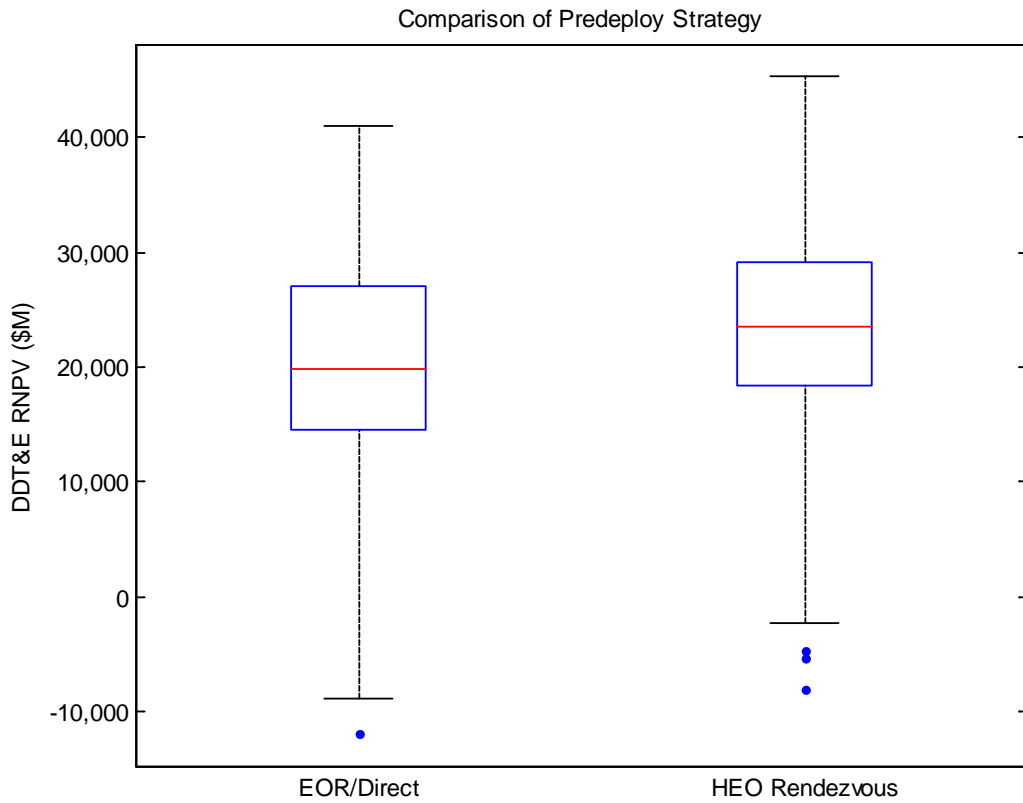


Figure C-31: Box and Whisker Plot of DDT&E RNPV for NEO System Architectures that Use Different Aggregation Strategies

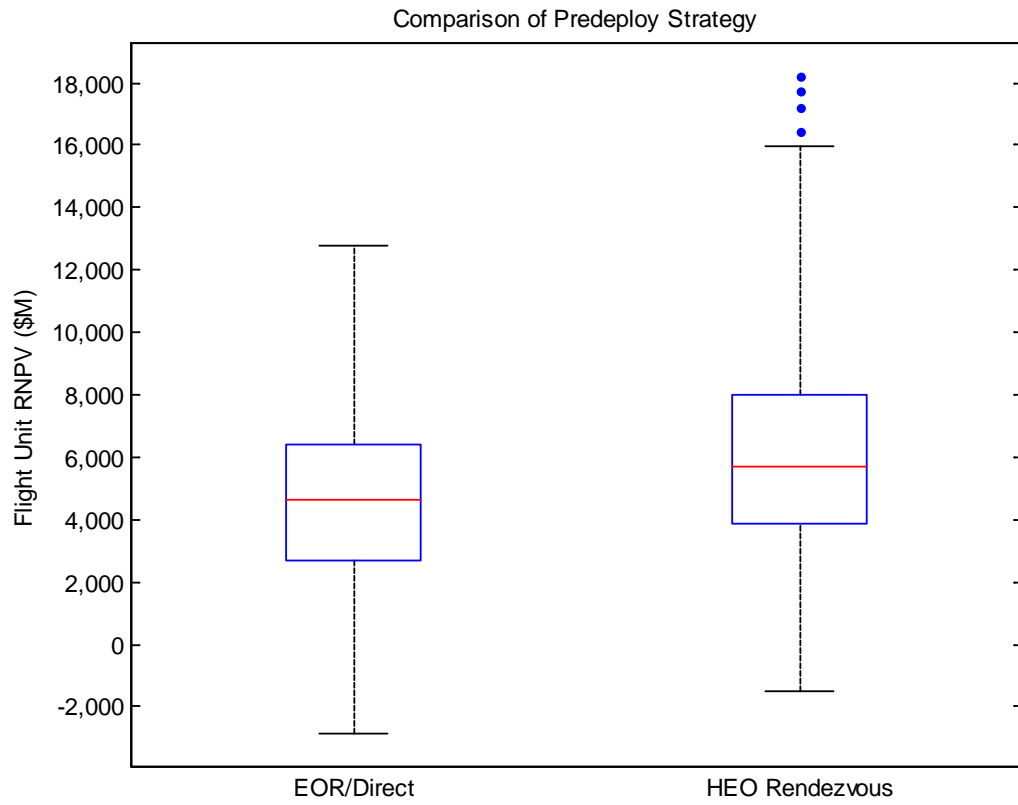


Figure C-32: Box and Whisker Plot of Flight Unit RNPV for NEO System Architectures that Use Different Aggregation Strategies

APPENDIX D

This appendix contains details on the sizing methods used for each of the individual system types. The system types are as follows:

- D.1. Crew
- D.2. Crew Capsule
- D.3. Lunar Descent Stage
- D.4. Lunar Ascent Stage
- D.5. Launch Vehicle
- D.6. Propulsive Stage
- D.7. Propellant Depot
- D.8. Surface Habitat
- D.9. In-Space Habitat

D.1. Crew

The crew system assumes that each crewmember weighs 93 kg, including clothing and other personal items.

D.2. Crew Capsule

The crew capsule model uses a photographically scaled version of the Block 2 Lunar Crew Exploration Vehicle presented in ESAS [7]. This system was volumetrically sized to accommodate a crew of six for a mission to the ISS, but the crew accommodations and life support consumables are for a crew of four. Also, the thermal protection is designed to accommodate lunar reentry velocity. The geometry for this vehicle is presented in Figure D-1.

