Enhancing the Economics of Satellite Constellations via Staged Deployment and Orbital Reconfiguration

by

Mathieu Chaize

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of
Master of Science in Aeronautics and Astronautics
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
May 2003

Author . . . .
Department of Aeronautics and Astronautics
May 22, 2003

Certified by . . . .
Olivier L. de Weck
Assistant Professor of Aeronautics and Astronautics and of Engineering Systems
Thesis Supervisor

Certified by . . . .
Richard de Neufville
Professor of Engineering Systems and of Civil and Environmental Engineering
Thesis Supervisor

Accepted by . . . .
Edward M. Greitzer
H.N. Slater Professor of Aeronautics and Astronautics
Chair, Committee on Graduate Students

Enhancing the Economics of Satellite Constellations via Staged Deployment and Orbital Reconfiguration

by

Mathieu Chaize

Submitted to the Department of Aeronautics and Astronautics
on May 22, 2003, in partial fulfillment of the
requirements for the degree of
Master of Science in Aeronautics and Astronautics

Abstract

The "traditional" way of designing constellations of communications satellites is to optimize the design for a specific global capacity. This approach is based on a forecast of the expected number of users and their activity level, both of which are highly uncertain. It can lead to economic failure if the actual demand is smaller than the one predicted. This thesis presents an alternative approach to the design process to reduce the economic risks. It proposes to deploy constellations in a staged manner, starting with a smaller, more affordable capacity that can be increased if necessary. When the capacity is increased, additional satellites have to be launched and the existing constellation needs to be reconfigured on orbit. Technically, it implies that particular design elements are initially embedded in the design to allow the reconfiguration. Such elements are called "real options" and give decision makers the right but not the obligation to increase the capacity of the system after its initial deployment.

This approach reframes the design objectives. Instead of determining an optimal design for a specific capacity, paths of architectures are sought in the trade space. A general framework is presented to identify the paths that offer the most flexibility given different demand scenarios. It is then applied to LEO constellations of communications satellites. Improvements in the life cycle costs on the order of 30% can be obtained for different discount rates and volatilities. This value of flexibility has to be compared to the actual price of the real options. A general method is proposed to study this problem and two technical solutions are proposed.

Thesis Supervisor: Olivier L. de Weck
Title: Assistant Professor of Aeronautics and Astronautics and of Engineering Systems

Thesis Supervisor: Richard de Neufville
Title: Professor of Engineering Systems and of Civil and Environmental Engineering

Acknowledgments

I would like to thank my advisors, Prof. Olivier L. de Weck and Prof. Richard de Neufville for their guidance and support. Prof. de Weck helped me throughout my entire experience at MIT and he cannot be thanked enough for this. My family and Julie Muller need to be acknowledged for their support and love. I would also like to thank the Messinger family, Sara and Laurent Levy, Roadkill Buffet (MIT's premiere improv' comedy troupe), Isabelle Pauwels and Sebastien Aknouche for their encouragement during the time this research was accomplished. Finally, members of the Space Systems Laboratory need to be thanked for their suggestions, in particular Prof. David Miller, Dr. Raymond Sedwick, Julien-Alexandre Lamamy, Dr. Cyrus Jilla, Darren Chang and Kin Cheong Sou.

This research was supported by the Alfred P. Sloan Fundation, grant 2000-10-20. Dr. Gail Pesyna served as project monitor. I would like to thank Dr. Joel Cutcher-Gershenfeld who is the leader of the Enginnering Systems Learning Center and the Industry System Study Project Leader, as well as Ann Tremelling who is the fiscal officer.

Contents

1 Introduction **21**

1.1 Motivation 21

1.2 Literature Review 24

1.2.1 LEO Constellations of Communications Satellites 24

1.2.2 Valuation of Flexibility for Space Systems 25

1.2.3 Staged Deployment for Space Systems 25

1.2.4 Orbital Reconfiguration 26

1.2.5 Valuation of flexibility in Other Domains 27

1.3 Organization of the Chapters 27

2 Value of Flexibility for Capacity Planning under Uncertainty **31**

2.1 Problem Definition 31

2.2 Traditional or Net Present Value Approach 35

2.2.1 Net Present Value (NPV) 36

2.2.2 Traditional Selection of an Architecture 37

2.2.3 Response to Uncertainty in Demand 40

2.3 Value of Flexibility 40

2.3.1 Economic Advantage of Flexibility 40

2.3.2 Issues Related to the Valuation of Flexibility 42

2.4 Valuing Flexibility Methods 43

2.4.1 Decision Analysis 43

2.4.2 Real Options Analysis 49

2.5 Valuation Framework for Staged Deployment 57

2.5.1 Presentation of the Problem 57
2.5.2 Paths of Architectures 58
2.5.3 Valuation of Flexibility 61
2.5.4 Implementation 63

3 Staged Deployment Strategy for Reconfigurable LEO Constellations 67
3.1 Problem Definition 67
3.1.1 Presentation of the Simulator 67
3.1.2 Definition of Capacity 71
3.1.3 Assumptions 74
3.2 Decomposition of the Design Vector 75
3.3 Identification of Paths 77
3.3.1 Evolution Rules 77
3.3.2 Representation of Paths 78
3.3.3 Ordering Families 80
3.4 Costs Decomposition 84
3.4.1 Flexibility and Reconfiguration 84
3.4.2 Evolution Costs 85
3.4.3 Transition Matrix 89
3.5 Definition of Demand 90
3.6 Life Cycle Costs 93
3.7 Optimization 95
3.7.1 Input Parameters 95
3.7.2 Stored Values 97
3.7.3 Path Optimization Process 98

4 Case Study 103
4.1 Introduction 103
4.2 Determination of x^{trad} 103
4.2.1 Capacity 104
4.2.2 Life Cycle Costs 104

4.2.3 Trade Space . . . 105
4.3 Determination of a Best Path and Parameter Study . . . 106
4.3.1 Parameters Selected . . . 106
4.3.2 Value of Flexibility . . . 106
4.3.3 Discount Rate r . . . 108
4.3.4 Volatility σ . . . 113
4.3.5 Other Parameters . . . 116
4.4 Conclusions . . . 117

5 Orbital Reconfiguration of Constellations 119
5.1 Problem Definition . . . 119
5.1.1 Reconfiguration . . . 119
5.1.2 Orbital Transfer Optimization Problem . . . 121
5.2 Constellation Reconfiguration Model . . . 122
5.2.1 Orbital Slots . . . 122
5.2.2 Assignment Problem . . . 124
5.2.3 Orbital Transfers . . . 127
5.3 Technical Solutions . . . 134
5.3.1 Additional Propellant . . . 134
5.3.2 Space Tug . . . 136
5.4 Remaining Issues . . . 141

6 Summary and Conclusions 143
6.1 Summary . . . 143
6.2 Conclusions and Recommendations . . . 145
6.2.1 Value of Staged Deployment . . . 145
6.2.2 Reconfiguration . . . 146
6.2.3 Other Opportunities for Constellations of Satellites . . . 147

A Estimation of the envisaged charge per minute 149

B Orbital Elements **153**

B.1 Position of the Spacecraft on the Ellipse 153

B.2 Position of the Ellipse with Respect to Earth's Frame 154

C Sub-synchronous and Super-synchronous Orbits **157**

C.1 Problem Definition 157

C.2 Calculating $T_{transfer}$ and ΔV 160

C.2.1 Circular Reference Orbit 160

C.2.2 Elliptic Reference Orbit 164

C.3 Performance of the Transfers 165

C.4 Constraints on ΔV and $T_{transfer}$ 166

C.4.1 Time Constraint 167

C.4.2 ΔV Constraint 167

List of Figures

1-1 Projection made in 1991 for US Terrestrial Cellular Subscribers between 1991 and 2000 (from [CGH+92]). 23
1-2 Roadmap of the thesis. 29

2-1 Adapted form of Michael Porter's five forces model for competitive strategy (from [Por98]). 32
2-2 Flexibility and Robustness as a function of the system's objectives and environment (from [Sal02]). 33
2-3 Pareto Frontier and Utopia point for a Trade Space composed of 5 architectures. 38
2-4 Traditional selection of an architecture for a desired capacity in the case of LEO constellations. 39
2-5 Response to uncertainty in demand of the traditional way of selecting a design. 41
2-6 Basic nodes of a decision tree. 44
2-7 Structure of the decision tree corresponding to the roulette example. . 45
2-8 Initial column of values of the decision tree. 46
2-9 Computation of the value of a chance node from the column of values. 47
2-10 Computation of the value of a decision node from the column of values. 47
2-11 Profit made with the option for different future values of the stock price. 50
2-12 Basic structure of the binomial model. 54
2-13 Example of a binomial tree and evolution of the value of the underlying asset over four periods. 55

2-14 Value of the option and suggested decision for the possible values of the stock price. 55
2-15 Example of a path of architectures in a Trade Space. The series of architectures (A_1, A_2, A_3, A_4) represents the path. 61

3-1 Architecture of a LEO constellation of communications satellites (from [dWC02]). 69
3-2 Geometric definition of the elevation angle. The coverage area of the satellite is shaded. S: satellite; U:user; O: Earth's center. 70
3-3 A decrease in altitude from position $S1$ to $S2$ decreases the coverage area of a satellite if ϵ is kept constant. 79
3-4 Example of a path in the trade space. 80
3-5 Difference between $\Delta C_{A\to B}$ and $LCC(B) - LCC(A)$ due to the launch costs. 88
3-6 Difference between the present value of $\Delta C_{A\to B}$ and $LCC(B) - LCC(A)$. 88
3-7 Representation of the elements of ΔC that are different from -1 and zoomed view of a ΔC^{family} submatrix. 90
3-8 Two possible evolutions of demand over two periods. 91
3-9 Example of one scenario out of $2^5 = 32$ in the binomial tree ($n = 5$). . 92
3-10 Example of a matrix D and a vector P to represent demand scenarios. 100
3-11 Link between the different steps of the optimization process. 101

4-1 Position of the Iridium constellation and the optimal traditional architecture x^{trad} for $Cap_{max} = 2.8 \cdot 10^6$. 105
4-2 Difference between the life cycle cost of the optimal traditional design x^{trad} and the average life cycle cost of the optimal path $path^\star$. 107
4-3 Binomial tree for $D_{initial} = 5.10^4$, $\sigma = 70\%$ and $\mu = 20\%$. 108
4-4 Value of flexibility in % of money saved with respect to r. 109
4-5 Five optimal paths obtained in the parameter study. 111
4-6 Life cycle costs of x^{trad} and the different optimal paths with respect to the discount rate. 112

4-7 Value of flexibility with respect to σ when $r = 25\%$. 113
4-8 Six optimal paths obtained for different values of σ. 115
4-9 Life cycle costs of the different optimal paths with respect to σ when $r = 25\%$. 116

5-1 Conceptual representation of the reconfiguration maneuvers to evolve from constellation A to constellation B. 120
5-2 Physical meaning of a, i, θ and Ω. 123
5-3 Network flow representation of the assignment problem. 125
5-4 Network flow for a reconfiguration between constellation A with $N(A)$ satellites and constellation B with $N(B)$ satellites. 126
5-5 Optimal assignment of the launched and transferred satellites in the orbital planes. 132
5-6 Example of a launch strategy for the second approach. 133
5-7 Example of a network flow diagram for the Traveling Salesman Problem with 8 cities. 138
5-8 Network flow proposed by the first approach. 139
5-9 Network flow proposed by the second approach. 140

B-1 Geometric meaning of a_0, e_0 and ν_0. 154
B-2 Geometric representation of i, Ω and ω_0. 155

C-1 Relative position of Sl with respect to Sp when the reference orbit is circular or elliptic. 158
C-2 Decomposition of a sub-synchronous transfer and rendezvous between Sp and Sl when the reference orbit is circular. 159
C-3 Representation of the notations used for the calculations of ΔV_{apogee}. 163
C-4 ΔV and $T_{transfer}$ for different transfers ($e_0 = 0$, $R_{ref} = 11400$ km and θ=270 deg). 166
C-5 Optimal ΔV for different values of θ when the reference orbit is circular with an altitude of 5000 km. 168

C-6 Optimal $T_{transfer}$ for different values of θ when the reference orbit is circular with an altitude of 5000 km. 168

List of Tables

1.1 History of the Iridium system (from [LWJ00]). 22

2.1 Comparison of financial options and real options (adapted from [Ram02]). 51

2.2 Call option cost and payoffs. 56

2.3 Stock and loan costs and payoffs. 56

3.1 Range of the different design variables used to generate the trade space. *Step* corresponds to the increment used. 71

3.2 Characteristics of constellations A and B. 72

3.3 Rank of architectures in a family. 82

3.4 Decomposition of the initial development costs and dependence with respect to x_{base} and x_{flex} (the symbol "+" is used when there is a dependence). 86

4.1 Values of the fixed parameters. 106

A.1 Prices obtained with the different equations. 151

Nomenclature

Abbreviations

CPM	Cost Per Minute
DA	Decision Analysis
EX	Exercise Option
FCC	Federal Communications Commission
GCI	Geocentric Inertial
GEO	Geosynchronous Equatorial Orbit
GINA	Generalized Information Network Analysis
GSM	Global System for Mobile communications
IFMIF	International Fusion Materials Irradiation Facility
ITU	International Telecommunication Union
KO	Keep Option Opened
LEO	Low Earth Orbit
MDO	Multidisciplinary Design Optimization
MEO	Medium Earth Orbit
NPV	Net Present Value
PV	Present Value
RAAN	Right Ascension of the Ascending Node
ROA	Real Options Analysis
TSP	Traveling Salesman Problem

Symbols

A_{user}	Average user activity

C	Constellation type
Cap	Capacity
Cap_{max}	Maximum capacity
C_{tot}	Lifetime capacity
D_A	Antenna diameter
$D_{initial}$	Initial value of demand
E_{Sl}	Eccentric anomaly of Sl
E_{Sp}	Eccentric anomaly of Sp
G	Newtonian constant of gravitation
IDC	Initial development cost function
ISL	Inter satellite links
Isp	Specific impulse
J	Objective function
LCC	Life cycle cost
LCC_{best}	Life cycle cost of the optimal path
M_{Earth}	Earth's mass
M_f	Final mass of the spacecraft
M_i	Initial mass of the spacecraft
M_p	Mass of propellant
$N_{channels}$	Instantaneous number of duplex channels of a constellation
N_{full}	Maximum number of satellites a launch vehicle can carry
N_{ground}	Number of satellites initially on the ground
N_{sats}	Number of satellites
N_{tugs}	Number of tugs
N_{user}	Maximum number of subscribers for a constellation
OM	Operations and Maintenance Costs function
P_t	Satellite transmit power
R_{Earth}	Mean radius of the Earth
R_{apogee}	Distance between the center of the Earth and the apogee of an orbit
$R_{perigee}$	Distance between the center of the Earth and the perigee of an orbit

S	Stock price or Present value of asset
Sl	Orbital slot
Sp	Spacecraft
T_{delay}	Time to orbit from Sl to the initial position of Sp
T_{ex}	Time to expiration date
T_{sys}	Lifetime of the system
$T_{transfer}$	Orbital transfer time
U_S	Global system utilization
V_{apogee}	Velocity of the spacecraft at the apogee of the transfer orbit
V_{ref}	Velocity of the spacecraft on the reference orbit
X	Strike or Exercise price
a	Orbital altitude
a_d	Disturbing acceleration
$a_{transfer}$	Semi-major axis of the transfer orbit
a_0	Semi-major axis
c	Vector of constant parameters
e_0	Eccentricity of the orbit
h	Massless angular momentum
i	Inclination
$path^{\star}$	Optimal path of architectures
p	Probability of going up in the binomial tree
q	Constraints vector
r	Discount rate
w	Optimization parameters vector
x	Design vector
x^{trad}	Best traditional design
ΔC	Transition matrix
ΔC^{family}	Transition matrix associated to a particular family
ΔV	Sum of the necessary changes of velocities of an orbital transfer
Δf_c	Per-channel bandwidth

Δt	Time step for the binomial tree
Π_{ref}	Period of the reference orbit
$\Pi_{transfer}$	Period of the transfer orbit
Ω	Longitude of the ascending node
ϵ	Minimum elevation angle
θ	Argument of latitude
μ	Expected return per unit time
ν_0	True anomaly
σ	Volatility
τ	Time reference in Kepler's equation
ω_0	Argument of perigee

Chapter 1

Introduction

1.1 Motivation

In November 1998, the first "big" Low Earth Orbit constellation of communications satellites, Iridium, initiated service. This constellation, with its 66 satellites placed in polar orbits, connected via inter satellite links, was considered a technical success. In 1991, the forecasts for the cellular telephone market were optimistic and up to 110 million subscribers[1] were expected in the US by the year 2000 (see US forecast in Figure 1-1). Iridium's original target market was the global business traveler, out of range of terrestrial cellular networks. Additionally, before the introduction of GSM in Europe, there were no common standards. The Iridium company believed it could attract about 3 million potential customers. However, by the time Iridium was deployed, the significant development of terrestrial cellular networks had transformed the marketplace. Thirteen months after it was launched, the Iridium constellation had attracted only 50,000 customers and had to file for Chapter 11 protection (Iridium's history is summarized in Table 1.1). A year later, the Globalstar constellation was deployed and had to declare bankruptcy for the same reasons although it did have different technical characteristics.

Those two cases reveal the high economic risks encountered by developers of com-

[1]The forecast in 1991 were actually wrong because they underestimated the future market. The number of subscribers in the US forecast for 2000 represented only 35% of the actual market at that time.

1987:	Start of concept design.
December 1990:	FCC filing for construction permit and frequency allocation.
1991:	Founding of the Iridium LLC company.
January 1995:	FCC license received (for construction, launch, and operation of Iridium). The usage was limited to the band 1621.35 MHz to 1626.5 MHz.
May 1997:	First satellite launch from Vandenberg, California (5 satellites on a Delta 2 rocket).
May 1998:	Full satellite constellation in orbit.
November 1998:	Start of operation (telephony, paging, and messaging).
End of 1998:	Problems with 12 satellites[2] in the constellation.
August 1999:	Iridium files for Chapter 11 bankruptcy protection, having debt in excess of US$ 4 billion.
December 1999:	Iridium has 50 000 subscribers.

Table 1.1: History of the Iridium system (from [LWJ00]).

mercial space systems. Indeed, many years (11 years in the case of Iridium) and many changes in the marketplace may separate conceptual design from deployment of such systems. The traditional way of designing large capacity systems, which optimizes the design for a specific global capacity, fails to deal with demand uncertainty. Indeed, if the selected capacity is too large compared to the actual demand, the important deployment costs (non-recurring costs) cannot be offset by revenues, thus leading to fast bankruptcy due to negative cash flows. On the other hand, if capacity is much lower than the actual demand, a market opportunity may be missed. Consequently, the traditional approach presents too much risk and another way of designing must be sought.

This thesis proposes an alternative approach to dealing with uncertainty in future demand. A "staged deployment" strategy is considered for which the capacity of the system is increased progressively. Technically, this can be achieved by introducing "real options" into the design. The "real options" are design elements that give the ability but not the obligation to decision makers to adjust the capacity of a system after its initial deployment. This approach is different from adding capacity as an after thought, i.e. without a priori planning. With such flexibility, it is possible to

[2] Iridium officially reported seven failures. The failures were related to stabilization problems or damaging of the satellites during the launch operations.

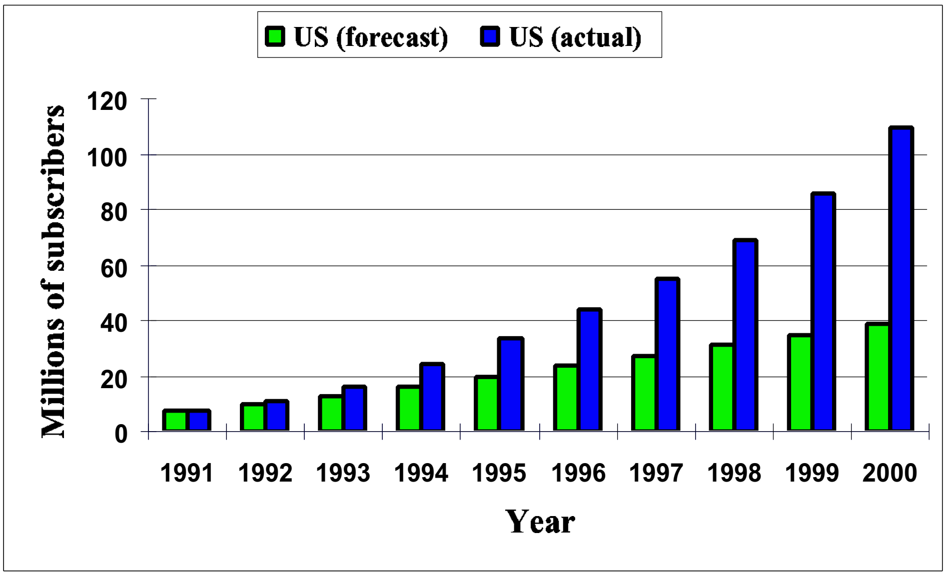

Figure 1-1: Projection made in 1991 for US Terrestrial Cellular Subscribers between 1991 and 2000 (from [CGH+92]).

reduce the economic risks by initially deploying a smaller, more affordable capacity that can be increased when the market conditions are good. This approach marks a distinct change in the design process, as it involves incorporating non-technical, market considerations in the specifications of the system. Indeed, given a demand probability distribution, an initial architecture for the system as well as a deployment strategy have to be found by the designers. Moreover, "real options" to achieve the changes in capacity have to be found.

To assess those problems, this thesis proposes a general framework. Its goal is to help designers to find potential real options in their designs and value the economic opportunity of the flexibility they provide. We applied the framework to study LEO constellations of communications satellites. To increase the capacity of constellations after their initial deployment, satellites need to be added. The on-orbit satellites thus need to be "reconfigured" to form a new constellation with the additional satellites[3]. The "real options" in this case would be a system to achieve the orbital reconfiguration of on-orbit satellites such as extra fuel or a space tug. Staged deployment reduces the

[3]This thesis focuses on inter-satellite reconfiguration rather than intra-satellite reconfiguration.

economic risk when the uncertainty in demand is significant and reconfiguration seems to have an important financial potential. Consequently, we also give an introduction to the problem of orbital reconfiguration and give recommendations for future research on the topic.

1.2 Literature Review

1.2.1 LEO Constellations of Communications Satellites

LEO constellations have been extensively studied since the 1980's. One of the main problems that has been solved was the design of constellations that could achieve global coverage while minimizing the necessary number of satellites. Walker [Wal77] proposed to organize the satellites using inclined circular orbital planes. Adams and Rider [AR87] suggested the use of polar orbits to achieve global coverage. Adams and Lang [AL98] compared those two types of constellations and provided a table of the optimal constellations that achieved global coverage. Other types of constellations have been proposed by Castiel, Brosius and Draim [CBD94] for zonal coverage involving elliptical orbits. They present their approach to optimize the design of the ELLIPSO mission. Elliptic orbits are more difficult to study than circular orbits and Elly, Crossley and Williams [ECW99] proposed to use genetic algorithms to optimize them.

Constellations of satellites are the only way to achieve global coverage. That is why they have been used for communications. The principles of space systems for communications and a presentation of the existing systems are presented by Lutz, Werner and Jahn [LWJ00]. In particular, the characteristics and milestones of the Iridium and Globalstar constellations are presented. Several graduate theses from MIT investigated satellite constellations for mobile phone and data communication. To compare the performances of constellations with different architectures, Gumbert [Gum96] and Violet [Vio95] developed a cost per billable minute metric. Six mobile satellite phone systems were analyzed with this metric [GVH97]. The set of systems studied contained LEO, MEO, GEO and elliptical orbits systems. A methodology named Generalized Information Network Analysis (GINA) to assess the performance of distributed satellite systems was developed by Graeme Shaw [Sha99]. He applied the methodology to broadband satellite systems. Jilla [Jil02] refined the GINA methodology to account for multidisciplinary design optimization (MDO) and proposed a case study of a broadband communications mission. Kashitani [Kas02] proposed an analysis methodology for satellite broadband network architectures based on the works of Shaw and Jilla. The study compared LEO, MEO and elliptic systems and showed that the best architecture depends on customer demand levels. An architectural trade methodology has been developed by de Weck and Chang [dWC02] for the particular case of LEO personal communication systems. A simulator of constellations has been used to generate 1800 different architectures. A benchmarking of the simulation was conducted with the Iridium and Globalstar constellations. This simulator has been used in our study and is presented in Chapter 3.

1.2.2 Valuation of Flexibility for Space Systems

The vast majority of space systems are designed without any considerations for flexibility. One of the main reasons is that operations in space are difficult and expensive. Several theses at MIT focused on the economic opportunity that flexibility could present for space systems. Saleh [Sal02] and Lamassoure [Lam01] studied on-orbit servicing for satellites. On-orbit servicing provides the flexibility to increase the lifetime of the satellites or upgrade their capabilities. Lamassoure proposed to consider the decision of using on-orbit servicing as a real option.

1.2.3 Staged Deployment for Space Systems

Staging the deployment of a space system to reduce the economic and technological risks has been envisioned for both military and scientific missions. Miller, Sedwick and Hartman [MSH01] studied the possibility of deploying distributed satellite sparse apertures in a staged manner. The first stage serves as a technology demonstrator.

Additional satellites are added to increase the capability of the system when desired. The Pentagon also plans to deploy its next generations of radars in a staged manner (see [Sin03]). A first constellation of space based radars will be launched in 2012, but will not provide full coverage. This will allow the tracking of moving targets in uncrowded areas. To enhance the capability of this constellation, a second set of satellites could be launched in 2015.

The Orbcomm constellation is an example of a constellation of communications satellites that was deployed in a staged manner. A short history of this system is presented in [LWJ00]. The Orbcomm constellation started its service even though all of its satellites were not deployed. Satellites were added through time, in accordance with a schedule defined in advance. The advantage of this approach is that the system started to generate revenue very early. However, decision makers did not take into account the evolution of the market and did not adapt their deployment strategy. Orbcomm finally had to file for Chapter 11 protection after a few years of service.

Kashitani [Kas02] compared the performances of systems in LEO and MEO orbit and their behavior with respect to different levels of demand. His conclusion was that the elliptic systems were more likely to adapt to market fluctuations because they could adjust their capacity by deploying sub-constellations. Having the ability to add "layers" to a constellation with sub-constellations could allow efficient staged deployment strategies.

1.2.4 Orbital Reconfiguration

The term "reconfiguration" for constellations of satellites has been principally used to designate the set of necessary maneuvers to recover service after the failure of a satellite. Ahn and Spencer [AS02] studied the optimal reconfiguration for constellation of satellites after the failure of one of the satellites. Reconfiguration as a form of flexibility has been presented by Saleh, Hastings and Newman [SHN01] with the Techsat21 example. This Air Force Research Laboratory program consists of a constellation of satellites able to reconfigure the geometry of the different clusters. By modifying this geometry, the system changes its capability from a radar mode to a geo-location mode. An application of reconfiguration for military purposes can use the work of Henry and Sedwick [HS01]. They propose to put the satellites into resonant orbits. The particularity of those orbits is that over an integer number of days, the satellites fly over the same regions of Earth. The orbital planes can drift using small variations of the altitudes of the satellites and exploiting perturbations caused by imperfections in Earth's geometry and mass distribution. In particular, the constellation could be reconfigured by drifting all the orbital planes to focus the capacity on a particular area. This flexibility could be very interesting with constellations of radars when a conflict arises on a particular region of the globe.

1.2.5 Valuation of flexibility in Other Domains

The staged deployment strategy has been considered in different domains. Ramirez [Ram02] studied the value of staged deployment for Bogota's water-supply system. Three valuation frameworks were compared: net present value (NPV), decision analysis (DA) and real options analysis (ROA). Kalligeros [Kal03] used real options analysis to study the opportunity of reorganizing the structure of a facility. The real options considered give to managers the flexibility to change the use made of the facility. In particular, in this model, decision makers can choose not to use a particular space. This framework was applied to the design of the Exploration Headquarters for British Petroleum in Aberdeen, Scotland. Takeuchi *et Al.* [TSN$^+$00] proposed to build the International Fusion Materials Irradiation Facility (IFMIF) in a staged manner. The full performance of the facility is achieved gradually in three phases. They showed that this approach reduces the overall costs from $\$M797.2$ to $\$M487.8$.

1.3 Organization of the Chapters

The thesis consists of six chapters. Chapter 2 focuses on the value of flexibility in a design. The traditional approach and the Net Present Value valuation is presented first. Then, the economic mechanisms that give value to flexibility when there is uncertainty in future demand are presented. Two methods to value flexibility, Decision

Analysis and Real Options analysis, are presented. Finally, we propose a framework based on those methods to quantify the economic opportunity of staged deployment for systems facing uncertainty in future demand.

Chapter 3 applies the framework to LEO constellations of communications satellites. The method is applied using an existing computer program that simulates the performance, cost and capacity of communications satellite constellations. The adaptations made to the simulator are presented first. Then, the different steps to apply the framework are presented. The flexibility studied is the ability to reconfigure the constellation after its initial deployment to increase the global capacity of the system. The optimization process over the different deployment strategies is finally presented. This chapter also provides recommendations to simplify the required computations.

Chapter 4 uses the framework to analyze the economic opportunity of staged deployment for a case similar in its requirements to the Iridium constellation. The sensitivity of the value of flexibility with respect to the discount rate and the uncertainty in future demand is presented and discussed.

Chapter 5 proposes a framework to price the reconfiguration process. A general modeling of the problem is proposed that can adapt to any possible technical solutions. Two technical solutions to allow the transfer of the satellites are considered. The first one proposes to add extra propellant to the satellites and the second one to use a space tug to maneuver the satellites. The way the modeling needs to be adapted to price those solutions is introduced. Finally, the problems that are not covered by the framework and that need to be taken into account in future works are listed.

Finally, Chapter 6 summarizes the findings, identifies the limits of the analysis and sets recommendations for future studies in this field.

The roadmap of this thesis can be found in Figure 1-2.

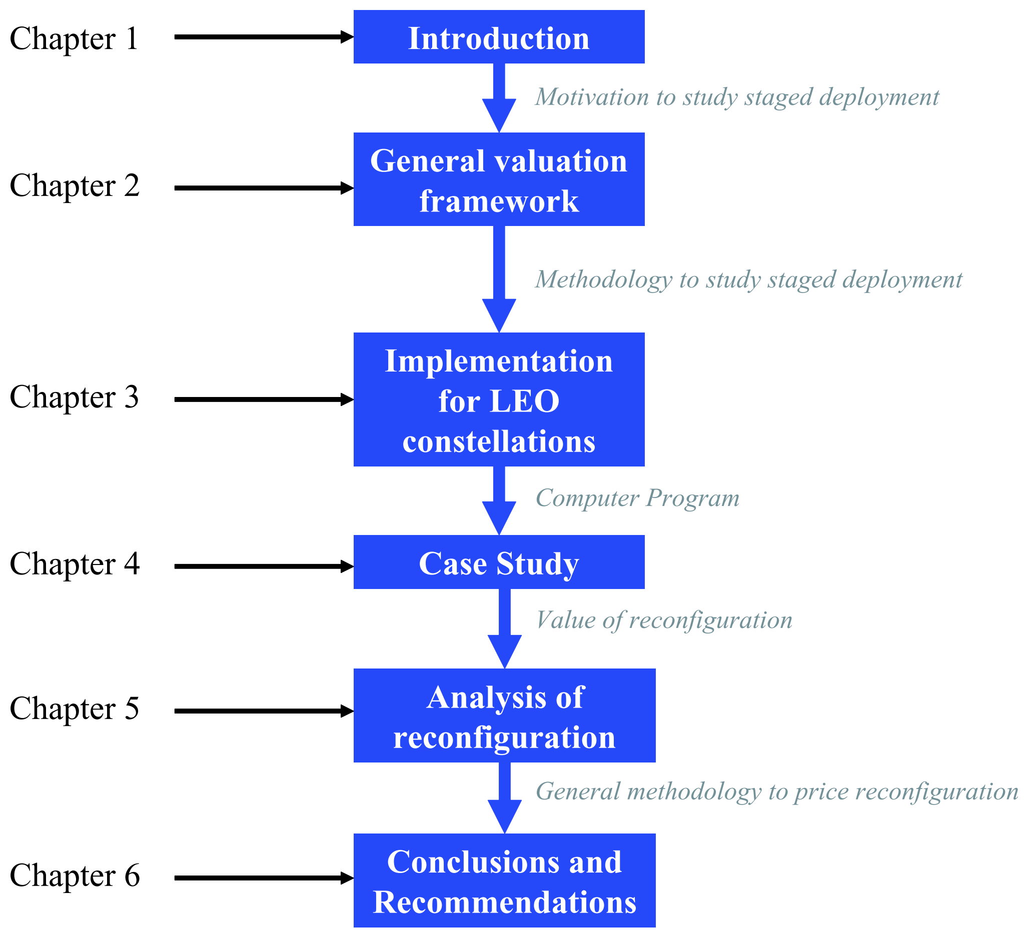

Figure 1-2: Roadmap of the thesis.

Chapter 2

Value of Flexibility for Capacity Planning under Uncertainty

2.1 Problem Definition

Large capacity systems often face high uncertainty in future demand. The main reason is that the development process can spread over many years during which many modifications in the targeted marketplace may occur. To identify the potential sources of uncertainty on the market place, it can be interesting to use Michael Porter's five forces model for competitive strategy (see Figure 2-1). For space systems, the forces can be described in the following way:

- **Buyers/Customers:** with time, the customers' needs are likely to change and the technology of the system may become obsolete by the time it is deployed. Moreover, the demand curves of customers may evolve quickly with time and the price people are willing to pay for a service decreases with time. If the prices have not been estimated correctly and are too high, only a few customers will be attracted.

- **Suppliers:** the suppliers for space systems are mainly spacecraft manufacturers and launch vehicles operators. They directly affect the performance of the system and the time required for development. Any delay in manufacturing or

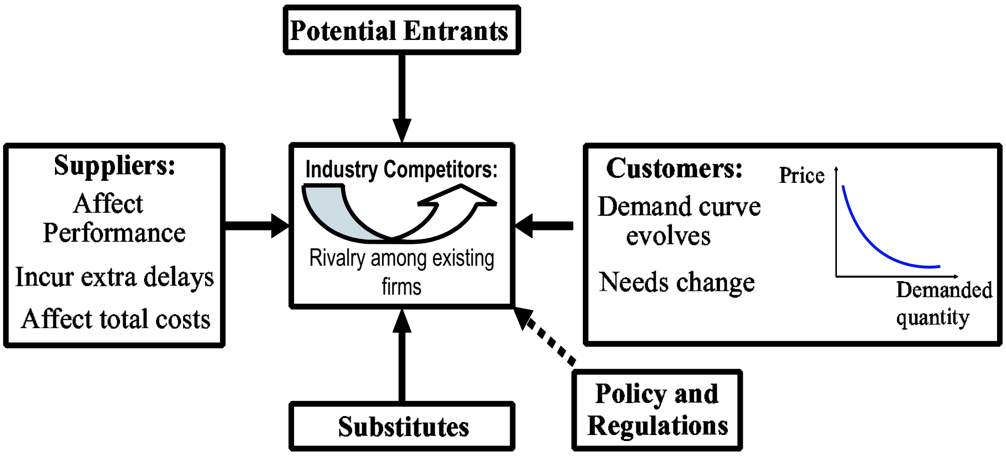

Figure 2-1: Adapted form of Michael Porter's five forces model for competitive strategy (from [Por98]).

any launch failure will extend the deployment time. Any flaw in the performance after the deployment of the system may result in a loss of customers.

- **Industry Competitors:** if the system is in competition with other systems, it can lose part of the market, thus reducing the expected number of customers.
- **Potential Entrants:** New systems with better performance or lower prices may attract a percentage of the market, thus reducing the number of customers.
- **Substitutes:** New technologies that can provide the same services as space systems can also attract an important portion of the market. A long development time for a system increases the possibility of substitutes arising in the marketplace. This is what happened to Low Earth Orbit Constellations with the successful development of terrestrial cellular networks.
- **Policy and Regulations:** This force was added on the side of the diagram because it affects all of the five main forces. Changes in policy or regulations concerning the technology may affect the development phase and impact delays. An example of policy impact is export control, which can prohibit the use of certain foreign launch vehicles. Another example is the allocation of frequencies for communications by the ITU and the FCC. However, it has to be noted that

certain policies could reveal helpful, even though it is unlikely.

