Preliminary design of a crewed Mars flyby Solar Electric Propulsion mission

Stijn De Smet-November 25, 2014





Preliminary design of a crewed Mars flyby Solar Electric Propulsion mission

MSC THESIS

For the degree of Master of Science in Aerospace Engineering at Delft University of Technology

Stijn De Smet - 4047141

Supervisors: ir. Ron Noomen - Delft University of Technology dr. Jeffrey S. Parker - University of Colorado

November 25, 2014

Faculty of Aerospace Engineering \cdot Delft University of Technology



Copyright © Astrodynamics and Space Missions All rights reserved



The work in this thesis was supported by the Colorado Center for Astrodynamics Research. Their cooperation is hereby gratefully acknowledged.

"I get by with a little help from my friends." — The Beatles

To my parents.

Acknowledgments

There are many people I would like to thank who helped me during the writing of this thesis. First of all, I would like to thank my supervisor Ron Noomen who convinced me to follow the astrodynamics and space missions master track through his inspirational lectures on astrodynamics. Furthermore, I would also like to thank him for the supervision during this thesis work, even though I was over 5000 miles away. I also owe a large thank you to Jeffrey Parker from the University of Colorado for letting me do my internship and thesis work in his research group and for repeatedly making time for me in his full schedule. I would also like to thank Jon Herman for helping me with the applications at the University of Colorado and for being my mentor while I was taking my first baby steps in the academic world. Besides thanking Ron Noomen, Jeffrey Parker and Jon Herman for their help throughout this thesis work, I would also like to thank them for their help writing and presenting my first two conference papers. I also want to thank the people in the low-thrust research group at the University of Colorado; Jeffrey Parker, Jon Herman, Nathan Parrish, Jonathan Aziz and Collin Bezrouk for the interesting weekly discussions. I also want to thank the numerous other people I met at CU and TU Delft for the valuable input and advice they had to this thesis. Thank you!

I also want to thank the grants provided by the TU Delft allowing me to do my internship at the University of Colorado, which ultimately led to me writing my thesis there. These are the Van der Maas funds and the Board of Directors fund.

Moving on to more personal things, I want to say thank you to all the amazing people I have met throughout my school going years. First and foremost, I would like to thank my close friends "de boys" Nick, Gertjan, Bruno and Sasja for all the great times we have had. Also, Chiro Sikambers, for 16 fantastic years of friendship and everlasting memories, the Belgian "gang" in Delft for the good times and fun car rides, the CU Hiking Club for the incredible trips I have been on and led and for the friendships formed in the mountains, the other CU Internationals for being such a good group of the most multi-cultural friends imaginable, and finally all other great people I have met during my years in Belgium, the Netherlands and the US. Thank you all for the unforgettable experiences.

Finally, I would like to thank my family, especially my mother and father for their unconditional support and love throughout every step in my life and for always being there for me. I could not have imagined a better mom for having a "coffee chat" or a better dad to go to concerts with and to ramble on for hours and hours about music. I love you!

Delft, University of Technology November 25, 2014 Stijn De Smet

iv

Summary

The goal of this thesis research has been the improvement of a validated, robust low-thrust optimization tool for interplanetary trajectory design, written during the author's internship. This improved tool has then been applied to design crewed Martian flyby missions. Such a mission is a crucial step within the flexible path scenario for human and robotic space exploration, formulated "in support of the Exploration Beyond LEO committee of the Review of US Human Space Flight Plans Committee commissioned by President Obama" [Korsmeyer et al., 2010]. This mission would provide critical experience in preparation for a Mars landing without the actual risk of the landing itself. Several technologies could be validated on this kind of mission such as new countermeasures for radiation shielding, better regenerative life support systems, further investigation of the effect of deep-space isolation on the human psyche, etc.

This thesis was the follow-up of previously conducted research during the author's internship for which a basic Sims-Flanagan based low-thrust optimization tool was written. The first goal of this thesis was the improvement of this existing code. Therefore, the code has been extensively profiled and analyzed to identify and remove bottlenecks. Furthermore, advantage has been taken from the sparsity of the Jacobian to further decrease the run time. Depending on the scenario, gains in run time of up to a factor 10 have been observed. Additionally, different representations of the Sims-Flanagan transcription have been investigated. It was found that compared to a throttled representation, the classical thrust representation is slower, but more robust.

The second goal of this thesis was the addition of time-optimization capabilities. Therefore, methods to analytically derive the Jacobian elements with respect to time have been established. However, numerical difficulties arose from this method. To circumvent this problem, a forward finite-difference method has been written. Furthermore, several representations to couple ephemeris to time have been set up and compared. These time-optimization capabilities have been tested on Earth-Mars-Earth flyby missions launching in 2018. Based on previous results, those time-optimization capabilities could be validated for two different objective functions: minimized launch mass and maximized final mass. Finally, the latter objective function has been selected for this research. The third goal of this thesis was the automation of the addition and optimization of additional legs. Therefore, automation algorithms have been set up throughout the program. These and the previously established time-optimization capabilities have been tested on Earth-Venus-Mars-Earth flyby missions. During these tests, issues with local optima arose. Therefore, a multi-start method has been implemented and tested. This multi-start method circumvents the majority of those local optima issues.