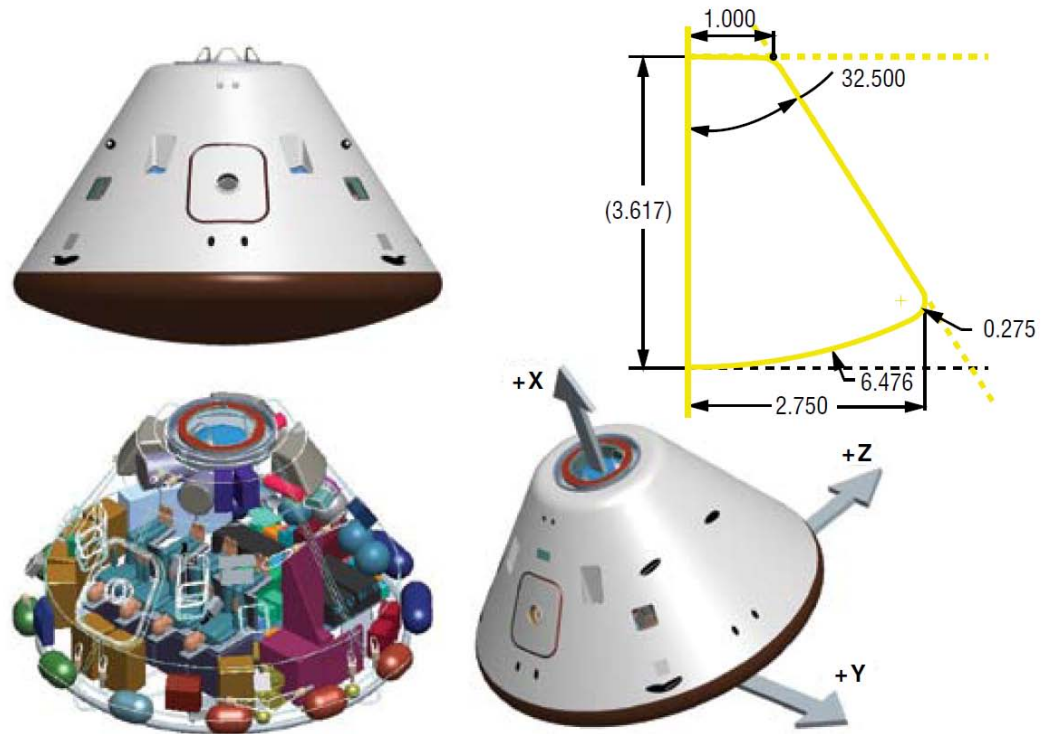


Figure D-1: Configuration of Block 2 Lunar Crew Exploration Vehicle [7]

This vehicle is photographically scaled to accommodate the number of crew for a given mission. Figure D-2 presents a screenshot of the Excel spreadsheet that performs the photographic scaling for this system, where changes in number of crew affect the volume and area ratio parameters, and changes in system gross mass affect the reaction control propellant load.

	Original (kg)	Modified (kg)	Sizing
Structure	1,882	1,330	
<i>Pressure Vessel, Windows</i>	1,105	737	Volume
<i>Heatshield Substructure</i>	777	593	Area
Protection (TPS)	894	682	Area
Propulsion	414	359	
<i>RCS Tanks, Lines, Pressurization</i>	266	231	delta-V
<i>RCS Thrusters + Installation</i>	148	128	Gross Mass Ratio
Power	819	819	Constant
Control	0	0	
Avionics	435	435	Constant
Environmental	1,089	972	
<i>ECLSS</i>	462	462	# Crew Ratio
<i>Active Thermal Control</i>	352	235	Volume
<i>Crew Accommodations</i>	275	275	# Crew
Other	1,160	1,067	
<i>Terminal Descent, Misc.</i>	703	610	Gross Mass Ratio
<i>LIDS Docking Mechanism</i>	457	457	Constant
Growth (20% to each subsystem)	1,339	1,133	
Dry Mass	8,032	6,796	
Non-Cargo	822	822	
<i>Personnel</i>	400	400	# Crew
<i>Crew Provisions</i>	189	189	# Crew
<i>Operational Supplies</i>	133	133	# Crew
<i>Food</i>	96	96	# Crew
<i>Residuals</i>	4	4	# Crew
Cargo (Ballast)	100	100	
Non-Propellant	367	367	
Inert Mass	9,321	8,085	
Propellant	184	160	
Gross Mass	9,505	8,245	
Baseline			
Number of Crew (volume)	6		
Number of Crew (accomm., ECLSS,	4		
RCS DV (m/s)	67.11825323		
Modified			
Number of Crew	4		
Volume Ratio	0.666666667		
Area Ratio	0.763142828		
Crew Load Ratio	1		
Prop Mass Ratio	0.86741022		
Gross Mass Ratio	0.86741022		

Figure D-2: Screenshot of Crew Capsule Sizing Spreadsheet

D.3. Lunar Descent Stage

The lunar descent stage model uses photographic scaling of two separate vehicles depending on the propellant type. Descent stages that use cryogenic propellant, such as LOX/LH2 or LOX/CH4, are based on the descent stage for the Lunar Surface Access Module (LSAM) presented in ESAS [7]. The geometry for this vehicle is presented in Figure D-3. The regression for inert mass fraction as a function of the stage gross mass that is used to predict the inert mass for a given propellant demand (as calculated by the rocket equation) is presented in Figure D-4.



Figure D-3: Configuration of ESAS LSAM Cryogenic Descent Stage [7]