There exist two main strategies to deal with uncertainty: robustness and flexibility. Robustness consists in designing a system that is not affected by variations in uncertain parameters. The system designed is fixed and is not modified to adapt to uncertainty. Flexibility gives the ability to a system to adapt to uncertain parameters. Saleh [Sal02] represented the relation between flexibility and robustness of a design as a function of the system's environment. It has been represented in Figure 2-2. The system is thus modified with the variations of the uncertain parameter. The distinction between flexibility and robustness is not always clear for designers. Robustness is often considered as a form of flexibility in the sense that it can endure any outcomes. Actually, there is a subtle conceptual difference as Ku [Ku95] explains:

> "Flexibility means the ability to change by (quickly) moving to a different state, selecting a new alternative or switching to a different production level. Robustness on the other hand is associated with not needing change. While flexibility is a state of readiness, robustness is a state of being. Flexibility and robustness are not opposite or the same, but two sides of the same coin, two ways of responding to uncertainty."

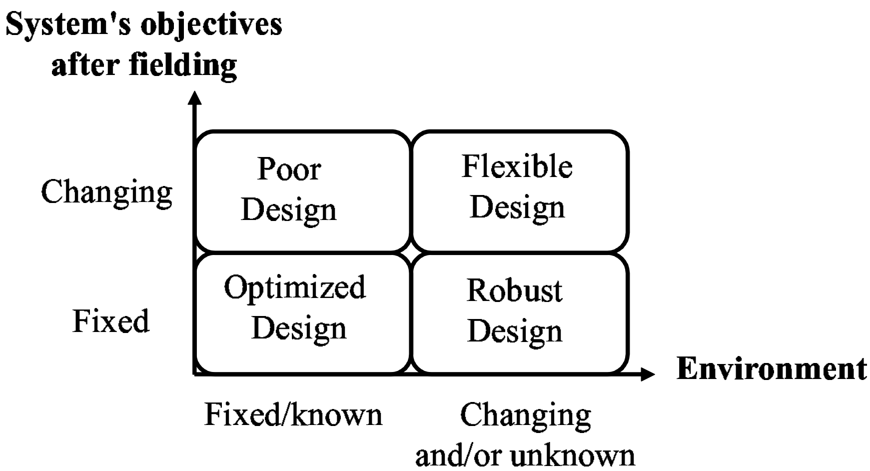

Figure 2-2: Flexibility and Robustness as a function of the system's objectives and environment (from [Sal02]).

The opposition between those two strategies is a classic problem illustrated in the 17th century by the French poet Jean de La Fontaine [dLF43] with *The Oak and the Reed*:

The Oak spoke one day to the Reed
"You have good reason to complain;
A Wren for you is a load indeed;
The smallest wind bends you in twain.
You are forced to bend your head;
While my crown faces the plains
And not content to block the sun
Braves the efforts of the rains.
What for you is a North Wind is for me but a zephyr.
Were you to grow within my shade
Which covers the whole neighbourhood
You'd have no reason to be afraid
For I would keep you from the storm.
Instead you usually grow
In places humid, where the winds doth blow.
Nature to thee hath been unkind."
"Your compassion", replied the Reed
"Shows a noble character indeed;
But do not worry: the winds for me
Are much less dangerous than for thee;
I bend, not break. You have 'til now
Resisted their great force unbowed,
But beware."
As he said these very words
A violent angry storm arose.
The tree held strong; the Reed he bent.
The wind redoubled and did not relent,

Until finally it uprooted the poor Oak
Whose head had been in the heavens
And roots among the dead folk.

The lesson taught by this poem is that the robust approach may fail to take into account certain outcomes and thus may not be able to adapt to them. On the other hand, the flexible approach giving the ability to adapt to unforeseen conditions may be able to deal with more situations. This shows how those approaches change the relationship between uncertainty and risk. The robust approach will not try to use uncertainty but will try to forecast all the outcomes as well as possible. If that were possible, there would not be any real uncertainty. That is why this approach fails to reduce risks. Flexibility will not try to understand uncertainty but admit that it exists and find ways to adapt to unforeseen outcomes. This adaptation reduces the risks of the project since many responses will be available for the system to particular conditions. Flexibility will benefit from uncertainty, while robustness will suffer from it. however, embedding flexibility does not generally come for free.

The way projects are valued will lead to one of those two approaches. It is thus important to know the different valuation methods that exist. This chapter first presents the traditional approach that tries to design for a fixed capacity system and discusses the influence of uncertainty on the selected designs. Then, it introduces the concept of flexibility for a system and analyzes the economic advantages it presents in uncertain contexts. To value flexibility of a system, the chapter presents two important methods: decision tree analysis and real options analysis. Finally, it proposes a framework to study the value of staged deployment for large capacity systems.

2.2 Traditional or Net Present Value Approach

This section presents the traditional approach that relies on discounted cash flow methods to valuate projects. The most common methods are Net Present Value and Internal Rate of Return. This section introduces only Net Present Value since it is widely used for investment decisions and explains how it influences the design process.

2.2.1 Net Present Value (NPV)

If one were to invest in a project, the first thing he or she would try to know is if the project is worthwhile, that is to say if it will bring more money than it will cost. An intuitive and simple method would be to look at the expected cash flow of the project, sum all the cash receipts then subtract all the expenditures and see if the final value obtained is negative or positive. Unfortunately, this approach fails to take into account the time value of money, that is to say that a dollar now is worth more than a dollar tomorrow. This difference is due to the productivity of money. If one has dollar now, he can invest and get greater amount of money later. To mathematically represent this time value of money, the discount rate r is introduced. The discount rate should equal the rate of return of equivalent investment alternatives in the capital market place. This means that if one chooses a discount rate of r percent per years, having Q dollars now is equivalent to receiving $Q(1+r)$ dollars in one year. Inversely, the present value (today) of Q dollars in one year is $\frac{Q}{(1+r)}$. Assuming that the discount rate r stays constant in time, a generalization of this statement consists in writing that the present value of a quantity Q whose payoff occurs in T years is:

$$PV(Q) = \frac{Q}{(1+r)^T} \tag{2.1}$$

This example concerned cash receipts but Present Value calculations are also acceptable in the case of expenditures, that is to say when $Q \leq 0$. In this case, the main argument is that a dollar spent tomorrow is worth more than a dollar spent next year since this money can be invested differently for one more year.

Now that different amounts can be compared with respect to their position in time, in order to decide if one should invest in a project, a discount rate r is selected and the present value of the expenditures is subtracted from the present value of the expected cash receipts. This is exactly what the Net Present Value method does and it can be summarized in the following way:

$$NPV(Project) = PV(RECEIPTS) - PV(EXPENDITURES) \tag{2.2}$$

Net Present Value offers a fast way to decide on whether or not to invest in a project: only the sign of its NPV has to be considered. It also allows an easy comparison of projects that are totally different, such as investing in the construction of a bridge or buying bonds. NPV is a very broadly used method because of the ease of the calculations and the simple selection rule it provides when comparing projects. But one has to keep in mind that the calculations of the Net Present Value of a project implies looking at a fixed cash flow. Consequently, the projects considered are fixed. This approach is often used when comparing different fixed architectures for a system and the next part shows how it affects the design process.

2.2.2 Traditional Selection of an Architecture

The traditional approach for designing a system performs an optimization to meet a set of requirements in order to obtain a certain capacity. This approach encourages the selection of architectures that are fixed over time because the requirements themselves are fixed. To illustrate the response to uncertainty of the traditional way of designing systems, this subsection introduces the concept of trade space of architectures. In particular, it shows which architecture is selected by the traditional approach in the trade space.

The objectives relevant to decision makers are the Net Present Value or the expected costs for an architecture but also its performance or capacity since it determines the size of the market that can be assessed. The trade space is a representation of the life cycle costs and capacities of a set of architectures. Creating a trade space involves having a simulation tool able to compute the objective vector, $\mathbf{J}$, given a design vector, $\mathbf{x}$, that mathematically represents an architecture. A framework to create such a simulation tool and implement a system architecture evaluation is given by Jilla [Jil02].

A trade space can contain an important number of architectures. However, for the traditional approach, only the ones that minimize the life cycle costs for a given capacity present an interest. Such architectures are not dominated by any of the architectures. To explain this notion a simple illustration is used. Five architectures, A_1, A_2, A_3, A_4 and A_5 are considered. Figure 2-3 represents their cost versus their performance. The selection of a fixed architecture is an optimization problem where

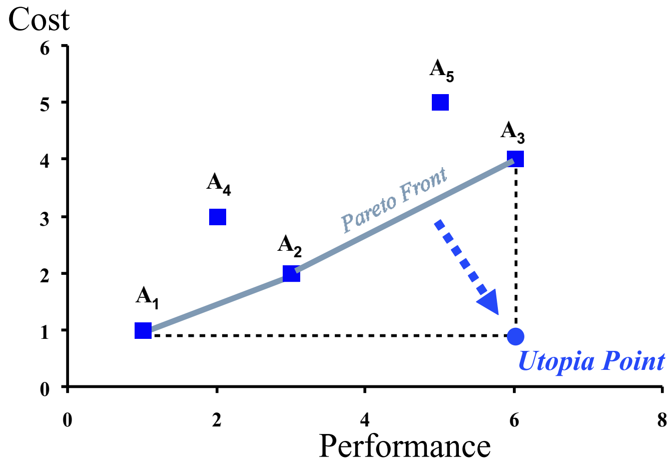

Figure 2-3: Pareto Frontier and Utopia point for a Trade Space composed of 5 architectures.

one tries to have maximum performance for minimum cost. This is represented by an arrow pointing in the bottom right direction toward an imaginary point called Utopia Point. Some architectures look less interesting than others: A_5 provides less performance than A_3 for a higher cost. With the traditional point of view, A_3 will always be preferred to A_5. Architecture A_5 is thus dominated. Similarly, A_4 is dominated by A_2. A formal definition of dominance is given by Steuer [Ste86] when there are k objectives. Let J^1, $J^2 \in R^k$ be two criterion (objective) vectors. Then, J^1 dominates J^2 weakly iff $J^1 \geq J^2$ and $\exists i$ such that $J_i^1 > J_i^2$. Architectures A_1, A_2 and A_3 are nondominated: it is impossible to reduce the cost while increasing the performance of those architectures simultaneously by selecting another architecture. Those nondominated architectures are called Pareto optimal. The set of all Pareto optimal architectures is called the Pareto set and is represented by connecting the architectures A_1, A_2 and A_3. The Pareto set is also known as the cost function (see [dN90]). It is really useful since it can reduces a vast trade space to a subset of relevant architectures.

The result of the traditional design can be illustrated with the concept of trade space. Given a capacity, designers seek an architecture that provides this capacity for a minimal cost. This is equivalent to projecting vertically this capacity on the Pareto front and look for a Pareto optimal architecture that is the closest from this projection and that provides a capacity at least equal to the one desired. An example for LEO constellations of communications satellites is represented in Figure 2-4. A capacity equal to 10^5 thousands of users is desired. It is projected on the Pareto front. The Pareto optimal architecture that satisfies the requirements and is the closest from this capacity is selected. From this architecture, the cost of the project is obtained. It is 9.5B$ in this particular example.

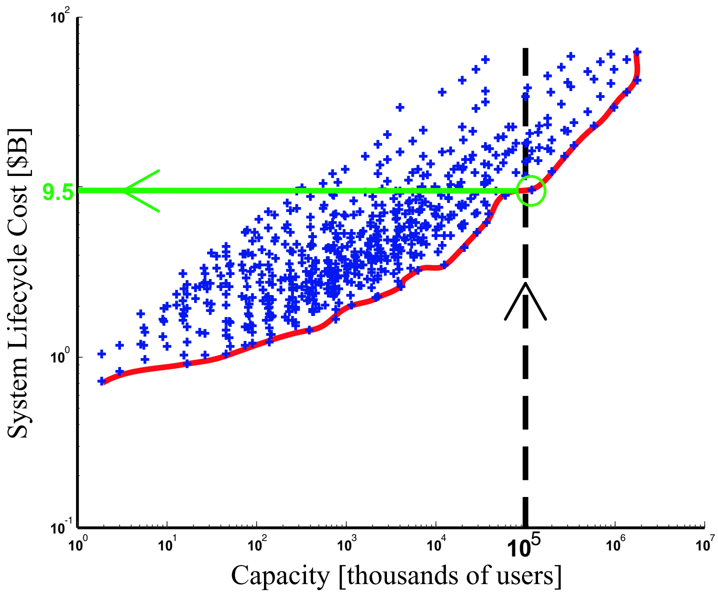

Figure 2-4: Traditional selection of an architecture for a desired capacity in the case of LEO constellations.

2.2.3 Response to Uncertainty in Demand

The traditional approach for selecting architectures implies designing for a fixed target capacity. When demand is uncertain, this capacity can be difficult to estimate and the risks appear to be significant. Indeed, if the capacity selected is too large and demand does not grow, the revenues may not be sufficient to balance the initial investment. This situation has been represented for the example of LEO constellations in Figure 2-5. In this example, future demand has a probability density function. If the actual demand is equal to 10^4 thousands of users, a system with a smaller capacity should have been designed. This results in a certain waste which is the difference between the life cycle costs of the system selected and the one of the system that is optimal for a capacity of 10^4 thousands of users. The waste in this case represents 6.3 $B. On the other hand, if the capacity is too small, a market opportunity may be missed. In the example developed, it corresponds to an actual demand of 4×10^5 thousands of users. The result is that a lot of money is invested in forecasts that do not reduce the risks mentioned. To respond to uncertainty, a fixed architecture must be able to cope with any predicted outcomes. Consequently, the Net Present Value approach will lead to a robust design that is to say a design that can deal with a "worst-case" scenario. The main risk of this approach is that the "worst-case" scenario for demand may not be forecast correctly. Moreover, this can lead to designing a large capacity with high initial deployment costs that may suffer from low revenues during the first years following the start of service. Consequently, by considering fixed architectures, the NPV method does not completely reduce the risks associated with the uncertainty in future demand.

2.3 Value of Flexibility

2.3.1 Economic Advantage of Flexibility

A design is said to be flexible if it can adapt to unforeseen conditions. It implies the existence of decision points through time. At a decision point, the values of

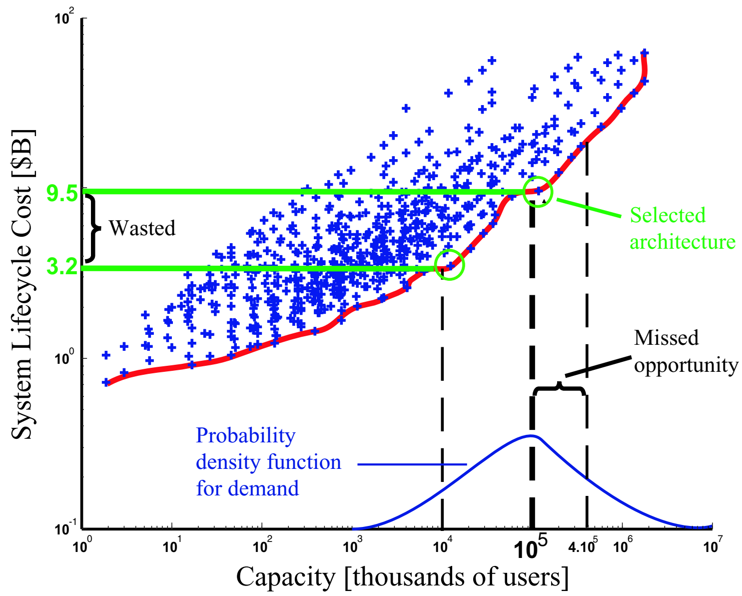

Figure 2-5: Response to uncertainty in demand of the traditional way of selecting a design.

parameters that were uncertain are analyzed. Depending on these values, a decision is made to adapt to them. Since uncertain parameters are observed through time, uncertainty is reduced, thus reducing the risks of the project. Decisions can be of various types, ranging from extending the life of a project to canceling its deployment.

This thesis focuses on the flexibility provided by staged deployment when demand is the uncertain parameter. This strategy deploys a system in a staged manner, starting with an affordable capacity that is increased when necessary. The decision in this case is about whether or not to move to the next stage in the deployment process. This decision is of course influenced by the market conditions but other circumstances can be taken into account such as the availability on new technologies that decision makers want to embed in the next stages or the presence of sufficient revenues to invest in the next stages. This approach presents an economic advantage compared to the traditional way of designing systems because it takes current market conditions into account. Two mechanisms explain this advantage and are described in this subsection. The staged deployment strategy tries to minimize the initial deployment costs by deploying an affordable system but, the expenditures associated with transition between two stages can be large. However, since those expenditures are pushed forward in time, they are discounted and are smaller in terms of present value. The first mechanism that staged deployment allows is that expenditures are spread in time. The second mechanism that is important is that the stages are deployed with respect to market conditions. If the market conditions are bad, there is no need to deploy the capacity further and nothing is done. The expenditures are kept as low as possible to avoid economic failure. On the other hand, if demand is large enough and revenues realized are sufficient, the capacity can be increased. The economic risks are considerably decreased with this approach since stages can be deployed as soon as they can be afforded and when the market conditions are good. The technological risks are reduced too since state of the art technologies could be integrated to the system as the stages are deployed.

This approach reframes the design selection process. It no longer designs for capacity but for flexibility. The questions that need to be solved are different and are discussed in the next subsection.

2.3.2 Issues Related to the Valuation of Flexibility

Designing for flexibility poses new issues for designers. Flexibility has a price so if it is not embedded wisely, the system may still be too expensive and result in an economic failure. Moreover, the price of flexibility is difficult to know. Indeed, the way flexibility is embedded in a system has to be distinguished from its effects. For instance, being able to deal with a flat tire is a flexibility that can be embedded in many different ways. Of course there could be a spare tire but another car or sufficient tools on board to repair the tire are other possible solutions.

Since the technology to embed flexibility may not be known, it can be interesting as a first step to look at the value flexibility can provide. This value can be defined as the economic advantage that flexibility provides over a fixed design. This number gives also the maximum price some people should be willing to pay for flexibility and may reveal an economic opportunity and encourage a further study on ways to embed flexibility. The valuation process is complicated because it takes into account uncertain parameters. A consequence is that the value of a form of flexibility is not inherent to it but depends on the context. To illustrate this, consider the example of a spare tire. If one lives in a suburban environment, a spare tire presents a certain value, a second spare tire has less value and a third spare tire has no value because the probability to have a flat tire is really low. On the other hand, in the desert, having one spare tire has less value than having three or four because the probability of having a flat tire is more significant. The value of flexibility depends on the underlying uncertainty that it tries to deal with. therefore, it needs to be represented accurately, which is a difficult task.

2.4 Valuing Flexibility Methods

This section focuses on two important methods to calculate the value of flexibility for a project or an architecture: decision analysis and real options analysis. They represent two different point of views concerning strategic planning. Decision analysis tries to look at all the possible outcomes for a project and determine the strategy that maximizes the average value of the project through time. On the other hand, real options analysis considers flexibility as an "option" that can be kept or abandoned and looks at its value with respect to time and uncertain parameters. It is important to understand these two point of views since the framework uses elements from both approaches.

2.4.1 Decision Analysis

Decision Tree

Decision Analysis (DA) looks at all possible scenarios for a project. A scenario designates a series of decisions that were made and events that occurred over the lifetime of a project. The set of all possible scenarios can be conveniently represented by a decision tree. The tree consists of nodes and branches. There exist two types of nodes (see Figure 2-6):

- **Decision nodes:** from this node leave as many branches as there are possible decisions. Decision nodes represent decision points in time where managerial flexibility is taken into account. A decision node cannot be connected to another decision node since a succession of two decisions can be gathered in a single decision.

- **Chance nodes:** branches leaving such nodes represent the different possible evolutions of the uncertain parameter considered. Each branch thus represents an outcome called event to which a probability of occurrence is attached. The events that are related to a same chance node have to be mutually independent. Moreover, these events have to represent all the possible outcomes. A consequence of those two properties is that, if P_i is the probability associated with event i and if there are k events, then $\sum_{i=1}^{k} P_i = 1$. Two chance nodes can be connected to enhance the representation of uncertainty of a parameter or to take several uncertain parameters into account.

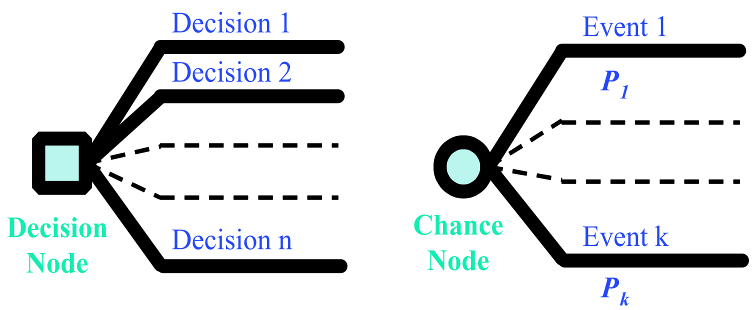

Figure 2-6: Basic nodes of a decision tree.

Decision nodes represent the flexibility of the system whereas chance nodes represent uncertainty. The tree has a unique initial node on the left that represents the initial situation for the project considered. A scenario is a path from the initial node to a terminal node on the right. A terminal node represents the situation at the end of life of a project for a particular scenario. The succession of nodes through which a path goes corresponds to the repartition in time of decisions and events for the associated scenario.

To illustrate the previous definitions, an example can be considered. In the roulette game, if bets are placed only on colors, there are three possible events: red (R), black (B) and green which corresponds to 0. There are 37 numbers in the roulette game, 18 are red, 18 are black, the last one being 0. If those numbers are considered evenly likely, the probability associated with the events R and B are the same: $P_R = P_B = \frac{18}{37}$. The probability of obtaining 0 is $P_0 = \frac{1}{37}$. The decisions that a gambler can make are red, black and 0. A gambler plays two roulette games and bets \$10 each time. If he wins, he gains 10 more dollars otherwise the bet is lost. A decision tree can be built for this case to represent all the possible scenarios. The basic structure of this decision tree is represented in Figure 2-7. The decision times correspond to the beginning of a game, when the player has to place a bet. A period corresponds to the bet and the outcome of the corresponding game. A scenario for which the player wins in the first game and loses in the second is also represented in the decision tree. The final gain for this scenario is equal to \$0 and this value is represented next to the corresponding terminal node.

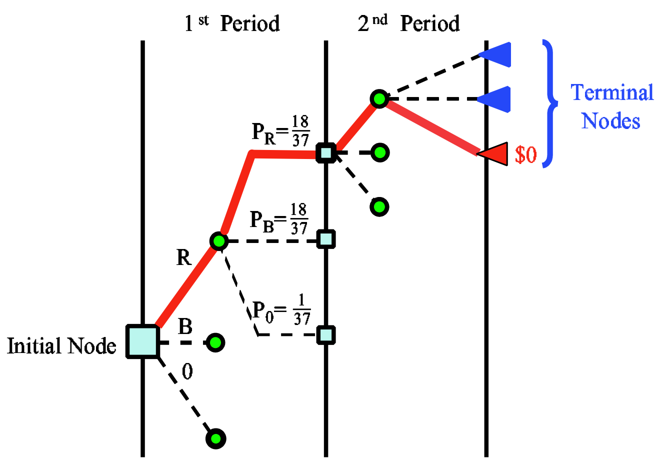

Figure 2-7: Structure of the decision tree corresponding to the roulette example.

The decision tree is a powerful way to represent all the possible scenarios for a project and manages to take into account both uncertainty and flexibility in the same representation. The next section presents how it determines the value of a project and a complete strategy.

Value of Flexibility and Strategy

For each scenario on the decision tree, the objective of interest can be computed. For instance, in the roulette example, the final gain of the gambler could have been computed. Decision analysis tries to find the decision that will optimize the expected value of the objective considered at each decision node. Consequently, the tree needs to be solved backwards with the following process:

1. **Initialization**: the objective that is maximized (resp. minimized) is computed for each scenario and written on the associated terminal nodes. This defines an initial column (see Figure 2-8). The algorithm calculates a new column of values at each iteration moving from the right of the tree to the left.

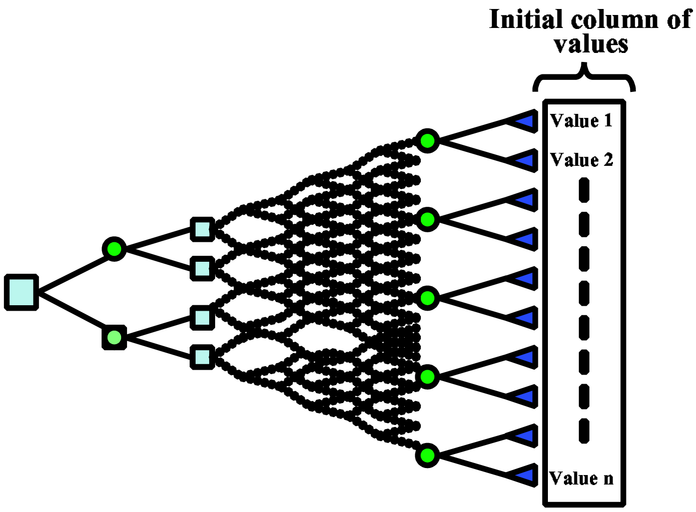

Figure 2-8: Initial column of values of the decision tree.

2. **Computations for chance nodes**: if the column of values considered is connected on the right to chance nodes, the expected value of the branches of each chance node are computed and written on top of the chance node (see Figure 2-9).

3. **Computations for decision nodes**: if the column of values considered are connected on the right to decision nodes, the branch that leads to a chance node with the maximum (resp. minimum) expected value for the objective is

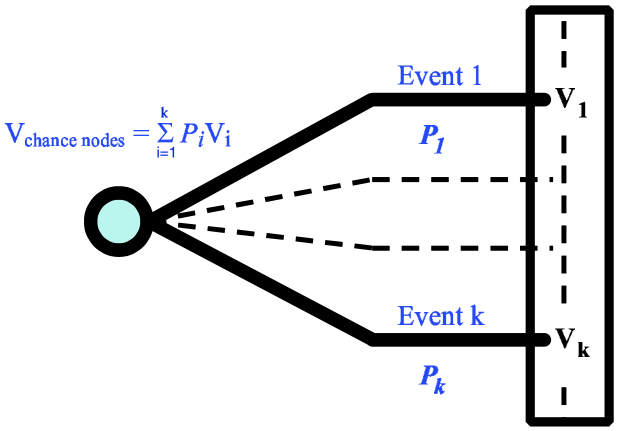

Figure 2-9: Computation of the value of a chance node from the column of values.

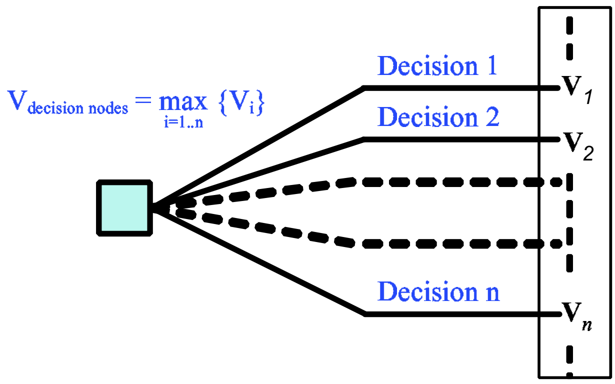

Figure 2-10: Computation of the value of a decision node from the column of values.

kept and this expected value is written on top of the decision node. The other branches are "cut" that is to say removed form the decision tree (see Figure 2-10).

4. **Termination**: the algorithm terminates when the initial node is reached. The value associated with this node represents the expected objective for the project.

At the end of this algorithm, the decisions that maximize the objective are the only ones remaining in the tree but all possible events are present. Consequently, once solved, the tree defines a best strategy: decision makers just have to follow the branches associated with the different events and make the decisions that are in the tree. This value can be compared to the one that a fixed design would give, thus providing the value of the flexibility studied.

Analysis of the Method

Decision analysis is easy to understand because it provides a simple decision rule and a clear representation of the relation between uncertainty and flexibility. But the complexity of the tree depends directly on the number of decisions available and the representation of uncertainty. The tree can easily get complex and impossible to read. Moreover, the more detailed the tree is, the more scenarios will have to be considered which can be expensive in terms of computation. If decision analysis is applied to compare a set of different architectures, it can be seen how complex the calculations will be because a tree will have to be built and solved for each of the architectures. Even if the tree does not have to be built manually, the computational time can become prohibitive.

The method minimizes the economic risk by always selecting the branches that maximize the expected value of the system. However, it fails to represent the exact value that may be expected from the project because it considers expected values. Indeed, it provides a weighted average of the possible outcomes but the value obtained may not represent any of the outcomes. This issue can be illustrated with a simple example. The price of a ticket to participate in a lottery is \$30. The probability of winning \$100 is $p_{win} = 0.7$ and the probability of losing the \$30 paid for the ticket is thus $p_{lose} = 1 - 0.7 = 0.3$. Decision analysis provides an expected gain equal to $0.7 * \$100 - 0.3 * \$30 = \$61$. However, the gain is never \$61 with any of the two outcomes. Actually, a participant could lose \$30. Consequently, it is not because the expected value is positive that the possible outcomes will provide a positive value. This is why decision analysis needs to be used with a lot of caution.

Consequently, decision analysis is really useful to find a best strategy for a system that is already known, but may be difficult to implement when comparing different projects. It provides a clear presentation of the link between uncertainty and managerial flexibility. However, the values provided by decision analysis need to be interpreted with caution because they represent the expected value from several possible outcomes.

2.4.2 Real Options Analysis

Option Theory

Real options analysis has been inspired by financial options theory. There exist two main types of options: calls and puts. A call option gives the right to the holder to buy a particular asset at a certain price by a certain date. Inversely, a put option gives the right to sell a certain asset at a certain price by a given date. The price is called the strike or exercise price and is fixed on the day the option is acquired. The date, also called expiration date, exercise date or maturity, is fixed too but can represent two distinct things whether the option is American or European. American options can be exercised at any time until the expiration date. European options can only be exercised on the expiration date. A complete description of options can be found in [Hul89]. Options are a powerful way to deal with the uncertainty of the price of an asset because they give the holder the right but not the obligation to take an action in the future. This flexibility creates an asymmetry in the profit the holder can expect from the underlying asset.

This asymmetry can be illustrated with an example. Consider an asset whose current price is \$ 100 and a European call option with an exercise price of \$ 105 that expires in one year. The price of this option is \$10. If the terminal price of the asset is over \$ 105, the owner of the option exercises it since he can get this asset for \$ 105. If the terminal price is below \$ 105, he does not exercise the option. The profit realized as a function of the future price is represented in Figure 2-11. Even though the profit is negative when the terminal price is below the exercise price plus the price of the option (\$115), this deficit is at most equal to the price of the option. The value of the flexibility provided by the option is equal to zero if the terminal price is smaller than the strike price. However, for future prices higher than the strike price, the value of the option increases with the future price. Consequently, a call option is an initial investment with an asymmetric value: its future value is higher or equal to

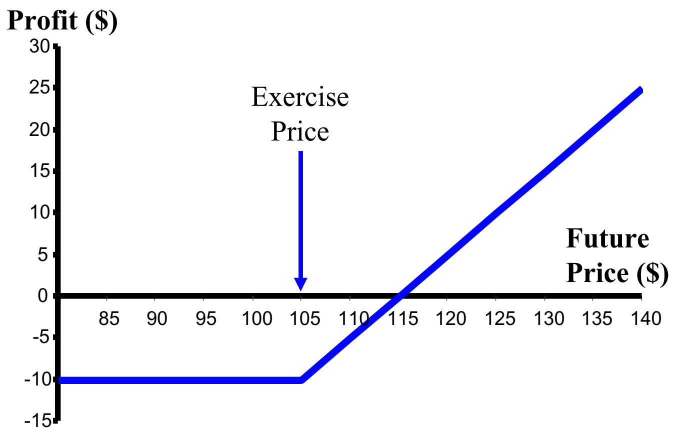

Figure 2-11: Profit made with the option for different future values of the stock price.

zero. Significant profits could be obtained from it and the maximum loss possible is equal to the price of the option.

This idea of having a right but not the obligation to take an action in the future is a form of flexibility. The fact that valuation methods existed in the financial domain for this flexibility encouraged seeking parallels with the engineering domain. Stewart Myers [Mye77] was the first to propose the term "real options" when using the financial concepts for the analysis of real assets. The next section presents the relationship between financial options and real options.

From Financial Options to Real Options

Financial theory proposes a different approach to flexibility. This approach has been translated into the engineering world to provide more accurate estimations of the value of flexibility. An option for a system would be a technical element embedded initially into the design that gives the right but not the obligation to decision makers to react to uncertain conditions. For instance, installing a docking device on a geostationary satellite to achieve on-orbit refueling and expand the life of a satellite if judged necessary could be an engineering option. Consequently, engineering options are physical elements. They have thus been called "real options in systems" to differentiate them from financial options or "real options on systems". This approach treats the physical artifact as a black box.

The analogy with the financial world can be pushed forward. For instance, technical reasons or the lifetime of the system may limit the use of the real option to a certain period of time. This corresponds to a time to expiration date T_{ex}. Moreover, the cost X to use the real option corresponds to an exercise price. The real option will be exercised only if the value it brings is higher than the exercise price. In the case of financial options, the comparison was made between the exercise price X and the actual stock price S that was subject to uncertainty. Real options compare the price to acquire the asset X with the value expected from it, S, that is also subject to uncertainty since it depends on external conditions. The parallel between the financial and engineering spheres is summarized in Table 2.1.

VARIABLE	REAL OPTION	FINANCIAL OPTION
S	Present Value of asset to be acquired	Stock or commodity price
X	Required investment to acquire option	Exercise Price (Strike)
T_{ex}	Time over which decision can be deferred	Time to expiration date

Table 2.1: Comparison of financial options and real options (adapted from [Ram02]).

In theory, the relationship between the financial and engineering world seems obvious. In reality, however, technical systems may take into account parameters that are not easily quantifiable and the use of real options can be difficult. In particular, S can be hard to define when it depends on parameters such as "users' satisfaction" and there is no clear methodology existing to solve this problem. The choice of the representation of S[1] is essential because it is through this variable that uncertainty is taken into account.

The main interest of this analogy with the financial world is that the existing valuation methodologies can be applied. The next section presents the basic underlying principles of option evaluation.

[1]In the particular case of LEO constellations of communications satellites, S will be used to represent uncertainty in future demand.

Principles of Option Valuation

The main argument that serves as a basis to options valuation is arbitrage pricing. Arbitrage consists in profiting from discrepancies between two markets on the price of an asset. When a difference is noticed between two prices, the asset is bought on the market offering the lowest price and sold on the other market. If the competitive markets are well-operated, such arbitrage opportunities should not exist. This implies that two assets that have the same risk distribution should trade at the same price and offer the same payoffs. This assumption allows a simplification of the pricing process for complex investments such as options.