Using the added and validated capabilities, several launch windows for crewed Martian flyby missions have been identified for different SEP power levels, different launcher configurations and different payload masses in 2018, 2019 and 2021. In addition, an opportunity for a crewed Venus and Martian flyby mission has been identified launching in 2021.

Table of Contents

	List	of Acronyms	xiii
	List	of Symbols	xiv
1	Intro	oduction	1
I	Mis	sion motivation and heritage	3
2	Miss	sion motivation	5
3	Miss	sion heritage and SEP systems	7
	3-1	Previous low-thrust missions	7
		3-1-1 Deep Space 1	7
		3-1-2 Dawn	8
		3-1-3 Jupiter Icy Moons Orbiter	8
	3-2	Solar Electric Propulsion mechanisms	9
		3-2-1 Electrothermal	10
		3-2-2 Electrostatic	10
		3-2-3 Electromagnetic	10
	3-3	Characteristics	10
	3-4	Realistic values for Solar Electric Propulsion	12
11	Pre	evious work performed during the author's internship	13
4	Intro	oduction to previous research	15

5	Sim	s-Flana	gan method and program structure	17
	5-1	Sims-F	lanagan method	17
	5-2	Structi	ure of the state vector	18
	5-3	Structi	ure of the program	19
6	Pro	pagatio	n module	21
	6-1	Structi	ure of the propagation	21
	6-2	State ⁻	Transition Matrix	22
	6-3	Propag	gation between manoeuvres	23
		6-3-1	DOPRI8 numerical integrator	23
		6-3-2	Two-body force model derivative function	25
	6-4	Applica	ation of the manoeuvre	26
7	Con	straints		27
-	7-1	Cost fi	unction	2
	7-2	Genera	ll constraints	2'
		7-2-1	Match point constraints	2'
		7-2-2	Thrust constraints	28
	7-3	Leg-sp	ecific constraints	28
		7-3-1	Departure-node constraints	29
		7-3-2	Flyby-node constraints	29
		7-3-3	Return-node constraint	30
8	Jaco	obian		3
	8-1	Cost fi	unction	3
	8-2	Mass r	natch point $F_{\Delta M}$	3.
	8-3	State r	match point constraints $F_{\Delta x}$, $F_{\Delta y}$, $F_{\Delta z}$, $F_{\Delta \dot{x}}$, $F_{\Delta \dot{y}}$, $F_{\Delta \dot{z}}$	32
		8-3-1	Derivatives with respect to the initial node's coordinates, velocities and hyperbolic excess velocities	32
		8-3-2	Derivatives with respect to the final node's coordinates, velocities and hy-	9
		0 2 2	Derivatives with respect to the forward velocity components	ം ച
		0-3-3	Derivatives with respect to the forward velocity components	ം ച
	0.4	0-3-4 Thurrent	Derivatives with respect to the backward velocity components	ა ი
	0-4 8 5			31
	0-0	reg-sh		0
9	Sun	nmary p	revious results	3
	9-1	Set-up	of the optimization problem and goals	3
	9-2	Operat	cional considerations	38
	9-3	Missio	n parameters and launcher configurations	39
	9-4	Initial	guess generation	40
	9-5	Results 9-5-1	s	4: 4
		9-5-2	Farth-Mars-Farth flyby mission in 2019	4'
		9-5-3	Farth-Mars-Farth flyby mission in 2021	4
	9_6	Conclu	sion	
	50	Conciu		т,

10	Thes	sis rese	arch goals	51
	10-1	Researc	ch goal 1	51
	10-2	Researc	ch goal 2	52
	10-3	Researc	ch goal 3	52
111	Re	search	goal 1: Improvements to previous work	53
11	Prof	iling		55
	11-1	One-leg	g scenario	55
		11-1-1	Nominal one-leg scenario	55
		11-1-2	Adaptation 1: Array allocation moved up in function hierarchy	58
		11-1-3	Adaptation 2: Array allocation moved out of loop	58
		11-1-4	Adaptation 3: Avoid usage of pow function	58
		11-1-5	Adaptation 4: Merge multiple arrays into memory pool	59
		11-1-6	Adaptation 5: Move step size initialization	59
		11-1-7	Time comparison	60
	11-2	Two-le	g scenario	61
		11-2-1	Nominal two-leg scenario	61
		11-2-2	Time comparison	61
	11-3	Grid se	arches	63
	11-4	Conclu	sion	64
12	Thro	ottled v	ersus thrust representation	65
	12-1	Thrott	led representation	65
	12-2	Compa	rison of the sparsity of the Jacobian	67
	12-3	Jacobia	an of the throttled representation of the Sims-Flanagan transcription	68
		12-3-1	Derivatives of the match point constraints with respect to the initial and final masses, coordinates, velocities and hyperbolic excess velocities, and the mass of the SEP system	68
		12-3-2	Derivatives of the match point constraints with respect to the throttle parameters and the specific impulses	70
		12-3-3	Derivatives of the throttle constraints with respect to the throttle parameters	72
	12-4	Compa	rison of throttled and thrust representations	73
		12-4-1	Nominal single-leg scenario	73
		12-4-2	Convergence issues throttled representation	73
		12-4-3	Validation and profiling of the nominal scenario	74
		12-4-4	Sparsity	74
		12-4-5	Grid search around nominal scenario	75
	12-5	Conclu	sion	76