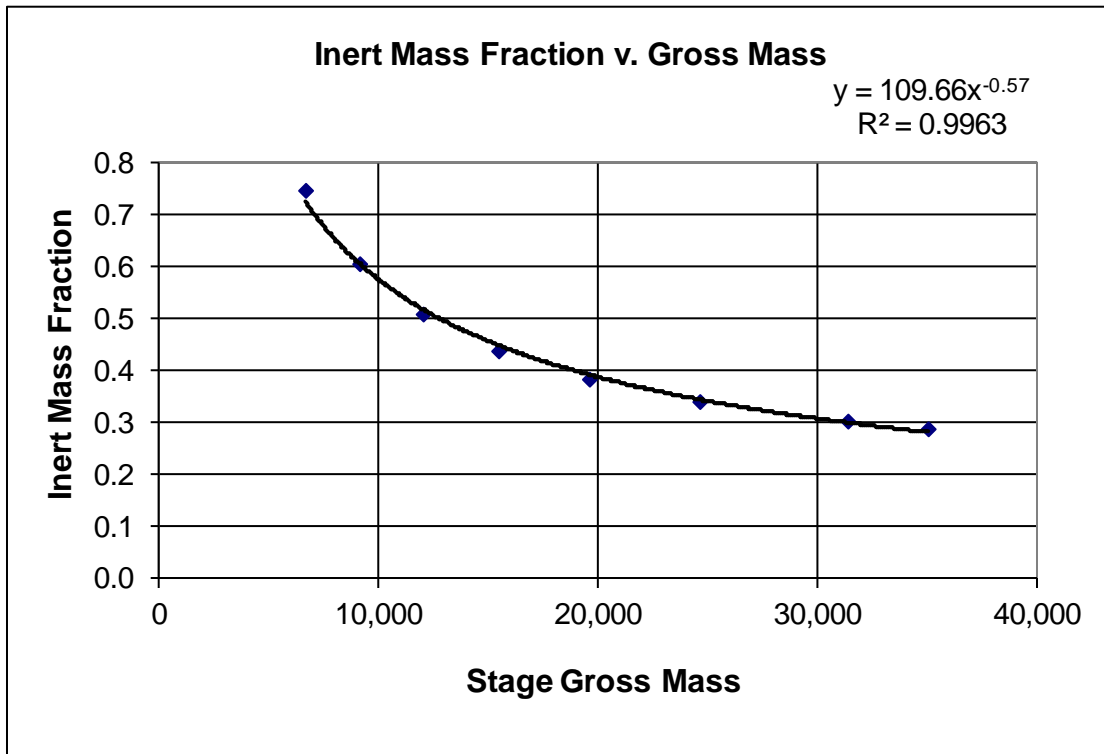


Figure D-4: Inert Mass Fraction Estimation for a Cryogenic Lunar Descent Stage

Descent stages that use storable propellant, such as NTO/MMH and LOX/RP-1, are based on the descent stage for the Apollo Lunar Excursion Module [40]. The geometry for this vehicle is presented in Figure D-5. The regression for inert mass fraction as a function of the stage gross mass that is used to predict the inert mass for a given propellant demand (calculated by the rocket equation) is presented in Figure D-6.

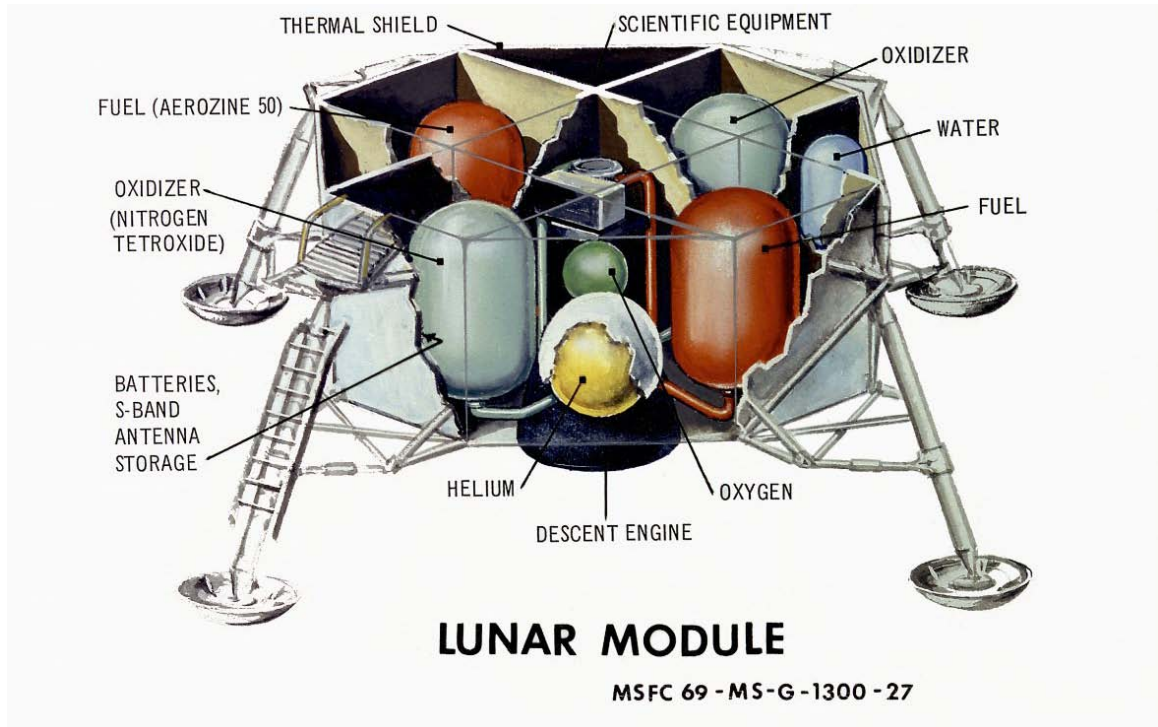


Figure D-5: Configuration of Apollo Lunar Excursion Module Hypergolic Descent Stage (Image: NASA)