A replicating portfolio that is subject to the same uncertainty as the underlying asset is created so that its return is independent of future outcomes. This portfolio has the exact same risk as the option that is considered. Therefore, the combinations of buying the portfolio and selling the option or selling the portfolio and buying the option are riskless. This method may look simple but it requires complex mathematical representations, in particular to represent the evolution of S with time. A classic assumption in finance is to say that the behavior of stock prices follows a generalized Wiener process also known as geometric Brownian motion. This section does not describe the mathematics of Wiener processes in detail but simply considers the discrete-time version of the model:

$$\frac{\Delta S}{S} = \mu \Delta t + \sigma \epsilon \sqrt{\Delta t} \tag{2.3}$$

S represents the current price of the stock, ϵ is a random variable with a standardized normal distribution and Δt the time step of the discretization. Consequently, $\frac{\Delta S}{S}$ represents the rate of change of the stock price during an small interval of time Δt. $\frac{\Delta S}{S}$ depends on a random variable consequently it is a random variable. μ and σ are constants in this formula. Their meaning can be understood with simple mathematical considerations. The expected value of the rate of change of S is:

$$E\left[\frac{\Delta S}{S}\right] = E\left[\mu \Delta t\right] + E\left[\sigma \epsilon \sqrt{\Delta t}\right] \tag{2.4}$$

$$= \mu\Delta t + \sigma\sqrt{\Delta t}\,\underbrace{E\left[\epsilon\right]}_{=0} \tag{2.5}$$

$$= \mu\Delta t \tag{2.6}$$

Consequently, if the current value of the stock is S, the expected variation of it in the time interval Δt is $\mu S\Delta t$. Therefore, μ is the expected return per unit time on the stock, expressed in a decimal form. Let's consider the variance of $\frac{\Delta S}{S}$:

$$var\left(\frac{\Delta S}{S}\right) = var\left(\mu\Delta t + \sigma\epsilon\sqrt{\Delta t}\right) \tag{2.7}$$

$$= \left(\sigma\sqrt{\Delta t}\right)^2 \underbrace{var\left(\epsilon\right)}_{=1} \tag{2.8}$$

$$= \sigma^2\Delta t \tag{2.9}$$

σ^2 is thus the variance rate of the relative change in the stock price. σ is usually called the volatility of the stock price. It is interesting to note that σ "scales" uncertainty in future prices. Moreover, the bigger the time step considered is, the bigger the variance of the relative change in stock prices will be. This mathematical property is a translation of the fact that uncertainty increases as the time horizon is farther from us. From this model, Black, Scholes and Merton (see [BS73] and [Mer73]) were able to derive a closed form solution for the price of an option given many assumptions. One of the main assumption behind the Black-Scholes equation is that the options considered are European options which is rarely the case with "real options". Most of the time, numerical procedures have to be used because no closed form solutions are available. The Wiener model involves random variables, consequently a common method is to run a Monte-Carlo simulation over the price of the stock. However, when the time intervals considered are big enough, a binomial model can be used as a time discrete representation of the stock price. The binomial model simplifies the Wiener model by stating that the stock price S can only move up or down during an interval or time leading to a new price Su or Sd (see Figure 2-12). There is a probability p to move up and a probability $1-p$ to move down. To be consistent with the Wiener model, this representation needs to provide the same expected return and variance

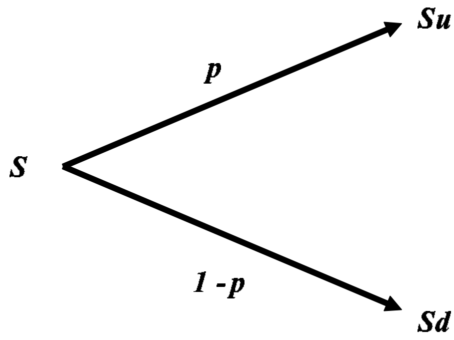

Figure 2-12: Basic structure of the binomial model.

when Δt gets close to zero. Hull [Hul89] demonstrates that it can be achieved by setting p, u and d in the following manner:

$$u = e^{\sigma\sqrt{\Delta t}} \quad (2.10)$$

$$d = \frac{1}{u} \quad (2.11)$$

$$p = \frac{e^{\mu\Delta t} - d}{u - d} \quad (2.12)$$

When used over many periods, the binomial model provides a tree for the price of the stock. An example of a binomial tree over four periods is represented in Figure 2-13. For each nodes of the tree, the value of the underlying asset has been computed. In this example, $\mu = 10\%$, $\sigma = 40\%$, $\Delta t = 1$ and $S = 100$. For those parameters, the values of u, d and p are $u = 1.49$, $d = 0.67$ and $p = 0.67$.

The binomial tree is useful for real options and allows an easy and systematic pricing methodology illustrated by an example. Consider a European call option on the opportunity to buy a stock in one year for \$21. The current price of the option is \$20 and, from its expected return and volatility, a simple binomial model is created. It is represented in Figure 2-14. The stock price will either move up to \$22 or \$18. If the stock price goes up, the value of exercising the option is \$1. If the price goes down, exercising the option does not have any value. From those considerations, the binomial tree is converted into an event tree. The event tree represents the best decision between keeping the option open (KO) or exercising the option (EX) at each

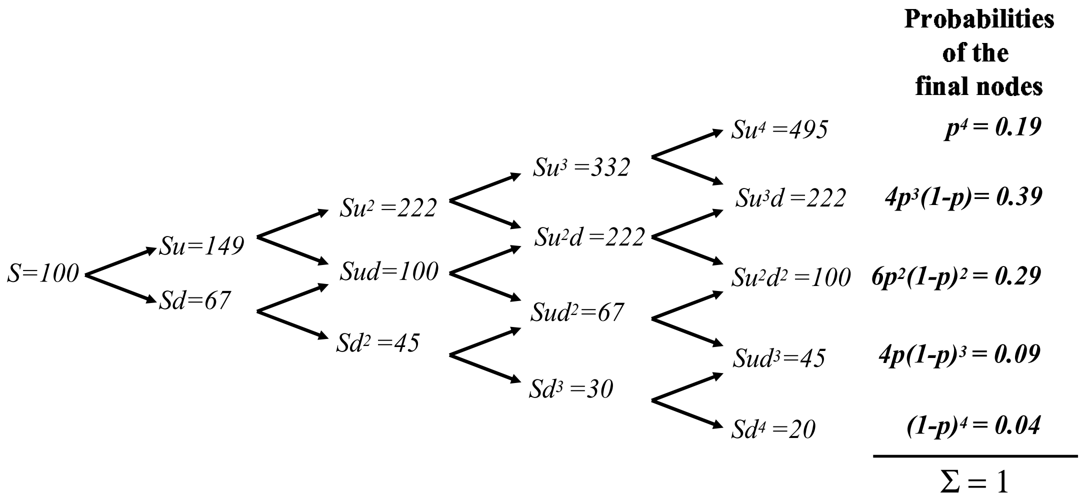

Figure 2-13: Example of a binomial tree and evolution of the value of the underlying asset over four periods.

node of the original binomial tree.

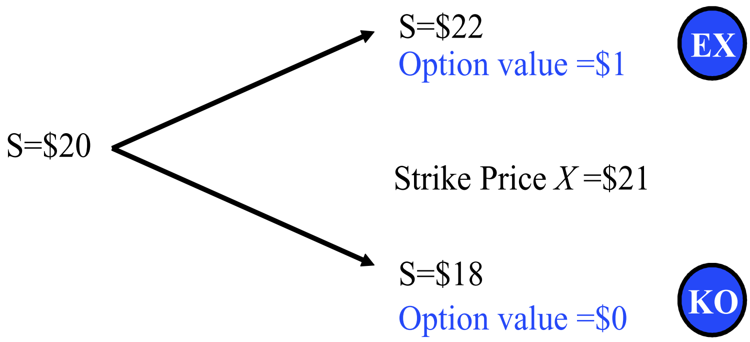

Figure 2-14: Value of the option and suggested decision for the possible values of the stock price.

The cost and payoff obtained with the call option considered are summarized in Table 2.2. To determine the price f of the option, a replicating portfolio with the same cost and payoff needs to be created with different assets. This portfolio consists of m shares of the stock and an amount B of money borrowed with an interest rate r_i. The stock and loan costs and payoffs have been represented for this portfolio in Table 2.3. This portfolio should provide the same costs and payoffs than the call option for

	Start (S=\$20)	End (S=\$18)	End (S=\$22)
Buy call option (X=\$21)	-f	0	1

Table 2.2: Call option cost and payoffs.

	Start (S=\$20)	End (S=\$18)	End (S=\$22)
Buy stock	-20m	18m	22m
Borrow money	B	$-B(1+r_i)$	$-B(1+r_i)$
Net	$-20m+B$	$18m-B(1+r_i)$	$22m-B(1+r_i)$

Table 2.3: Stock and loan costs and payoffs.

both outcomes. This leads to the following set of equations:

$$-f = -20m + B \tag{2.13}$$

$$0 = 18m - B(1+r_i) \tag{2.14}$$

$$1 = 22m - B(1+r_i) \tag{2.15}$$

$$\tag{2.16}$$

From Equations 2.15 and 2.16 the values of m and B are obtained: $m = 0.25$ and $B = \frac{9}{2(1+r_i)}$. The final value of f is obtained from Equation 2.14:

$$f = 20m - B \tag{2.17}$$

$$= 5 - \frac{9}{2(1+r_i)} \tag{2.18}$$

$$\tag{2.19}$$

For an interest rate $r_i = 10\%$, the price of the option is $f = \$0.909$.

It is interesting to point out that this determination of the price of the option does not use the probabilities associated with the down and up movements. Indeed, this method only considers the payoffs of the outcomes and not their probabilities of occurrence. It is an important result that marks a difference with decision analysis. The value of flexibility is consequently represented more accurately. Moreover, it provides a strategy represented by decision points where the real option should be exercised or not. The valuation for real options is more complicated but the principles are the same. The main difficulty is to find the volatility of the asset considered. A presentation of the issues and methods to solve them can be found in [Tri96]. Good examples of "real options" valuations can be found in [Lam01] and [Ram02].

The real options analysis presents many advantages but the implementation may seem sometimes difficult. Moreover, ROA is based on a financial framework that can be difficult to explain to engineers because the creation of a portfolio to value an option is less intuitive than the clear decision rules provided by Decision Analysis or NPV. However, without going through all the different implementation methods, the concept used for real options calculations can enforce a different way of considering flexibility. In general, the implementation of real options calculations depends on the particular aspect of a problem but the principle remains the same. Flexibility is considered as an initial investment and the opportunities it provides to decision makers is estimated. This represents a big change in the way flexibility is considered by decision makers. A "real options" approach in adopted in the framework proposed to study the economic opportunity of staged deployment. This framework is presented in the next section.

2.5 Valuation Framework for Staged Deployment

2.5.1 Presentation of the Problem

Staged deployment is a particular way of introducing flexibility in a system. It reduces the economic risks of a project by deploying it progressively, starting with a smaller and more affordable capacity than the one proposed by the traditional approach.

When there is enough money to increase the capacity or if demand for the service provided goes over the current capacity, the system is upgraded to a new stage with a higher capacity. This approach does not design for a target capacity but tries to find an initial architecture that will give to system managers the most flexibility to adapt to market conditions. However, it poses new challenges to designers. A first issue is that the possible evolution from an initial architecture has to be identified and understood. An ideal staged deployment would follow the Pareto Front but this is not necessarily feasible. Fundamentally this is true because Staged deployment implies the use of legacy components (the previously deployed stages) which reduces the number of degrees of freedom in the system. Consequently, the structure of the Trade Space and the relations existing between architectures have to be clearly defined and modeled. A second issue is that the price to pay to embed flexibility into the design is not known. The reason is that the technologies involved may not be known or accurately modeled.

The framework proposed solves all of those issues with different assumptions and models that are presented in the next sections. The notion of paths of architectures on which the method relies is first introduced as well as definitions used in the description of the framework. Then, the assumptions and principles of this valuation method are exposed. Finally, the general steps necessary for the implementation of the framework are detailed.

2.5.2 Paths of Architectures

When the concept of trade space was first introduced in the description of the traditional approach, the architectures were considered fixed over their lifetime. Staged deployment considers these architectures in a different manner: evolutions in the trade space are allowed. The evolutions that are possible from a given architecture depend on the flexibility that was embedded. In particular, only particular architectures can be obtained through evolution. Consequently, the initial architecture is not sufficient information to fully describe the system. To solve this problem, the framework considers only allowed evolutions called “paths”. This section introduces the concept of "paths" of architectures and other definitions that are used in the description of the framework.

A trade space is a mapping between a design space and an objective space. The architectures are represented by design vectors x and an associated objective $J(x)$. For instance, if the system considered is a bridge, the design variables could be the length, the width and the objectives the number of cars per days the bridge can support and its life cycle cost. However, the architectures are not fixed and can evolve through time in the trade space. This evolution is possible only if certain design variables can be modified after the deployment. This leads to a decomposition of the design vector into two parts (see Equation 2.20):

- x_{flex} gathers all the design variables that provide flexibility to the system. This means that a real option is considered that allows a modification of those variables.
- x_{base} represents all the design variables that cannot be modified after the deployment of the system. Those variables may not be changed for physical or for strategic reasons: system designers may not want to consider the potential flexibility provided by certain variables.

$$x = \begin{pmatrix} x_{flex} \\ x_{base} \end{pmatrix} \tag{2.20}$$

From an initial architecture x^0, evolution is possible only toward architectures x such that $x_{base} = {x^0}_{base}$. For this reason, the trade space is divided into sets of architectures that share the same vector x_{base}. Those particular sets are called families. The set of all families form a partition of the trade space that sometimes allows simplifications in the implementation process of the framework. The variables of x_{flex} may be changed according to certain evolution rules. It is important to identify them since they will define feasible evolutions. Those rules can concern the variables itself. For instance, a design variable may be modified only by increasing its value or decreasing it. The evolution rules can also concern the relations between variables. In fact, problems of priority or simultaneity may arise between certain variables that need to be taken into account.

This thesis focuses on staged deployment to adapt the capacity of a system to market conditions. This can be done in two different ways. A first way would be to increase the capacity when necessary, a second would be to aim at a perfect adaptation, increasing and decreasing the capacity to get as close as possible to the actual demand. The first approach is interesting when the development costs and the cost of evolution are significant compared to operation costs. In this situation, decreasing the capacity does not present any value since the reduction in operation costs expected may be smaller than the investment necessary to decrease capacity. The second approach is applied when the operations costs represent the most important part of the expenditures. An example would be a taxi company for which capacity depends on the number of available drivers or cars and expenditures are mainly the wages and the fuel. The systems considered in this thesis are large capacity systems with low relative operations costs. Consequently, the first approach will always be the one considered in the framework. This implies that if $Cap(x)$ is the capacity of a given architecture, one will always consider evolutions from an architecture x_i to an architecture x_j such that $Cap(x_i) < Cap(x_j)$. "Path" will thus designate a series of architectures $(x_1, x_2, \ldots, x_n)$ such that:

- all of the architectures are in the same family, i.e. $(x_i)_{base} = (x_j)_{base}$ $\forall i, j \in \{1, 2, \ldots, n\}$ and $i \neq j$;
- $\forall i \in \{1, \ldots, n-1\}$, the evolution from x_i to x_{i+1} does not violate any evolution rules;
- $\forall i \in \{1, \ldots, n-1\}$, $Cap(x_i) < Cap(x_{i+1})$.

An example of a path is represented in Figure 2-15. Because of all the constraints that a path needs to satisfy, it is usually impossible to have a path that follows exactly the Pareto frontier. Therefore, the valuation will have to compare many possible paths and perform an optimization over them. The selection of a system differs from the traditional approach in this framework. Fixed architectures are not considered anymore but rather paths that represent the available capacities for the system and the deployment strategy at the same time.

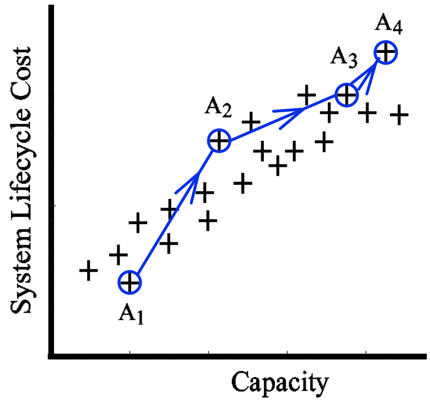

Figure 2-15: Example of a path of architectures in a Trade Space. The series of architectures (A_1, A_2, A_3, A_4) represents the path.

2.5.3 Valuation of Flexibility

Assumptions

The framework proposes the value that staged deployment offers when market conditions are uncertain. To solve this problem, several assumptions are made. First, the system should be able to provide service for a maximum demand. This means that a maximum capacity Cap_{max} is set and that a path can be considered if and only if at least one architecture has a capacity higher or equal to Cap_{max}. This assumption allows a selection of the paths that will be considered but also defines the traditional design with which the final solution is going to be compared with. A second assumption is that the system tries to adapt to demand. As soon as demand is higher than the current capacity, the next stage of the system is deployed. This corresponds to a worst-case strategy for staged deployment since it corresponds to the maximum expenditures for the system. Consequently, if an opportunity is revealed for a worst-case scenario, it can safely be assumed that the real option considered presents value. This assumption is natural for systems that provide a service without trying to generate profits. For instance, for a water supply system, it is vital that the system adapts to demand. A third assumption is that the price to embed flexibility into existing designs is not taken into account. Eventually, the cost of the optimal staged deployment solution will be compared to the cost given by the traditional design. If the staged deployment strategy is cheaper, then the difference with the traditional design will give an estimate of the maximum price one should be willing to pay to embed such flexibility. Consequently, this approach tries to reveal opportunities rather than studying their technical details. This can motivate research for technically embedding flexibility into systems. A last assumption is that demand follows a geometric Brownian motion. In particular, the binomial model can be used.

Valuation Principles

The goal of the framework is to compare the life cycle costs obtained with flexibility with those obtained with a traditional design. The traditional design is a fixed architecture that minimizes the life cycle costs and with a capacity at least equal to Cap_{max}. Once this design and its objectives are determined, each path is considered. Demand scenarios are generated with the geometric Brownian motion model and the life cycle costs are estimated for a path when adapting to demand. The expected value of those life cycle costs is then compared to the life cycle cost obtained with the traditional approach. If the path provides a smaller life cycle cost, then an economic opportunity is revealed. Otherwise, flexibility does not present any value and the proposed path should not be considered for staged deployment. Considering many paths, an optimization over life cycle costs can be done to find a best path. This best path will define the best deployment strategy since it describes the series of architectures that should be followed. The difference between the cost of the traditional design and the optimal life cycle cost obtained with staged deployment gives an approximation of the maximum price designers should be willing to pay to embed flexibility into the design. Therefore, even though the technical way to embed flexibility may not be known, an estimate of the maximum price one should be investing to develop this technology can be obtained.

This valuation method tries to reveal economic opportunities of "real options" that may not be known in detail. This is one of the main interest of the approach. To complete it for commercial systems, a decision analysis or a real options analysis could be carried out to see how the profits could be maximized with the optimal path. This really simplifies the process since only one decision tree has to be created and the decision nodes are easy to represent since they have a maximum of two branches: one representing the fact that the system is deployed to the next architecture in the path, the other represented the decision to keep the design the way it is.

2.5.4 Implementation

This section describes the main steps through which the implementation of the framework has to go through. Those steps have to be adapted to the particular system that is studied but the principle remains the same.

Identification of Flexibility

The sources of flexibility that designers want to study have to be identified. Once the design variables that provide flexibility are identified, the design vector should be decomposed as in Equation 2.20. For the design variables in x_{flex}, evolution rules have to be identified and modeled. Finally, the Trade Space should be divided in families of architectures.

Transition Matrix and Initial Development Costs

From the trade space, the costs for development and deployment of each architecture should be computed. $IDC(x)$ will designate the initial development costs of an architecture represented by a design vector x. The evolution costs should be also estimated for each possible evolution. A convenient way to store those costs is the introduction of a transition matrix ΔC. The transition matrix is defined in the following manner: if x_i and x_j are two architectures, then $(\Delta C)_{(i,j)}$ is equal to the expenditures necessary to go from x_i to architecture x_j if this evolution is feasible and to -1 otherwise. -1 is one way to represent infeasible evolution but any other convention can be used. Since evolutions between architectures of different families are infeasible, it is recommended to define transition matrices only for families instead of one global matrix almost completely filled with -1's. IDC and the transition matrices are the two necessary elements to calculate life cycle costs. To take into account maintenance and operations costs that occur yearly, or more generally in a time period Δt, a function OM can be also considered. For a given architecture x, $OM(x)$ represents the recurring costs of the architecture over one year.

Demand Representation and Cap_{max}

To describe the evolution of demand through time with a geometric Brownian motion, σ and μ need to be defined. From there, different demand scenarios are generated and their probabilities of occurrence are estimated. The time step used to represent demand will be considered equal to the period between two decisions. If there are n scenarios, $\left(scenario^1, \ldots, scenario^n\right)$ with probabilities $(p_1, \ldots, p_n)$, then:

$$\sum_{i=1}^{n} p_i = 1 \tag{2.21}$$

The maximum capacity Cap_{max} has also to be defined. With this capacity and using the Pareto Front of the Trade Space, an optimal traditional architecture x^{trad} is identified with which the best path will be compared.

Creation of Paths

For each design vector x, the possible paths starting with x are generated following the evolution rules identified previously. If the decision rules do not change as well as the design variables considered, this process may be done only once and all the possible paths can be stored. To this effect, it can be interesting to associate an index with each design vector and represent paths by a series of indices. A selection is made among the paths to see which one can provide a capacity higher or equal to Cap_{max}.

Expected Life Cycle Costs

For each selected path, the life cycle costs that are necessary to adapt to the demand scenarios are calculated. Then, the expected life cycle cost of the path is computed. If $LCC\left(scenario^{i}_{path_j}\right)$ is the life cycle cost of $path_j$ for $scenario^i$ then, the expected life cycle cost of this path is:

$$E\left[LCC\left(path_j\right)\right] = \sum_{i=1}^{n} p_i LCC\left(scenario^{i}_{path_j}\right) \tag{2.22}$$

Optimization and Termination

The minimum over all possible paths of the expected life cycle costs is found and the path that provides this value is called $path^{\star}$. $E\left[LCC\left(path^{\star}\right)\right]$ and $LCC\left(x^{trad}\right)$ are finally compared to see if an economic opportunity has been revealed.

This framework has been applied to the particular of LEO constellations of communications satellites.

Chapter 3

Staged Deployment Strategy for Reconfigurable LEO Constellations

The framework introduced at the end of Chapter 2 has been applied to the particular case of LEO constellations of communications satellites. Recent constellations such as Iridium or Globalstar were designed with a traditional approach which resulted in an economic failure for both systems. By adapting the framework to this particular case, the economic opportunity that a staged deployment strategy may have provided was studied. The basic tool for this study was the simulator developed by de Weck and Chang [dWC02]. This chapter first presents the simulator and then explains how the different steps of the framework have been implemented. Throughout the sections, recommendations to improve the computations or simplify the process are given. The final section describes the different steps of the optimization process.

3.1 Problem Definition

3.1.1 Presentation of the Simulator

To conduct an architectural trade study of LEO constellations of communications satellites, de Weck and Chang [dWC02] developed a simulator. The role of a simulator is to map a design space containing design vectors x to an objective space, containing the objective vectors $J(x)$. Other important inputs are usually added such as a constant parameters vector c and a constraints vector q[1]. De Weck and Chang also took into account a vector p reflecting policy decisions. The mapping between the design space and the objective space is represented by the following expression:

$$x \overset{mapping}{\rightarrow} J = f\left(x, c, p, q\right) \tag{3.1}$$

The architecture of a constellation of communications satellites is represented in Figure 3-1. The design vector of the simulator captures the essential elements of a constellation. Namely, those elements are:

- **Constellation type C:** there exist two main types of constellations to achieve global coverage with circular orbits at low altitudes. The first one is called a Walker constellation. It was developed by J.G. Walker [Wal77] and is composed of inclined planes with ascending nodes equally spaced along the equator. The second type of constellation is called polar. The term polar is used because the inclination of the orbital planes is close to ninety degrees and the satellites go over the poles as they orbit the Earth. An introduction to the characteristics of both types of constellations can be found in [LWJ00]. This study only considered polar constellations.

- **Orbital altitude a:** the altitude is constant for satellites in circular orbits. The acronym LEO designates orbits with altitudes ranging from 200 km to 2000 km.

- **Minimum elevation angle ϵ:** it is the angle between the satellite and the horizon from which a user on the ground sees the satellite. This angle and the altitude of the satellite are sufficient to define the area covered by the satellite. The geometric meaning of ϵ has been represented in Figure 3-2.

- **Satellite transmitter power P_t**

[1]The notation used for the constraints vector differs from [dWC02] where r is used to avoid confusion with the discount rate.

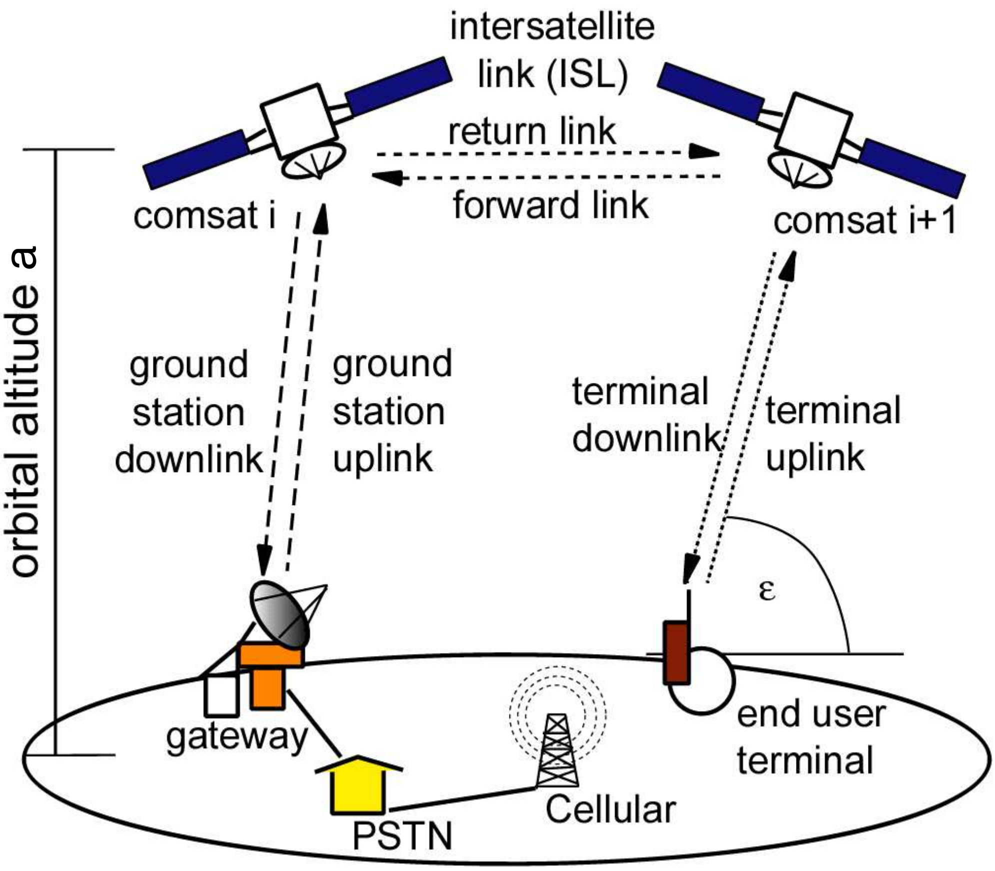

Figure 3-1: Architecture of a LEO constellation of communications satellites (from [dWC02]).

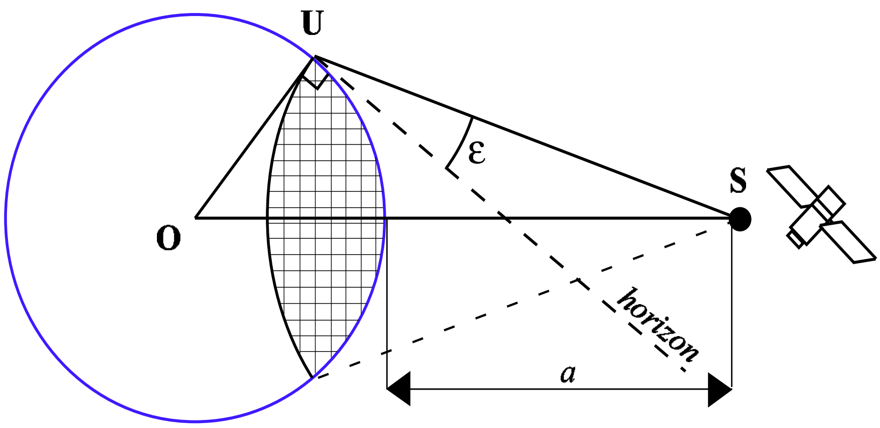

Figure 3-2: Geometric definition of the elevation angle. The coverage area of the satellite is shaded. S: satellite; U:user; O: Earth's center.

- **Antenna diameter** D_A
- **Inter satellite links** ISL**:** this technology may ($ISL = 1$) or may not ($ISL = 0$) be embedded in the constellation. Inter satellite links allow transmissions of data directly between the satellites. This connectivity reduces the number of ground stations necessary to transmit information. This technology has been used on the Iridium constellation. It represented a technical challenge because a satellite must transmit information to the neighboring satellites on its own orbital plane, but also to the closest satellites in adjacent planes.
- **Per-channel bandwidth** Δf_c**:** in this study, Δf_c has been fixed to 40 kHz.
- **System lifetime** T_{sys}

Those variables are the elements of each design vector. The simulator computes the objective vector J for each design vector. This vector consists of six elements but only three of them are of interest for this study:

- **Instantaneous number of duplex channels of the system** $N_{channels}$**:** when a subscriber uses the service, one duplex channel is assigned.
- **Lifetime capacity** C_{tot}**:** it corresponds to the total number of billable minutes over the lifetime of the constellation. It is thus expressed in minutes.

- **Life cycle cost** LCC

The simulator is used to create a trade space representing C_{tot} and LCC. Table 3.1 presents the range of the different design variables used to generate the trade space. A total of 1800 architectures have been simulated. From this trade space and the

Variable	Unit	Range	Step
C	-	Polar	Fixed
a	km	400 - 2000	400
ϵ	deg	5 - 35	15
P_t	W	200 - 1800	400
D_A	m	0.5 - 3.5	1
Δf_c	kHz	40	Fixed
ISL	-	0 - 1	1
T_{sys}	years	5 - 15	5

Table 3.1: Range of the different design variables used to generate the trade space. *Step* corresponds to the increment used.

data provided by the simulator, the valuation framework can be applied. However, certain modifications are necessary in order to study the uncertainty in the number of future users. The next subsection discusses this issue.

3.1.2 Definition of Capacity

The metric used by the simulator to define capacity is the lifetime capacity, C_{tot}, that is to say the total number of billable minutes over the lifetime of the constellation. From this capacity, the cost per minute (CPM) of a constellation can be obtained. Indeed, the cost per minute is computed dividing the life cycle costs of the system by the total number of billable minutes of the system. De Weck and Chang [dWC02] showed with the trade space that the CPM of the constellations decreased as capacity was increased. This is due to economies of scale and learning curve effects. The CPM seems to be an interesting metric to compare architectures since it gives the necessary investment to provide a unit of service. However, this metric should be used with caution because it does not take any market considerations into account. The consequences of this approach are illustrated with an example. Two constellations, A and

B, are considered. The life cycle costs, capacities and CPM of both constellations are presented in Table 3.2. Constellation A has a larger capacity than constellation

	Constellation A	Constellation B
LCC (B$)	5	1
C_{tot} (min)	$5 \cdot 10^{12}$	10^{11}
CPM ($/min)	0.001	0.01

Table 3.2: Characteristics of constellations A and B.

B and its CPM is lower. A priori, constellation A seems more interesting than constellation B. However, the CPM metric makes sense to compare architectures if and only if the full capacity is eventually used. Indeed, if the actual demand is such that only 10^{10} minutes of service are billed over the lifetime of both constellations, the actual cost per minute is 0.5$/min for constellation A and 0.1$/min for constellation B. Consequently, constellation B is cheaper and has a lower cost per minute for this demand. It should thus be selected rather than constellation A. Therefore, the CPM metric encourages the selection of systems with large capacities and high life cycle costs. But, it implicitly assumes that the entire capacity of the system will be used and fails to take into account market uncertainties. The system selected with this approach are among the most expensive but their actual cost per minute can reveal higher than expected if demand stays low. This case leads to an economic failure since the system selected requires high initial investments and a high price for the service needs to be set which limits the number of potential users. This is what happened with the Iridium constellation that proposed an airtime charge between $1.5 and $7 per minute according to Lutz, Werner and Jahn [LWJ00] while terrestrial networks provided the same service for less than $1 per minute. Formulas have been proposed to compute the service charge of a constellation of communications satellites (see [CGH+92]). This thesis introduces one in Appendix A. However, it relies on assumptions that do not take the uncertainty of future demand into account and lead to the same issues as the CPM metric.

To compute the CPM of a constellation, the total number of minutes the system can provide needs to be determined. To know this number of minutes, the future behavior of the system needs to be known perfectly. Consequently, only fixed systems can be considered. It is not the case in this study: the staged deployment strategy change the capacity with respect to uncertain market conditions. To study adaptation to demand, it would be convenient to define capacity as the maximum number of users the system can support since it can be easily related to demand. The simulator provides the instantaneous number of duplex channels, $N_{channels}$, for an architecture. To relate $N_{channels}$ to the maximum number of subscribers the system can support, N_{user}, the following parameters need to be introduced:

- A_{user}: average user activity in minutes per **month**. It is assumed constant over the lifetime of the systems.

- U_S: global system utilization in percent. This parameter is fixed by decision makers and determines the target percentage of capacity that can really be used at each moment. It is similar to a target load factor $0 \geq U_S \leq 1$.

The number of channels that are made available to subscribers is not $N_{channels}$ but $U_S N_{channels}$. To know the maximum number of subscribers the system can have, the number of available duplex channels needs to be divided by the average activity of users:

$$N_{user} = \frac{U_S N_{channels} \left(\frac{365}{12} \cdot 24 \cdot 60\right)}{A_{user}} \tag{3.2}$$

With those formulas, N_{user} can be computed for each architecture. N_{user} is the metric used for capacity throughout the study. Of course, values for U_S and the average activity level have to be set first. According to [LWJ00], typical values for those parameters are:

- U_S: between 5 and 15%;

- A_{user}: between 100 and 150 min/month;

The Iridium case can be used to check the validity of those values. For this constellation, $N_{channels}$ = 86000 and 3 million subscribers were expected. If U_S=10% and

A_{user}=125 min/month, then:

$$N_{user} = \frac{0.1 \times 86000 \left(\frac{365}{12} \cdot 24 \cdot 60\right)}{125} = 3.013 \cdot 10^6 \tag{3.3}$$

This maximum number of users is close to 3 million predicted users and the range proposed for the parameters can be considered valid. From now on, capacity will be designated as the maximum number of users N_{user} for a constellation and will be be noted $Cap = N_{user}$.

3.1.3 Assumptions

To apply the framework to the particular case of LEO constellations of communications satellites, the assumptions it implies need to be reviewed. First, the system studied is assumed to be designed for a certain maximum demand. So, the system should be able to provide a particular capacity Cap_{max} if necessary. Consequently, a maximum number of subscribers will have to be set and it will be equal to Cap_{max}. For instance, if the Iridium constellation is studied, Cap_{max} would be set to 3×10^6 users.

The second assumption of the framework is that the system adapts to demand. This means that, if demand is over the maximum number of subscribers that the system can support at a decision point, the system evolves to the next architecture in the path.

A third assumption is that the price to embed flexibility is not taken into account. The technical way to embed flexibility is not known in advance and the framework only tries to reveal the economic opportunity of this flexibility. The interpretation of this particular assumption for this study is discussed in the next sections.

A last assumption is that demand follows a geometric Brownian motion. Consequently, a volatility will be set for the number of users and demand scenarios will be generated.

3.2 Decomposition of the Design Vector

A first step of the framework consists in identifying the sources of flexibility that will be evaluated. For staged deployment, this means that the design variables that can be modified after the deployment of a first architecture need to be identified. The particularity of space systems is that on-orbit modification of the satellites is virtually impossible even though there is an increasing interest in reconfigurable spacecrafts. On-orbit servicing is not sufficiently developed to consider that the hardware of a satellite can be easily modified once in space. Consequently, design variables such as P_t, D_a or ISL have to be considered fixed. Moreover, the lifetime of the constellation cannot be considered as a design variable anymore. In fact, T_{sys} will be a fixed parameter since the behaviors of different systems with respect to the evolutions of demand over a certain period are compared. Of course, the economic opportunity of flexibility for different values of T_{sys} could be investigated but for a single analysis, it needs to be fixed. Δf_c and C are considered constant in this model and they are not considered as design variables. If the simulator could take into account Walker constellations, C would have been a good candidate as a flexible variable since changing the type of constellation does not necessarily imply any modifications of the satellites themselves.

The remaining design variables are a and ϵ. The altitude a can be changed after the satellites are deployed because it does not necessitate any changes in the hardware of the system. The variables a and ϵ are only affected by the arrangement of the fleet of satellites that form the constellation. Such arrangement will be called a configuration. The variable ϵ corresponds to the minimum angle the satellite should have with respect to the horizon to be able to provide communication to users. If a user can communicate with satellites when they are above 5 deg for instance, it will also be possible to communicate with them when they are above 35 deg. Consequently, ϵ does not depend on the satellite hardware and can easily be changed. However, a first evolution rule appears. Indeed, ϵ can be increased but cannot necessarily be decreased. A user that can see a satellite above 35 deg may not see it when it is below 35 deg. Thus, ϵ can only be increased.

Using notations of Equation (2.20), the design vectors x can now be partitioned in two parts:

$$x_{flex} = \begin{pmatrix} a \\ \epsilon \end{pmatrix} \tag{3.4}$$

$$x_{base} = \begin{pmatrix} P_t \\ D_A \\ ISL \end{pmatrix} \tag{3.5}$$

The ranges used for the different design variables are the same as those presented in Table 3.1. A family of architectures corresponds to a certain vector x_{base} that is to say to particular values of P_t, D_A and ISL. This means that a family of architectures consists of constellations with satellites that share the exact same hardware but that are in different configurations. Since P_t can take 5 different values, ISL 2, and D_a 4, there are 40 different families. Each is composed of architectures with different altitudes a and minimum elevation angles ϵ. The altitude a can take five different values and ϵ only three different values so there are fifteen different constellations per family and a total of 600 different constellations[2].