IV	Research goal 2: Control node's time optimization	77
13	Coupling between time and ephemeris	79
	13-1 Option 1: Ephemeris constraints	79
	13-1-1 Analytical derivatives of ephemeris constraints	79
	13-1-2 Analytical derivatives with respect to time	80
	13-1-3 Numerical difficulties ephemeris constraints	81
	13-2 Option 2: Remove coordinates and velocities from the state vector	82
	13-3 Conclusion	82
14	Sparsity pattern implementation	83
15	Two-leg Earth-Mars-Earth test case	85
	15-1 Representation 1: Minimized launch mass	85
	15-1-1 Validation	86
	15-1-2 Two-dimensional grid search in departure date and TOF	90
	15-2 Representation 2: Maximized final mass	91
	15-2-1 Validation	91
	15-2-2 Two-dimensional grid search in departure date and TOF	91
	15-3 Conclusion	92
v	Research goal 3: Addition of dynamic legs	93
16	Structure of the automated leg addition	95
17	Three-leg Earth-Venus-Mars-Earth test case	97
	17-1 Initial trajectory issues	97
	17-2 Local optima issues	100
	17-3 Multi-start method	105
	17-3-1 Working principles	105
	17-3-2 Analysis	106
	17-3-3 Conclusion	112
VI	Validation, results, conclusions and recommendations	113
18	Validation	115
	18-1 Basic version code developed during author's internship	115
	18-2 Time optimization capabilities	116
	18-3 Additional legs/EVME	116

19	Resu	ılts		117
	19-1	Earth-l	Mars-Earth flyby mission in 2018	117
		19-1-1	SLS 4RL launch configuration	117
		19-1-2	SLS 1RL launch configuration	129
	19-2	Earth-l	Mars-Earth flyby mission in 2019	131
	19-3	Earth-l	Mars-Earth flyby mission in 2021	134
	19-4	Earth-	Venus-Mars-Earth flyby mission in 2021	136
	19-5	Conclu	sion	138
20	Con	clusions		141
	20-1	Resear	- ch goal 1	141
	20-2	Resear	ch goal 2	141
	20-3	Resear	 ch goal 3	142
	20-4	Launch	windows	142
01	Dee			112
21	21_1	Earth (dations	143 1/3
	21 1	Elvby a	altitude limit	1/3
	21-2	Higher	fidelity of the trajectory design	140
	21-5	Global	ontimization	144
	21-4	Progra		144
	21-6	Ontimi		144
		optim		
. /1			•	145
VI	IA	ppena	ices	143
Α	TNC	C transf	formation	147
в	TNC	trans	formation matrix	149
-				1.0
С	Jaco	bian		157
	C-1	Cost fi		157
	C-2	Mass r	match point $F_{\Delta M}$	157
	C-3	State r	match point constraints $F_{\Delta x}$, $F_{\Delta y}$, $F_{\Delta z}$, $F_{\Delta \dot{x}}$, $F_{\Delta \dot{y}}$, $F_{\Delta \dot{z}}$	158
		C-3-1	Derivatives with respect to the initial node's coordinates, velocities and hyperbolic excess velocities	158
		C-3-2	Derivatives with respect to the final node's coordinates, velocities and hyperbolic excess velocities	161
		C-3-3	Derivatives with respect to the forward velocity components $\ . \ . \ .$.	163
		C-3-4	Derivatives with respect to the backward velocity components $\ . \ . \ .$.	165
	C-4	Thrust	constraints	166
		C-4-1	Constant power	166
		C-4-2	Power dependent on heliocentric distance	170

	C-5	Leg-specific constraints	178
		C-5-1 Departure-node constraints	178
		C-5-2 Flyby-node constraints	178
		C-5-3 Return-node constraint	180
D	Exte	ended TNC transformation matrix	181
	D-1	Forward propagation	181
	D-2	Backward propagation	183
Е	Exte	ended TNC matrix	185
	E-1	Forward propagation	185
	E-2	Backward propagation	186
F	Cha	nges to the Jacobian for the throttled representation	187
G	Ana	lytical derivatives of ephemeris constraints with respect to time	191
н	Glob	oal Trajectory Optimization Competition	195
	H-1	Problem statement	195
	H-2	Strategy	196
	H-3	Validation of single-leg approximation tools	196
	H-4	Validation of asteroid tours	197
	H-5	Building entire probe solutions	199
	H-6	Conclusion	199
I	Eart	h-Asteroid-Earth rendez-vous mission	201
VI		Bibliography and index	205
			205
	Bibl	iography	205

Glossary

List of Acronyms

CATO	Computer Algorithm for Trajectory Optimization	7
CEV	Crew Exploration Vehicle	5
СОМ	Center of Mass	147
DSM	Deep Space Manoeuvre	
FEEP	Field Emission Electric Propulsion	10
бтос	Global Trajectory Optimization Competition	
JIMO	Jupiter Icy Moons Orbiter	8
RIT	Radiofrequency Ion Thruster	
SEP	Solar Electric Propulsion	
SEPTOP	Solar Electric Propulsion Trajectory Optimization Program	7
SLS	Space Launch System	
SQP	Sequential Quadratic Programming	19
STM	State Transition Matrix	
TNC	Tangential, Normal and Cross-track	147