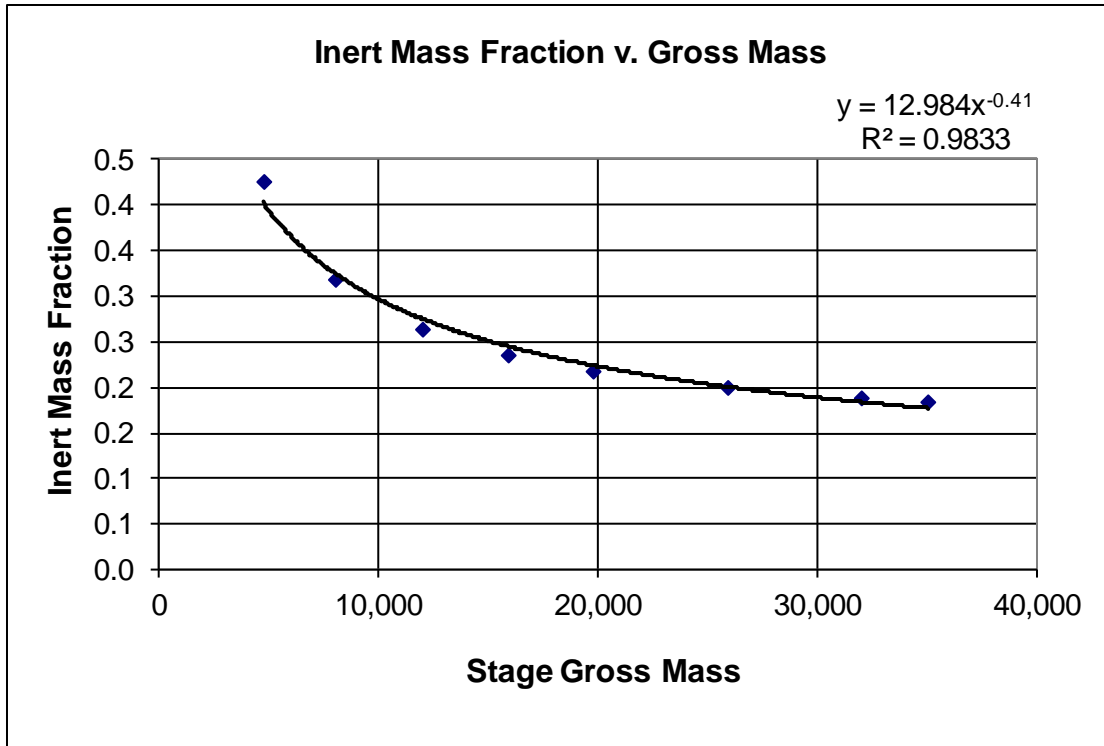


Figure D-6: Inert Mass Fraction Estimation for a Storable Lunar Descent Stage

D.4. Lunar Ascent Stage

The lunar ascent stage model uses a photographically scaled model of the propulsive elements of the LSAM ascent stage, including tanks, tank support, insulation, and power. Figure D-7 presents the full LSAM ascent stage including the surface habitat, which is excluded from the ascent stage analysis. Figure D-8 presents a screenshot of the Excel spreadsheet that performs the photographic scaling for this system, where changes in performance requirements (ΔV , T/W, and payload mass) and system implementation (propellant type) affect the sizing of the structure, protection, and propulsive system masses.

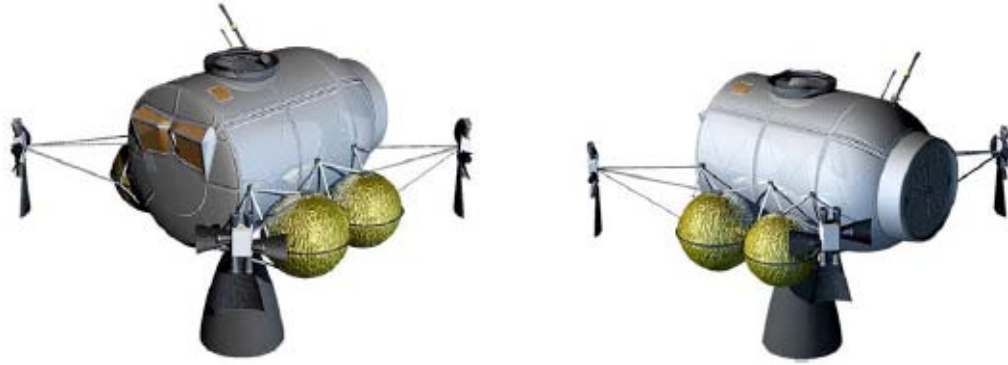


Figure D-7: Configuration of ESAS LSAM Lunar Ascent Stage and Surface Habitat [7]

INPUTS		MASS BREAKDOWN			
Required DV 1	1968 m/s	Subsystem	Lvl 1 Mass	Lvl 2 Mass	Sizing
Required DV 2	0 m/s	Structure	36		
Required DV 3	0 m/s	Tank Support		36	Volume
Required DV 4	0 m/s	Protection (insulation)	98		Area
Required DV 5	0 m/s	Propulsion	667		
Payload Mass 1	5067.2 kg	Propellant Tanks, Lines, etc.		524	Volume
Payload Mass 2	0 kg	OMS Engine		44	T/W
Payload Mass 3	0 kg	RCS Thrusters		99	Gross Mass
Payload Mass 4	0 kg	Power	0		
Payload Mass 5	0 kg	Control	0		
System T/W	1.97	Avionics	0		
Engine Type	NTO/MMH	Growth (20% to each subsystem)	160		
Planet	Moon	Dry Mass	961		
PHOTOGRAPHIC SCALING		Non-Propellant	284		Volume
Gross Mass Ratio	0.640079	Inert Mass	1,245		
Bulk Density	1225.254	Propellant	5,672		Volume
Density Ratio	1.482229	Gross Mass	6,917		
Volume Ratio	0.811659	ENGINE SPECS			
Area Ratio	0.870126	Engine Type	Isp	T/m	O/F
Required Thrust	38483.54 N	LOX/LH2	443	596	6
BULK DENSITY		LOX/CH4	353	473.40426	3.6
LOX	1141 kg/m ³	LOX/RP-1	337	784	2.6
LH2	71 kg/m ³	NTO/MMH	313	869	2.6
CH4	415 kg/m ³				
NTO	1443 kg/m ³				
RP-1	820 kg/m ³				
MMH	880 kg/m ³				
LOX/LH2	361.887 kg/m ³				
LOX/CH4	826.6296 kg/m ³				
LOX/RP-1	1029.096 kg/m ³				
NTO/MMH	1225.254 kg/m ³				

Figure D-8: Screenshot of Lunar Ascent Stage Sizing Spreadsheet

D.5. Launch Vehicle

Unlike the other system types used in this analysis, the launch vehicle systems were not sized to meet a certain performance requirement that was embedded in an edge. The analysis used existing or planned launch vehicles with defined payload capabilities and packaged the flight hardware systems into that capability. The launch vehicle types considered are presented in Figure D-9, where the vehicles are shown to scale and compared to the Saturn V launch vehicle from the Apollo program. The Heavy Lift Launch Vehicles (HLLVs) used in this analysis are photographically scaled versions of the Ares V from ESAS, which can deliver approximately 150 mt to LEO [7]. This vehicle has a LOX/LH2 core with Space Shuttle Main Engines and two solid rocket boosters. The second stage is an Earth Departure Stage that performs both suborbital and in-space burns. The other two launch vehicles, the Delta IV-H and Falcon Heavy use the quoted performance from the manufacturer [44],[63].