The real options considered in this study are technical devices that offer the flexibility to change a and ϵ after a constellation is deployed. Those design variables depend only on the configuration of the constellation. Therefore, the real options give the opportunity to reconfigure constellations after they have been deployed. To move the satellites, the propulsion systems of the satellites could be used if the propellant available is sufficient. Moreover, since the altitudes of the satellites may vary, phased array antenna have to be used to adjust the spot beams. Consequently, there exist different technical solutions to embed this flexibility. However, they will not be considered in the implementation of the framework. Indeed, the framework focuses only on the value of this flexibility and not on its price. This principle is contained

[2] The original simulator generated 1800 architectures but since the lifetime of the satellites T_{sys} is no longer a design variable, this number is reduced to 600.

in the assumptions of the framework, when it is stated that the price of real options is not taken into account. Therefore, the technical way to embed flexibility does not need to be discussed in this analysis but should be the subject of future research. Reconfiguration are studied in detail in Chapter 5.

This section identified two candidates to provide flexibility to the architectures: a and ϵ. The next section describes the different evolution rules of those design variables and establishes the conditions that paths of architectures need to satisfy.

3.3 Identification of Paths

3.3.1 Evolution Rules

To identify paths, the evolution rules of the different variables of x_{flex} need to be defined first. Two types of evolution rules should be distinguished though. A first type of evolution rule concerns individual variables. Those evolution rules will be called intra-variable rules. For instance, if the height of a wall that is progressively built is considered, a first intra-variable rule is that this height can only be increased. A second intra-variable rule is that the increment to increase the height has to be a multiple of the height of the bricks used. The second type of evolution rule concerns the relationships between the variables. Those evolution rules will thus be called inter-variables rules. For instance, if the length of a plane is increased to increase its capacity, its mass is increased. Therefore, the wingspan will also have to be increased to adjust to this new mass and generate additional lift. On the other hand, if the wingspan is increased, the length of the plane may not be affected. When those evolution rules are identified, it is important to see if the flexible variables selected can increase the capacity of the system while respecting the evolution rules. If it is possible, the variables are relevant, otherwise, it is not necessary to consider them as flexible. In this section, the different evolution rules for a and ϵ are identified.

It has already been explained that ϵ can only be increased. However, it was not specified if an increase in ϵ had an interesting effect on the capacity of the constellation. As discussed before, if the elevation angle is constrained to be 35 deg instead of 5 deg, a user for which the satellite is at 5 deg above the horizon cannot communicate anymore. So, the area covered by a single satellite is decreased when ϵ is increased. Constellations of communications satellites have to achieve global coverage, which implies that more satellites are necessary. Each satellite represents a certain number of duplex channels $N_{channels}$ that depend on the hardware. Consequently, having more satellites increases the capacity of the constellation. An increase in ϵ will increase the capacity of the constellation and the minimum elevation angle appears to be a relevant flexible variable.

There are no restrictions on the evolution of the altitude of the satellites. However, if ϵ is kept the same for a satellite that is moved from a position $S1$ to a position $S2$ by lowering its altitude, its coverage area decreases as illustrated in Figure 3-3. Consequently, to achieve global coverage, more satellites will be necessary for a constellation with a lower altitude if ϵ is kept the same. The capacity of the constellation is thus increased when altitude is lowered. This is consistent with the finding by de Weck and Chang [dWC02]. However, there exist cases where the capacity of the constellation increases when altitude is increased and ϵ is increased. Even though an increase of a does not always seem relevant, it cannot be forbidden by an evolution rule. Consequently, there are no evolution rules concerning a.

Finally, inter-variable evolution rules need to be considered. a and ϵ are physically independent and can be changed without affecting each other. Consequently, there is a unique decision rule that states that ϵ cannot be decreased for an evolution.

3.3.2 Representation of Paths

Now that the evolution rules have been identified, the paths can be completely defined. A path of architectures in the trade space is a series of design vectors $(x_1, x_2, \ldots, x_n)$ such that:

- All the architectures are in the same family: $\forall i \in \{2, \ldots, n\}$, $(x_i)_{base} = (x_1)_{base}$
- The evolutions do not violate any evolution rules: $\forall i \in \{1, \ldots, n-1\}$, $\epsilon_i \leq \epsilon_{i+1}$

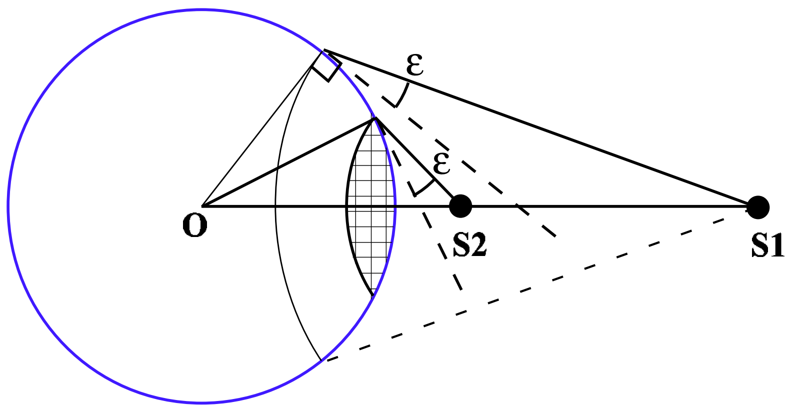

Figure 3-3: A decrease in altitude from position $S1$ to $S2$ decreases the coverage area of a satellite if ϵ is kept constant.

- An evolution corresponds to an increase in capacity: $\forall i \in \{1, \ldots, n-1\}$, $Cap(x_i) < Cap(x_{i+1})$[3].

When the design vectors consist of many variables, the representation of a path as a series of design vectors can be difficult to manipulate. This is why it is often necessary to index each of the architectures. This way, an architecture is not represented by a design vector but by an integer. Of course, this implies the creation of a function that provides the design vector associated with this index. This function is called x_{all}. If i is the index of architecture x, then $x_{all}(i) = x$. In the same manner, it can be interesting to transform functions such as Cap so that the index of the architecture is the only input. This reduces the number of inputs to only one integer compared to many design variables. With this indexation, a path becomes a series of indexes such as $(i_1, i_2, \ldots, i_n)$ which is easier to manipulate. For this reason, it is considered that a unique integer i is associated with a given architecture x and that $x_{all}(i) = x$. Also, to simplify notations, the capacity of a given architecture will be noted indifferently $Cap(i)$ or $Cap(x_{all}(i))$ and paths will be represented by a series of indexes or design vectors.

[3]This study only allows increase in capacity because the non-recurring costs are larger than the operations and maintenance costs for constellations of communications satellite. Consequently, decreasing capacity to adapt to demand is not worthwhile and capacity can only be increased.

As in Chapter 2, paths of architectures are represented in the trade space. This has been done in Figure 3-4. For each architecture on the path, the altitude a, the minimum elevation angle ϵ and the number of satellites N_{sats} have been indicated. Two comments have to be made about this figure. First, note that an evolution implies an increase in the number of satellites and, consequently, the launch of additional satellites. This is one of the main costs for an evolution. Secondly, the reconfiguration costs are not the difference between the life cycle costs of the initial and the final architectures. Actually, the costs that are represented are the life cycle costs when fixed architectures are considered. Paths are represented in the trade space to show the evolution from one architecture to another and how capacity evolves but the figure does not represent the evolution of expenditures correctly. The way to estimate the costs will be detailed in Section 3.4.

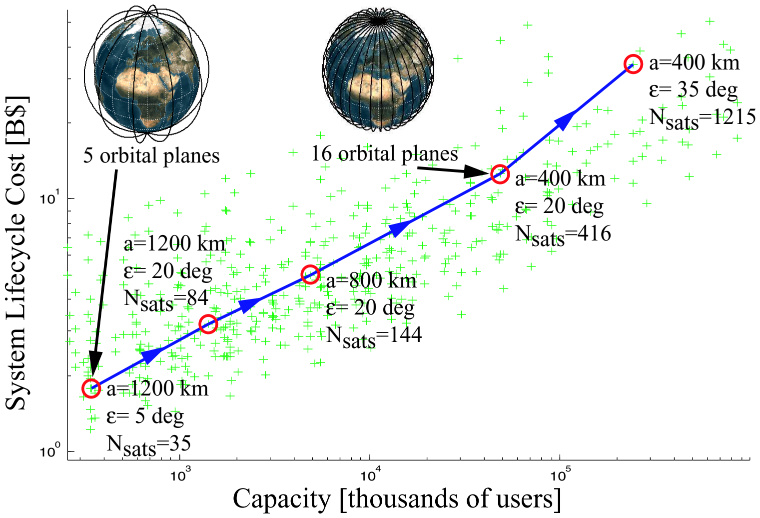

Figure 3-4: Example of a path in the trade space.

3.3.3 Ordering Families

The concept of families of architectures provides a convenient partition of the trade space. It can be interesting to index architectures with respect to their families. Indeed, the previous subsection proposed to associate an index to architectures but no particular ordering was exposed. There are 40 families, each consisting of 15 architectures. Architectures could be indexed so that family k consists of architectures $15(k-1)+1$ to $15k$. Particular ways to rank the architectures inside families could be sought to simplify certain operations. It could be interesting for the generation of paths to rank architectures so that it is impossible to evolve from any architecture to one with a lower rank, i.e. the ranking has embedded in it the evolution rules. In fact, this ranking limits the number of evolutions to consider to evolutions from an architecture to the ones with higher ranks. Using the particular structure of the problem, a ranking system that exhibits this property can be determined. The only decision rule is that ϵ can only be increased. Consequently, architectures need to be ordered so that the elevation angle increases with the rank. Moreover, Subsection 3.3.1 showed that when ϵ is kept constant, decreasing the altitude will correspond to an increase in capacity. An evolution needs to increase capacity to be feasible, consequently, it is interesting to order architectures with a similar elevation angle from the higher altitude to the lowest. With this ranking, evolutions from a rank to higher ones are the only one to consider. Those particular evolutions satisfy the evolution rules so there is no need to verify them. However, the capacity may not increase for those evolutions and this is the only condition that needs to be checked. From those different remarks, families are ordered with respect to ϵ and a as shown in Table 3.3. The number of satellites N_{sats} associated with a particular rank is given to show that the number of satellites, and thus the capacity, does not necessarily increase as a rank is increased.

An essential characteristic of this classification is that rank 1 corresponds to the minimum capacity of the family and rank 15 to the maximum capacity. Indeed, as explained before, increasing the minimum elevation angle with a constant altitude or decreasing the altitude with a constant ϵ will increase the capacity of the system. The minimum capacity will thus be provided by a constellation with the smallest ϵ possible and the maximum altitude, that is to say with rank 1. On the other hand, the maximum capacity will be provided by the constellation with a the highest ϵ and the lowest altitude which corresponds to rank 15. Three remarks can be made about those extremes:

Rank	ϵ [deg]	a [km]	N_{sats}
1	5	2000	24
2	5	1600	28
3	5	1200	35
4	5	800	54
5	5	400	112
6	20	2000	45
7	20	1600	60
8	20	1200	84
9	20	800	144
10	20	400	416
11	35	2000	98
12	35	1600	135
13	35	1200	198
14	35	800	375
15	35	400	1215

Table 3.3: Rank of architectures in a family.

- It is possible to evolve toward the architecture with maximum capacity (highest rank) from any architecture in the family.

- From the architecture with minimum capacity(lowest rank), it is possible to evolve toward any architecture in the family.

- There is no possible evolution form the architecture with highest rank.

One of the main consequences of the first remark is that it can be told in advance if, from an architecture, it is possible to evolve over a certain targeted capacity. Any architecture can evolve toward the architecture with maximum capacity in the family. If this capacity is smaller than the one targeted, all the architectures of the family can be ruled out. This principle has been used in the optimization to reduce the number of computations (see Section 3.7). The second remark can be used to simplify the computation and storage of the different paths within a family. Assume the different paths leaving architecture 1 were generated and stored in a matrix, each row of the matrix representing a path [4]. If the rows of the matrix are correctly arranged, a

[4] The different paths leaving an architecture may have different lengths. To store them in a matrix, zeros need to be added at the end of the paths that are shorter than the number of columns of the matrix.

matrix A_1 can be obtained that exhibits the following organization:

$$A_1 = \begin{pmatrix} 1 & 0 & \cdots & \cdots \\ 1 & 2 & \cdots & \cdots \\ \vdots & \vdots & \cdots & \cdots \\ 1 & 2 & \cdots & \cdots \\ 1 & 3 & \cdots & \cdots \\ \vdots & \vdots & \cdots & \cdots \\ 1 & 14 & \cdots & \cdots \\ 1 & 15 & 0 & \cdots \end{pmatrix} \tag{3.6}$$

The first column is filled with 1's because all the paths considered leave the architecture with rank 1. The first row corresponds to the path containing only architecture 1. This is a particular path that corresponds to the case for which 1 is considered fixed, i.e. the constellation is initially deployed but it is never changed. The following rows are ranked in order to gather all the paths leaving 1 and going to a particular architecture i on the first evolution. The last row will thus correspond to the particular evolution from architecture 1 to 15 that cannot go any further because architecture 15 is the last architecture with maximum capacity in the family. Note that the set of all paths leaving 1 and evolving to 2 on the first evolution can also be obtained by adding the evolution from 1 to 2 to each of the paths leaving from 2. This motivates the introduction of the submatrices $(A_i)_{i=2...15}$, A_i representing the set of all possible paths leaving architecture i. Matrix A_1 can thus be decomposed in the following way:

$$A_1 = \begin{pmatrix} 1 & 0 \cdots 0 \\ 1 & A_2 \\ \vdots & A_3 \\ \vdots & \vdots \\ \vdots & A_{14} \\ 1 & A_{15} \end{pmatrix} \tag{3.7}$$

So, from the paths leaving architecture 1, all the paths leaving the different architectures of the family can be obtained. This reduces the necessary memory to store the paths.

To generate paths from architecture 1, many algorithms can be imagined. The basic principle is to go from architecture to architecture, trying to consider all the possible evolutions that respect the evolution rules and increase the capacity. The particular ordering of the family is useful to find the potential evolutions because from an architecture i, there are no acceptable evolutions toward architectures with a lower rank in the family. Consequently, the number of cases to consider is limited to the ranks higher than i. As explained previously, when evolutions toward higher ranks are considered, the evolution rules are always satisfied. Therefore, the only condition that needs to be verified for an evolution to be relevant is that the capacity of the system increases with the evolution.

3.4 Costs Decomposition

3.4.1 Flexibility and Reconfiguration

This framework tries to reveal the economic opportunity of introducing flexibility in the original design. In the case of LEO constellations, it means having the ability to change x_{flex} that is to say a or ϵ after the initial deployment of a constellation. One of the main assumptions of the framework is that the price to embed this flexibility does not have to be taken into account. The reason is that the technical way to embed this real option may not be known in detail. Future work will add fidelity in this respect. However, there are certain costs related to evolutions that can be estimated from the model and that will be taken into account. As noticed in Figure 3-4, during an evolution, the increase in capacity is the effect of an increase of the number of satellites. Instead of launching an entire set of new satellites during an evolution, the on-orbit satellites are reconfigured and the necessary number of additional satellites are launched. Hybrid (multi-altitude) constellations are not considered in this study.

As explained previously, the price of the real options necessary to reconfigure the satellites will not be taken into account. Consequently, only the price to launch the additional satellites is taken into account in the evolution costs.

3.4.2 Evolution Costs

The price to pay to evolve from a constellation to another is not the difference between the life cycle costs of the two constellations. Indeed, costs that are directly linked to the development and research of the hardware will occur only for the first constellation in a path but will not be necessary for the next configuration[5]. This cost belongs to a first class of costs: the ones that depend only on the elements of x_{base}. Those costs are the same for all architectures in the family and need to be taken into account for the first architecture in the constellation but not in the evolution costs. The second class of costs correspond to the ones that also depend on x_{flex} that is to say on the particular characteristics of architectures inside a family. The modification of x_{flex} during an evolution will effect those costs and they need to be taken into account. The way the life cycle costs of the different constellations have been calculated in the original model thus needs to be understood. One of the key consequences is that the simulator cannot be considered anymore as a "black box".

The cost module of the simulator computes the initial development costs IDC and the operations and maintenance costs OM for a given architecture. The operations and maintenance costs do not need to be taken into account in the evolution costs but they are necessary for the calculations of the life cycle costs of a path. Consequently, from the cost module, the vector OM of operations and maintenance cost for all architectures will have to be built. To estimate the evolution costs, the decomposition of IDC has to be studied as well as the relation between each one of its components with x_{flex} and x_{base}. The cost module is based on a parametric study that relates the costs to certain characteristics of the constellations. The characteristics are not always design variables and are obtained via different modules. A certain amount of work is necessary to find the origin in terms of design variables of the different

[5] The situation is different in cases where evolution is an after thought.

costs. Table 3.4 presents the different elements of the initial deployment cost and their relation to x_{flex} and x_{base}. The software costs are assumed constant in the model. To compute the evolution costs, the individual costs that depend on x_{flex}

Individual Cost	x_{base}	x_{flex}	*constant*
Development of hardware	+		
Manufacturing of satellites	+	+	
Aerospace ground segment equipment	+		
Total program level cost	+		
Launch operations and orbit support	+	+	
Flight software			+
Ground software			+
Ground segment development	+	+	
Launch vehicles	+	+	

Table 3.4: Decomposition of the initial development costs and dependence with respect to x_{base} and x_{flex} (the symbol "+" is used when there is a dependence).

need to be considered. Those costs are:

- **Manufacturing of additional satellites:** satellites are launched during an evolution. So, during an evolution from constellation i to constellation j, $N_{sats}(j) - N_{sats}(i)$ will have to be manufactured and the associated costs will have to be taken into account in the evolution cost.

- **Launch Operations and Orbit support:** these costs corresponds to corrections in the orbits of the additional satellites once they are launched.

- **Launch Vehicle:** during an evolution, additional satellites are deployed and the cost to launch them has to be taken into account.

- **Ground Segment Development:** the model considers that the constellation needs one ground station per orbital plane when there are inter satellite links. An evolution may imply additional orbital planes and extra ground stations will have to be built.

Those four individual costs can be calculated using the simulator. Those computations have to be done for each one of the possible evolutions. It may be convenient to store the evolution costs for each evolution. The transition matrix is useful for this purpose; it is presented in the next subsection.

From the decomposition of the evolution costs presented and the remarks made, it seems that the evolution costs depend linearly on the additional number of satellites. If it was the case and if the operations and maintenance costs were neglected, the evolution costs would be exactly equal to the difference of life cycle costs between the final and initial architectures. However, it may not be the case. Indeed, the costs corresponding to the launch of additional satellites depend on the strategy used to deploy the satellites. The simulator finds a minimum number of launch vehicles to deploy a given set of satellites. Since a certain number of satellites can go in a particular launch vehicle, the costs do not depend linearly on the number of satellites. The staged deployment strategy may actually increase those costs compared to the traditional approach. This problem is illustrated with an example. Consider a constellation B containing 28 satellites. The optimal launch vehicle that the simulator proposes can launch 4 satellites simultaneously. A total of 7 launches will thus be necessary to deploy this constellation directly. Assume the staged deployment strategy proposes to deploy a constellation A with 18 satellites first and then deploy 10 more satellites to evolve toward constellation B. The first series of launch will require the use of 5 launch vehicles. Later, to launch the additional 10 satellites, 3 vehicles will be necessary. Consequently, the staged deployment uses a total of 8 launch vehicles and the costs associated are higher than the ones obtained with the traditional approach. The evolution costs will thus be higher than the difference of life cycle costs between A and B: $\Delta C_{A \to B} \geq LCC(B) - LCC(A)$. This situation is represented in Figure 3-5. From this perspective, staged deployment seems to be an expensive strategy. However, the economic mechanisms that were presented in Section 2.3 apply in this case. Therefore, the fact that the evolution from A to B occurs after the deployment of constellation A implies that its cost is discounted. In terms of net present value, the evolution costs can thus be smaller than the difference of life cycle costs:

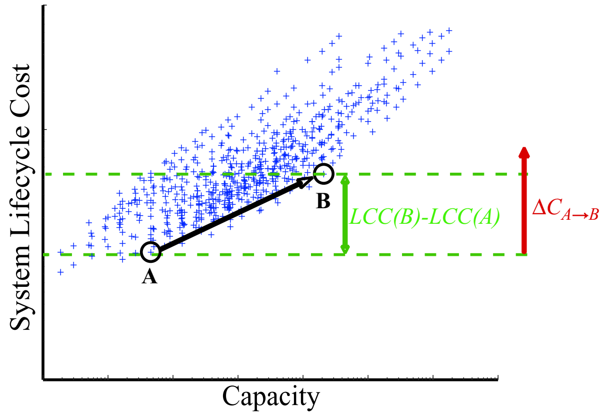

Figure 3-5: Difference between $\Delta C_{A\to B}$ and $LCC(B) - LCC(A)$ due to the launch costs.

$PV(\Delta C_{A\to B}) \leq LCC(B) - LCC(A)$. This situation is represented in Figure 3-6. Moreover, the evolution from A to B may never be necessary. In this case, there is no evolution costs and the savings achieved via the staged deployment strategy are exactly equal to $LCC(B) - LCC(B)$. Consequently, even though the evolution costs may be higher than the difference of life cycle costs between two architectures, it has to be kept in mind that evolutions may not occur and that, when they are decided, the costs associated are discounted.

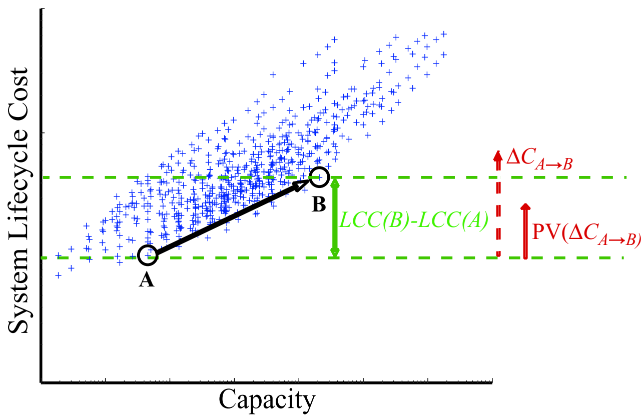

Figure 3-6: Difference between the present value of $\Delta C_{A\to B}$ and $LCC(B) - LCC(A)$.

3.4.3 Transition Matrix

The cost of evolutions are essential to calculate the life cycle costs of paths and may have to be used several times during computations. To decrease the computation times, it is convenient to compute them in advance and store them. The transition matrix is a convenient tool to store those data. If ΔC is the transition matrix, then, the evolution cost between architecture i and architecture j is equal to $\Delta C(i, j)$ when the evolution is feasible and to -1 otherwise[6]. Consequently, the transition matrix is a square matrix whose size is equal to the number of architectures considered. If three architectures are considered, an example of a transition matrix would be:

$$\Delta C = \begin{pmatrix} 0 & 4 & -1 \\ -1 & 0 & 5 \\ 6 & -1 & 0 \end{pmatrix} \tag{3.8}$$

From the first row of this matrix, it can be seen that the evolution from architecture 1 to 1 does not cost anything, the evolution from 1 to 2 has a cost equal to 4 and the transition from 1 to 3 is impossible.

If the number of architectures is large, ΔC can be cumbersome. In the case studied, it is a 600 by 600 matrix square matrix. The elements of ΔC that are different from -1 have been represented in Figure 3-7. Only few elements of ΔC are different from -1. Indeed, the transitions from an architecture to another from a different family being infeasible, most of the transitions are infeasible and many -1 appear in ΔC. Because of the particular ordering of families used, ΔC is a block diagonal matrix. Each block matrix corresponds to the transition matrix associated with a particular family that will be noted ΔC^{family}. Using only one transition matrix to represent all the different cases does not relevant since the matrix obtained is almost empty in terms of useful information. It would be much simpler to use only the matrices ΔC^{family} and benefit once again from the family approach. A

[6] -1 was used as an internal convention. Other conventions can be used to represent infeasibility but 0 should be used with care to avoid confusion with between infeasible evolutions and evolutions with no costs such as evolving from i to i.

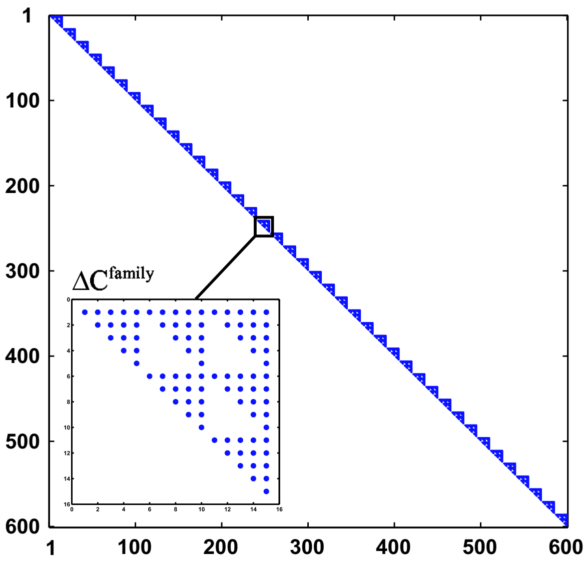

Figure 3-7: Representation of the elements of ΔC that are different from -1 and zoomed view of a ΔC^{family} submatrix.

zoomed view of such matrix is included in Figure 3-7. Those matrices are upper triangular because of the way architectures were indexed. As explained in Section 3.3, families are ordered so that an evolution from a certain rank in the family to a lower is infeasible. The result of this ordering was presented in Table 3.3. Since the rank can only be increased with an evolution, $\Delta C^{family}(i, j) = -1$ if $i > j$ and the matrix is upper triangular. This reduction of ΔC into smaller matrices that contain the essential data about the cost to reconfigure reduces the memory needed and uses the family approach.

3.5 Definition of Demand

In this study, demand is represented by the number of subscribers to the service. To represent the fact that it is uncertain, the assumption was made that it follows a geometric Brownian motion[7]. The binomial tree is used as an approximation of this property for the ease of use it provides. However, the framework also assumes that the system adapts to demand. This means that if the capacity of the system is smaller

[7]The parameter A_{user} is also a random variable in reality. However, in this study, it is assumed to be fixed.

than demand at a particular decision time, an evolution to the next architecture on the path is decided. Consider two possible evolutions of demand over two periods represented in Figure 3-8. The system has initially a capacity Cap_0 and the next

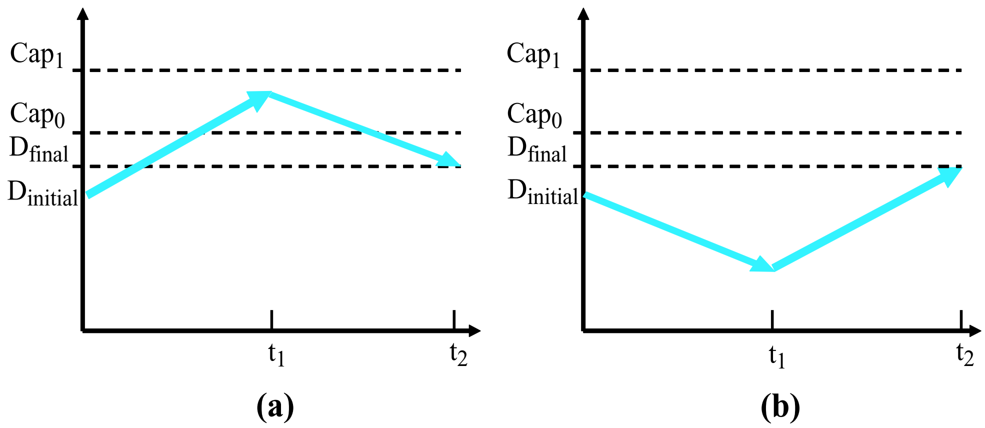

Figure 3-8: Two possible evolutions of demand over two periods.

evolution increases the capacity to Cap_1. In case (a), demand increases during the first period and gets over Cap_0. Consequently, at t_1, there will be an evolution to the next constellation. Then, demand decreases and no evolution is necessary. In case (b), demand decreases first and then increases, reaching the same level D_{final} as in case (a). However, demand never gets higher than Cap_0 in case (b) and no evolution is necessary during the two periods. Consequently, even though case (a) and (b) lead to the same level of demand D_{final} at time t_2, they do not lead to the same evolution of the system. Therefore, in the binomial tree, the value of demand at a particular node will not be considered, but the entire progression of demand over time. Scenarios in the binomial tree will thus be considered. If there are n periods, a scenario will be a series of n up and down movements of demand. Consequently, there are 2^n possible scenarios. Since the up and down movements are independent events in terms of probability, the probability of a scenario will be:

$$P(scenario) = p^k(1-p)^{n-k} \tag{3.9}$$

where p is the probability of going up (given by Equation (2.12)) and k the number of up movements in the scenario. An example of a scenario in the binomial model is given in Figure 3-9. The binomial tree now needs to be built. The volatility of demand,

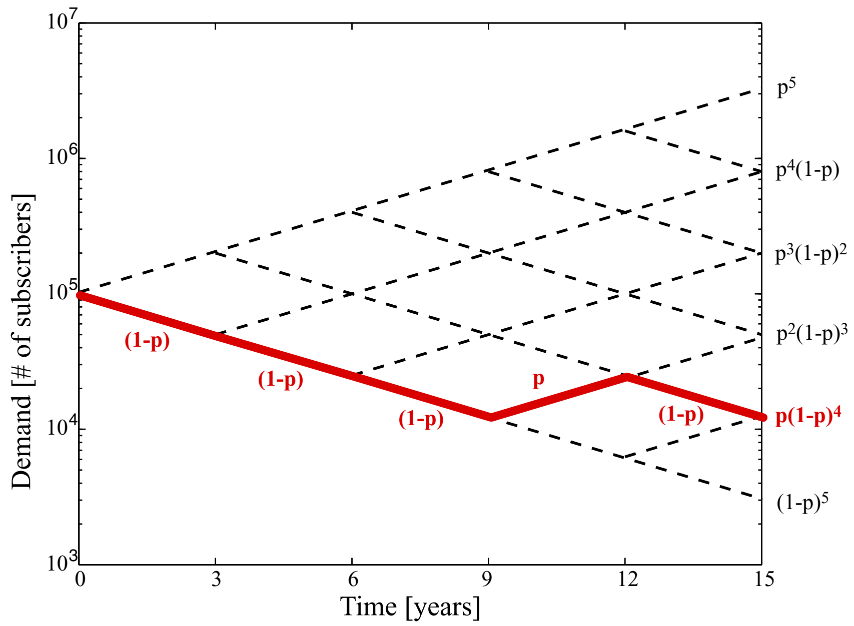

Figure 3-9: Example of one scenario out of $2^5 = 32$ in the binomial tree ($n = 5$).

σ, its expected return per unit time,μ, and an initial value of demand, $D_{initial}$, have to be defined. Also, the length of periods has to be set. This length will define the time between two decisions and needs to be chosen with care. A short time length will create a very dense binomial tree and will exponentially increase the number of scenarios to consider (2^n, where n is the number of periods). A long time length will reduce the number of decision points and also the value they present. However, the necessary time to achieve an evolution is on the order of one or two years because many satellites need to be launched and the lifetime usually considered for a system are comprised between 5 and 20 years. Therefore, it is not necessary to decrease the length of the periods below this limit and a maximum of 10 decision points should be considered. In this particular case, $2^{10} = 1024$ scenarios will be considered.

3.6 Life Cycle Costs

The life cycle cost of a path is the expected value of the life cycle costs for each scenario (see Equation (2.22)). Indeed, each path-scenario pair has a life cycle cost. From the cost module and with appropriate modifications, the following functions can be obtained:

- **IDC:** if i is the index of an architecture, $IDC(i)$ is the initial development cost of this architecture.
- **OM:** if i is the index of an architecture, $OM(i)$ is the sum of the operations and maintenance cost of the architecture for one year.
- **ΔC:** if i and j are the indexes of two architectures such that the evolution from i to j is feasible, $\Delta C(i, j)$ is the evolution cost between i and j.

It has been explained how the function ΔC function can be decomposed with respect to families. The IDC and OM functions can be computed in advance for each one of the architectures and stored into vectors. From those three functions, the life cycle cost of a path given a demand scenario can be computed. If a path $(i_1, i_2, \ldots, i_k)$ is considered, the rules to compute its life cycle cost are:

1. In the first year, the expenditures are $IDC(i_1) + OM(i_1)$. All the development and deployment of constellation $index_1$ are assumed to occur on the first year. also, operations are assumed to start directly.
2. At a decision point, the corresponding value of demand in the scenario and the capacity of the system are considered. This capacity is equal to $Cap(i_j)$ if i_j is the current architecture. If demand is over the current capacity and if further deploy is possible, capacity is increased. For the year corresponding to the decision point, the expenditures are $\Delta C(i_j, i_{j+1}) + OM(i_{j+1})$. Consequently, deployment delays of the next stage are not taken into account. If the system cannot be deployed further or if demand is below the current capacity, the expenditures are just $OM(i_j)$. No change in architecture occurs in that last case.

3. In a year which does not correspond to a decision point, the expenditures are just the ones corresponding to operations and maintenance that is to say $OM(i_j)$ where i_j is the current architecture.

Once the repartition of expenditures is calculated, their net present value is computed. A discount rate r thus needs to be defined. Using the notations of Equation (2.22), this present value is called $LCC\left(scenario_{(i_1,i_2,\ldots,i_k)}\right)$. For each one of the scenarios of the binomial tree, this value is computed using the same discount rate r. If there are n periods, a total of 2^n scenarios and their probabilities need to be considered. The life cycle cost of the path $(i_1, i_2, \ldots, i_k)$ is the expected value over all the different scenarios and is given by:

$$LCC\left(path\right) = \sum_{l=1}^{2^n} P(scenario^l) LCC\left(scenario^l_{(i_1,i_2,\ldots,i_k)}\right) \tag{3.10}$$

To associate a life cycle costs to the paths, this approach has to consider an average value over the scenarios. Consequently, the value obtained may not reflect the life cycle costs that could be obtained with the different scenarios and the results will have to be interpreted with caution. In particular, the fact that the average life cycle cost of a path is smaller than the life cycle cost of the optimal traditional design does not imply that it is true for all the scenarios of demand. However, this framework always consider a worst-case situation for staged deployment since a reconfiguration is decided every time demand gets higher than the capacity of the system. In reality, decision maker take into account other parameters than the value of demand and a reconfiguration will be decided only if it is judged profitable. Therefore, even thought the life cycle costs can get higher than the traditional life cycle cost for certain demand scenarios, one has to keep in mind that in reality, the decision of deploying a next stage depends and other parameters than the actual value of demand. Moreover, if a deployment is decided, this implies that demand got over the capacity of the system and that there is a significant number of potential customers. Even thought a reconfiguration implies a cost, the fact that the level of demand is large ensures that revenues will be large too.

For the validity of the comparisons between the traditional and the flexible approaches, the life cycle costs of the different architectures considered fixed need to be calculated the same way. In the case of fixed architectures, there is no need to take market conditions or evolutions into account. First, it is considered that the costs associated with the initial development and the beginning of operations occur on the first year. Then, every year over the lifetime of the system, operation and maintenance costs are taken into account. Finally, those expenditures are discounted and their present value is computed. This present value is actually the life cycle cost of the fixed architecture.

3.7 Optimization

The optimization process consists in looking for the "best" path that is to say the one that will respect the requirements while minimizing the average life cycle costs $LCC(path)$ (see Equation(3.10)). In the previous sections, many inputs corresponding to certain requirements were introduced. They need to be defined to achieve the computations. Once defined, particular parameters that remain constant throughout the calculations can be computed and stored to reduce the optimization time. Finally, the computations can be run to find an optimal path of architectures. Therefore this section lists the different variables that need to be defined and the different parameters that could be computed in advance. It can be seen as a "check-list": all those parameters or values need to be known to run the optimization. Then, the different steps of the optimization process and the different modules they require are described.

3.7.1 Input Parameters

Along the previous sections, some inputs were introduced. Designers need to define them before the optimization can be achieved. Those parameters are the following:

- U_S and A_{user}: those parameters are necessary to convert the number of channels of the system into a maximum number of subscribers (see Subsection 3.1.2).

- Discount rate, r: it needs to be defined for the calculations of the life cycle costs.

- μ and σ: those parameters are defined to generate the binomial tree and the probabilities associated with the up and down movements (see Subsection 2.4.2).

- Initial demand $D_{initial}$: the level of demand at the beginning of the first year needs to be defined to generate the binomial tree (see Section 3.5).