List of Symbols

Roman Symbols				
Symbol	Description	\mathbf{Units}		
a	semi-major axis	m		
C_3	launch energy	m^2/s^2		
e	eccentricity	-		
F	derivative function	-		
g_0	gravitational acceleration at the Earth's surface	m/s^2		
H	specific angular momentum	m^2/s		
h_{fly}	altitude of the flyby	m		
h_n	step size	S		
I_{sp}	specific impulse	\mathbf{S}		
i	inclination	rad		
k_{P_0}	power to mass ratio of the SEP system	$\mathrm{kg/W}$		
L	mean longitude	rad		
M	mean anomaly	rad		
M_0	initial mass before manoeuvre	kg		
M_f	final mass after manoeuvre	kg		
M_{P_0}	mass of propulsion unit	kg		
\dot{m}	mass flow	$\rm kg/s$		
N	half of the number of segments in one leg	_		
P_0	electrical power supplied by the power source at 1 AU	W		
$P_{\rm jet}$	power of exhaust jet	W		
\dot{R}	heliocentric distance	m		
T	thrust	Ν		
TOF_a	time of flight leg a	S		
u	throttle unit vector element	-		
V_{j}	exhaust velocity of propellant	m/s		
$\dot{V_{\infty}}$	hyperbolic excess velocity	m/s		
W	Lambert's function	-		
x(t)	state vector	-		

Greek Symbols

Symbol	Description	Units
α	flyby deflection angle	rad
Δt	duration of a segment	\mathbf{S}
ΔV	change in velocity	m/s
ϵ	fudge factor	-
$\eta_{ m jet}$	power conversion efficiency	-
θ	true anomaly	rad
κ	general TNC coordinate T, N or C	-
μ	standard gravitational parameter	m^3/s^2
П	longitude of the perihelion	rad
ϱ	general Cartesian coordinate x, y, or z	-
Φ	state transition matrix	-

Ω	longitude of the ascending node	rad
ω	argument of perihelion	rad

Superscripts

Symbol	Description	Units
	first derivative	-
	second derivative	-
Ι	inbound leg of a flyby	-
II	outbound leg of a flyby	-
_	state before a manoeuvre	-
+	state after a manoeuvre	-

Subscripts

\mathbf{Symbol}	Description	\mathbf{Units}
0	initial control node	-
f	final control node	-

Chapter 1

Introduction

Mankind has always been fascinated with our planetary neighbor Mars. A long history of speculation about life on Mars exists. When the planet was mapped through telescopes, a series of what looked like channels were observed [Schiaparelli, 1895]. It was thought that these were dug out by an intelligent life form. Furthermore, before the first flybys of the red planet in the sixties, scientists were convinced that there was vegetation on Mars. This belief was based on Mars' spectrum in the near infrared, which appeared to show signs of chlorophyll [liveScience, 2012]. This belief in alien life forms on Mars was further intensified by science fiction. Remember for instance the famous radio play 'The War of the Worlds' by Orson Welles in 1937. However, after almost 60 years of satellite and robotic exploration of Mars, the question of life on Mars and many others remain unanswered.

Another source of fascination for Mars lies within the very nature of mankind itself. Since the dawn of mankind, humans have always had the urge to explore, which has made us leave Africa's cradle and sail the oceans. This resulted in human presence on each continent, even Antarctica. Through the ages, technology enabled us to go faster and further. We developed steam boats, cars, planes and even spacecraft. The latter enabled us to set foot on the Moon in the late sixties. The logical big next step seems to be the human exploration of Mars.

To achieve this goal, some trajectory design problems must be overcome to which the author would like to contribute. During the author's thesis work, the trajectory for a crewed Solar Electric Propulsion (SEP) flyby mission to Mars will be designed. Such a mission would provide critical experience in the preparation for a Mars landing without the actual risk of the landing itself. Several technologies could be validated on this kind of mission such as new countermeasures for radiation shielding, better regenerative life support systems, further investigation of the effect of deep-space isolation on the human psyche, etc. The motivation to fly a crewed Martian flyby mission will be elaborated further upon in Chapter 2.

No mission can be designed without "standing on the shoulder of giants". Therefore, Chapter 3 will investigate previously flown or designed low-thrust missions. This chapter will also

investigate different forms of SEP systems and will give an overview of important SEP-related concepts along with an overview of realistic capabilities of SEP.

As this thesis work is a follow-up study, the previous research will be introduced in Chapter 4. The utilized method, the Sims-Flanagan low-thrust transcription, will be described in Chapter 5. The developed tool has three main modules: the propagation module, the constraint module and the Jacobian module which will be discussed in Chapters 6, 7 and 8 respectively. Readers familiar with the Sims-Flanagan method can skip Chapter 5 up to 8. This will be followed by a discussion on the main results from this previous work in Chapter 9. From this discussion, the main thesis goals can be derived in Chapter 10.

The first research goal that will be identified is the improvement of the previously written code. Therefore, in Chapter 11, bottlenecks in the existing code will be identified and removed to improve the run time of the code. In Chapter 12, a comparison between two different representations of the Sims-Flanagan transcription, throttled versus thrust representation, will be made and one representation will be selected.