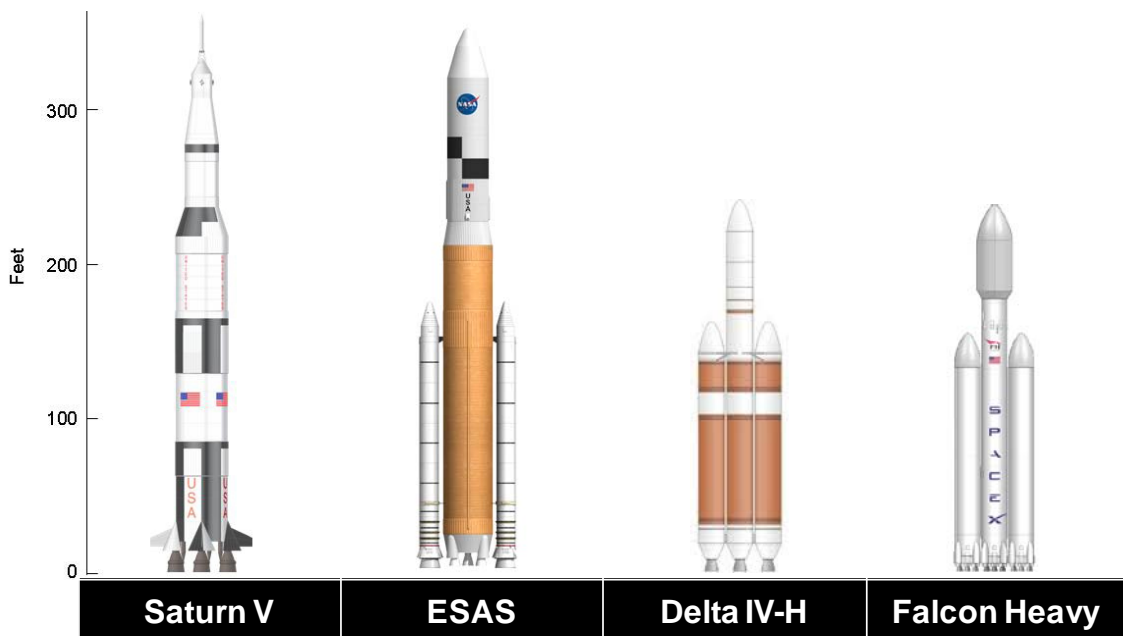


Figure D-9: Comparison of Launch Vehicles Used in System Architecture Analysis

D.6. Propulsive Stage

The propulsive stages are sized using a combination of regressions of existing propulsive stages and response surface equations. The inputs for the propulsive stage are ΔV (for each burn), payload mass (for each burn), propellant combination (fuel, oxidizer, oxidizer-to-fuel ratio), and system T/W. The inert mass of the propulsive stages used in this analysis is a regression through existing upper stages and in-space stages. An overview of the properties of each of these systems is given in Table D-1 [7],[44]. The regressions created from this data are presented in Figure D-10. Each propellant type has a separate curve, where the differences primarily stem from the different bulk density (total density of oxidizer and fuel combined) of the selected propellant combination. Systems with a propellant mass below 20 mt are modeled as service module configurations, while systems above 20 mt of propellant are modeled as upper stages or in-space stages.

Table D-1: Overview of Existing and Designed Propulsive Stages [7],[44]

Stage Name	Usage	Propellant Type	Propellant Mass (kg)	Gross Mass (kg)	Inert Mass (kg)	IMF
Ariane V Aestus	In-Space	Storable	9,700	10,900	1,200	0.110
Titan II US	In-Space	Storable	27,000	30,000	3,000	0.100
Fregat Soyuz Stage	In-Space	Storable	5,350	6,435	1,085	0.169
Titan IV US	In-Space	Storable	35,000	39,500	4,500	0.114
Titan II Core	Launch	Storable	118,000	122,018	4,018	0.033
Titan IV Core	Launch	Storable	155,000	163,000	8,000	0.049
Apollo SM	SM	Storable	18,410	24,520	6,110	0.249
ESAS CEV SM	SM	LOX/CH4	9,071	13,647	4,576	0.335
Delta IV 4-m US	In-Space	LOX/LH2	20,400	24,170	3,770	0.156
Delta IV 5-m US	In-Space	LOX/LH2	27,200	30,710	3,510	0.114
Delta II US	In-Space	LOX/LH2	16,820	19,300	2,480	0.128
ESAS EDS	In-Space	LOX/LH2	224,788	247,837	23,049	0.093
Ariane V Core	Launch	LOX/LH2	155,000	170,000	15,000	0.088
Delta IV Core	Launch	LOX/LH2	200,000	218,030	18,030	0.083
Soyuz 3rd Stage	In-Space	LOX/RP-1	22,800	25,300	2,500	0.099
Molniya 4th Stage	In-Space	LOX/RP-1	5,500	7,360	1,860	0.253
Falcon 9 US	In-Space	LOX/RP-1	90,719	95,254	4,535	0.048
Atlas V Core	Launch	LOX/RP-1	284,100	304,800	20,700	0.068
Delta II Core	Launch	LOX/RP-1	95,550	104,380	8,830	0.085
Soyuz 2 Strap-Ons	Launch	LOX/RP-1	39,200	44,500	5,300	0.119
Soyuz 2 Core	Launch	LOX/RP-1	90,000	99,400	9,400	0.095

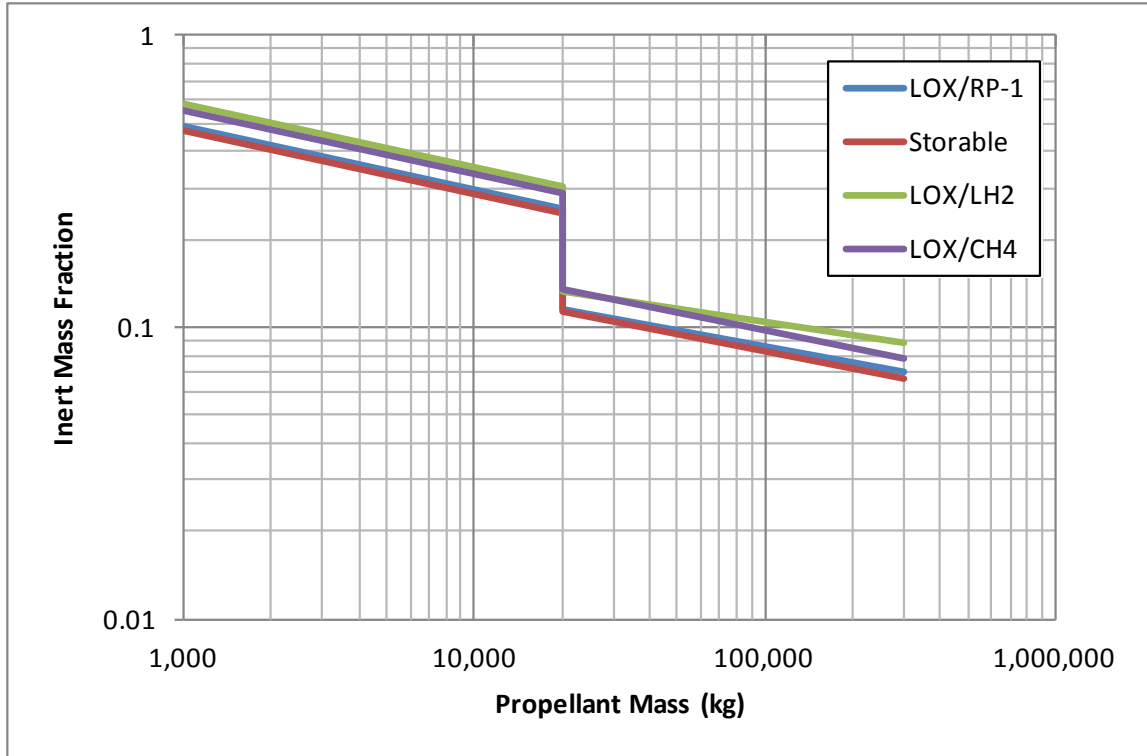


Figure D-10: Regression of Propulsive Stage Inert Mass Fraction for Different Propellants