- Maximum number of subscribers expected Cap_{max}: this value corresponds to the size of the market targeted by designers. The system does not have to provide this capacity after the initial deployment but it has to be able to provide this capacity after several evolutions at least within n time periods (see Subsection 3.1.3).

- Minimum capacity of the system Cap_{min}: designers may require a minimum capacity that the system should always provide. This means that for a given path, the first architecture should provide a capacity at least equal to Cap_{min}. If designers do not want to define a minimum capacity, Cap_{min} can be set to zero. However, it is better to have $Cap_{min} \geq D_{initial}$ otherwise the initial architecture of some of the paths may not provide a sufficient capacity on the first year.

- Lifetime of the system T_{sys}: this parameter is necessary to generate the binomial tree but also to calculate the life cycle costs of the fixed architectures. Setting a particular value of T_{sys} does not prevent the system from being operated beyond this time in practice.

- Period between two decision points Δt: this parameter associated with T_{sys} will define the number of decision points throughout the lifetime of the system. This parameter is necessary to generate the binomial tree.

For computations, those parameters can be gathered inside a vector such as the vectors c and q introduced in Section 3.1. If this vector containing the parameters of the optimization is noted w, its elements are:

$$w = \begin{pmatrix} U_s \\ A_{user} \\ \mu \\ \sigma \\ D_{initial} \\ r \\ Cap_{max} \\ Cap_{min} \\ T_{sys} \\ \Delta t \end{pmatrix} \tag{3.11}$$

3.7.2 Stored Values

Many values are used several times throughout the optimization and are not affected by changes in the parameters. It can be interesting to compute those values before the optimization and store them to avoid calculating them several time. Moreover, if the optimization has to be done several times for different values of the parameters, time is saved by having stored those values. These values are:

- Cap and LCC: they need to be obtained from the simulator with the appropriate modifications for each one of the architectures. Cap needs to be expressed in thousands of users and LCC should be computed with the same rules used to calculate the life cycle costs of a path.

- IDC and OM: for each of the architectures, the initial deployment costs and costs for operations and maintenance need to be known. They can be calculated in advance from the cost module and stored into the vectors IDC and OM.

- ΔC: the transition matrices can be computed in advance for each family. Those costs only depend on the design vectors and capacities of the initial and final architectures considered in an evolution. Consequently, they do not depend on any of the elements of w and need to be computed only once. Consequently, even though the transition matrices can be computationally expensive, they only need to be computed once and can then be used for any values of the parameters considered.

It should be pointed out that different approaches can be considered to generate paths. The paths of architectures depend only on evolution rules and architectures themselves but not on any of the parameters contained in the vector w. Consequently, all of them could be generated and stored in advance. The optimization process will then have to go through each one of them and compare them. However, the number of paths in a family is approximately 2000 and there are 40 families. Storing all these paths could be relatively costly and most of them may be unnecessary because they may not meet the particular requirements set by the parameters of the problem. for instance, they may not be able to ultimately achieve Cap_{max}. Generating the paths concurrently with the optimization can add extra computation time if the algorithm used is not efficient. An intermediate solution can be found in between by storing the paths leaving the first architecture of each family. As explained in Section 3.3.3, from those paths, all the paths of the family can be generated easily. Consequently, all the paths can be generated with only a few computations and the necessary memory to store paths can be reduced. An efficient algorithm to generate the paths can also be sought to avoid storing them.

3.7.3 Path Optimization Process

This subsection presents the optimization process in a step by step manner. It is assume that the parameters introduced in this section have been defined by designers and that the necessary values for computations are stored or can be obtained.

Determine x^{trad}

The goal of the framework is to compare the staged deployment strategy with the traditional approach. The traditional approach will select the architecture on the

Pareto Front that is closest to Cap_{max} but with a higher capacity. The first step of the framework is to determine this architecture noted x^{trad}. The life cycle cost of this architecture is $LCC(x^{trad})$ and it will eventually be compared with the life cycle cost of the best path obtained through the optimization. If the Pareto front is already known, x^{trad} can be easily obtained. Otherwise, an algorithm can determine x^{trad} by looking at the architecture with minimal life cycle costs among the architectures with a capacity greater than Cap_{max}.

Identify Relevant Paths

The parameters Cap_{min} and Cap_{max} set limitations of the trade space. In particular, the first architecture of a path should provide a capacity greater than Cap_{min} and the last architecture a capacity greater than Cap_{max}. This constraint reduces the number of paths to consider. In particular, if an architecture has a capacity smaller than Cap_{min}, it is not necessary to consider the paths that it leaves. Moreover, if a path has a maximum capacity smaller than Cap_{max}, it is not necessary to consider it. Actually, an entire family can be ruled out with the same argument by considering the last architecture of the family. If its capacity is smaller than Cap_{max}, none of the paths inside the family will be able to provide a sufficient capacity in case it is needed and the architectures of this family should not be considered. Other paths can be excluded too. Indeed, if an architecture has a capacity that is strictly higher than $Cap(x^{trad})$, then it is not necessary to consider the paths leaving it because they will always lead to life cycle costs that are higher than $LCC(x^{trad})$ because x^{trad} is Pareto optimal. Consequently, the initial architectures that are relevant are included between Cap_{min} and $Cap(x^{trad})$ and the paths that they leave need to be considered if and only if their maximum capacity is higher than Cap_{max}. Those restrictions limit the number of cases to consider and allow a determination the set of paths to consider.

Generate Demand Scenarios

From the parameters T_{sys}, Δt, μ and σ, the different scenarios for demand can be generated. The probability of each scenario can be computed too using Equation (3.9). A scenario being a series of values of demand at each decision point, each scenario can easily be represented by a row vector with those values and all the scenarios can be gathered in a matrix D, each row representing a scenario and each column representing a decision time. The corresponding probabilities for each scenarios can be gathered in a column vector P for which the i-th element corresponds to the probability of the i-th row in D. An example is presented in Figure 3-10.

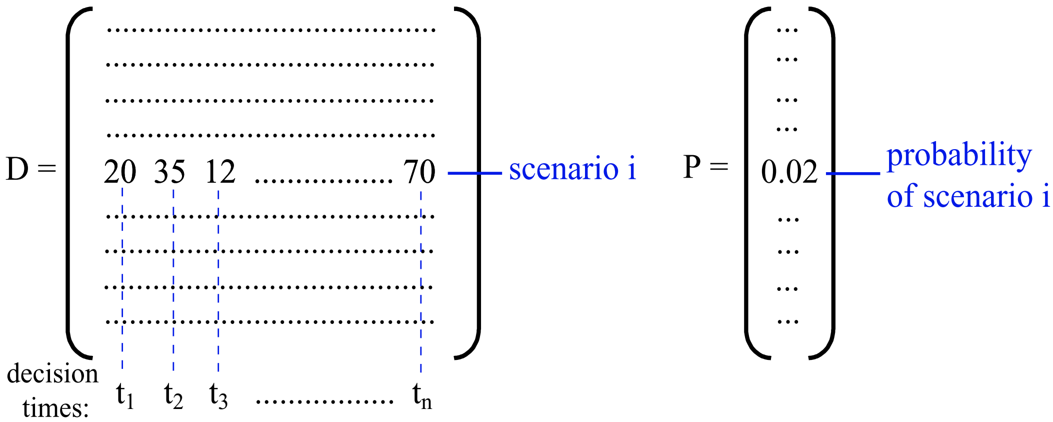

Figure 3-10: Example of a matrix D and a vector P to represent demand scenarios.

Compute Life Cycle Costs

Using the method described in Section 3.6, the average life cycle costs of the paths that satisfy the requirements are computed. The path that provides the minimum average life cycle cost is identified. This life cycle cost is noted LCC_{best} and the optimal path is called $path^{\star}$.

Compare Approaches

$LCC(x^{trad})$ and LCC_{best} are compared. The difference between those two values reveal the magnitude of the economic opportunity associated with staged deployment.

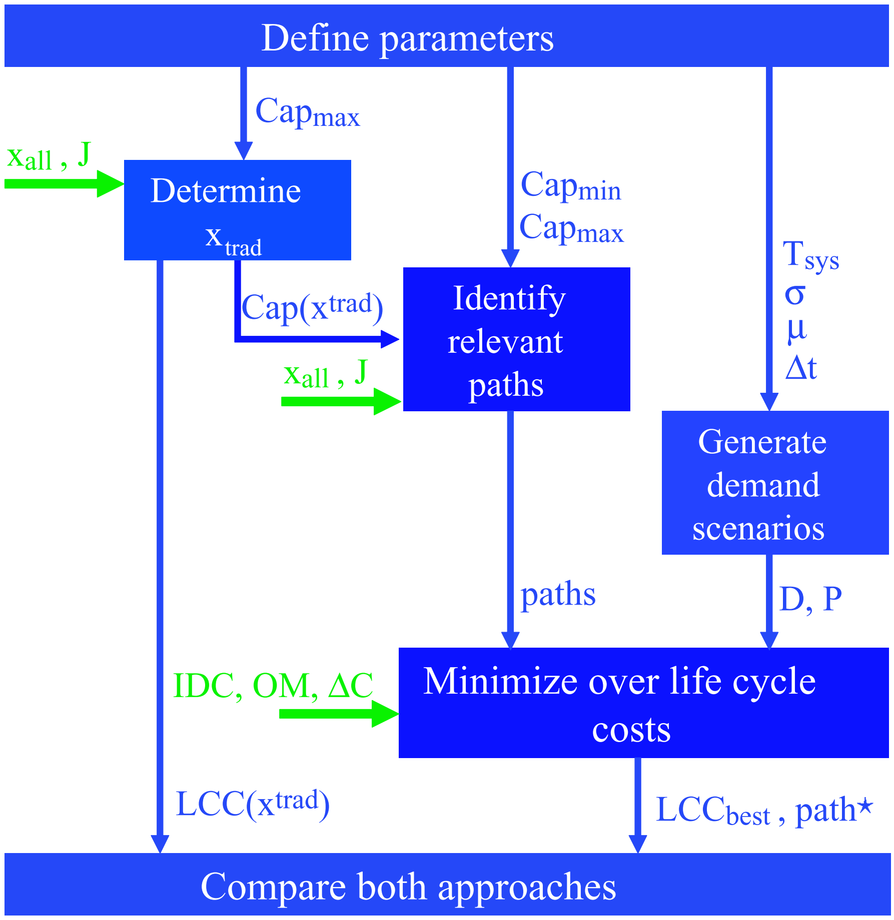

Figure 3-11: Link between the different steps of the optimization process.

This optimization process has been summarized in Figure 3-11. The results obtained with this adaptation of the framework will be presented in the next chapter.

Chapter 4

Case Study

4.1 Introduction

In this chapter, the framework developed for LEO constellations of communications satellites is applied to a particular case. The economic opportunity of staged deployment is evaluated when the size of the market targeted is close to the one Iridium originally expected. The average life cycle costs obtained will not be compared with the actual one of Iridium for two reasons. First, as explained by de Weck and Chang [dWC02], LCC is difficult to determine exactly. Moreover, according to the same study, the Iridium constellation is not Pareto optimal and the framework compares the staged deployment strategy with a best traditional design. Consequently, an architecture that is Pareto optimal and that can provide the same capacity as the Iridium constellation needs to be determined first. This architecture will then be compared to the relevant paths of architectures. The influence of different parameters on the best paths will be analyzed. Finally, the different results obtained will be presented.

4.2 Determination of x^{trad}

To identify a best traditional design, certain parameters need to be set to estimate the capacity of the different systems and their life cycle costs. Once determined, the trade space can be generated and the Pareto optimal architecture x^{trad} that meets the capacity requirement can be identified. This section presents the different values of the parameters for this case study.

4.2.1 Capacity

To define a capacity for the architectures, U_S and A_{user} need to be defined. In Section 3.1.2, it was shown that the capacity of the Iridium constellation was $N_{user} = 3.013 \cdot 10^6$ subscribers for $U_S = 10\%$ and A_{user}=125 min/month. The same values of U_S and A_{user} will be used throughout the study. To find a system equivalent to the Iridium constellation in terms of capacity, the targeted capacity for the system was set to $Cap_{max} = 2.8 \cdot 10^6$ [1].

4.2.2 Life Cycle Costs

To determine the life cycle costs of the architectures, the discount rate r used and the lifetime of the systems considered need to be known. The Iridium constellation was originally designed with a 10 years lifetime. T_{sys} will thus be set to 10 years. Setting the discount rate to a particular value is not interesting since its value is going to be changed several time to study its influence on optimal solutions. The most important part of the life cycle costs for a fixed architecture is the initial development costs. Those costs are discounted at the same time and in the same manner for all architectures given a discount rate. Consequently, if an architecture is more expensive than another for a given discount rate, it will still be more expensive for any other value of r. The architectures, when considered fixed, can thus be compared using any value of r. So, to determine the best optimal architecture, the trade space is generated by setting r to a particular value.

[1] The targeted capacity is not set exactly to $3.013 \cdot 10^6$ to take into account architectures with a capacity close from this number but with a lower capacity.

4.2.3 Trade Space

To represent the trade space and calculate the life cycle costs, r is set to zero. The trade space obtained has been represented in Figure 4-1. A vertical line represents the capacity targeted Cap_{max} and from there the optimal traditional architecture x^{trad} is identified. x^{trad} is represented in a square. Its capacity is $Cap(x^{trad}) = 2.82 \cdot 10^6$ and its life cycle cost is $LCC(x^{trad}) = 2.03$ \$B for $r = 0$. According to Lutz, Werner

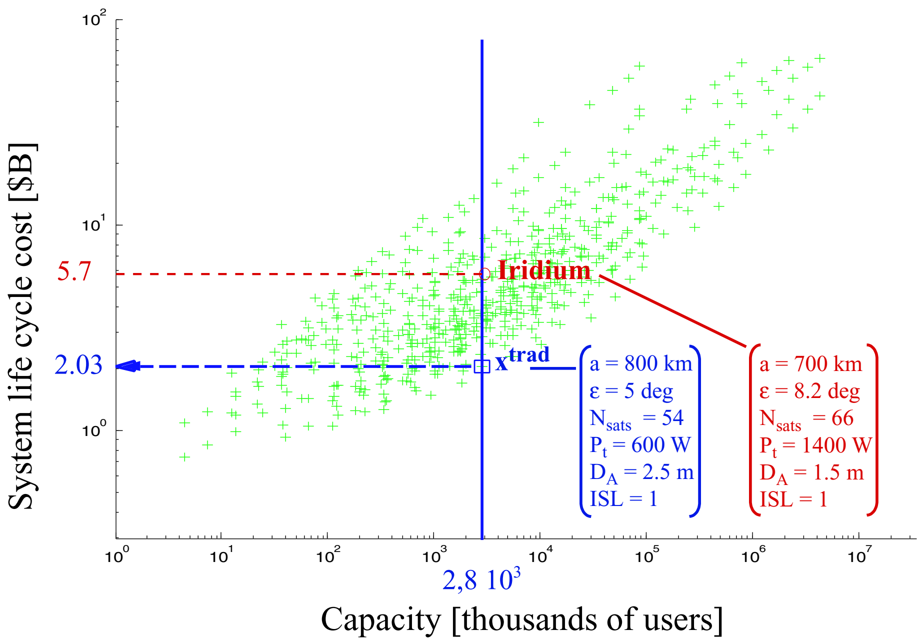

Figure 4-1: Position of the Iridium constellation and the optimal traditional architecture x^{trad} for $Cap_{max} = 2.8 \cdot 10^6$.

and Jahn [LWJ00], the total cost of the Iridium constellation was estimated at \$B5.7. For this value of LCC, the Iridium constellations has been represented in the trade space. The characteristics of x^{trad} and Iridium are also given in this figure.

This particular design will be compared to the staged deployment strategy. To achieve the necessary optimization, as explained in Chapter 3, certain parameters need to be defined. The next section explains how the influence of those parameters on the economic opportunity have been studied.

4.3 Determination of a Best Path and Parameter Study

4.3.1 Parameters Selected

To achieve the optimization, r needs to be set and a binomial tree has to be generated. To study the influence of r, this parameter will be set to several different values. For the binomial tree, the time step Δt is set to 2 years. Consequently, decisions concerning the deployment of the constellation will be taken every two years. Moreover, the expected increase in demand per time unit μ is assumed constant. It is set to $\mu = 20\%$ per year. The remaining parameters that need to be set are the initial demand $D_{initial}$ and the volatility of demand σ. The influence of σ will also be studied so this parameter will take several different values. The Iridium constellation only had 50000 subscribers after almost one year of service (see Table 1.1). This value will be considered for $D_{initial}$. Also, the initial architectures are constrained to deliver a capacity at least equal to the initial demand. So $Cap_{min} = D_{initial}$. Table 4.1 summarizes the different constant values that were have assigned to the parameters for the case study.

Parameter	Value	Unit
U_S	10	%
A_{user}	125	min/month
Cap_{max}	2.8	millions of subscribers
T_{sys}	10	years
Δt	2	years
μ	20	%
Cap_{min}	50000	subscribers
$D_{initial}$	50000	subscribers

Table 4.1: Values of the fixed parameters.

4.3.2 Value of Flexibility

The parameter study will consider the value of flexibility. This value of flexibility can a priori be defined as the discounted money saved compared to the traditional approach that is to say the difference between the life cycle cost of x^{trad} with the average life cycle cost of the optimal path $path^{\star}$. This difference has been represented in Figure 4-2. However, for different values of the discount rate r or for different methods to estimate the life cycle costs, this value may change. To achieve the parameter study concerning the discount rate or the volatility, it is thus necessary to scale this value with respect to the life cycle cost of the traditional design. Consequently, the value of flexibility will now designate the percentage of money saved with respect to the traditional approach:

$$Value = \frac{LCC(x^{trad}) - LCC(path^{\star})}{LCC(x^{trad})} \tag{4.1}$$

With this definition of the value of flexibility, the parameter study can now be done.

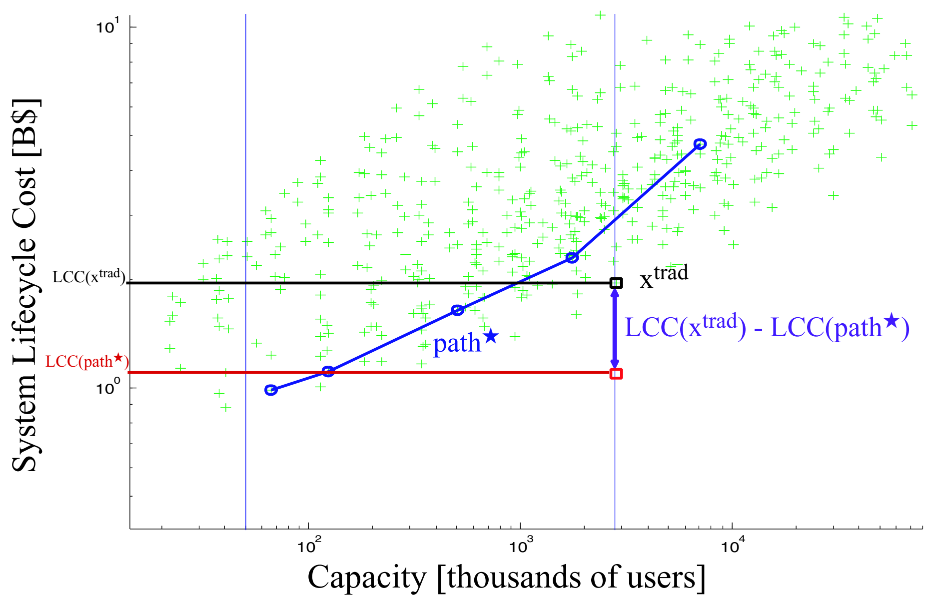

Figure 4-2: Difference between the life cycle cost of the optimal traditional design x^{trad} and the average life cycle cost of the optimal path $path^{\star}$.

4.3.3 Discount Rate r

To study the parameter of a solution with respect to the discount rate, the volatility is set to $\sigma = 70\%$. For this value, the binomial tree obtained is represented in Figure 4-3. σ directly affects the span of the binomial tree. For values of σ smaller than 70%, the maximum demand that can be attained is smaller than Cap_{max} and the probability to have a demand higher than the one targeted is equal to zero. That is why σ is set to this particular value. In the next section, when different values of σ are studied, it will be shown how the fact that the targeted capacity cannot be reached affects the value of flexibility.

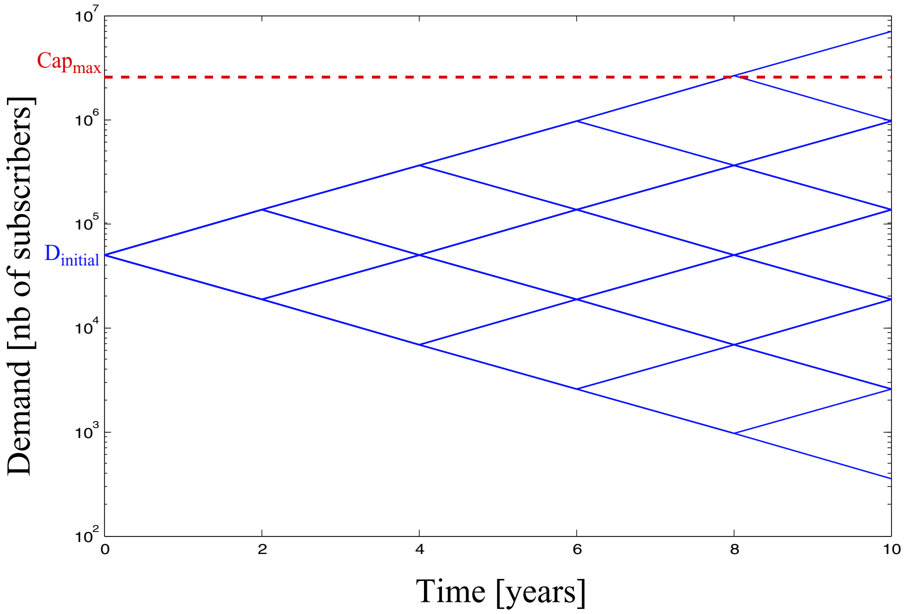

Figure 4-3: Binomial tree for $D_{initial} = 5.10^4$, $\sigma = 70\%$ and $\mu = 20\%$.

To study the influence of the discount rate on the economic opportunity of staged deployment, an optimization is run over the paths of architectures for values of r ranging from 0% to 100% with a step of 5%. For each value of the discount rate, the optimal path is obtained and the value of flexibility associated to it. The results obtained are presented in Figure 4-4. It can be seen that the value of flexibility increases with the discount rate. The reason is that the higher r is, the less expensive a reconfiguration appears in terms of net present value and the more valuable it

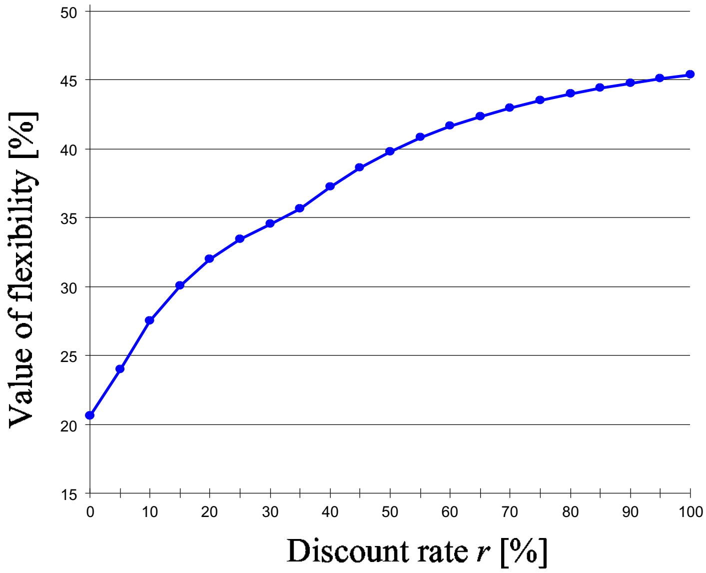

Figure 4-4: Value of flexibility in % of money saved with respect to r.

is. This is the first mechanism that justifies the value of flexibility. For a discount rate equal to zero, the staged deployment strategy still presents an economic value. The reason is that average life cycle costs are considered. Consequently, for certain demand scenarios, a reconfiguration may not be necessary and the life cycle costs correspond to the one of the initial architecture that are smaller than the life cycle cost of the traditional architecture. However, even though the average life cycle cost considered are smaller than $LCC(x^{trad})$, the final life cycle cost may be higher for certain demand scenarios. This does not imply that flexibility has no value in this case. Indeed, the framework considers a worst case scenario for staged deployment that is to say that adaptation to demand is done when possible. In reality, the decision of increasing the capacity depends on other factors, in particular the availability of money to achieve the evolution. Consequently, even though the staged deployment solution could reveal more expensive for a discount rate of 0% and important increase in demand, it should be kept in mind that the decision to deploy the next architecture belongs to the managers and is not an automatic process. Another comment that needs to be done about this figure is that the value of flexibility can be over 30% which is a significant value. Moreover, it is on the average which means that for situations where demand does not grow, the economic risk is lowered at least by 30%. For high discount rates, the curve flattens out. Indeed, even though the value of flexibility increases with the discount rate, it cannot go over a certain limit. Indeed, since the architectures considered are constrained to provide an initial capacity at least equal to $Cap_{min} = D_{initial}$, the life cycle costs of the paths will always be greater or equal to the life cycle cost of the Pareto optimal architecture that provides a capacity greater than $D_{initial}$. Consequently, there is an asymptotic value for flexibility.

The optimal path to consider depends directly on the discount rate. When r ranges from 0% to 100%, five different paths are obtained. Path 1 is optimal when $r = 0\%$; Path 2 is optimal when $r = 5\%$; Path 3 is optimal when $10\% \leq r \leq 30\%$; Path 4 is optimal for $r = 35\%$; Path 5 is optimal when $40\% \leq r \leq 100\%$. Their position in the trade space with respect to $D_{initial}$ and Cap_{max} has been represented in Figure 4-5. The number of evolutions in a path increases with the discount rate. As explained previously, the higher the discount rate is, the less expensive a reconfiguration will be in terms of present value. Having the possibility to achieve many reconfiguration allows a more precise adaptation to demand and reduces the overall costs if reconfiguration are not too epxensive. For instance, Path 2 and Path 3 have the same initial and final architectures but, since Path 3 has more architectures than Path 2, it can adapt to demand with a higher precision.

From those five paths, it can also be observed that the capacity of the initial architecture of an optimal path decreases with the discount rate. The reason is that reconfigurations are more discounted when they occur late in time. If the initial capacity is low, the need for a reconfiguration may occur early after the initial start of service. If the discount rate is low, this reconfiguration will be expensive. Consequently, when r is low, the reconfiguration should occur as late as possible. If the initial architecture has a high capacity, the demand will have to grow significantly before getting over this capacity thus delaying the need for a reconfiguration. On the other hand, if the discount rate is high, reconfiguration do not seem too expensive and the best way to lower the life cycle costs is to lower the initial development costs as much as possible. That is why an initial capacity with a low capacity should be sought when r is large.

The ideal case for staged deployment would be to follow the Pareto front when increasing the capacity. However, because of the particular decomposition of the trade space into families and the technical or physical limitations that drive the evolution rules, it cannot generally be done. The optimal paths that are obtained sometimes have architectures on the Pareto front but they do not perfectly follow it. The staged deployment strategy does not look for architectures that are Pareto optimal anymore but for architectures that provide the maximum flexibility in an affordable manner. Consequently, some paths may have an initial architecture that is off the Pareto front, such as path 2 or 3.

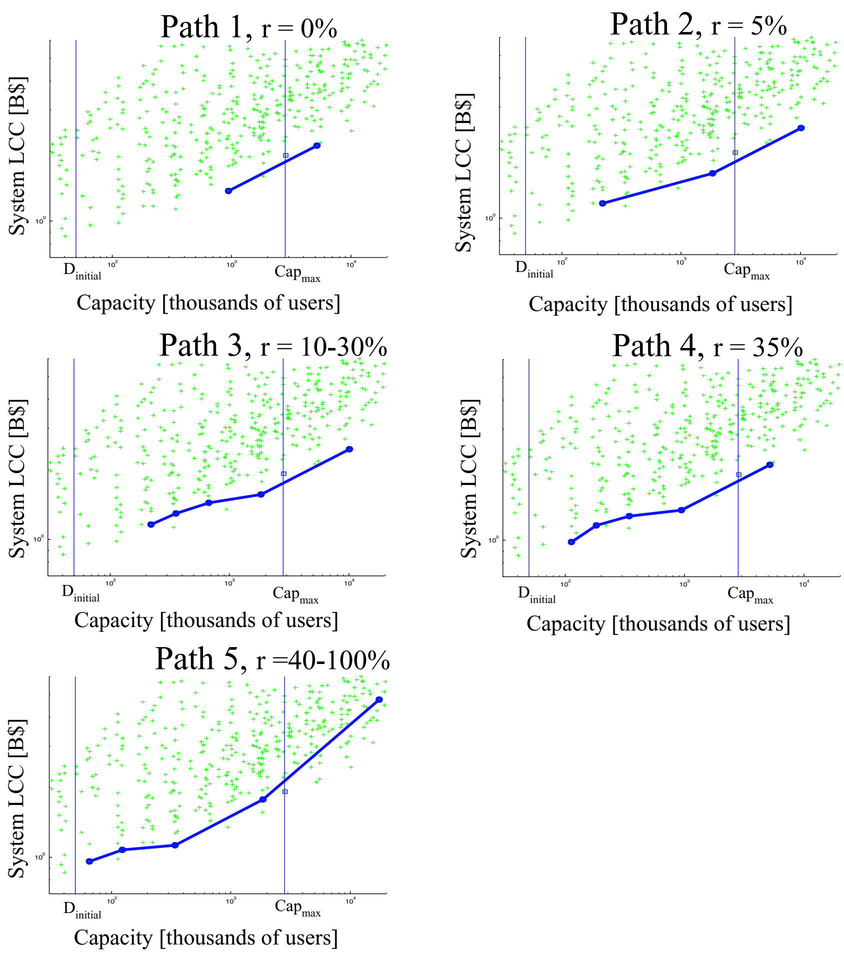

Figure 4-5: Five optimal paths obtained in the parameter study.

So far, the value of the paths were considered when they are optimal. Comparing the values of the different paths for the range of r considered could reveal paths that are robust with respect to variations in r. The life cycle costs of the different paths obtained with the parameter study with respect to the discount rate have been represented in Figure 4-6. The life cycle costs of x^{trad} have been represented too. $LCC(x^{trad})$ decreases slowly as r increases. This is because the most important part of the expenses for fixed architectures corresponds or initial investment which are not discounted since they occur early in time. Path 1 consists only of two architectures, consequently, the effect of the discount rate on its life cycle cost is not important. Paths 4 and 5 can lead to very low life cycle costs but for low values of r, they do not present any economic opportunity. Finally, paths 2 and 3, even though they are not always optimal, always present an important economic value for any given discount rate. Consequently, it is possible to find a path that presents an opportunity for any discount rate when the volatility, σ, is fixed. Figure 4-6 also reveals the crossover points between paths. Given the fact that $10\% \geq r \leq 50\%$ for these types of projects, Path 3 seems to be the most interesting.

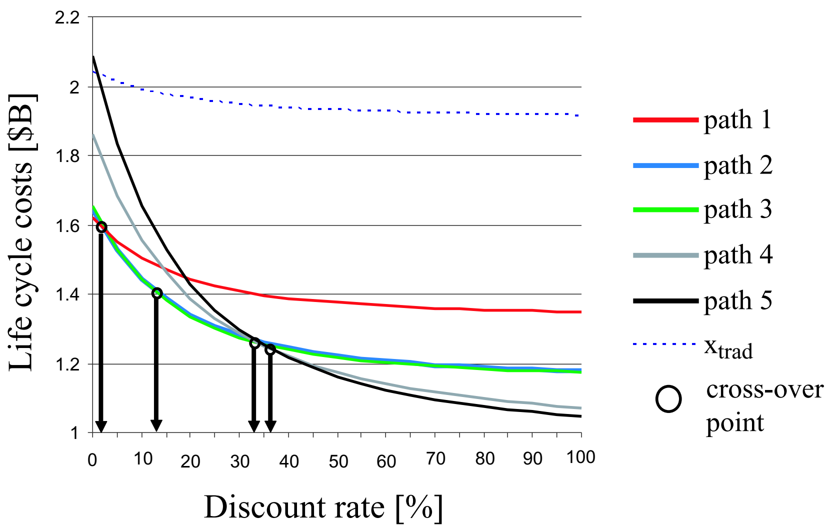

Figure 4-6: Life cycle costs of x^{trad} and the different optimal paths with respect to the discount rate.

4.3.4 Volatility σ

To study the parameter of the optimal paths to volatility, a discount rate $r = 25\%$ was used. The binomial representation sets constraints on the values that σ can take. In the binomial tree, the probability of going up is given by Equation (2.12). This probability of going up cannot be smaller than zero or greater than one. The first case cannot occur but the second can. To prevent that, the following equations have to be verified:

$$\frac{e^{\mu\Delta t}-d}{u-d} \leq 1 \tag{4.2}$$

$$\Leftrightarrow \quad e^{\mu\Delta t} \leq u \tag{4.3}$$

$$\Leftrightarrow \quad e^{\mu\Delta t} \leq e^{\sigma\sqrt{\Delta t}} \tag{4.4}$$

$$\Leftrightarrow \quad \mu\sqrt{\Delta t} \leq \sigma \tag{4.5}$$

Consequently, σ needs to be greater or equal to $\mu\sqrt{\Delta t}$. In the present case, $\mu = 20\%$ and $\Delta t = 2$ years so σ needs to be greater or equal to $0.2 \times \sqrt{2} = 0.2828$. The values that will be considered for σ will range from 0.3 to 1.5.

The value of staged deployment as a function of σ has been represented in Figure 4-7. The value of flexibility is always higher than 30%. Everytime a volatility is

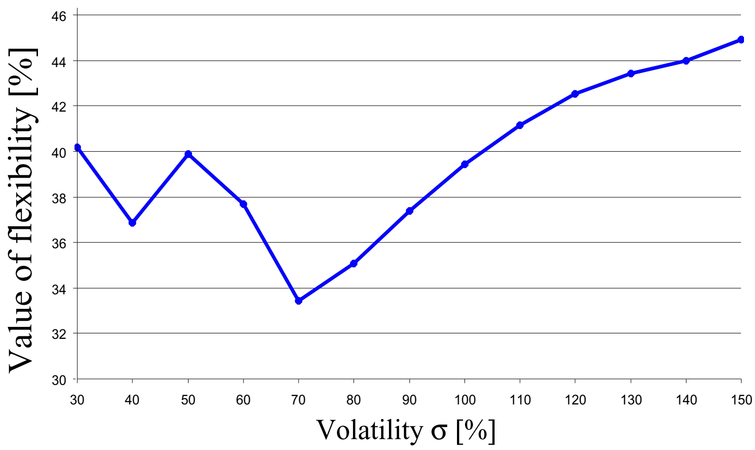

Figure 4-7: Value of flexibility with respect to σ when $r = 25\%$.

considered, a new binomial tree has to be generated with associated demand scenarios.

For values of volatility smaller than 70%, the value of flexibility tends to decrease as volatility is increased which is initially counter-intuitive. As explained in the previous section, the span of the different binomial trees generated is lowered when σ gets smaller and demand does not reach Cap_{max} in any of the scenarios if $\sigma < 70\%$. Having the capability to reach Cap_{max} will not present much value for those cases since this case will never occur. However, the paths considered are constrained to ultimately provide this capacity if necessary. Consequently, when σ is smaller than 70%, the capacity that is targeted is not relevant since it is never attained. The value of flexibility being measured with respect to it, the results obtained are relevant either. When σ is higher than 70%, the scenarios reach important levels of demand and in particular, in many of the scenarios, the demand gets over Cap_{max}. It can be seen that the value of flexibility increases with volatility. The value of flexibility thus increases with uncertainty in future demand.

Six different optimal paths are obtained when σ is changed. Their position in the trade space has been represented in Figure 4-8. For values of σ greater than 70%, as volatility increases, the capacity of the initial architecture decreases. The reason is that the probability of going up, p, decreases as σ increases when μ is fixed. Consequently, the probability of the optimistic scenarios decreases and the best strategy is to start with a capacity as small as possible. The life cycle costs of the optimal paths with respect to σ have been represented in Figure 4-9. It is more difficult in this case to find a path of architecture that offers an interesting value for any level of volatility. However, paths such as paths B and D are almost optimal for any value of σ and could be good candidates. When studying the parameter of the paths with respect to r, a path that offered an interesting value for all r was identified, Path 3. This path is path C in this case. Even though this path is not optimal, it always provides the same value for any level of volatility but is not as interesting as paths B or D. It is thus difficult to find a path that will present value for any values of r and σ and, in order to achieve such choice, an idea of the ranges of those two parameters will be necessary.