The second identified research goal is the addition of time-epoch optimization capabilities. Crucial for time of flight optimization is the proper connection between time and ephemeris. Therefore, in Chapter 13, two options to couple time and ephemeris will be set up and compared. After this chapter, the structure of the state vector and the constraint vector, and hence of the Jacobian, will be fixed. Hence, the sparsity of the Jacobian can then be implemented to further reduce the run time of the code. This will be done in Chapter 14. Finally, the added capabilities will be tested and validated in Chapter 15 on an Earth-Mars-Earth flyby mission. This will be done for two different objective functions, after which a comparison between them will be made and one objective function will be selected.

The third identified research goal is the automation of the addition of extra legs. Therefore, several different types of control nodes will be set up in Chapter 16. The entire problem structure including the state vector set-up and propagation, constraints and Jacobian set-up and calculation will then be automated based on the type of control nodes and the number of legs in an input file. This automation will be tested on an Earth-Venus-Mars-Earth scenario in Chapter 17 combined with an additional test case for the time-optimization capabilities of the tool. In this chapter, problems due to convergence on local optima will arise. Therefore, it also explains the implementation of a multi-start method to overcome these problems.

Throughout the performed research, continuous validation effort has been performed. Chapter 18 will summarize these efforts. Then, Chapter 19 will show how the previously added and validated capabilities have been used to generate the final results. Here, launch windows identified in Chapter 9 are elaborated further upon by making the total time of flight variable. Furthermore, a scenario with a Venus and Martian flyby will be discussed. Chapter 20 will summarize the conclusions. Finally, Chapter 21 will identify future work and will list recommendations on how to realize this suggested work.

Besides applying and testing the written code on crewed Martian flyby missions, the tool has also been extensively used during the 7th Global Trajectory Optimization Competition [Casalino and Colasurdo, 2014]. Appendix H introduces the problem, explains how it provided an additional test bed for the tool and how the tool has been used to find a solution.

Part I

Mission motivation and heritage

Chapter 2

Mission motivation

In 2009, a flexible path scenario for human and robotic space exploration was formulated "in support of the Exploration Beyond LEO committee of the Review of US Human Space Flight Plans Committee commissioned by President Obama" [Korsmeyer et al., 2010]. The goal of this flexible path is to gradually increase experience in human exploration for missions ranging from weeks to several years, which ultimately would enable a crewed Mars surface mission [Korsmeyer et al., 2010]. The committee has identified a nominal sequence for this flexible path exploration composing of 8 steps, listed in Korsmeyer et al. [2010]. Note that all of these missions shown below are assumed to be crewed, except as noted.

- 1. An uncrewed circumlunar mission utilizing only the Orion Crew Exploration Vehicle (CEV) and an in-space propulsion stage.
- 2. A circumlunar mission utilizing only the Orion CEV and an in-space propulsion stage.
- 3. A mission to the Earth-Moon L1 Lagrange point, to demonstrate the ability to perform crewed operations and emplace and service assets stationed there.
- 4. A mission to the Sun-Earth L2 Lagrange point, to demonstrate the ability to emplace and service scientific assets stationed there.
- 5. A mission to the Sun-Earth L1 Lagrange point for additional deep-space exploration preparation (i.e. experience with the interplanetary radiation environment outside the Earth's magnetosphere, which is not the case at the Sun-Earth L2 point) and to emplace and service scientific assets stationed there.
- 6. Several missions to rendezvous with NEOs of different composition (e.g. metallic, carbonaceous chondrites, etc.) for exploration, scientific instrument emplacement and the return of samples.
- 7. A free return mission to Mars, with a flyby, but no major manoeuvring in the vicinity of Mars.

8. A mission to the moons of Mars (Phobos and/or Deimos) combined with the return of samples from a robotic mission to the Martian surface.

The main step of interest for this thesis work is step 7, a Martian flyby mission. Such a Martian flyby mission would be a reduced-risk crewed Mars mission. This mission would provide critical experience in preparation for a Mars landing without the actual risk of the landing itself [Korsmeyer et al., 2010]. Several technologies could be validated on this kind of mission such as new countermeasures for radiation shielding, better regenerative life support systems, further investigation of the effect of deep-space isolation on the human psyche, etc.

Besides governmental interest for a visit to Mars, a number of private companies have similar plans. Based on the promising discussion of free-return trajectories to Mars discussed in Patel et al. [1998] and based on flexible path step 7 describing a free-return flyby mission to Mars [Korsmeyer et al., 2010], Inspiration Mars wants to perform a crewed Mars flyby mission launching as soon as 2018 [Tito et al., 2013]. At the moment of writing, Inspiration Mars only considers the use of chemical propulsion. Taking into account that this mission relies on a very specific planetary alignment to achieve a feasible final payload mass, the next launch opportunity would only be in 2031 [Tito et al., 2013].

Meanwhile, major improvements in Solar Electric Propulsion (SEP) are expected by the end of the decade. NASA even states that the top technical challenge for in-space propulsion is the "development of high-power electric propulsion system technologies to enable high ΔV missions with heavy payloads" [Meyer et al., 2012]. This improvement in propulsion systems could be used to aid the planetary exploration of Mars.