Residual propellant mass and propellant boil-off mass are modeled separately from the curve presented in Figure D-10. Residual mass is given in Equation (16), and is a function of propellant mass flow rate and bulk density.

$$Residual\ Propellant = 1782.8 \left(\frac{\text{propellant mass flow rate}}{\text{bulk density}} \right)^{0.6678} \quad (16)$$

Boil-off propellant is assumed to have a constant percent of the total stage propellant load per day for different propellants. For LOX, boil-off is assumed to be 0.025% per day; for LH2, boil-off is assumed to be 0.185% per day; for CH4, boil-off is assumed to be 0.033% per day; and for all other propellants, boil-off is assumed to be zero [41]. These values are a function of the thermal properties of the stage and location in space, which would require more detailed analysis.

D.7. Propellant Depot

The propellant depot model consists of regressions and response surfaces to build up a propellant depot that is derived from a propulsive stage (contains propulsion capability, common structural design, etc.). Figure D-11 presents the configuration used for this model. The inputs to the model are propellant mass required (after boil-off and transfer losses), fuel and oxidizer type, oxidizer-to-fuel ratio, and on-orbit time. The bottoms-up mass of each subsystem is then predicted using regressions and response surface equations that were derived from the literature [41],[64]. The equations used to develop this model are presented, along with the mass breakdown structure for this system in Table D-2. The means in which this model calculates boil-off and residual propellant is equivalent to the propulsive stage model presented in Section D.6.

Figure D-11: Configuration of Propellant Depot (Derived from Propulsive Stage)

Table D-2: Overview of Propellant Depot System Sizing Relationships

Mass Element	Estimation Method	Source
1. Structure		
1.1. Main Structure	$m_{struct} = 4.951 * Area^{1.15}$	[40]
1.2. Fuel Tank	$m_{fuel\ tank} = 5.4949 * Vol_{fuel}^{1.063}$	[41]
1.3. Oxidizer Tank	$m_{ox\ tank} = 5.4949 * Vol_{ox}^{1.0318}$	[41]
1.4. Thrust Structure	$m_{thrust\ struct} = 0.0947 * T[kN]^{1.1488}$	[41]
1.5. Docking & Fluid Transfer Mechanism	Assumed 400 kg	[64]
2. Main Propulsion		
2.1. Engine mass	$m_{engine} = \frac{T/W * m_{gross} * g_0}{Engine\ T/m}$	[41]
2.2. Fuel System	$m_{fuel\ sys} = 45.208 * \dot{m}_{fuel}^{0.5999}$	[41]
2.3. Oxidizer System	$m_{ox\ sys} = 15.294 * \dot{m}_{ox}^{0.6388}$	[41]
2.4. Engine Control	$m_{engine\ control} = 0.1897 * T[kN]^{0.9179}$	[41]
3. Power		
3.1. Array Mass	$m_{array} = \frac{Power\ Required}{Array\ Specific\ Power}$	[41]
3.2. Battery Mass	$m_{array} = \frac{Energy\ Required}{Battery\ Specific\ Energy}$	[41]
3.3. Power Management & Distribution	5% of total power subsystem mass	[64]
4. Thermal Control (cryocoolers + insulation)	$m_{cryocooler} = 0.0122 * m_{fuel} + 6.219$ $m_{insulation} = 0.0451 * Area + 0.0458$	[64]
5. Avionics	Assumed 200 kg	[64]
6. Growth	Add 20% to each subsystem mass	

D.8. Surface Habitat

The surface habitat model is a photographically scaled version of the surface habitat used on the LSAM from ESAS, as presented previously in Figure D-7. The

habitat scales with number of crew and stay time. The pressure vessel, power, life support system, consumables, and thermal control systems scale with the ratio of total crew days. Crew accommodations scale with the number of crew, and the command, control, and data handling system remains constant. A screenshot of the Excel spreadsheet that performs the photographic scaling for the surface habitat system is presented in Figure D-12.

INPUTS			MASS BREAKDOWN		
Number of Crew	4		SUBSYSTEM	LEVEL 1 MASS	LEVEL 2 MASS
Stay Time	7	days	Structure (Pressure Vessel, Windows)	980	
			Power	579	
			Command, Control, & Data Handling	385	
			Environmental	895	
			<i>ECLSS</i>		521
			<i>Active Thermal Control</i>		283
			<i>Crew Accommodations</i>		91
			Other	382	
			Growth	644	
			Dry Mass	3,865	
			Non-Cargo	834	
			Cargo	0	
			Inert Mass	4,699	
			Propellant Mass	0	
			Gross Mass	4,699	

Figure D-12: Screenshot of Surface Habitat Sizing Spreadsheet

D.9. In-Space Habitat

Finally, the in-space habitat is a photographically scaled version of the deep space habitat developed for the Human Exploration Framework Team [11] to support a crew of four for 365 days. The geometry of this habitat is presented in Figure D-13. This system scales with number of crew and mission duration, and Figure D-14 presents a screenshot of the Excel spreadsheet that performs the photographic scaling.

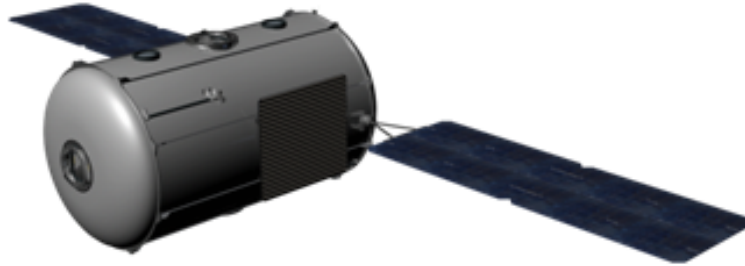


Figure D-13: Configuration of Deep Space Habitat [11]