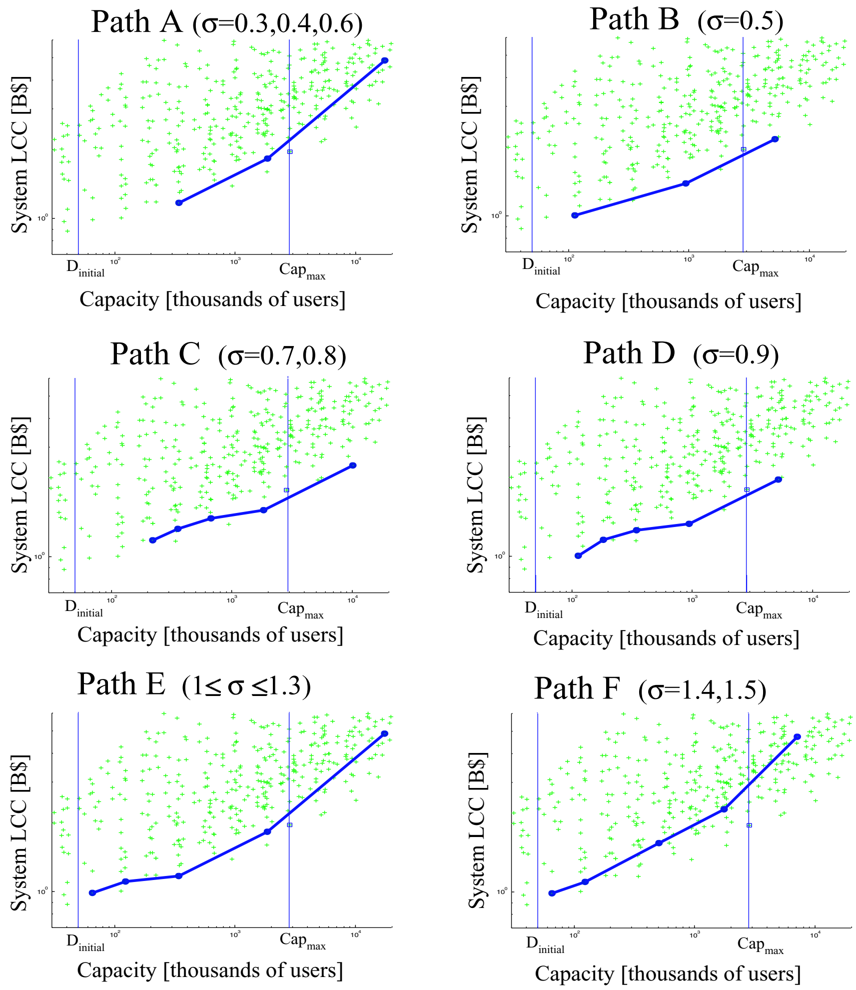

Figure 4-8: Six optimal paths obtained for different values of σ.

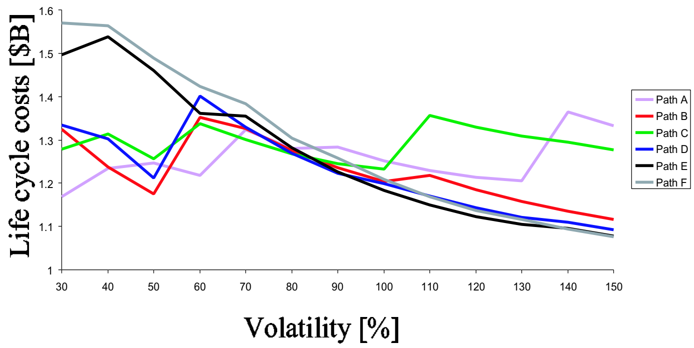

Figure 4-9: Life cycle costs of the different optimal paths with respect to σ when $r = 25\%$.

4.3.5 Other Parameters

The influence of $D_{initial}$ and μ has not been studied in details. However, a few comments can a priori be made about those parameters. It was shown that the value of flexibility is limited when a minimum capacity Cap_{min} is set for the system. If this capacity is constrained to be equal to $D_{initial}$, the value of flexibility will then be limited by $D_{initial}$. The higher this parameter is, the lower the value of flexibility is. Indeed, as $D_{initial}$ gets closer to Cap_{max}, the staged deployment solution does not seem relevant since the actual demand is significant at the beginning of the service. The parameter μ does not affect the span of the binomial tree. Indeed, from Equations (2.11), (2.12) and (2.12) it can be seen that it only affects p. As μ increases, the probability of going up increases too. This parameter thus represents the confidence in the future of the decision makers. Very high values of μ will imply that decision makers consider that the demand will grow with a high probability. Consequently, flexibility will not present value since the future is considered well known.

Setting $D_{initial}$ or μ to high values lowers the value of flexibility. However, doing that is equivalent to considering that the capacity targeted will be reached with a high probability. Consequently, it is equivalent to considering that there is no uncertainty in future demand which is not the case. That is why the influence of these parameters have not been studied much further.

4.4 Conclusions

This case study revealed that staged deployment presents an important economic opportunity since decreases in the life cycle costs between 20% and 45% are obtained. This implies that staged deployment can lower the effective Pareto frontier: the expected life cycle costs of the paths of architectures are lower than the fixed architectures even though they can provide the same capacity. As explained, this strategy also reduces the economic risk associated with the deployment of a large capacity systems. This improvement in the life cycle cost should motivate designers to think about the trade space differently. First, the architectures may not be always considered fixed. Then, as noticed in this case study, the best architectures for staged deployment are not necessarily the ones on the Pareto front but the one that give the maximum flexibility in the future capacity.

The flexibility that evolutions provide to architectures has a price because it needs to be technically embedded. So far, the value of this flexibility has not been taken into account. This solution is relevant if and only if its cost appears to be lower than its a priori value. Chapter 5 proposes a framework to study the problem of reconfiguration and estimate the costs of the reconfiguration process.

Chapter 5

Orbital Reconfiguration of Constellations

The previous chapters revealed the economic opportunity orbital reconfiguration represents for LEO constellations of communications satellites. For this opportunity to be valid, it would be interesting to see if the price to pay for flexibility is smaller than the maximum price one should be willing to pay for it, i.e. on the order of 20-45% of the LCC of a fixed architecture. Therefore, the technical ways to embed this flexibility need to be stuied in detail in order to estimate the extra costs associated with it. This chapter proposes a general method to study orbital reconfiguration. Two ways of embedding flexibility and the associated frameworks to study them are proposed. Finally, the end of the chapter presents several issues that will need to be assessed in future works.

5.1 Problem Definition

5.1.1 Reconfiguration

In orbital mechanics, at a given time, the movement of a spacecraft in an elliptic orbit is defined by its six orbital elements: Ω, ω_0,i, a_0, e_0 and ν_0. They are presented in Appendix B. Therefore, a constellation could be described by giving at a certain time

(Epoch) the orbital elements of each of its satellites. Those positions of the satellites will be called "slots"[1]. A reconfiguration is the set of orbital maneuvers to evolve from a constellation A to a constellation B. If the number of satellites is $N_{sats}(A)$ for A and $N_{sats}(B)$ for B, since the evolutions considered only increase capacity, the number of satellites in B should be higher than in A: $N_{sats}(B) > N_{sats}(A)$. A reconfiguration will thus consist of two types of maneuvers:

- The orbital transfer of $N_{sats}(A)$ on-orbit satellites to slots of constellation B.
- The launch of $N_{sats}(B) - N_{sats}(A)$ satellites to occupy the remaining slots of B.

A conceptual representation of those maneuvers is given in Figure 5-1.

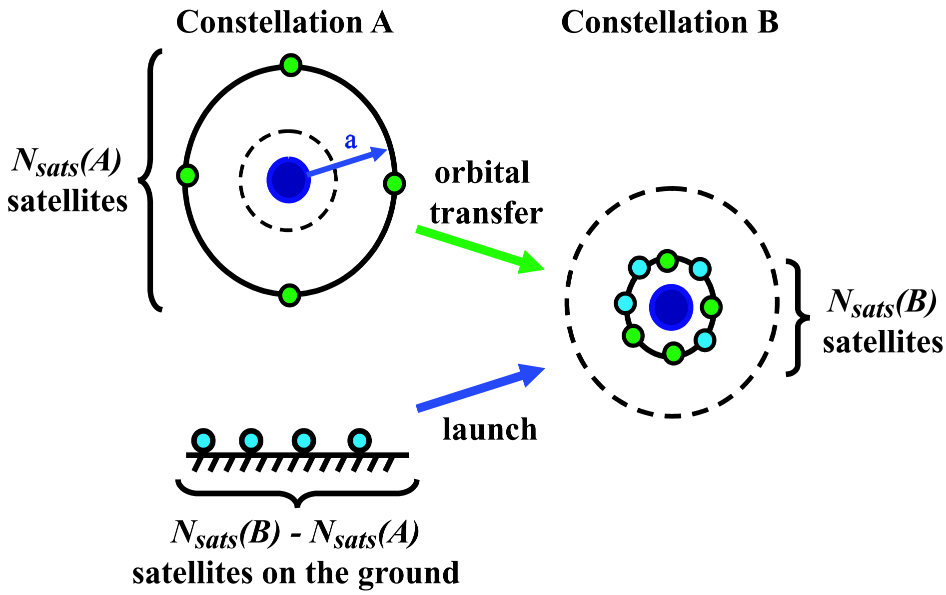

Figure 5-1: Conceptual representation of the reconfiguration maneuvers to evolve from constellation A to constellation B.

In Chapter 3, a cost associated with the launch of additional satellites was taken into account. However, the cost to achieve the orbital transfer of on-orbit satellites was not considered. To do so, different technical ways to achieve those transfers need to be studied and priced. The most important problem with reconfiguration is that the number of possible transfers is large. Some transfers are more efficient than

[1]A slot for each spare satellites can be taken into account.

others in terms of time or energy so those transfers have to be optimized. The next subsection introduces the main issues related to this optimization.

5.1.2 Orbital Transfer Optimization Problem

To study a reconfiguration process in detail, many orbital transfers have to be considered for two reasons. First, since a reconfiguration from constellation A to constellation B consists in ranking a total of $N_{sats}(B)$ satellites that are initially on the ground or on-orbit in $N_{sats}(B)$ orbital slots, the number of possible reconfigurations that exist for an evolution is $(N_{sats}(B))!$. Then, to move a satellite to an orbital slot, there exists many different ways to achieve the transfer. Indeed, many different propulsion systems could be used that do not allow the same types of trajectories. Moreover, the transfer itself can be done by changing all the orbital parameters at the same time or by changing them in a certain order one by one. All those transfers need to be considered because the cost associated with the reconfiguration depends on the ones selected. Indeed, an orbital transfer can be characterized by two parameters. The first one is the time of the transfer $T_{transfer}$ and the second one is the energy that needs to be provided. The exact energy to achieve the transfer requires a knowledge of the mass of the spacecraft considered which may be difficult to obtain. A more convenient way to compare transfers in terms of energy consists in computing the changes of velocities that needs to be done. The changes of velocity are independent of the mass of the spacecraft but, from them and a good knowledge of the propulsion system used, the necessary mass of fuel to achieve the transfer can be obtained . This metric is called the ΔV of the transfer. $T_{transfer}$ and ΔV will directly affect the cost for the transfer. Indeed, if the time of transfer is too long, the satellite will not provide service for a long time which may result in direct loss of revenues corresponding to outage costs. Moreover, if the ΔV of the transfer is too large, a lot of fuel will have to be provided to satellites which will represent an extra cost.

The problem considered is thus an optimization problem where the constraints and objectives depend on ΔV and $T_{transfer}$ and the variables are the different transfers that are available. Reconfiguration need to be modeled in a manner flexible enough to take into account different ways to achieve transfers as well as different objectives. Indeed, a military mission and a commercial mission will not have the same requirements concerning reconfiguration and their objectives will differ. The next sections will present the framework proposed to study reconfiguration and how it solves the problems that were discussed.

5.2 Constellation Reconfiguration Model

5.2.1 Orbital Slots

As explained previously, an orbital slot in a constellation is defined by 6 orbital elements $(a_0, e_0, i, \Omega, \omega_0, \nu_0)$ and the time t at which they are considered. It is important to understand the role of each one of those elements to study reconfiguration. The inclination of the orbit with respect to the equator i and the longitude of the ascending node Ω define the orbital plane in which the satellite evolves. In a constellation providing global coverage, the satellites are equally distributed in several orbital planes. The eccentricity e_0 and the semi-major axis a_0 define the characteristics of the elliptical orbit of the satellite. The argument of perigee ω_0 describes the position of the perigee of the ellipse on the orbital plane and thus gives the orientation of the ellipse in the orbital plane. Finally, the true anomaly ν_0 gives the position of the satellite on the ellipse with respect to the perigee at the time considered. When circular orbits are considered, a_0 and e_0 take particular values: $e_0 = 0$ and $a_0 = R_{Earth} + a$ where R_{Earth} is the mean radius of the Earth and a the altitude of the orbit. Moreover, it is impossible to define a perigee because all the points on the orbit have the same altitude. ω_0 and ν_0 cannot be defined anymore and are replaced by the argument of latitude θ which represents the angle between the position of the satellite and the line of nodes. Consequently, circular orbits simplify the representation of orbital slots and only 4 orbital elements need to be known at a given time. Those elements are the altitude a, the argument of latitude θ, the inclination i and the longitude of the ascending node Ω (RAAN). They have been represented in Figure 5-2.

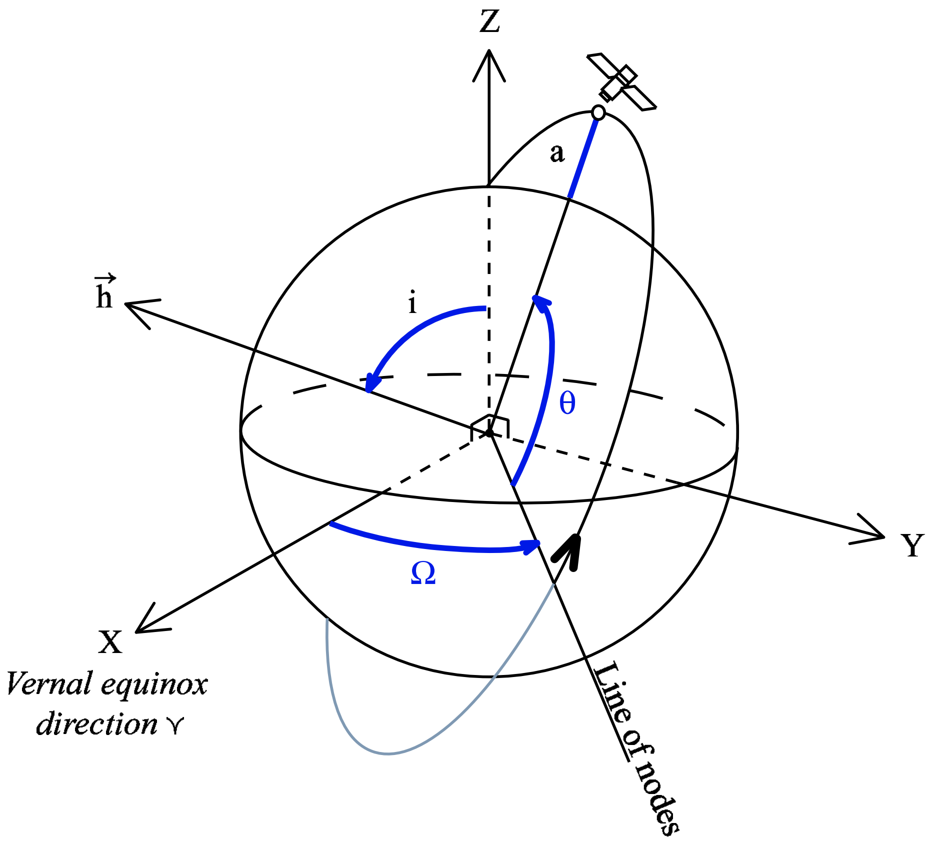

Figure 5-2: Physical meaning of a, i, θ and Ω.

A transfer for a reconfiguration will consist in changing those 4 parameters for a satellite. Since a transfer takes a certain time $T_{transfer}$, as a satellite is transfered to a particular slot, this orbital slot moves. To know $T_{transfer}$, the exact location of the orbital slot needs to be known but, to know the position of this slot, $T_{transfer}$ also needs to be known. Consequently, to determine the times of transfer, an algorithm may have to be created adding several iterations to the process. This difficulty can be avoided by splitting the transfer into two steps. The first step consists in changing i, Ω and a to put the satellite on the same orbit as the orbital slot. After this first step, the slot and the spacecraft may have different arguments of latitude θ. A second step consists in moving the spacecraft to the orbital slot (phasing). This rendezvous maneuver can be done efficiently with respect to time or ΔV. In the case of chemical propulsion, particular orbits called sub-synchronous and super-synchronous orbits offer a range of different possible transfers. In particular, given a maximum time for the rendezvous (respectively a maximum ΔV), it is possible to find the maneuver with minimum ΔV (respectively $T_{transfer}$) that is independent on the relative position of the spacecraft with respect to the orbital slot. Those orbits are presented in Appendix C. Consequently, given this maximum time of transfer or ΔV, the transfer can be seen as a change of orbital plane to which a time or ΔV penalty is added for phasing.

A reconfiguration now consists of transfers from orbital planes to orbital planes with possible changes of altitudes. Additional time and ΔV penalties to achieve a final rendezvous will have to be taken into account but those are easier to estimate than the exact time of transfer. Moreover, those penalties are the same for all transfers since they only depend on the characteristics of the final constellation. The next subsection describes how this decomposition of transfers can be used to find an optimal way to maneuver satellites to their final orbital slots.

5.2.2 Assignment Problem

The reconfiguration problem now consists in transferring and launching satellites to fill orbital planes that have a given number of orbital slots. An analogy can be done between this problem and a classic network flow problem called the assignment problem (see [BT97]). The assignment problem considers n projects $proj_1$, $proj_2$,..., $proj_n$, and n contractors $cont_1$, $cont_2$,...,$cont_n$. The contractors are interested in obtaining a single project. For each project $proj_j$ there is a reward p_j associated with running the project. But, for any contractor $cont_k$, there is a cost c_{kj} associated with accepting project $proj_j$ so that the net profit is $p_j - c_{kj}$. To represent this problem as an assignment problem, a node is created for each contractor and each project. n arcs leave each one of the contractors nodes to connect them to the projects. The cost associated with the arc leaving $cont_k$ and going to project $proj_j$ is c_{kj}. The network flow obtained is represented in Figure 5-3. The assignment problem looks for an assignment of projects to contractors that will maximize the global profit. The reconfiguration problem can be seen as an assignment problem where the satellites are the contractors and the projects the orbital slots. An assignment will thus correspond

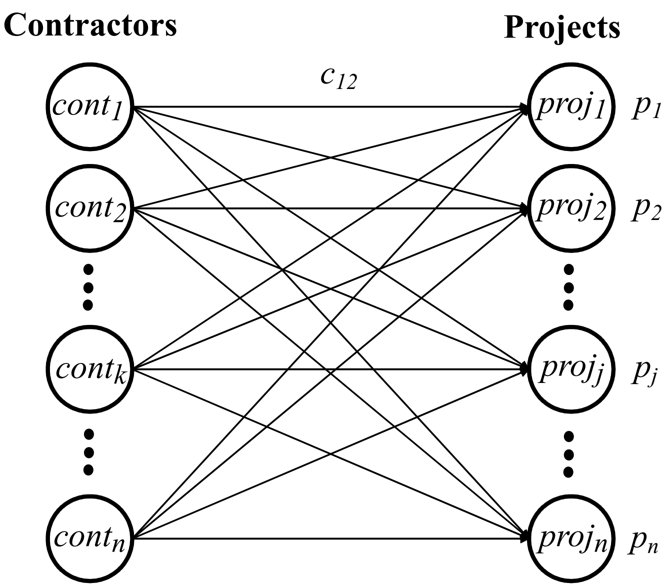

Figure 5-3: Network flow representation of the assignment problem.

to an orbital transfer. The cost of this assignment will be the objective designers want to optimize. To complete the analogy with the assignment problem, it can be assumed that the slots are equivalent and that their reward is zero.

Representing reconfigurations as an assignment problem gives the flexibility to change the objectives with respect to designers priorities. However, for the particular case of LEO constellations of communications satellites, the costs of the transfers should be represented by ΔV for two reasons. The first one is that reducing ΔV is one of the goals since it will eventually reduce the costs for reconfigurations. The second reason is that taking into account the times of transfer will not represent correctly the total time for reconfiguration. Indeed, since many transfers can be done at the same time, the sum of the times of transfer may be different from the total time necessary for the reconfiguration. A good way to take into account time in the assignment problem is by adding a constraint. A maximum time needs to be set for individual transfers and only transfers that are below this time limit are allowed to appear as arcs between the satellites and the slots. A limit on ΔV can actually be added too by removing arcs whom ΔV is over a particular value thus forbidding the considered transfer [2]. From now on, the reconfiguration problem will be modeled as an assignment problem where the costs are the ΔV of the transfers. For other cases of reconfiguration, an adapted cost function can of course be created. The network flow obtained in the case of reconfiguration is represented in Figure 5-4. This representation takes into account the launched satellites and the on-orbit satellites at the same time.

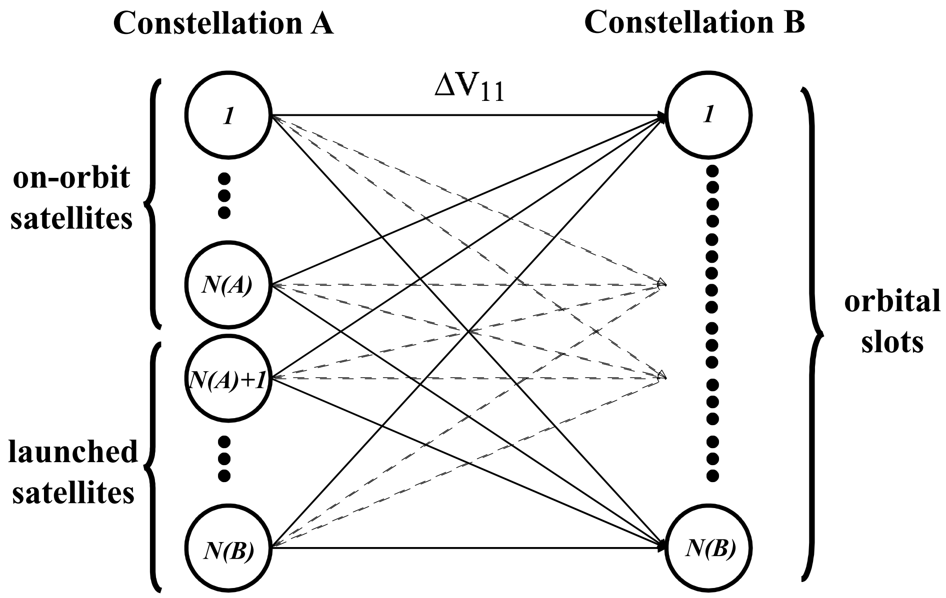

Figure 5-4: Network flow for a reconfiguration between constellation A with $N(A)$ satellites and constellation B with $N(B)$ satellites.

One of the main interests of the representation proposed is that there exist efficient algorithms to determine the best assignment. One of the easiest one to compute is the auction algorithm that is presented in [BT97]. This algorithm develops an auction process during which the contractors bid for the projects that appear best for them until all the contractors are assigned a project. The speed of this algorithm depends on the number of projects and contractors n but also on the value of the highest cost for a project c_{max}. The overall complexity of the algorithm is polynomial since it runs in time $O(n^4 c_{max})$.

[2] To discard an arc from a network flow problem without having to remove it, its cost has to be set to a very high value. This way, this arc will never be used because it is too expensive.

The assignment problem allows an optimization of the reconfiguration with respect to ΔV without having to consider all the existing transfers. It also divides the problem into two parts. First, the ΔV for each possible transfer needs to be studied and compared with certain time or energy constraints. Then, given the best transfers for all the satellites, the assignment problem is solved. To complete this study of the reconfiguration process, the way to determine the ΔV and time of orbital transfers thus needs to be known. This is a difficult problem to solve because transfers can be achieved in many different ways. Moreover, the value of ΔV will differ from on-orbit satellites and launched satellites in the way it is calculated. The next subsection discusses the different problems that need to be addressed to study orbital transfers.

5.2.3 Orbital Transfers

On-orbit Satellites

The transfers considered for on-orbit satellites consist in changing i, Ω and a if there is an altitude change in the evolution. There exist several ways to achieve those transfers because many different trajectories can be taken into account but also because the necessary energy can be provided in different ways. This subsection proposes to discuss the problems that need to be solved to study the characteristics of orbital transfers and exposes the different existing trade-offs.

It is essential to consider all propulsion systems in order to see the possible technical ways to achieve reconfiguration. In particular, chemical propulsion and electric propulsion should be compared. Chemical propulsion allows fast transfers but requires a lot of fuel in terms of mass. On the other hand, electric propulsion does not need a lot of fuel to achieve the same transfer but is very slow and requires a lot of electrical power. Those two types of propulsion systems should be compared separately because the equations for transfer that they use are different. In the case of chemical propulsion where the impulse given to the spacecrafts are considered instantaneous, the equations giving ΔV and $T_{transfer}$ are well known. They can be found in books such as [WL99] or [BMW71]. For electric propulsion, numerical models need to be developed because the thrust that is applied to the spacecraft is continuous.

For each propulsion system considered, different trajectories have to be studied for the transfers. The order in which the orbital elements are changed dramatically affects ΔV and $T_{transfer}$ for a given transfer. Considering all the trajectories possible for a given transfer can represent a lot of computations. However, some principles may provide only relevant cases. To derive those principles, it is necessary to understand how to change the orbital parameters. Battin [Bat99] gives general equations relating the rate of change of i and Ω with the disturbing acceleration applied to the spacecraft $\vec{a_d}$.Those equations were adapted for the particular case of circular orbits:

$$\frac{d\Omega}{dt} = \frac{\sqrt{a + R_{Earth}}\sin\theta}{\sqrt{G \cdot M_{Earth}}\sin i}\vec{a_d} \cdot \vec{i_h} \tag{5.1}$$

$$\frac{di}{dt} = \frac{\sqrt{a + R_{Earth}}\cos\theta}{\sqrt{G \cdot M_{Earth}}}\vec{a_d} \cdot \vec{i_h} \tag{5.2}$$

G is the Newtonian constant of gravitation G and M_{Earth} the mass of the Earth. $\vec{i_h}$ is a unit vector orthogonal to the orbital plane of the orbit and oriented in the same direction as the massless angular momentum of the spacecraft $\vec{h}$. Some conclusions concerning the changes of planes can be done from those two equations. The disturbing acceleration is the most efficient when the scalar product of $\vec{a_d}$ with $\vec{i_h}$ is maximal that is to say when $\vec{a_d}$ is orthogonal to the orbital plane. To affect i or Ω part of the acceleration has to be applied orthogonally. From Equation (5.1), it can be seen that both i and a play a role in the change of the longitude of the ascending node. The more the plane is inclined, the more acceleration will have to be given to the spacecraft. Moreover, the greater a is, the easier it is to change Ω. The worst-case will thus correspond to LEO satellites with polar orbits for which i=90 deg which is the case studied. That is why Ω needs to be changed in the most efficient manner. If i and Ω have to be changed in a transfer, Ω should be changed when the inclination is the smallest to minimize the necessary ΔV for the transfer. Moreover, if a has to be changed too, Ω has to be changed when a is the highest. Equation (5.2) brings the same type of conclusions for i. Changes in i will only depend on a and should be done only when a is the highest possible. These conclusions already give principles concerning the order in which i, a and Ω have to be changed. However, another effect could be taken into account to reduce ΔV. Earth's oblateness induces a force on satellites. If the first order effect of this force is considered, it is called the J_2 effect. In particular, Ω drifts with time if no corrections are done to the orbit of a satellite. The average variation of Ω in degrees per days for circular orbits is given by [Bat99]:

$$\overline{\frac{d\Omega}{dt}} = -9.96\Big(\frac{R_{Earth}}{R_{Earth}+a}\Big)^{3.5}\cos i \tag{5.3}$$

For polar orbits the J_2 effect does not have any effect but for lower inclinations, Ω can be changed without propellant. However, this effect is slow for LEO satellites. If a=1000 km and i=85 deg for instance, the rate of change is $\overline{\frac{d\Omega}{dt}}$=-0.5214 degrees/day. A change of 90 deg will thus take about 6 months with the J_2 effect. But, even if this effect is slow, it does not cost anything in terms of ΔV which can be a great advantage.

To calculate the cost of the different transfers, a module has to be created. Given the initial and final orbital planes, time and ΔV constraints and the type of propulsion system used, this module should provide the optimal transfer possible by exploring the different possibilities exposed. The basic principles presented can reduce the number of cases to consider.

Launched Satellites

The additional satellites are launched from the ground in this model to achieve the evolution. Even though the assignment problem gathers those launched satellites and the on-orbit satellites in the same framework, both maneuvers need to be considered differently. In particular, a time of transfer for launched vehicles cannot be defined and the necessary ΔV to put the satellites in orbit is mainly provided by launch vehicles. This subsection will see how those problems can be addressed.

With launched satellites, the time to launch all the satellites is considered rather than the transfer time. This time depends on the launch vehicles that are used and on the strategy used to launch satellites. It is very difficult to estimate it precisely and an estimation can be sufficient. Indeed, this time is a limiting factor for the reconfiguration because there is no need to transfer the on-orbit satellites if the launched satellites are not already put into orbit. If the on-orbit satellites were transfered before the additional satellites are launched, the service would be disrupted for a long time. Moreover, launching the additional satellites before transferring any of the on-orbit satellites reduces the risk of failure for the reconfiguration because the main risk is that a launch fails. So, it is not necessary to consider the time that it takes to launch a satellite but the time to launch all of them should be estimated to know the delay between the actual beginning of service of the new constellation and the time the decision to evolve is taken. Those estimations could be based on actual cases. For example, it took 2 years and 2 months for Iridium to launch 66 satellites plus 6 spares.

A launch vehicle can generally put several satellites at the same time in the same orbital plane. The necessary ΔV to achieve this transfer is 0 for the satellites because the launch vehicle maneuvers the satellites. Consequently, as a first approximation, in the assignment problem, the arcs leaving the nodes of on the ground satellites should have a ΔV equal to zero. However, this would be true if the assignment problem and the launch strategy perfectly matched that is to say if all the satellites in each launch vehicles were assigned to the same orbital planes. This may not always be the case. Different ways to take this problem into account are proposed in the nest paragraphs. For this, one needs to understand how the simulator determines the number of satellites that are carried by launch vehicles.

To know how many launch vehicles will be necessary to deploy the additional satellites, the simulator considers a database of launch vehicles. For each of them, given the mass and volume of the satellites, it estimates the number of launch vehicles that will be necessary and the associated cost. The launch vehicle that provides the lower cost is considered to be the only vehicle that will be used for the launch of the additional satellites [3]. If there are N_{ground} satellites to launch and if the best solution

[3] To take into account different properties of the launch vehicles, other methods can be used such as the one proposed by Jilla [Jil02].

proposes a vehicle that can carry N_{full} satellites, N_{ground} can be decomposed in the following way:

$$N_{ground} = N_{full}k + r, 0 \leq r < N_{full} \tag{5.4}$$

k and r are integers. If $r = 0$, exactly k launch vehicles will be necessary to achieve the transfer and all the launch vehicles will be full. If $r \neq 0$, k launch vehicles that will be full plus a vehicle carrying r satellites will be necessary. As explained, the problem that may arise is that the assignment problem does not assign a multiple of N_{full} additional satellites in each orbital plane.

To illustrate this problem and the solutions proposed, a simple example is used. Consider an evolution from constellation A with 24 satellites to constellation B that contains 35 satellites divided into 5 orbital planes. Each orbital plane contains 7 satellites. The assignment problem is solved for a given propulsion system and constraints assuming initially that the ΔV associated with launched satellites is zero. The assignment obtained proposes to launch a certain number of additional satellites in particular planes. The distribution obtained is represented in Figure 5-5. 11 satellites need to be launched and the simulator proposes to use a launch vehicle with a maximum capacity N_{full} of 4 satellites per vehicle. Consequently, in this case, 2 vehicles are necessary with 4 satellites in them and a launch vehicle with only 3 satellites. This last vehicle can be used to fill the third orbital plane and another vehicle can start filling the first plane. But, the last vehicle can only place satellites in one orbital plane and satellites are needed in the first, second and fourth plane. Three ways to solve this type of problem are proposed and illustrated with the example that has been introduced.

A first approach would be to change the way the launch strategy is obtained. Instead of looking for a single type of launch vehicle to launch the N_{full} satellites, a best vehicle is sought for each one of the orbital planes. For the example, this means that a best launch vehicle to launch 5 satellites in the first orbital plane will be sought. Then a best vehicle to launch two satellites in the second orbital plane will be sought and so on. This approach may increase the launch costs but the satellites will be

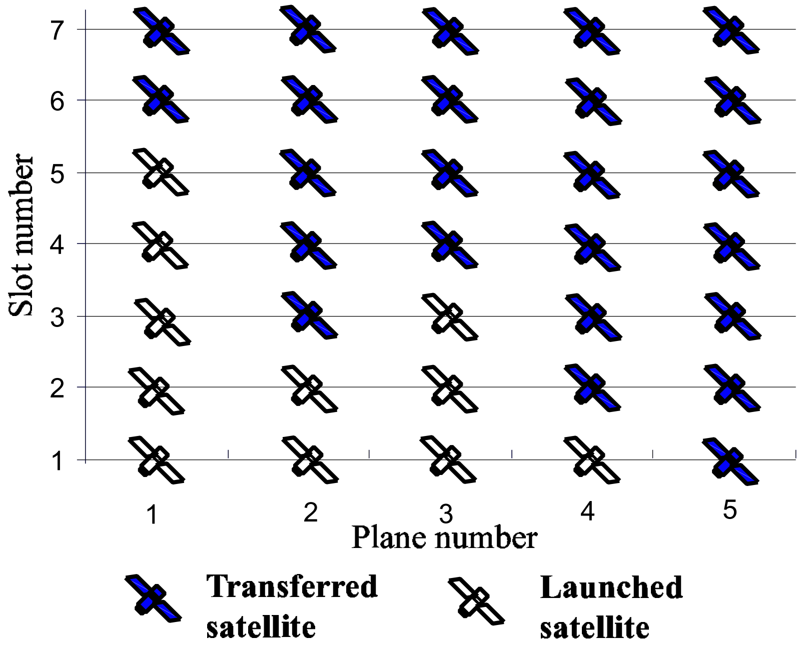

Figure 5-5: Optimal assignment of the launched and transferred satellites in the orbital planes.

launched directly to the correct orbital plane and the assignment solution will not have to be modified. Another advantage is that the module used to find the launch strategy can still be used. But, instead of using it once, it will be used once for each orbital plane where additional satellites need to be launched. This approach gives a priority to the on-orbit satellites and does not imply any penalty on the launched satellites in terms of ΔV. The only drawback is that it may lead to a launch strategy that is not optimal, thus increasing the launch costs.

A second approach can be to launch the satellites respecting the launch strategy and move the satellites to different orbital planes if necessary. For instance, a launch vehicle could launch 4 satellites in the first plane, a second 4 satellites in the second plane and a last three satellites in the third plane. Then two satellites of the second plane are transferred, one to the first plane and the other to the fourth (see Figure 5-6). This strategy respects the optimal launch strategy and the on-orbit satellites assignment. But, it will imply a penalty on the launched satellites in terms of ΔV. However, an optimization needs to be carried out to minimize the total ΔV of the additional transfers. For this, all the possible ways to assign the orbital planes to the launch vehicles need to be studied which can represent a lot of cases.