This research will therefore investigate how solar electric propulsion could be used to facilitate crewed missions much like Inspiration Mars. This research will as such focus on Mars flyby missions and will contribute to the trajectory design of Inspiration Mars or a similar mission. Such a mission can be used as a test bed or a technology trigger to develop and test several different new technologies that will later enable humankind to land on Mars, without having the risk of the actual Mars landing.

Chapter 3

Mission heritage and SEP systems

In this chapter, previous low-thrust missions will be discussed. This discussion will be followed by an explanation of the working principles of solar electric propulsion. First of all, the different types of low-thrust mechanisms and hardware utilized to produce the low thrust will be discussed. Afterwards, a brief discussion on important concepts and current capabilities of low-thrust systems will be given.

3-1 Previous low-thrust missions

In this section, previous low-thrust missions will be discussed. Each mission will be shortly described in terms of trajectory, the method of trajectory design (if available), SEP capabilities for this mission, etc.

3-1-1 Deep Space 1

The Deep Space 1 mission was the first mission to use ion propulsion as a primary form of propulsion. During this mission, ion propulsion and 11 other technologies were validated [Rayman and Williams, 2002]. It was planned for launch in July or August 1998, after which it would perform an asteroid flyby on McAuliffe and perform a comet flyby on West-Kohoute-Ikemura where it would arrive after 2 years. This trajectory, visualized in Figure 3-1, was designed with the **Solar Electric Propulsion Trajectory Optimization Program (SEPTOP)**, which optimizes the thrust profile, the magnitude and direction of the manoeuvres, as a function of time. This initial estimate obtained by SEPTOP was then later refined by the Computer Algorithm for Trajectory Optimization (CATO) [Rayman et al., 1999]. The actual mission was slightly different: it was launched in October 1998, and flew by asteroid 9969 Braille and the Borelly comet [Rayman and Williams, 2002].

An interesting feature of this mission that is also of importance for the crewed Mars flyby mission is the notion that the power to the spacecraft was not constant due to its varying



Figure 3-1: Planned trajectory of Deep Space 1, actual dates differ [Rayman et al., 1999].

heliocentric distance. This in combination with the required thrust profile for the mission requires the engine to be throttled. Since the power, thrust and specific impulse are directly related for a SEP system, as will be explained in Section 3-3, this power throttling affects the thrust and the specific impulse of the engine. The engine selected for this mission could be throttled between 525 and 2500 W for which it delivered 19 and 92 mN of thrust with a specific impulse of 1900 and 3100 s respectively [Rayman and Williams, 2002]. For the mission baseline mass of 480 kg, this thrust level translates to an acceleration of between $4.0 \cdot 10^{-5}$ and $1.9 \cdot 10^{-4}$ m/s².

3-1-2 Dawn

Another groundbreaking mission regarding low-thrust propulsion is the Dawn mission. This mission visits the two largest asteroids: Vesta and Ceres. The mission trajectory can be seen in Figure 3-2. The trajectory was designed using the **Mystic** tool [Rayman et al., 2006].

Just like the Deep Space 1 mission, the engines were throttled. At the maximum power input of 2.6 kW, a thrust of 92 mN is developed, while for 0.5 kW, the thrust is 19 mN. The specific impulse ranges from 3200 to 1900 s [Rayman et al., 2006]. These numbers are quite similar to those of Deep Space 1, as the ion engines for the Dawn mission were inherited from the NSTAR engines validated during the Deep Space 1 mission. For the spacecraft mass of 1240 kg, this translates to an acceleration of between $1.5 \cdot 10^{-5}$ and $7.4 \cdot 10^{-5}$ m/s².

3-1-3 Jupiter Icy Moons Orbiter

The Jupiter Icy Moons Orbiter (JIMO) mission is a canceled NASA mission, which was designed to orbit Callisto, Ganymede and Europa. The immense ΔV budget required for



Figure 3-2: Planned trajectory of Dawn, actual dates differ [Rayman et al., 2006].

this mission, which is in the order of 30 km/s, makes the use of a highly efficient propulsion system necessary [Whiffen, 2003]. This mission relies on an engine which we are not yet able to design. It had to deliver 3.353 N using 148 kW of power with a specific impulse of 9000 s. Using the mass estimate of 9 tons, the acceleration would be around $3.8 \cdot 10^{-4}$ m/s². In Whiffen [2003], the trajectory design process, which has been done using the **Mystic** tool, has been explained along with the difficulties that were encountered. One of these difficulties was that multiple capture and escape trajectories around Jupiter's moons had to be included in order to enter and leave low-altitude science orbits around those moons [Sims, 2006]. This means that they had to optimize an interplanetary trajectory with a planetcentered spiral trajectory. As such, there are large time and distance scaling differences, which create difficulties for optimization. Furthermore, a multi-body force model had to be included adding further optimization difficulty. However, the inclusion of multi-body force models also has advantages: it can show opportunities to increase performance compared to nominal two-body force models. To minimize the problems with optimization of this complex trajectory, it was optimized in parts.