INPUTS		BASELINE MASS BREAKDOWN				MASS BREAKDOWN	
Number of Crew	4	Subsystem	Lvl 1 Mass	Lvl 2 Mass	Scaling	Lvl 1 Mass	Lvl 2 Mass
Stay Time	359 days	Structure	4,539.0		Area	4,370	
PHOTOGRAPHIC SCALING		Protection	2,297.0		Area	2,212	
		Propulsion	0.0			0	
		Power	1,286.0		Crew	1,286	
		Control	0.0			0	
Crew Volume Ratio	1	Avionics	453.0		Contant	453	
Duration Ratio	0.9447	Environmental	9,917.0			9,558	
Crew-Days Ratio	0.9447	<i>ECLSS</i>		4,582.0	Crew, Crew-Days		4,333
		<i>Active Thermal</i>		943.0	Crew, Crew-Days		834
		<i>EVA Systems</i>		253.0	Days, Crew-Days		251
		<i>Crew Accommodations</i>		4,139.0	Crew		4,139
		Dry Mass	18,492.0			17,879	
		Growth	5,547.6		30% Dry Mass	5,364	
		Dry Mass with Growth	24,039.6			23,243	
		Consumables	4,302.0			4,064	
		<i>Spares</i>		2,021.0	Days		1,909
		<i>Food</i>		2,281.0	Crew-Days		2,155
		Gross Mass	28,341.6			27,307	
		BASELINE MASS BREAKDOWN					
		Subsystem	Lvl 1 Mass	Lvl 2 Mass	Scaling		
		<i>ECLSS Model</i>	4,582				
		<i>Air Subsystems</i>		1,028	Crew-Days		
		<i>Water Subsystem</i>		1,969	Crew-Days		
		<i>Food Storage</i>		571	Crew-Days		
		<i>Human Accommodations</i>		84	Crew		
		<i>Other</i>		930	Crew-Days		

Figure D-14: Screenshot of In-Space Habitat Sizing Spreadsheet

REFERENCES

1. Review of U.S Human Spaceflight Committee. *Seeking a Human Spaceflight Program Worthy of a Great Nation*. October 2009.
2. "Fiscal Year 2011 Budget Estimates" NASA, Washington, DC. February 2010.
3. Hastings, Daniel and Weigel, Annalisa. "Lecture 1: Space Systems and Definitions." Course notes from ESD.353J: Space System Architecture and Design. Massachusetts Institute of Technology, Fall 2004.
4. Crawley, Edward. "Definitions, the Architect and Architecting, and Deliverables of the Architect." Course notes from ESD.34: System Architecture. Massachusetts Institute of Technology, January 2007.
5. Crawley, Edward et al. "The Influence of Architecture in Engineering Systems." *Engineering Systems Monograph*. March 2004.
6. *Manned Lunar Landing Program Mode Comparison*. NASA TM-X-74752. July 1962.
7. *NASA's Exploration Systems Architecture Study Final Report*. NASA TM-2005-214062. November 2005.
8. *The Vision for Space Exploration*. NASA NP-2004-01-334-HQ. February 2004.
9. Mazanek, Daniel D. et al. "Near-Earth Object Crew Mission Concept Status." NASA Internal Study, 2005.
10. Abell, Paul and Landis, Rob. "New NASA Initiatives for the Exploration of Near-Earth Objects." Presented at 3rd Small Bodies Assessment Group Meeting. August 2010.
11. Human Architecture Team Steering Council. *HEFT Phase I Closeout*. NASA Internal Study, September 2010.
12. Portree, David S.F. "Humans to Mars: Fifty years of Mission Planning, 1950-2000." *Monographs in Aerospace History #21*. NASA SP-2001-4521. February 2001.
13. Hoffman, Stephen J. and Kaplan, David I., ed. *Human Exploration of Mars: The Reference Mission of the NASA Mars Exploration Study Team*. NASA SP-6107. July 1997.
14. Drake, Bret G., ed. *Reference Mission Version 3.0: Addendum to the Human Exploration of Mars: The Reference Mission of the NASA Mars Exploration Study Team*. NASA SP-6107-ADD. June 1998.
15. Borowski, Stanley K., Dudzinski, Leonar A., and McGuire, Melissa L. *Vehicle and Mission Design Options for the Human Exploration of Mars/Phobos Using 'Bimodal' NTR and LANTR Propulsion*. NASA TM-1998-208834. December 1998.
16. Drake, Bret G., ed. *Human Exploration of Mars Design Reference Architecture 5.0: Addendum*. NASA SP-2009-566-ADD. July 2009.
17. Mavris Dimitri. "A 'Paradigm Shift' in Complex System Design." Course notes from Advanced Design Methods I. Georgia Institute of Technology, August 2006.
18. Fabrycky, Wolter J. and Blanchard, Benjamin S. *Life Cycle Cost and Economic Analysis*. Prentice-Hall, NJ. 1991.

19. Speller, Thomas H. *Algorithmic Approach to System Architecting Using Shape Grammar-Cellular Automata*. Ph.D. Thesis, Massachusetts Institute of Technology. June 2008.
20. de Weck, Olivier et al. "State-of-the-Art and Future Trends in Multidisciplinary Design Optimization." AIAA Paper 2007-1905. April 2007.
21. Taylor, Christine. *Integrated Transportation System Design Optimization*. Ph.D. Dissertation, Massachusetts Institute of Technology. February 2007.
22. Komar, D.R., Hoffman, Jim, Olds, Aaron, and Seal, Mike. "Framework for the Parametric System Modeling of Space Exploration Architectures." AIAA Paper 2008-7845. September 2008.
23. Simmons, Willard L., Koo, Benjamin H. Y., and Crawley, Edward F. "Architecture Generation for Moon-Mars Exploration Using an Executable Meta-Language." AIAA Paper 2005-6726. August 2005.
24. Komar, D. R., Personal conversation on February 3, 2010.
25. Dasgupta, S., Papadimitriou, C.H., and Vazirani, U.V. *Algorithms*. McGraw Hill, 2007. URL: <<http://highereducation.mcgraw-hill.com/sites/0073523402/>>.
26. Cormen, Thomas H., Leiserson, Charles E., Rivest, Ronald L., and Stein, Clifford. *Introduction to Algorithms, 2nd Ed.* The MIT Press, 2001.
27. Moore, B., Braun, R.D., and Kroo, I.M. "Use of Collaborative Optimization Architecture for Launch Vehicle Design." *6th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, 1996.
28. Macal, Charles M., and North, Michael J. "Introduction to Agent-based Modeling and Simulation." Presented to MCS LANS Informal Seminar, 2006.
29. Shankar, A. Udaya. *Discrete-Event Simulation*. University of Maryland, 1991.
30. West, Douglas B. *Introduction to Graph Theory*. Prentice-Hall, Inc. Upper Saddle River, NJ. 1996.
31. Bhadra, Dipasis and Hogan, Brendan. "A Preliminary Analysis of the Evolution of US Air Transportation Network." AIAA Paper 2005-7414. September 2005.
32. Lawler, E. L., Lenstra, J. K., and Rinnooy Kan, A. H. G. *The Traveling Salesman Problem*. John Wiley & Sons, Inc. Hoboken, NJ. 1985.
33. Toth, Paolo and Vigo, Daniele, ed. *The Vehicle Routing Problem*. Society for Industrial and Applied Mathematics, Philadelphia, PA. 2002.
34. Taylor, Christine, Song, Miao, Klabjan, Diego, de Weck, Olivier L., and Simchi-Levi, David. "Modeling Interplanetary Logistics: A Mathematical Model for Mission Planning." AIAA Paper 2006-5735. June 2006.
35. Bounova, Gergana A., Ahn, Jaemyung, Hofstetter, Wilfried, Wooster, Paul, Hassan, Rania, and de Weck, Olivier L. "Selection and Technology Evaluation of Moon/Mars Transportation Architectures." AIAA Paper 2005-6790. September 2005.
36. Garg, Naveen. Lecture 25 on Data Structures and Algorithms. India Institute of Technology, YouTube EDU. URL: <<http://www.youtube.com/watch?v=hk5rQs7TQ7E>>. Accessed April 29, 2010.
37. Gross, Jonathan J. and Yellen, Jay, ed. *Handbook of Graph Theory*. CRC Press LCC, Boca Raton, FL. 2004.
38. Villeneuve, Frederic. *A Method for Concept and Technology Exploration of Aerospace Architectures*. Ph.D. Dissertation, Georgia Institute of Technology. August 2007.