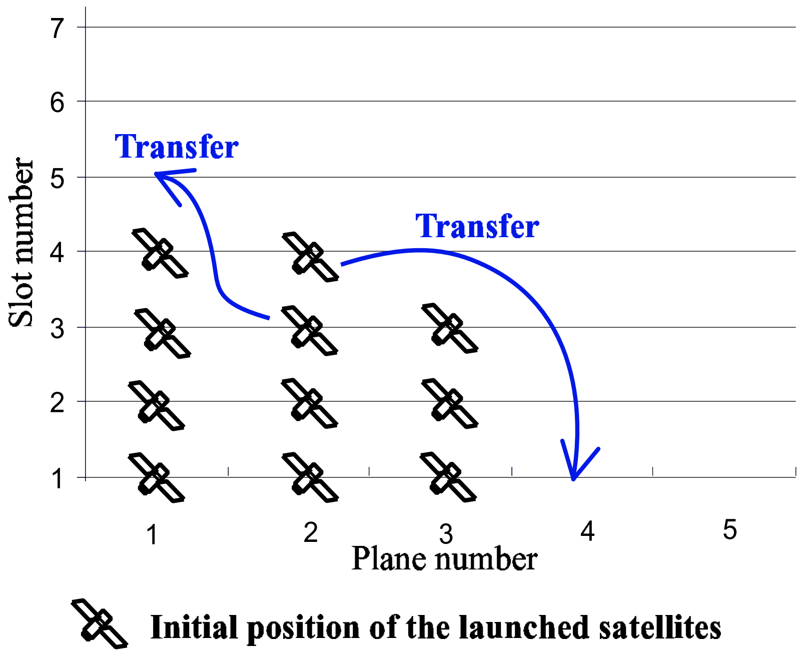

Figure 5-6: Example of a launch strategy for the second approach.

A last approach consists in respecting the launch strategy and change the optimal assignment. This implies that orbital slots are reserved for the launched satellites so that all the satellites of a vehicle can be in the same orbital plane. The on-orbit satellites are thus denied the access to particular slots. To do that the assignment problem needs to be modified by removing the arcs coming from on-orbit satellites going to the reserved orbital slots. For instance, in the example, 4 slots in the first plane could be reserved as well as 4 slots in the second plane and 3 in the third plane to respect the launch strategy. This approach will imply that the assignment is solved twice. A first time to see how the satellites are optimally assigned and a second time after the slots are reserved. However, a best way to reserve the slots needs to be found. This may be difficult to compute because many cases need to be considered. Moreover, this approach will add an extra ΔV to on-orbit satellites because some of them will have to be transferred to slots in a non-optimal manner.

With all those considerations, a module can be created to provide a final assignment and the ΔV associated with each satellites. To know the cost to embed flexibility, the technical way to provide ΔV to the satellites needs to be known. The next section provides a method to study to different technical possibilities based on the model.

5.3 Technical Solutions

This section will propose two technical opportunities to achieve reconfiguration and proposes a method to derive their costs. The technical solutions that are sought should allow maneuvers of the satellites after their initial deployment, sometimes several times. There are two ways to do that. The satellites can move using their own propulsion system or can be moved by another spacecraft. In the first case, the extra cost comes from the fact that satellites will need to carry more fuel. Their masses will thus be increased and more launch vehicles may have to be used. For the second solution, the cost is associated with the spacecraft that will move the satellites. Such spacecrafts are called *space tugs*. If the space tug is a service provided by another company, there is a price associated with it that needs to be taken into account. A third solution could be imagined that uses aspects from those two approaches. Indeed, the satellites could be designed to be "refuelable". A fuel servicer is then needed to provide the extra fuel for reconfiguration to the satellites before evolutions. This section presents how the first two solutions can be studied but does not consider the intermediate solution.

5.3.1 Additional Propellant

From the assignment problem, the necessary ΔV associated with each satellite for transfers can be obtained. This ΔV has to be linked to the necessary mass of fuel to add to each satellites. The relationship between the mass of a spacecraft, the propulsion system used and ΔV is given by the rocket equation that can take two forms:

$$
\begin{aligned}
M_p &= M_f \left[e^{(\Delta V / Isp \cdot G)} - 1 \right] && (5.5) \\
M_p &= M_0 \left[1 - e^{(-\Delta V / Isp \cdot G)} \right] && (5.6)
\end{aligned}
$$

In those equations, M_p is the necessary mass of propellant for the considered transfer, M_0 is the mass of the satellite before the transfer, M_f the final mass of the satellite and Isp is the specific impulse of the propellant used. The specific impulse is a measure of the energy content of the propellant and how efficiently it can be converted into thrust. The specific impulse has the dimension of a time. The higher the specific impulse is, the less mass of propellant is necessary for a given transfer. With electric propulsion, Isp's between 300s and 3000s can be obtained whereas chemical propulsion will provide Isp's on the order of 250s. Consequently, even though electric propulsion is slow because small thrust levels imply small accelerations, it can reduce the mass of fuel to carry compared to a chemical propulsion system.

The rocket equation shows that, to know the necessary mass of propellant for a transfer, the initial or the final mass of the spacecraft considered need to be known. The simulator gives the mass of the satellites without any flexibility. This mass should be the same as the final mass of the satellite after it has gone through all the necessary evolutions. Consider a path of architectures $(i_1, \ldots, i_n)$ and suppose the different assignment problems were solved for each one of the evolutions. For each one of the satellites of the last constellation i_n, the ΔV that has been necessary to go through all the evolutions is known. With the rocket equation and assuming that the final mass of the satellites should be equal to the original mass of the satellites obtained with the simulator, the mass of propellant for each one of the satellites can be found. Adding this mass M_p to the final mass of the satellites M_f, the initial mass of the satellite is obtained.

The fact that the initial mass of the satellites now differs from the one in the original model implies a revision the way the launch costs are calculated once more. Now, to find the best launch strategy, the new masses of the satellites need to be taken into account. The original way to find the launch strategy assumed that all the satellites had the same mass. A different mass for each satellite now needs to be considered. Of course, if a certain approach has been used with launch vehicles to solve the assignment problem, it still needs to be used to calculate the launch costs with the new masses. However, to define a launch strategy in the case where the satellites have to carry extra fuel, the most convenient approach seems to be the first. This approach finds a best strategy for each orbital plane and does not affect the optimal assignment. Even though it will not give an optimal launch strategy, it can easily take into account new masses for the satellites and provide the total launch costs for each evolution.

This method allows a complete estimation of the extra cost for evolutions when the technical solution is to provide some extra fuel to the satellites. It is important to note that the transition cost between architectures are now dependent on the initial and final architectures but also on the path. For instance, since a path such as $(i_1, \ldots, i_{n-1}, i_n)$ as one more evolution than $(i_1, \ldots, i_{n-1})$, there will be different transition and deployment costs for those paths. The price of the real options which is the extra fuel will thus depend on the paths considered and a transition matrix anymore cannot be used anymore. Indeed, the Markov property is lost in this case. Consequently, the optimization process will have to consider the transition costs differently and use a different value for each paths. This does not have any impact on the time of computations but it will increase the necessary memory to store all the evolution costs.

5.3.2 Space Tug

To maneuver the on-orbit satellites without providing extra fuel, a space tug can be used. It is not a classic real option since no initial investment has to be done to use the space tug unless the constellation owner wants to own and operate the tug. However, technical changes may have to be overcome to allow the docking of the tug with the spacecrafts. The cost associated with this change is difficult to estimate but can be consider smaller than the price to pay to use a space tug. Currently, no such space tug exists so it is impossible to use a database of prices to estimate the cost of this solution. This subsections proposes a different approach. Since the price one should be willing to pay for this real option is known, certain requirements concerning the tug can be derived. In particular, it would be interesting to know how much fuel a tug needs to carry or how many satellites can be maneuvered by a single tug that carries a certain amount of propellant. Once again, the network flow model will reveal itself as useful.

To present the framework developed to study the space tug, only one evolution will be considered. Then, it will be explained how to generalize the method to several evolutions. First, the method to obtain the necessary propellant of a single tug will be presented. Even though it may lead to an infeasible solution, this approach will eventually show how to split the work between many tugs. The optimal way for a single tug to move the on-orbit satellites is sought. It is assumed that it does not maneuver the launched satellites at all. As soon as the tug has transfered a satellite to its final orbital slot, it needs to maneuver to another on-orbit satellite. Consequently, a ΔV to move from an orbital slot to an on-orbit satellite will also need to be taken into account. Two possible ways can be explored to solve this problem. A first could consist in solving the original assignment problem and then find an optimal way for the tug to respect this assignment. The second would be to solve the assignment problem and find an optimal strategy for the tug at the same time. The next paragraphs explain how both methods should be represented and solved.

The first approach proposed uses the optimal assignment. Once the assignment problem is solved, it is known which orbital slots will be occupied by satellites that were initially on-orbit. If this assignment is respected, the tug places all the on-orbit satellites in the correct slots. An analogy can be made with the traveling salesman problem (TSP) to represent this problem as a network flow problem. With the TSP, a salesman has to visit several cities once and he wants to minimize the total expenses to visit them all. This problem can be represented as a network flow problem. Each city is represented by a node and arcs are created between the nodes every time a travel is possible. The cost associated with an arc corresponds to the cost of the travel. An example of TSP network is represented in Figure 5-7. The main interest of the TSP

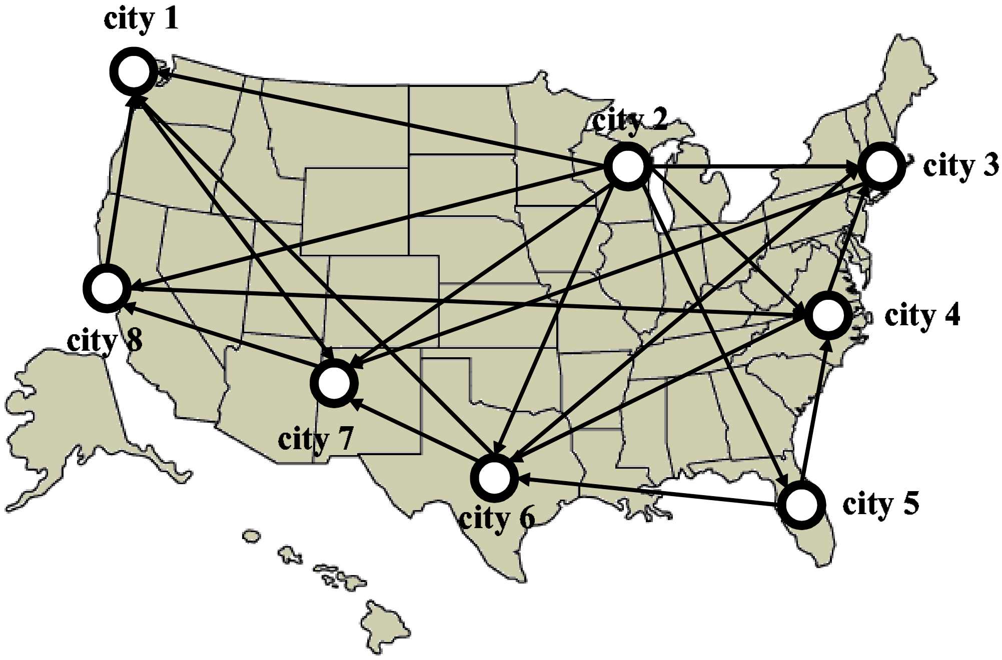

Figure 5-7: Example of a network flow diagram for the Traveling Salesman Problem with 8 cities.

representation is that efficient algorithms exist to solve it (see [BT97]. The space tug problem is a particular type of TSP because the space tug has to respect the optimal assignment. To represent that, a network flow with particular connections has to be created. From the nodes representing on-orbit satellites, only one arc leaves the node. This arc connects the node to the assigned orbital slot. The cost of this arc is the ΔV of the transfer that has already been calculated. From the nodes representing orbital slots assigned to on-orbit satellites, there are arcs connecting the nodes to all the nodes for on-orbit satellites except for the node to which it is assigned. The cost associated with this arc corresponds to the necessary ΔV for the transfer of the tug and should be calculated with the same method that has been used to compute all the ΔV in the assignment problem. The nodes of the launch vehicles are not considered as well as the nodes corresponding to the orbital slots to which they were assigned. The final network flow obtained to go from a constellation A to a constellation B is represented in Figure 5-8. With the network built, as soon as the tug gets to an on-orbit satellite, it has no choice but going to the assigned orbital slot. The assignment solution will thus be respected. Moreover, if the tug gets to an orbital slot, it can only go to on-orbit satellites for its next transfer. From this network representation, the TSP can be solved. The optimal path for the tug that respects the optimal assignment and the ΔV of its transfers are finally obtained.

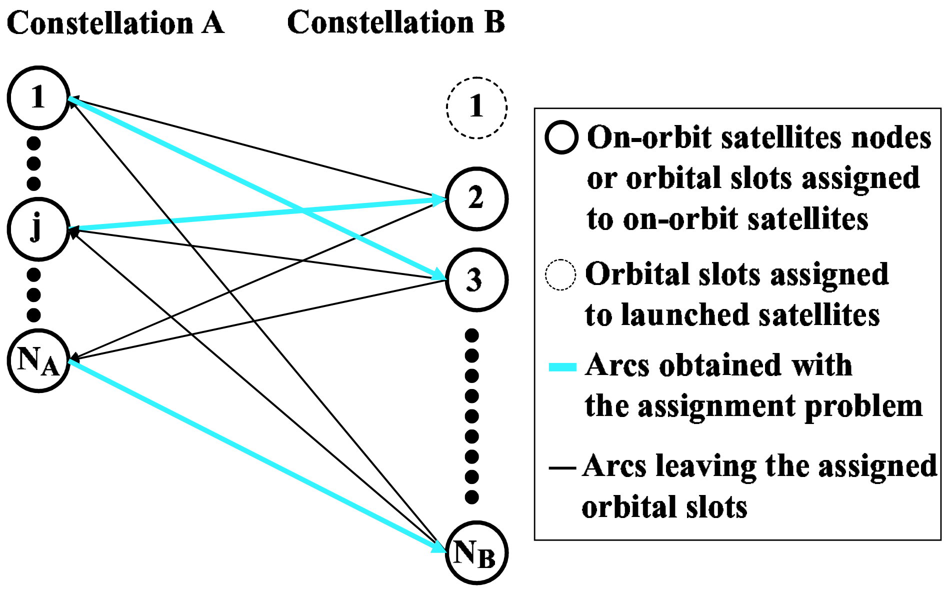

Figure 5-8: Network flow proposed by the first approach.

The second approach proposes to solve directly the assignment problem and the traveling salesman problem. To do that, the network of the assignment problem is considered but, arcs leaving the nodes of the orbital slots are added to all the nodes that represent the contractors in the assignment problem. Consequently, the same network as in the assignment problem is obtained except that the arcs go both ways. Since the tug is not supposed to take into account the launched satellites, this network needs to be adapted. To do that, one just has to consider that to a launched satellite from an orbital slot does not have any cost and that going from a launched satellite to any orbital slot does not have any cost either in terms of ΔV(see Figure 5-9). Transfers to launched satellites will appear in the final path of the tug but their impact on the ΔV of the tug will be equal to zero. From this network, the traveling salesman problem can be solved for the space tug and a best path is obtained as

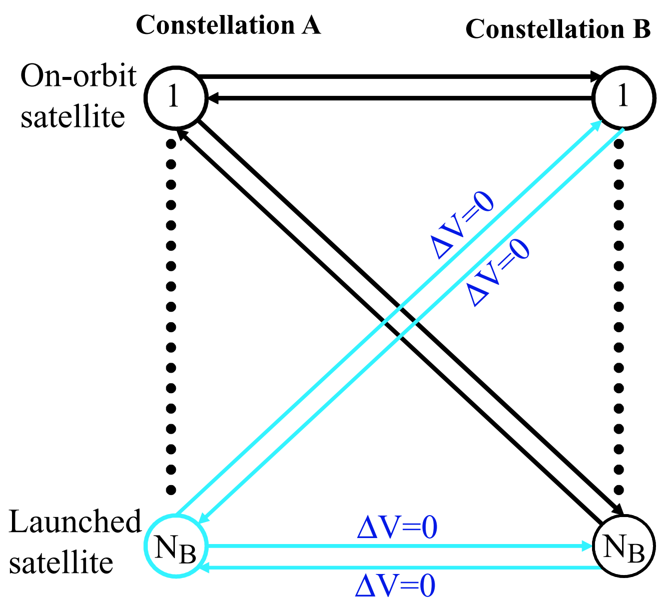

Figure 5-9: Network flow proposed by the second approach.

well as the total ΔV that is necessary. The segments in the path corresponding to visits of launched satellites should be removed. From the path, the assignment of satellites can be easily found since the tug always goes from an on-orbit satellite to an orbital slot. If a single tug was used for the transfers, the ΔV obtained could be infeasible. Several tugs may have to be used to reduce the ΔV of each vehicle. To know how many tugs are necessary, the path followed by the theoretical single tug should be divided in parts so that the ΔV associated with each smaller path is below a certain limit ΔV_{max}. The number of parts necessary to fall below this limit will give the necessary number of tugs. Another problem that needs to be solved is to find the necessary capacity of the tugs when there are have several evolutions. The approaches presented can be followed the same way to get the necessary ΔV of a single tug when there are many evolutions but the final position of the tug after an evolution need to be set as an initial position of the tug for the next evolutions. If many tugs are taken into account, the same principle should apply.

The masses of the tugs is an important parameter because it will directly define the price necessary to launch them. To determine them, it is of course necessary to use the rocket equation. The total ΔV of the tugs cannot be considered to get the mass of propellant when using the rocket equation. Indeed, if the tug is transferring a satellite, the final mass that needs to be considered in the rocket equation is the final mass of the tug plus the mass of the satellite. If the tug is moving to an on-orbit satellite, the final mass to consider is only the one of the tug. Consequently, the exact decomposition in terms of transfers needs to be known for each one of the tugs.

The space tug will thus be a much more difficult problem to solve but the model can provide insights concerning the minimum number of tugs that is necessary to achieve several evolutions given a maximum ΔV. The masses of the tugs can also be obtained for a given propulsion system. The use of a space tug removes the need to provide extra fuel to the on-orbit satellites. Consequently, the transition costs that were previously used are not affected in this case. As a first approximation, given a certain number of space tugs N_{tugs}, one can consider that the maximum price to pay for a tug is the ratio of the economic opportunity revealed with N_{tugs}. The model can thus be used to study the potential of reconfiguration for space tugs since it provides a maximum price companies may be willing to pay to have their spacecrafts reconfigured.

5.4 Remaining Issues

The framework proposed can solve several problems about reconfiguration. However, many issues still need to be considered that are discussed in this section.

The best strategy to evolve from a constellation is to launch the additional satellites first. Once all the launched satellites are placed in their final orbital slots, the on-orbit satellites are transfered. There are two issues related to this process that need to be studied in more detail. The first one is that launch vehicles need to be reserved one or two years in advance to be available. If the reservation is not done, there will be an extra delay before the reconfiguration can be done. A way to deal with this problem would be to assume that the launch vehicles are reserved every year for a possible evolution and if the evolution is not necessary, the reservation is sold. This can be seen as an option on launching the satellites and can be taken into account in the framework. A second problem is that the additional satellites need to be manufactured and that can cause an extra delay. A solution can be to assume that the satellites for the next evolution are built in advance in case of a reconfiguration. This is an extra cost that has to be taken into account.

Once the additional satellites are launched, the on-orbit satellites need to be maneuvered. As they are transfered, they may not be able to provide any service anymore and there may be gaps in the coverage. If the final constellation does not start service as the initial constellation is transfered, it might be impossible to provide any service at some point which is unacceptable. Consequently, both constellations, even though they are partially filled by satellites, should provide service simultaneously. The impact of the quality and availability of the service on the customers should be assessed. If a method can estimate this, a best schedule to move the satellites can be found that reduces the gaps of the constellations.

The last thing that needs to be done with this framework is the link with the economic framework. Once the reconfiguration process is studied in detail and that the price of the real option is known, it needs to be taken into account in the cost calculations. The optimization process is the same except that the transition matrix cannot be used anymore. This final step will show whether the technical solutions proposed are cheaper than the economic opportunity revealed or not.

Chapter 6

Summary and Conclusions

6.1 Summary

The goal of this thesis was to develop a framework to value the economic opportunity provided by staged deployment and apply it to the particular case of LEO constellations of communications satellites. In Chapter 1, the failure of the traditional approach for designing LEO constellations of communications satellites has been presented. Then, the new approach the thesis proposes has been introduced. Finally, an overview of the thesis was given via a thesis roadmap.

In Chapter 2, classic valuation frameworks were presented as well as a particular framework to study staged deployment. In Section 2.1, the potential sources of uncertainty for a system were detailed. Two ways of dealing with uncertainty were presented and compared: flexibility and robustness. In Section 2.2.1, the Net Present Value approach was introduced. It was shown how it fails to value correctly flexible designs and encourages the selection of robust architectures without reducing the economic risk. In Section 2.3, the way flexibility can reduce the economic risk of a project was presented. The issues related to the valuation of flexibility were also exposed. In Section 2.4, two methods to value flexibility were analyzed: Decision Analysis and Real Options Analysis. Finally, Section 2.5 presented the assumptions of the proposed framework to value staged deployment as well as the detailed process to implement it.

In Chapter 3, the implementation of the framework to the particular case of LEO constellations of communications satellites was presented. In Section 3.1, the simulator that has been used for the study was presented. In Section 3.2, the design vector is analyzed to identify the potential sources of flexibility. Two design variables, the altitude and the minimum elevation angle were selected as flexible variables. Then, the possible paths of architectures that could be created from the variations of those variables were analyzed in Section 3.3. This section also revealed the advantage of ordering the families of architectures to implement the framework. Section 3.4 explained how the evolution costs could be determined from the simulator from the decomposition of the design vector. The effects on the shape of the transition matrix of the particular ordering of the architectures were also presented. Section 3.5 shown how the demand scenarios and their probabilities could be obtained from the binomial tree. In Section 3.6, the assumptions used to calculate the life cycle costs of a path given a scenario for demand are presented. Finally, in Section 3.7, the different steps of the optimization process are detailed and recommendations to reduce the necessary computations are given.

In Chapter 4, a particular case study was presented that is inspired by the economic failure of the Iridium constellation. Section 4.2 sets some of the parameters for the study and determined a best fixed architecture that satisfied the same requirements as the Iridium constellation. In Section 4.3, a parameter study was performed with the discount rate and the volatility. The value of flexibility as well as the way best paths of architectures behave for different values of those parameters were analyzed. Significant economic opportunities were obtained for certain values of the discount rate and the volatility. On the order of 20-45% reduction in discounted life cycle cost was consistently shown.

Chapter 5 proposed a general framework to price the reconfiguration process. In Section 5.1, the reconfiguration process was introduced as well as the motivations to optimize it with respect to different objectives. Section 5.2 proposed to model reconfiguration as an assignment problem between satellites and orbital slots. Several considerations concerning orbital transfers were presented. Also, different methods to take into account the launched satellites correctly in the assignment problem were proposed. In Section 5.3, two technical solutions are proposed to transfer the satellites during reconfiguration. The first solution proposed is to add extra fuel to the on-orbit satellites. It is explained how to find the exact penalty in launch costs of this solution and price correctly the solution. The second solution proposed consists in using one or several space tugs to transfer the satellites. It was shown how a maximum price to pay for each space tug could be determined. Finally, Section 5.4 exposed the problems that need to be solved to complete the study of orbital reconfiguration for constellations of satellites.

6.2 Conclusions and Recommendations

6.2.1 Value of Staged Deployment

This thesis showed the economic value of staged deployment for LEO constellations of communications satellites. In particular, through a case study, it has revealed how this strategy could lower the Pareto front with respect to the life cycle costs. This approach asks designers to think differently about the trade space. Indeed, it proposes to seek for paths of architectures in the trade space instead of Pareto optimal architectures. Moreover, it takes into account technical data and a probabilistic representation of the evolution of demand through time. Consequently, the staged deployment strategy represents a real challenge for designers. It does not ask them to design a fixed system from a specific set of requirements but to design a flexible system that can adapt to highly uncertain market conditions. This flexibility needs to be embedded before the deployment of the system. This implies that a real options thinking is adopted. Real options are not necessarily used after the deployment of the system. Designing a system that may not be used is opposed to the traditional approach. So, designers will need to understand the value of designing a system with real options and decision maker the value of investing in such flexibility.

The principles of the approach presented in this thesis do not concern constellations of satellites only. The general framework can be applied to systems with similar characteristics. Future studies could focus on the value of staged deployment for different systems facing a high uncertainty in future demand with important non-recurring costs. If economic opportunities are revealed, such studies could motivate the seek of innovative technical solutions to embed flexibility in the systems.

6.2.2 Reconfiguration

Orbital reconfiguration revealed valuable for constellations of satellites. To determine if this solution is relevant, it has to be priced. A framework was developed in chapter 5 to this effect. However, many considerations that do not appear in this method should be taken into account to complete the study. In particular, the fact that during a transfer, a satellite cannot provide service needs to be modeled and integrated in the optimization in the form of service outage costs. There will certainly be a trade-off between fast but expensive transfers of the satellites and slow transfers that do not require a large mass of fuel but may result in a loss of performance during transfers. Moreover, if the altitude of the satellites is changed, the hardware of the satellites requires modifications. In fact, to produce a particular beam pattern on the ground, the characteristics of the antenna vary with the altitude of the satellites. Reconfiguration within the satellites themselves will thus have to be considered. Actually, significant interest now exists for intra-satellite flexibility (within a single spacecraft) and this topic may reveal more challenging technically. The method should also be completed by taking into account Walker constellations in addition to polar constellations. The variety of configurations this type of constellation offer may provide sets of architectures that are easiest to reconfigure.

The method developed in Chapter 5 could be applied to other staged deployment problems. Indeed, if the capacity is provided by identical elements that are distributed in a particular manner, when capacity is increased, the network flow representation proposed could be applied with necessary adaptations. The systems that could benefit from this approach range from buildings and infrastructures to aircraft fleets. The main interest is that efficient algorithms and software exist to optimize those networks problem and this representation is flexible and adapts well to new constraints.

6.2.3 Other Opportunities for Constellations of Satellites

The flexibility that was studied concerned a possible increase of the capacity of the constellations. It would be interesting to study the value of having the flexibility to change the capability of the constellations (type of service offered) and develop a particular framework for this. Indeed, the Iridium constellation was designed only for mobile phones communications at a data rate of 4.8Kbps per duplex channel. If it had the flexibility to change its bandwidth, it could have provided a totally different type of service such as worldwide access to the internet. The problem that needs to be assessed is to know if it would have saved the Iriridum constellation and what the price for this flexibility is.

If reconfiguration appears too expensive, another type of constellations could be considered to achieve staged deployment. For instance, hybrid constellations that consist of multiple layers of satellites at different altitudes could be considered. Those constellations could be deployed in a staged manner, one layer at a time. Moreover, layers could be deployed to increase the capacity only for certain parts of the globe, thus adapting to the variations in the geographic distribution of demand. Many problems yet need to be studied about those constellations. The capacity has to be defined differently if the coverage differs between regions. Also, if inter satellite links are used, links between the layers need to be created which is a very challenging problem.

This thesis is not about the past, but about the future. It is not claimed that Iridium would have been an economic success if the method proposed here had been employed. The amount of economic damage to shareholders/investors would have been significantly lower, possibly on the order of 20-30% of $B 5.7 thus resulting in $B 1.1-1.7 smaller loss.

Appendix A

Estimation of the envisaged charge per minute

This appendix proposes a formula for the envisaged price per minute of a constellation of communications satellites. It is ultimately compared to the formula proposed in [CGH+92]. The following parameters will be used:

- I [US-$]: net investment for the system. The totality of I is spent at the beginning of operations (its present value is I).
- k [%]: annual interest rate. It is assumed to be equal to the discount rate r.
- T [years]: period until amortization.
- A_{user} [min/month]: average user activity.
- N_{user} [-]: number of subscribers in the system.
- P [US-$-min]: service charge per minute airtime.

For the minimum service charge per minute, the revenues after T years should be equal to the initial investments in terms of present value:

$$I = PV(Revenues) \tag{A.1}$$

For a given year, the revenue made corresponds to the number of billed minutes over the year multiplied by the service charge. The number of billed minutes is $A_{user} \times N_{user} \times 12$. Consequently, the present value of the revenue for the year i is:

$$PV(revenue_i) = \frac{12 \times N_{user} \times A_{user} \times P}{(1+k)^i} \tag{A.2}$$

The present value of the revenues over T years will thus be:

$$\begin{aligned}
PV(Revenues) &= \sum_{i=1}^{T} \frac{12 N_{user} A_{user} P}{(1+k)^i} && \text{(A.3)}\\
&= 12 N_{user} A_{user} P \left(\frac{1-(1+k)^{T+1}}{1-(1+k)} - 1 \right) && \text{(A.4)}\\
&= 12 N_{user} A_{user} P \frac{(1+k)^{T+1} - (1+k)}{k} && \text{(A.5)}\\
&= 12 N_{user} A_{user} P \frac{(1+k)}{k} \left((1+k)^T - 1 \right) && \text{(A.6)}\\
&= I && \text{(A.7)}
\end{aligned}$$

The minimum price for the service is thus:

$$P = \frac{kI}{12 N_{user} A_{user} (1+k) \left((1+k)^T - 1 \right)} \tag{A.8}$$

The formula proposed in [CGH+92] is:

$$P = \frac{I(1+k)^T}{(365 \cdot 24 \cdot 60) T N_{channels} U_S} \tag{A.9}$$

The relationship between A_{user}, U_S, C_s and N_{user} is given by the following equation:

$$N_{channels} U_s = \frac{N_{user} A_{user}}{\frac{365}{12} \cdot 24 \cdot 60} \tag{A.10}$$

Consequently, the minimum charge for service can also be expressed as:

$$P = \frac{12 \cdot I(1+k)^T}{T N_{user} A_{user}} \tag{A.11}$$

Those two equations have been compared for three different constellations. The results obtained are presented in Table A.1. The prices obtained with the equation

System:	*Globalstar*	*Iridium*	*Teledesic*
k	0.22	0.22	0.22
I [US $B]	2.5	5	9
T [years]	3	3	3
$N_{channels}$	$65 \cdot 10^3$	$86 \cdot 10^3$	$7.2 \cdot 10^6$
U_S	0.1	0.1	0.05
Price with Equation (A.9)[US-$/min]	0.44	0.67	0.03
Price with Equation (A.8)[US-$/min]	0.16	0.25	0.01

Table A.1: Prices obtained with the different equations.

proposed in this appendix are on the same order of magnitude than the prices obtained with Equation (A.9). The difference between the results obtained seems to be the difference in the time reference used by the equations. Equation (A.9) computes P with the end of life as a reference, while Equation (A.8) is based on the present value (PV) at time of deployment. However, those two formulas rely on different assumptions that may not be true in reality. In particular, it is assumed that the total capacity of the system is used through time which may not be the case. Moreover, the average user activity is supposed constant. These assumptions do not take into account market considerations and a low demand will force decision makers to increase the service charge.

Appendix B

Orbital Elements

This appendix will present the six orbital elements that are used to position a spacecraft at a given time when it is in an elliptic orbit. First, the orbital elements that determine the position of the spacecraft on the ellipse will be given and then the orbital elements giving the position of the ellipse with respect to the central body will be presented.

B.1 Position of the Spacecraft on the Ellipse

The geometric characteristics of the ellipse on which the spacecraft is can be obtained from two parameters: the semi-major axis a_0 and its eccentricity e_0. For an elliptic orbit, we have $0 \leq e_o < 1$ [1]. The point on the ellipse that is the closest from the central body is called the periapsis and the point that is the farthest is called the apoapsis. If Earth is the central body, those points are called the perigee and the apogee. From now on, it will be considered that Earth is the central body. a_0 is equal to half the distance between the apogee and the perigee. The distance between the perigee and Earth's center is equal to $a_0\,(1 - e_0)$ and the distance between the apogee and Earth is equal to $a_0\,(1 + e_0)$. If $e_0 = 0$, those two distances are equal and all the points of the orbit are at the same distance with respect to Earth. This particular case will thus correspond to a circular orbit. The position of the spacecraft on its

[1] $e_0 = 1$ corresponds to a parabolic orbit and $e_0 > 1$ to a hyperbolic orbit.

orbit at a given time is given through the true anomaly ν_0. The true anomaly is the angle between the perigee and the spacecraft measured from Earth. All the elements and distances introduced in this section are summarized in Figure B-1.

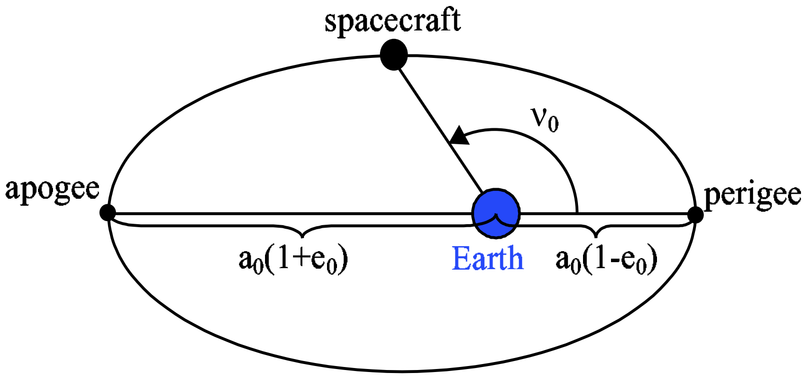

Figure B-1: Geometric meaning of a_0, e_0 and ν_0.

B.2 Position of the Ellipse with Respect to Earth's Frame

Earth's frame[2] is defined with respect to fixed stars. The X axis is also called the vernal equinox and points in the direction of the constellation of the Ram. It is represented with the zodiacal symbol Υ. The Z axis points from the center of the Earth to the geographic North pole (not the magnetic North pole). The Y axis is obtained from those two axes: $\vec{Y} = \vec{X} \times \vec{Z}$. The plane defined by the X axis and the Y axis is the equatorial plane. To position the elliptic orbit with respect to this frame, 3 orbital elements are necessary. The first orbital element determines the inclination of the orbit with respect to the equator. If $\vec{h}$ is the massless angular momentum of the spacecraft, this vector is perpendicular to the orbital plane. The inclination of the orbital plane is thus defined as the angle i between the Z axis and the vector $\vec{h}$. The second element positions the intersection of the orbital plane with respect to the equator. If the orbital plane is inclined, the orbit of the spacecraft intersects the

[2]This frame corresponds to Earth-centered inertial coordinates that are frequently called Geocentric Inertial (GCI).

equatorial plane in two points. In one case, the spacecraft goes through the equatorial plane from the South pole to the North pole. The associated point of intersection will thus be called the ascending node. In the other case, the spacecraft goes from the North pole to the South pole and the corresponding intersection will be called the descending node. The angle between the X axis and the ascending node is called the longitude of the ascending node (also called right ascension of the ascending node RAAN) and is noted Ω. Ω and i are sufficient to define the orbital plane. To position the ellipse on this orbital plane, only the relative position of the perigee with respect to the ascending node needs to be known. The third orbital element is thus the angle between the ascending node and the perigee. It is called the argument of perigee and is noted ω_0. The three orbital elements i, Ω and ω_0 have been represented in Figure B-2.

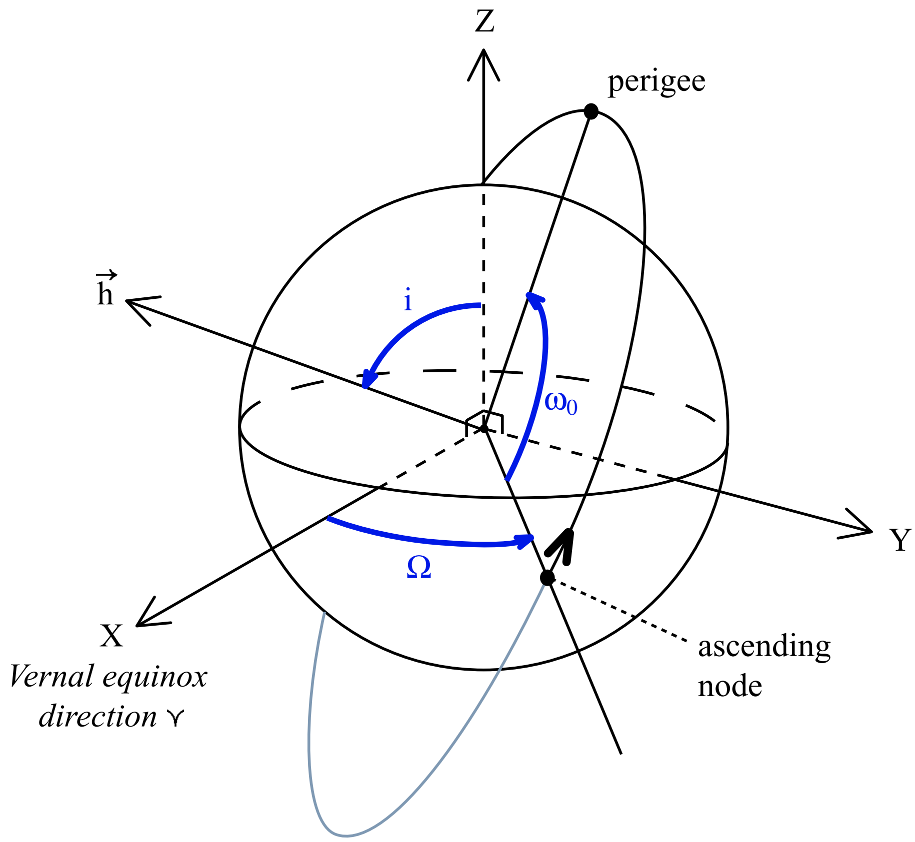

Figure B-2: Geometric representation of i, Ω and ω_0.