3-2 Solar Electric Propulsion mechanisms

For classical chemical propulsion systems, the propellant is at the same time the power source through the chemical reaction and the medium that is being expelled [Wakker, 2010]. However, for electric propulsion, there is a separate power source with which the exhaust medium is being accelerated [Wakker, 2010]. How this medium is accelerated depends on the type of electric propulsion. In this section, the focus will be on Solar Electric Propulsion (SEP), as other forms of power such as nuclear are currently inapplicable to space applications. Therefore, several technologies listed in Table 3-1 require power levels that are not yet achievable on spacecraft.

3-2-1 Electrothermal

Electrothermal thrusters increase the enthalpy of the expellant, which is then converted into kinetic energy using a nozzle [Fortescue et al., 2003]. Simply put, propellant is heated electrically after which the hot gas can expand and accelerate through a nozzle [Wakker, 2010]. There are two main types of electrothermal propulsive systems: resistojets and arcjets [Wakker, 2010]. In resistojets, the propellant is heated through electrically heating high-resistance metal parts. Arcjets heat the propellant directly by passing it through an electric arc discharge [Wakker, 2010]. The commonly used propellants for electrothermal thrusters are hydrogen, nitrogen, ammonia and hydrazine [Fortescue et al., 2003].

3-2-2 Electrostatic

In electrostatic thrusters, the positively charged propellant particles (ions) are accelerated in a static electric field between the ion source and accelerating electrode(s) [Fortescue et al., 2003, Wakker, 2010]. After acceleration of the ions, the ions are neutralized using electrons in the exhaust beam. This is necessary to avoid a negative charge build-up on the spacecraft [Fortescue et al., 2003]. The ions can be created by electron bombardment or by passing atoms through an extremely thin slit in an emitter in which an electrical field causes the propellant atoms to become unstable and ionize [Wakker, 2010]. The most well-known electrostatic thrusters are the Kaufman thruster, the Radiofrequency Ion Thruster (RIT) and the Field Emission Electric Propulsion (FEEP) thruster. Common propellants are inert gasses such as argon and xenon [Fortescue et al., 2003].

3-2-3 Electromagnetic

Electromagnetic thrusters use highly ionized propellant plasma [Wakker, 2010]. Crossed electric and magnetic fields induce a Lorentz force on this plasma, causing it to accelerate. Again, after the acceleration, the ions are neutralized to avoid a charge build-up on the spacecraft. Hall thrusters use the Hall effect to generate this electrostatic field. Magnetoplasmadynamic thrusters use an electric arch discharge, pulsed plasma thrusters utilize the interaction between an electric arc current and a self-induced magnetic field. Pulsed inductive thrusters use the interaction between a current and a magnetic field from a coil current [Wakker, 2010].

3-3 Characteristics

First of all, some major characteristics of low-thrust systems will be explained.

Thrust Thrust is the force that propels the spacecraft. The thrust T in a vacuum can be calculated from

$$T = \dot{m}V_j \tag{3-1}$$

where \dot{m} and V_j are the mass flow and the exhaust velocity of the propellant respectively.

Exhaust velocity and specific impulse As can be seen from Equation 3-1, the exhaust velocity has a linear effect on the thrust. A parameter that has a similar effect is called the specific impulse I_{sp} .

$$I_{sp} = \frac{\int T dt}{\int \dot{m} dt} = \frac{V_j}{g_0} \tag{3-2}$$

where g_0 is the gravitational acceleration at the Earth's surface.

Quite often, the exhaust velocity and specific impulse are used as parameters indicating the mass efficiency of the rocket as can be seen by rewriting the Tsiolkovsky equation:

$$\frac{M_f}{M_0} = e^{-\frac{\Delta V}{V_j}} = e^{-\frac{\Delta V}{g_0 I_{sp}}}$$
(3-3)

with M_f and M_0 the final mass after and the initial mass before the manoeuvre respectively. From this equation, one can see that for a given ΔV , a higher exhaust velocity or specific impulse results in a larger M_f/M_0 fraction. Therefore, the greater the exhaust velocity or specific impulse, the less propellant is required for a certain manoeuvre.

Power-related characteristics For chemical propulsion, the exhaust velocity is limited and dependent on the type of chemical reaction between the propellants. For electric propulsion, there is no theoretical upper limit on the exhaust velocity. However, in practice, the achievable exhaust velocity is limited due to a limit in available power, as can be seen from the following equation from Wakker [2010], which has been established by combining the law of conservation of energy and the thrust equation in Equation 3-1:

$$P_{\rm jet} = \eta_{\rm jet} P_0 = \frac{1}{2} \dot{m} V_j^2 = \frac{1}{2} T V_j = \frac{1}{2} T g_0 I_{sp}$$
(3-4)

where P_{jet} is the **jet power**, the power of the exhaust jet, P_0 is the electrical power supplied by the power source at 1 AU and η_{jet} is the **power conversion efficiency**, also known as the **jet efficiency** assumed to be 60% [Jacobson et al., 2005].

Since the available electrical power is limited, the exhaust velocity is limited as well. However, to obtain a high mass efficiency, the exhaust velocity should be high. Therefore, SEP operates at low thrust levels.

The mass of the SEP system M_{P_0} can be estimated based on the value of P_0 :

$$M_{P_0} = P_0 \cdot k_{P_0} \tag{3-5}$$

where k_{P_0} is the **power-to-mass ratio** of the SEP system.