39. Mavris, Dimitri. "Introduction to Design of Experiments and Response Surface Methods." Course notes from Advanced Design Methods I. Georgia Institute of Technology, October 2, 2006.
40. Heineman, Jr., Willie. *Design Mass Properties II: Mass Estimating and Forecasting for Aerospace Vehicles Based on Historical Data*. JSC-26098. November 1994.
41. Wilhite, A. W., Gholston, S. E., Farrington, P. A., and Swain, J. J. "Evaluating Technology Projections and Weight Prediction Method Uncertainty of Future Launch Vehicles." *Journal of Spacecraft and Rockets*, Vol. 45, No. 3, pp. 587-591. May-June 2008.
42. Larson, Wiley J. and Pranke, Linda K., ed. *Human Spaceflight: Mission Analysis and Design*. McGraw-Hill, 1999.
43. Arney, Dale and Wilhite, Alan. "Orbital Propellant Depots Enabling Lunar Architectures without Heavy-Lift Launch Vehicles." *Journal of Spacecraft and Rockets*, Vol. 47, No. 2. March-April 2010.
44. Isakowitz, S., Hopkins, J., and Hopkins, Jr., J. *International Reference Guide to Space Launch Systems*, 4th Ed. AIAA, 2004.
45. 2008 NASA Cost Estimating Handbook. NASA Headquarters Cost Analysis Division, 2008.
46. Krevor, Zachary C. *A Case Study of the STS Indirect and Support Costs: Lessons to be Learned for the Next Generation Launch System*. AE 8900 Project, Georgia Institute of Technology. April 2004.
47. Young, David A. *An Innovated Method for Allocating Reliability and Cost in a Lunar Exploration Architecture*. Ph.D. Dissertation, Georgia Institute of Technology. May 2007.
48. McAfee, Julie, Culver, George, and Naderi, Mahmoud. *NASA/Air Force Cost Model (NAFCOM): Capabilities and Results*. 2011 JANNAF MSS/LPS/SPS Meeting, Huntsville, AL. December 2011.
49. Koelle, Dietrich E. *Handbook of Cost Engineering for Space Transportation Systems (Revision 2) with Transcost 7.1 Statistical-Analytical Model for Cost Estimation and Economical Optimization of Launch Vehicles*. TCS-TransCostSystems. Ottobrunn, Liebigweg, Germany. February 2006.
50. Shishko, Robert. *NASA Systems Engineering Handbook*. NASA SP-610S, June 1995.
51. Vanderplaats, Garret N. *Numerical Optimization Techniques for Engineering Design: with Applications*. McGraw-Hill, 1984.
52. Spall, James C. *Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control*. John Wiley & Sons, Inc. Hoboken, NJ. 2003.
53. Venter, Gerhard and Sobieszczanski-Sobieski, Jaroslaw. "Particle Swarm Optimization." *AIAA Journal*, Vol. 41, No. 8. August 2003.
54. Venter, G. and Sobieszczanski-Sobieski, J. "Multidisciplinary optimization of a transport aircraft wing using particle swarm optimization." *Structural Multidisciplinary Optimization*, Vol. 26, pp. 121-131. 2004.
55. Volovoi, Vitali. "GA Paradigm." Course notes from Advanced Design Methods II. Georgia Institute of Technology, March 2007.
56. Evolver 5.5.1, developed by Palisade Corporation. URL: <http://www.palades.com/evolver/>. Accessed April 29, 2010.

57. Global Optimization Toolbox 3.0, developed by MathWorks for MATLAB. URL: <<http://www.mathworks.com/products/global-optimization/>>. Accessed April 29, 2010.
58. Donahue, Benjamin B. and Cupples, Mike L. "Comparative Analysis of Current NASA Human Mars Mission Architectures." *Journal of Spacecraft and Rockets*, Vol. 38, No. 5, pp. 745-751. Sept.-Oct. 2001.
59. Hickman, Joseph W., Wilhite, Alan, and Stanley, Douglas. "Optimization of the Mars Ascent Vehicle for Human Space Exploration." *Journal of Spacecraft and Rockets*, Vol. 47, No. 2, pp. 361-370. March-April 2010.
60. Duke, Michael B., Keaton, Paul W., Weaver, David, Roberts, Barney, Griggs, Geoffrey, and Huber, William. "Mission Objectives and Comparison of Strategies for Mars Exploration." AIAA Paper 1993-0956. February 1993.
61. Newnan, Donald G., Eschenbach, Ted G., and Lavelle, Jerome P. *Engineering Economics Analysis, 9th Ed.* Oxford University Press, USA. February 2004.
62. Office of Management and Budget, White House. *OMB Circular No. A-94 Appendix C*. December 2011. URL: <http://www.whitehouse.gov/omb/circulars_a094_a94_appx-c/>. Accessed May 27, 2012.
63. Falcon Heavy Website, Space Exploration Technologies Website, URL: <http://www.spacex.com/falcon_heavy.php>. Accessed May 27, 2012.
64. Street, David. *A Scalable Orbital Propellant Depot Design*. AE 8900 MS Special Problems Report, Georgia Institute of Technology. April 2006.
65. Young, James J. *A Value Proposition for Lunar Architectures Utilizing On-Orbit Propellant Refueling*. Ph.D. Dissertation, Georgia Institute of Technology. May 2009.
66. Wilhite, Alan, Stanley, Douglas, Arney, Dale, and Jones, Chris. "Near Term Space Exploration with Commercial Launch Vehicles plus Propellant Depot." March 2011. URL: <<http://nasawatch.com/archives/2011/03/using-commercial.html>>. Accessed May 27, 2012.