Appendix C

Sub-synchronous and Super-synchronous Orbits

This appendix introduces two particular orbital transfers: sub-synchronous and super-synchronous transfers. They are used when spacecrafts are placed on the correct orbit but need to achieve a final maneuver in order to be at a particular position on the orbit at a given time (phasing). Those transfers only concern spacecrafts with a high thrust propulsion system. It is shown how to obtain the ΔV and time of transfer $T_{transfer}$ and the trade-off existing between those two objectives. Also, principles to determine a priori time or ΔV penalties for the transfer are presented.

C.1 Problem Definition

Let's consider a spacecraft Sp on an elliptic orbit at a given time. This orbit will be called the reference orbit. We want this spacecraft to move to another orbital slot Sl on this same reference orbit. The relative position of this orbital slot with respect to the spacecraft will be noted in two different manners depending on the eccentricity e_0 of the orbit. If $e_0 = 0$, the orbit is a circular orbit and we position Sl with respect to Sp by considering the difference between their arguments of latitude[1] which is

[1] θ can also be seen as the argument of latitude of Sl when the argument of latitude of Sp is equal to zero.

constant through time. If $0 < e_0 < 1$, we consider the true anomaly ν_0 of Sl when Sp is at the apogee of the orbit. The different notations have been represented in Figure C-1.

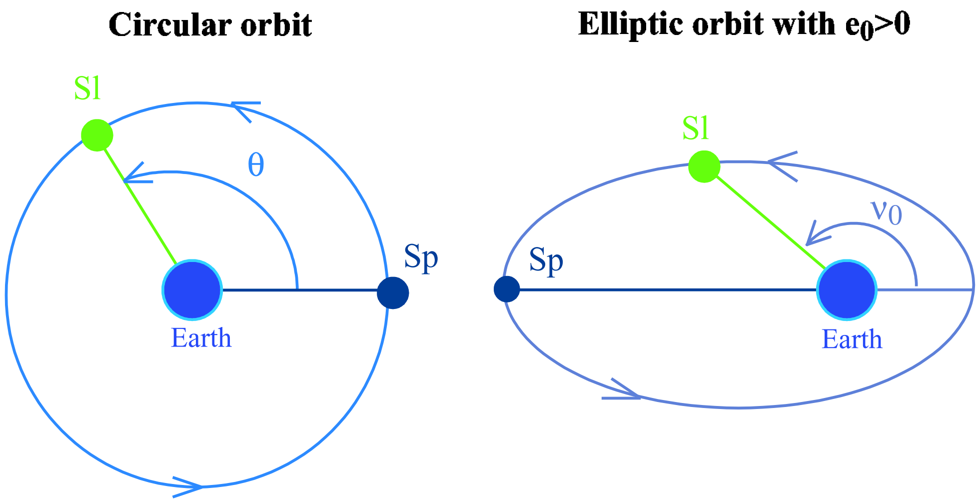

Figure C-1: Relative position of Sl with respect to Sp when the reference orbit is circular or elliptic.

To achieve the transfer of Sp to Sl, we consider two particular types of transfers: sub-synchronous and super-synchronous transfers. The first type of transfer has been decomposed in Figure C-2 in the case of a circular reference orbit. During a sub-synchronous transfer, an impulse is initially given to place Sp on a new orbit with a lower perigee but with the same apogee (step (a) in Figure C-2). Such orbit will be called a transfer orbit (represented in step (b) of Figure C-2). To do so, the impulse has to be given at the apogee of the orbit and tangentially to the velocity. With a circular orbit, the apogee and the perigee are the same so the impulse can be given at any point in time. The effect of lowering the perigee is that Sp “accelerates” with respect to the reference orbit and consequently relatively to Sl. Since the reference orbit and the transfer orbit share the same apogee, after one period of the transfer orbit, Sp is on the apogee of the reference orbit. If the perigee of the transfer orbit is selected correctly, after a certain number of periods, Sl and Sp rendezvous at the apogee of the reference orbit (step (c) in Figure C-2). A final impulse is then given to Sp to place it back on the reference orbit (step (c) and (d) in Figure C-2).

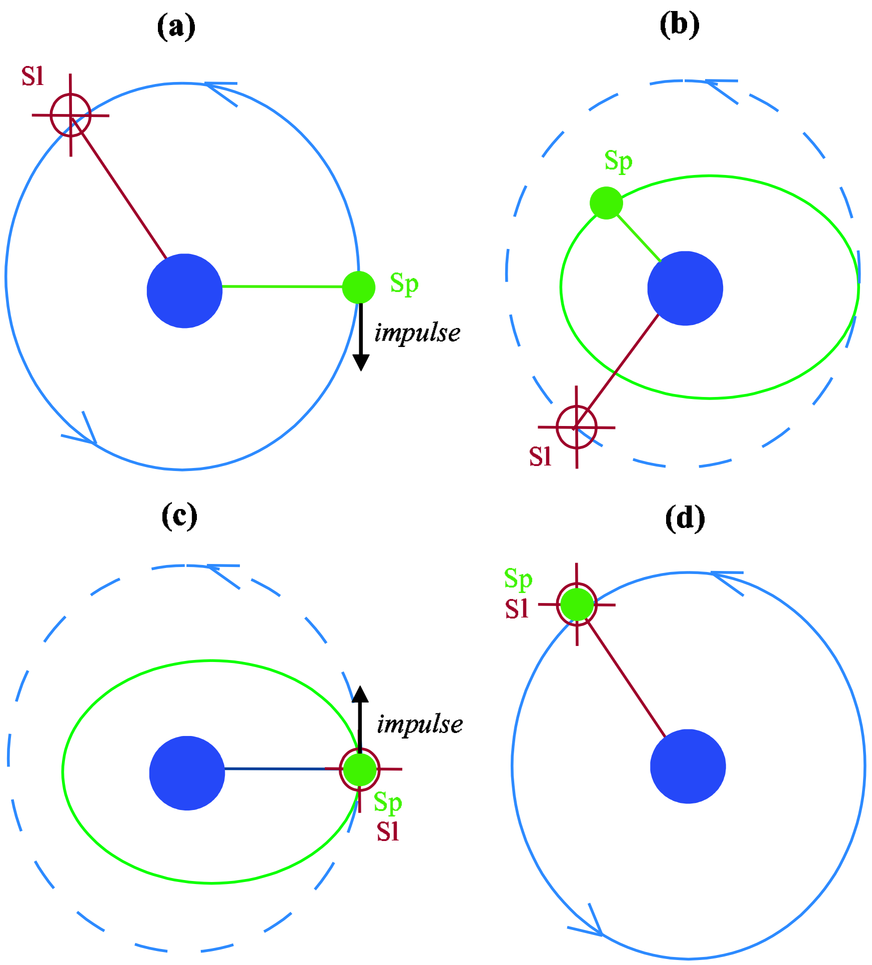

Figure C-2: Decomposition of a sub-synchronous transfer and rendezvous between *Sp* and *Sl* when the reference orbit is circular.

Super-synchronous rely on the same principle except that the impulse is given to the spacecraft so that the perigee is increased. The result is that *Sp* seems to slow down compared to *Sl*. If the transfer orbit is selected correctly, the rendezvous can be achieved after a certain number of periods. The next section will explain how the total ΔV for such transfers can be calculated as well as the necessary time for transfer.

C.2 Calculating $T_{transfer}$ and ΔV

To explain the necessary calculations for sub-synchronous and super-synchronous orbit, we will consider separately the two cases relative to the eccentricity of the reference orbit that we already introduced. The circular case will be explained first and then the way it can be generalized to the elliptic case will be introduced.

C.2.1 Circular Reference Orbit

Transfer Time

Selecting the good transfer orbit for Sp is a synchronization problem. The time reference that will be used corresponds to the instant at which the first impulse is given to Sp. The initial position of Sp will be the only point shared by the reference and the transfer orbit and thus will be the only position where the rendezvous can occur. To know the time of transfer, we should consider the trajectories of Sp and Sl separately before the rendezvous. Sp initially gets to the transfer orbit that has a period $\Pi_{transfer}$. We assume it orbits k_{Sp} times before the final rendezvous. The time of transfer for Sp will thus be:

$$T_{transfer} = k_{Sp} \times \Pi_{transfer} \tag{C.1}$$

The rendezvous can only occur at the initial position of Sp. We can decompose the trajectory of Sl into two parts. First, Sl gets to the initial position of Sp. The necessary time for this maneuver will be called T_{delay}. The angle between Sl and the initial position of Sp is θ at time $t = 0$. To get to the initial position of Sp, Sl needs to achieve a rotation of $\pi - \theta$ radians. Consequently, we have:

$$T_{delay} = \frac{\pi - \theta}{\pi} \times \Pi_{ref} \tag{C.2}$$

Π_{ref} corresponds to the period of the reference orbit. The second part in the trajectory of Sl corresponds to a certain number of full orbits before the rendezvous maneuver.

If there are k_{Sl} orbits, it corresponds to a time equal to $k_{Sl} \times \Pi_{ref}$. consequently, the time of transfer will also be equal to $T_{delay} + k_{Sl} \times \Pi_{ref}$. We thus have:

$$T_{transfer} = T_{delay} + k_{Sl} \times \Pi_{ref} = k_{Sp} \times \Pi_{transfer} \tag{C.3}$$

For given values of k_{Sl} and k_{Sp}, synchronizing Sl and Sp corresponds to determining the value of $\Pi_{transfer}$ that will satisfy Equation (C.3). The possible choices for k_{Sl} and k_{Sp} seem infinite. However, two particular cases only need to be considered: $k_{Sp} = k_{Sl} + 1$ and $k_{Sp} = k_{Sl}$. In the first case, we can also write $k_{Sl} = k_{Sp} - 1$ to see that the spacecraft orbits one more time than the orbital slot. The spacecraft is necessarily on a sub-synchronous orbit since it "accelerates". If k_{Sp} gets higher than $k_{Sl} + 1$, the spacecraft orbits many more times than Sl but needs to be more accelerated than for $k_{Sl} = k_{Sp} - 1$. The necessary impulse will thus be larger than for $k_{Sp} = k_{Sl} + 1$. However, the time of transfer will still be the same since it only depends on k_{Sl}. So, the same time of transfer is obtained with a higher necessary impulse. Consequently, the cases for which $k_{Sp} > k_{Sl} + 1$ are not relevant and don't need to be considered. $k_{Sp} = k_{Sl}$ is the second case we need to study. In this case, the orbital slot and the spacecraft achieve the same number of orbits but the orbital slot also has the time to achieve the $\pi - \theta$ rotation. Consequently, the spacecraft is slower than the orbital slot and the transfer orbit is a super-synchronous orbit. If $k_{Sp} < k_{Sl}$, Sp needs to be more decelerated but $T_{transfer}$ remains the same. Consequently, the transfers corresponding to $k_{Sp} < k_{Sl}$ should not be considered.

Finally, for a given value of k_{Sl}, only two values of k_{Sp} need to be considered: $k_{Sl} + 1$ that corresponds to a sub-synchronous orbit and k_{Sl} that corresponds to a super-synchronous orbit. We will thus replace k_{Sl} by k and set k_{Sp} to $k + 1$ if a sub-synchronous orbit is considered or to k if a super-synchronous orbit is considered. From Equation (C.3), it can be seen that the time of transfer for a given k is:

$$T_{transfer} = \left(\frac{\pi - \theta}{\pi} + k\right) \Pi_{ref} \tag{C.4}$$

ΔV of the Transfer

The ΔV budget for Sp consists of two impulses: one to place Sl on the transfer orbit, one to bring it back to the reference orbit during the rendezvous. Those two impulses have the same absolute value so only one of them needs to be computed. The calculations for the sub-synchronous transfer are going to be described. The formula for the super-synchronous transfer will be given directly but the principles used are the same.

We call $\Delta V_{initial}$ the necessary impulse to place Sp on the transfer orbit. $\Delta V_{initial}$ is the difference between the velocity of the spacecraft on the reference orbit V_{ref} and its velocity at the apogee of the transfer orbit V_{apogee}. The velocity of the spacecraft on the reference orbit can be obtained directly from the radius R_{ref}. Indeed, we have:

$$V_{ref} = \sqrt{\frac{GM_{Earth}}{R_{ref}}} \tag{C.5}$$

The velocity of the spacecraft at the apogee of the transfer orbit can be obtained using the Vis-Viva Integral:

$$V_{apogee}{}^2 = GM_{Earth}\left(\frac{2}{R_{apogee}} - \frac{1}{a_{transfer}}\right) \tag{C.6}$$

In this equation, $a_{transfer}$ is the semi-major axis of the transfer orbit and R_{apogee} the distance between the center of the Earth and the apogee. From Figure C-3, we note that $R_{apogee} = R_{ref}$. Consequently, to determine V_{apogee}, we only need to determine $a_{transfer}$. It can be obtained from the period of the transfer orbit:

$$\Pi_{transfer} = 2\pi\sqrt{\frac{a_{transfer}^3}{GM_{Earth}}} \tag{C.7}$$

$\Pi_{transfer}$ can be related to the period of the reference orbit using Equations (C.3) and (C.4) and by replacing k_{Sl} with k and k_{Sp} with $k+1$. We obtain:

$$\Pi_{transfer} = \left(\frac{\pi - \theta}{\pi} + k\right)\frac{\Pi_{ref}}{k+1} \tag{C.8}$$

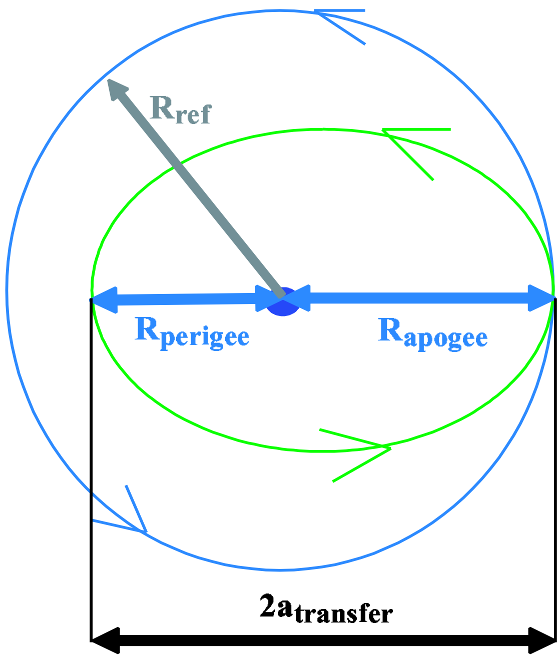

Figure C-3: Representation of the notations used for the calculations of ΔV_{apogee}.

The period of the the reference orbit is related to R_{ref} through the following equation:

$$\Pi_{ref} = 2\pi\sqrt{\frac{R_{ref}^3}{GM_{Earth}}} \tag{C.9}$$

Consequently, from Equations (C.7), (C.8) and (C.9) we can write:

$$2\pi\sqrt{\frac{a_{transfer}^3}{GM_{Earth}}} = \left(\frac{\pi-\theta}{\pi}+k\right)\frac{2\pi\sqrt{\frac{R_{ref}^3}{GM_{Earth}}}}{k+1} \tag{C.10}$$

This last last equation can be simplified to express $a_{transfer}$ as a function of k, θ and R_{ref}:

$$a_{transfer} = \left(\frac{\pi-\theta}{\pi}+k\right)^{\frac{2}{3}}\frac{R_{ref}}{(k+1)^{\frac{2}{3}}} \tag{C.11}$$

By replacing $a_{transfer}$ in Equation (C.6), we obtain a final expression for V_{apogee}:

$$V_{apogee} = \sqrt{GM_{Earth}\left(\frac{2}{R_{ref}} - \left(\frac{\pi-\theta}{\pi}+k\right)^{-\frac{2}{3}}\frac{(k+1)^{\frac{2}{3}}}{R_{ref}}\right)} \tag{C.12}$$

As explained previously, the total ΔV for the transfer is twice the difference between V_{ref} and V_{apogee}. Consequently, we have:

$$\Delta V_{transfer}^{sub} = 2\sqrt{GM_{Earth}}\left(\sqrt{\frac{1}{R_{ref}}} - \sqrt{\frac{2}{R_{ref}} - \left(\frac{\pi-\theta}{\pi}+k\right)^{-\frac{2}{3}}\frac{(k+1)^{\frac{2}{3}}}{R_{ref}}}\right) \quad \text{(C.13)}$$

The same process can be followed to derive the equation giving $\Delta V_{transfer}^{super}$. In this case, it has to be noted that $R_{perigee} = R_{ref}$ if $R_{perigee}$ is the perigee of the transfer orbit. Moreover, the impulse is not given in the same direction so the signs are not the same in the final expression. Finally, $k_{Sp} = k$ for a super-synchronous orbit. We all those considerations, the equation giving the necessary ΔV for a super-synchronous transfer can be derived directly:

$$\Delta V_{transfer}^{super} = 2\sqrt{GM_{Earth}}\left(\sqrt{\frac{2}{R_{ref}} - \left(\frac{\pi-\theta}{\pi}+k\right)^{-\frac{2}{3}}\frac{k^{\frac{2}{3}}}{R_{ref}}} - \sqrt{\frac{1}{R_{ref}}}\right) \quad \text{(C.14)}$$

C.2.2 Elliptic Reference Orbit

A generalization of the discussion we made can be done for reference orbits that are elliptic. We will not give the exact equations of those cases but give the necessary elements to achieve the derivations.

With elliptic orbits, the parameter T_{delay} is more difficult to calculate. For this, we need to introduce the eccentric anomaly of the orbital slot at time $t = 0$ that we will note E_{sl}. Kepler's equation links the eccentric anomaly of Sl to the time t and a time reference τ:

$$\frac{2\pi}{\Pi_{ref}}(t-\tau) = E_{Sl} - e_0 \sin E_{Sl} \quad \text{(C.15)}$$

e_0 and Π_{ref} are the eccentricity and the period of the reference orbit. T_{Delay} is the time for Sl to go from its initial eccentric anomaly to the initial position of Sp. Since Sp is initially considered at the apogee, its eccentric anomaly is equal to π. Consequently, Kepler's equation gives us the relationship between T_{delay} and E_{Sl}:

$$T_{delay} = \frac{\Pi_{ref}}{2\pi}\left[(\pi - e_o \sin \pi) - (E_{Sl} - e_o \sin E_{Sl})\right] \tag{C.16}$$

$$= \frac{\Pi_{ref}}{2\pi}\left[\pi - (E_{Sl} - e_o \sin E_{Sl})\right] \tag{C.17}$$

t=To determine T_{delay}, the initial eccentric anomaly of Sl needs to be determined. There exists a relationship between the true anomaly ν_0 of a spacecraft and its eccentric anomaly. It is given by Battin [Bat99]:

$$E_{Sl} = 2\arctan\left(\sqrt{\frac{1-e_0}{1+e_0}}\tan\frac{\nu_0}{2}\right) \tag{C.18}$$

From those equations the time of transfer can be obtained for elliptic orbits. From the time of transfer, the characteristics of the transfer orbits can be obtained as well as the necessary ΔV.

C.3 Performance of the Transfers

The advantage of the sub-synchronous and super-synchronous transfers is that only two impulses are necessary. This reduces the necessary ΔV to achieve the transfer. Another way to achieve the rendezvous consists in using two Hohmann transfers. Indeed, in a first time, Sp could be transferred with a Hohmann transfer to a lower orbit to synchronize with Sl. Then, when the synchronization is achieved, Sp is transfered with another Hohmann transfer to the reference orbit and the rendezvous is realized. Each Hohmann transfer requires two impulses. The time for transfer and ΔV depend on the altitude used for the transfer. The equations necessary to compute those objectives can be found in [WL99]. To show the advantage of the proposed transfers, we represented $T_{transfer}$ and ΔV for sub-synchronous and super-synchronous transfers as well as for the transfers involving two Hohmann transfers. The results are presented in Figure C-4. The double Hohmann transfers are represented with a line because these transfers depend on the altitude of the low orbit considered which is

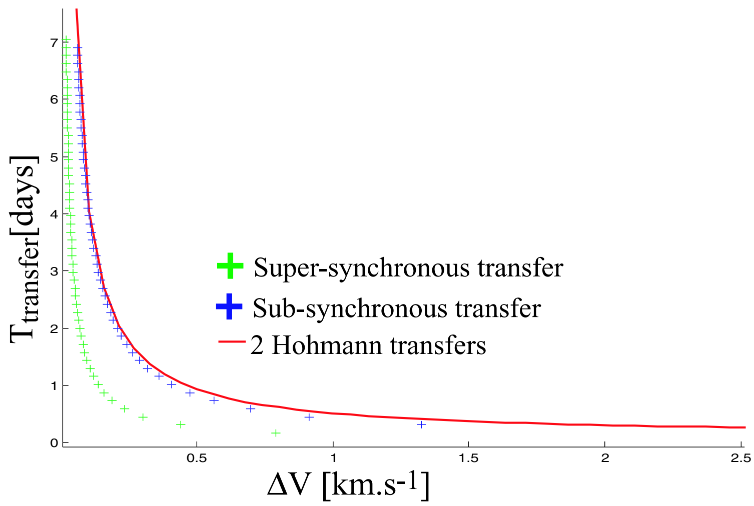

Figure C-4: ΔV and $T_{transfer}$ for different transfers ($e_0 = 0$, $R_{ref} = 11400$ km and θ=270 deg).

continuous. Sub-synchronous and super-synchronous transfers depend on k which is an integer and are thus in discrete quantities. We can see from this example that the transfers we propose are more performant with respect to ΔV and $T_{transfers}$ than any of the double Hohmann transfers. In this example, the sub-synchronous transfers appear to be less performant than the super-synchronous transfers. This is because the value of θ is between 180 deg and 360 deg. In this case, it is easier to decelerate Sp than accelerate it to rendezvous with Sl. When θ is between 0 deg and 180 deg, sub-synchronous transfers appear to be more performant. The worst-case for our transfers will thus correspond to $\theta = 180$ deg. The next section will explain how this property can be used to determine upper limits on ΔV or $T_{transfer}$ for a transfer when θ is not known.

C.4 Constraints on ΔV and $T_{transfer}$

Having the possibility to choose between different transfers to achieve a rendezvous allows an optimization with respect to some constraints. The two cases that we propose to study are the optimization of the time of transfer when a maximum ΔV is set and the minimization of ΔV when a maximum time for the transfer is set. In particular, we want to estimate the maximum time or ΔV for a transfer given some constraints and no a priori knowledge of θ. In this part, the circular case will be the only one considered but the considerations can be generalized to the elliptic case.

C.4.1 Time Constraint

Let's assume a constraint is set on the time of transfer and that we want to minimize the ΔV of the transfers. For each possible angle θ, we compare sub-synchronous and super-synchronous transfers for different values of k to find a transfer that minimizes ΔV. Figure C-5 gives an example of the value of ΔV obtained with respect to θ for an altitude of 5000 km and a maximum time for the transfer of 6 days. We see that the worst case in terms of ΔV corresponds to $\theta = 180$ deg. The reason is that sub-synchronous transfers are efficient for angles lower than θ and super-synchronous transfers for angles greater than 180 deg. But both transfers are the less efficient when θ is equal to 180 deg. For this value of θ, the distance between Sp and Sl is the largest possible. Consequently, if a time constraint is set, an upper limit for the ΔV of the transfer can be found considering the ΔV for $\theta = 180$ deg. When the angle for the transfer is not known but a time constraint for the transfer is set, this particular value of ΔV can be considered as a time penalty for the transfer. The same argument can be used with elliptic orbits. However, the notations being different, the worst-case scenario will correspond to $\nu_0 = 0$.

C.4.2 ΔV Constraint

We now assume that a constraint has been set on the ΔV of the transfer. In this case, we want to reduce the necessary time to achieve the transfer. The time of transfer as a function of the angle of transfer θ has been represented in Figure C-6 for a circular orbit at an altitude of 5000 km when the maximum ΔV is set to $400 m \cdot s^{-1}$. To respect the constraint set by the maximum ΔV, k has to be increased as θ is increased. Then, around $\theta = 170$ deg, super-synchronous orbits are more efficient than sub-synchronous orbits and the number of orbits k is decreased as θ is increased. Parameter k can only

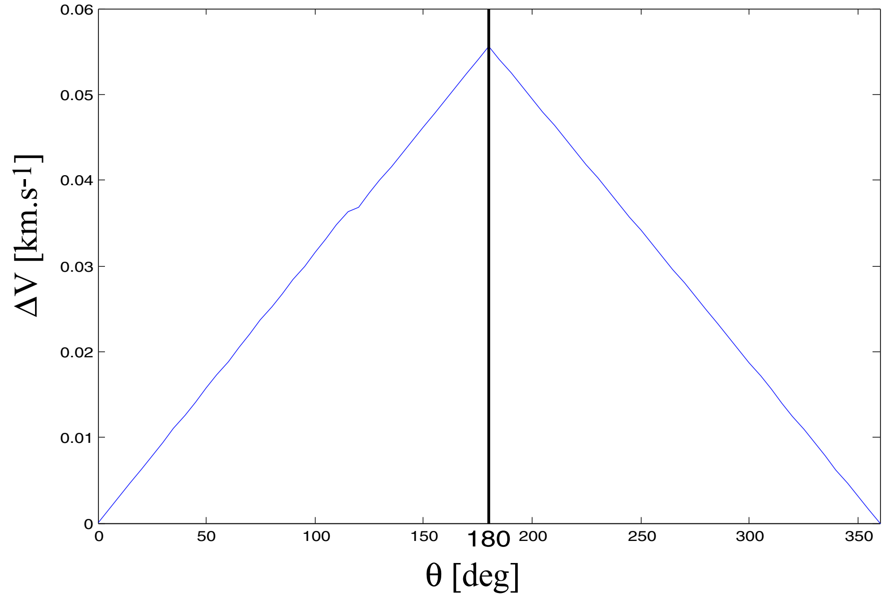

Figure C-5: Optimal ΔV for different values of θ when the reference orbit is circular with an altitude of 5000 km.

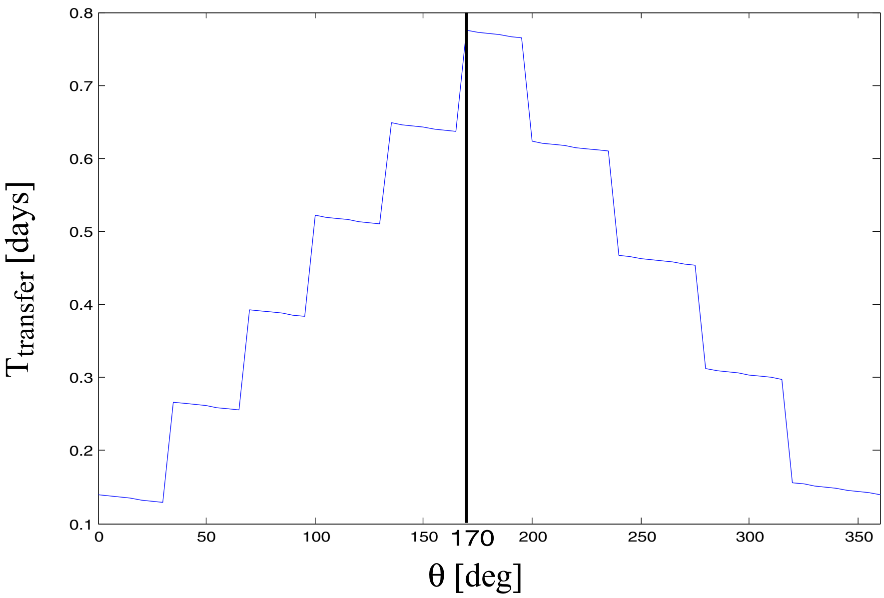

Figure C-6: Optimal $T_{transfer}$ for different values of θ when the reference orbit is circular with an altitude of 5000 km.

be increased in a discrete manner and the final function is almost pyramidal. We see that the angles around θ=180 deg correspond to worst-case scenarios. Once again, to determine a time penalty for the transfer when θ is not known but a maximum ΔV is known, considering the time of transfer to rendezvous with Sl when $\theta = 180$ deg seems a good approximation. For elliptic orbit, the time of transfer for $\nu_0 = 0$ has to be considered.

Bibliography

[AL98] W. S. Adams and T. J. Lang. A Comparison of Satellite Constellations for Continuous Global Coverage. In Jozef C. Van der Ha, editor, *Mission Design and Implementation of Satellite Constellations*, number 1 in Space Technology Proceedings, pages 51–162, Darmstadt, Germany, October 1998. ESA/ESOC, Kluwer Academic Publishers.

[AR87] W.S. Adams and L. Rider. Circular polar constellations providing single or multiple coverage above a specified latitude. *The Journal of the Astronautical Sciences*, 35(2):155–192, April 1987.

[AS02] Y. T. Ahn and D. B. Spencer. Optimal reconfiguration of a formation-flying satellite constellation. 53rd International Astronautical Congress of the International Astronautical Federation, October 2002.

[Bat99] Richard H. Battin. *An Introduction to the Mathematics and Methods of Astrodynamics*. AIAA, Reston, VA, 1999.

[BMW71] Roger R. Bate, Donald D. Mueller, and Jerry E. White. *Fundamentals of Astrodynamics*. Dover Publications, Inc., New York, 1971.

[BS73] F. Black and M. Scholes. The pricing of options and corporate liabilities. In *Journal of Political Economy*, volume 81, pages 637–654, May 1973.

[BT97] Dimitri Bertsimas and John N. Tsitsiklis. *Introduction to linear programming*. Athena Scientific, Belmont, Massachusetts, 1997.

[CBD94] D. Castiel, J. W. Brosisus, and J. E. Draim. Ellipso - coverage optimization using elliptic orbits. In *Proceedings of the 15th AIAA International Communications Satellite Systems Conference*, San Diego, CA, Feb 28–Mar 3 1994. AIAA-1994-1098.

[CGH+92] W. J. Ciesluk, L. M. Gaffney, N. D. Hulkover, L. Klein, P. A. Oliver, M. S. Pavloff, R. A. Pomponi, and W. A. Welch. An evaluation of selected mobile satellite communications systems and their environment. MITRE Corportation, ESA contract 92BOOOOO60, April 1992.

[dLF43] Jean de La Fontaine. *Fables, édition complète.* Les Éditions Variétés, Montréal, Canada, 1943.

[dN90] Richard de Neufville. *Applied Systems Analysis.* McGraw-Hill Publishing Company, New York, 1990.

[dWC02] Olivier L. de Weck and Darren D. Chang. Architecture trade methodology for LEO personal communication systems. In *Proceedings of the 20th AIAA International Communications Satellite Systems Conference*, Montréal, Québec, May 12–15 2002. AIAA-2002-1866.

[ECW99] T. A. Elly, W. A. Crossley, and E. A. Williams. Satellite constellation design for zonal coverage using genetic algorithms. *The Journal of the Astronautical Sciences*, 47:207–228, July 1999.

[Gum96] Cary C. Gumbert. Assessing the Future Growth Potential of Mobile Satellite Systems Using a Cost per Billable Minute Metric. Master's thesis, MIT, Department of Aeronautics and Astronautics, 1996.

[GVH97] C. C. Gumbert, M. D. Violet, and D. E. Hastings. Cost per billable minute metric for comparing satellite systems. *Journal of Spacecraft and Rockets*, 34(6), November-December 1997.

[HS01] Janisa J. Henry and Raymond J. Sedwick. Effects of small altitude variations on coverage statistics near resonant altitudes. Internal Memorandum of the Space Systems Laboratory, MIT, August 2001.

[Hul89] John Hull. *Options, Futures and Other Derivatives Securities.* Prenctice Hall, Englewood Cliffs, NJ, 2nd edition, 1989.

[Jil02] Cyrus Jilla. *A Multiobjective, Multidisciplinary Design Optimization Methodology for the Conceptual Design of Distributed Satellite Systems.* PhD dissertation, MIT, Department of Aeronautics and Astronautics, May 2002.

[Kal03] Konstantinos C. Kalligeros. Framework for the Optimal Design of Corporate Facilities for Contracting Operations. 16.888 Multi-Disciplinary System Design Optimization term paper, MIT, May 2003.

[Kas02] Tatsuki Kashitani. Development and application of an analysis methodology for satellite broadband network architectures. In *Proceedings of the 20th AIAA International Communications Satellite Systems Conference*, Montréal, Québec, May 12–15 2002. AIAA-2002-2019.

[Ku95] Anne Ku. *Modeling Uncertainty in Electricity Capacity Planning.* PhD dissertation, London Business School, UK, 1995.

[Lam01] Elisabeth S. Lamassoure. A Framework to Account for Flexibility in Modeling the Value of On-orbit Servicing for Space Systems. Master's thesis, MIT, Department of Aeronautics and Astronautics, May 2001.

[LWJ00] E. Lutz, M. Werner, and A. Jahn. *Satellite Systems for Personal and Broadband Communications.* Springer, Berlin, Germany, 2000.

[Mer73] R. C. Merton. Theory of rational option pricing. In *Bell Journal of Economics and Management Science*, volume 4, pages 141–183, 1973.

[MSH01] David W. Miller, Raymond J. Sedwick, and Kate Hartman. Evolutionary growth of mission capability using distributed satellite sparse apertures: application to NASA's soil moisture mission (EX-4). Internal Memorandum of the Space Systems Laboratory, MIT, 2001.

[Mye77] Stewart Myers. Determinants of corporate borrowing. *Journal of Financial Economics*, 5(2):6–13, November 1977.

[Por98] Michael E. Porter. *Competitive Strategy, Techniques for analyzing industries and competitors.* The Free Press, New York, 1998.

[Ram02] Natalia Ramirez. Valuing Flexibility in Infrastructure Developments: the Bogota Water Supply Expansion Plan. Master's thesis, MIT, Technology and Policy Program, May 2002.

[Sal02] Joseph H. Saleh. *Weaving time into system architecture :new perspectives on flexibility, spacecraft design lifetime, and on-orbit servicing.* PhD thesis, MIT, Department of Aeronautics and Astronautics, 2002.

[Sha99] Graeme B. Shaw. *The generalized information network analysis methodology for distributed satellite systems.* PhD thesis, MIT, Department of Aeronautics and Astronautics, 1999.

[SHN01] Joseph H. Saleh, Daniel E. Hastings, and Dava J. Newman. Extracting the essence of flexibility in system design. 3rd NASA/DoD workshop on evolvable hardware, July 2001.

[Sin03] Jeremy Singer. Space based tracking radar will have hurdles. *Space News*, April 2003.

[Ste86] Ralph E. Steuer. *Multiple Criteria Optimization: Theory, Computation, and Application.* John Wiley & Sons, New York, 1986.

[Tri96] L. Trigeorgis. *Real Options - Managerial Flexibility and Strategy in Ressource Allocation.* MIT Press, Cambridge, MA, 1996.

[TSN+00] H. Takeuchi, M. Sugimoto, H. Nakamura, T. Yutani, M. Ida, S. Jitsukawa, T. Kondo, S. Matsuda, H. Matsui, T. E. Shannon, R. A. Jameson, F. W. Wiffen, J. Rathke, C. Piaszczyk, S. Zinkle, A. Möslang, K. Ehrlich, F. Cozzani, H. Klein, J-M. Lagniel, R. Ferdinand, E. Daum, M. Martone, I. S. Ciattaglia, and V. Chernov. Staged Deployment of the International Fusion Materials Irradiation Facility (IFMIF). In *Proceedings of the 18th International Atomic Energy Agency Fusion Energy Conference*, Sorrento, Italy, October 4–10 2000. IAEA-CN-77-FTP2/03.

[Vio95] Michael D . Violet. The development and application of a cost per minute metric for the evaluation of mobile satellite systems in a limited-growth voice communications market. Master's thesis, MIT, Department of Aeronautics and Astronautics, 1995.

[Wal77] J. G. Walker. Continuous whole-earth coverage by circular-orbit satellite patterns. Technical Report 77044, Royal Aircraft Establishment, Farnborough, UK, 1977.

[WL99] James R. Wertz and Wiley J. Larson. *Space Mission Analysis and Design.* Microcosm Press, El Segundo, California, third edition, 1999.