Trade-off between mass of the propellant and of the power system As has been explained, a high exhaust velocity results in less propellant mass. However, to achieve higher exhaust velocities, more power is required, which means an increase in power system mass. Therefore, a trade-off needs to be done to determine the optimum mission exhaust velocity where the

combined mass of the propellant and of the power system are minimal. This concept has been visualized in Figure 3-3.



Figure 3-3: Trade-off between mass of the propellant and of the power system [Elvik, 2004].

3-4 Realistic values for Solar Electric Propulsion

Realistic values for different types of SEP have been listed in Table 3-1, categorized by the type of thruster. Despite being based on a document from 2003, it is still accurate and lists similar capabilities as the more recent source Oh et al. [2013], which shows the capabilities of existing engines and is hence very specific. Therefore, it has been decided to show the more general listing from Frisbee [2003].

Thruster	Typical electric power range	I_{sp} (s)	$V_j (km/s)$
Electrothermal			
Resistojets	100s of W	300-400	3-4
Arcjets			
Hydrazine	kW	500-600	5-6
Hydrogen	10s of kW	900-1200	9-12
Ammonia	kW to $10s$ of kW	600-800	6-8
Electrostatic (Xe propellant)			
Gridded ion engines	W to 100 kW	2000-10000	20-100
Stationary plasma thrusters	$100\mathrm{s}$ of W to $10\mathrm{s}$ of kW	1000-2500	10-25
Thruster with anode layer	$100\mathrm{s}$ of W to $10\mathrm{s}$ of kW	1000-4000	10-40
Electromagnetic			
Magnetoplasmadynamic			
Steady-state, lithium	100s of kW to MW	3000-9000	30-90
Steady-state, hydrogen	>MW	9000-12000	90-120
Pulsed plasma thruster	10s to $100s$ of W	1000 - 1500	10 - 15
Pulsed inductive thruster	10s of kW	3000-8000	30-80

Table 3-1: Capabilities of different Solar Electric Propulsion technologies [Frisbee, 2003].

Part II

Previous work performed during the author's internship

Chapter 4

Introduction to previous research

Low-thrust trajectory optimization can be done using a multitude of different methods and tools. A few examples are SEPTOP/VARITOP [Sauer, 2003], Sims-Flanagan/MALTO [Sims et al., 2006], Mystic [Whiffen, 2006], etc, each with their own level of complexity and accuracy [Polsgrove et al., 2006]. The previously performed research during the author's internship required a tool to perform preliminary, proof-of-concept studies. Hence, the speed with which the developed software could scan the search space was considered more important than the level of fidelity of the method. Therefore, the usage of a fast, low-fidelity, low-thrust optimization procedure was preferred. The selected optimization procedure is known in literature as the Sims-Flanagan method [Sims et al., 2006]. The most well-known implementation of this method is JPL's MALTO tool. Due to governmental ITAR (International Traffic in Arms Regulations) restrictions, the usage of MALTO is restricted to US citizens. Therefore, the author implemented a self-written version of the Sims-Flanagan method based on the work of Herman [2012]. For reasons that will be explained in the next chapter, the author developed a, to his knowledge, new variant of the Sims-Flanagan transcription using a TNC coordinate system instead of a Cartesian coordinate system. The developed code has then be validated using existing low-thrust tools.

This previous research resulted in a conference paper entitled Preliminary Design of a Crewed Mars Flyby Mission using Solar Electric Propulsion [De Smet et al., 2014]. The research method and results of this previously conducted research will be explained in the next six chapters: Chapter 5 will explain the utilized low-thrust optimization method: the Sims-Flanagan method. Chapters 6, 7 and 8 will each treat one of the main modules of the software: the propagation module, the constraints calculation module and the Jacobian module. Chapter 9 will summarize the results of the previous research. Finally, Chapter 10 will explain how the previously conducted research fits in with this thesis research and will set up the research goals.

Chapter 5

Sims-Flanagan method and program structure

In this chapter, the Sims-Flanagan method will be explained. Additionally, the structure of the state vector and of the program will be explained.

5-1 Sims-Flanagan method

The Sims-Flanagan method discretizes the **thrust profile** using multiple impulsive manoeuvres as an approximation of a continuous thrust profile. The trajectory is cut up into different **legs**, which are bounded by **control nodes** that allow for a constrained discontinuous state. Such a control node can have any physical meaning such as a rendez-vous or a flyby of a celestial body, a probe being released in deep space, etc. This has been visualized in Figure 5-1 for a trajectory consisting of two legs where the control nodes represent the spacecraft's encounters with planetary bodies. The control nodes bounding the first leg represent the launch from Earth and the arrival at Mars. Between the first and second leg of the trajectory, a Martian flyby is modeled as an instantaneous change in the hyperbolic excess velocity. The control nodes bounding the second leg represent the departure from Mars and the arrival at Earth.

Each of those legs is discretized into 2N segments. The thrust in a segment is represented by an impulsive manoeuvre at the midpoint of that segment, of which two have been depicted in Figure 5-1. These impulsive manoeuvres influence the forward and backward propagation of the leg starting respectively at the initial and final control node of the leg. These two propagations meet each other in the middle of the leg at a so-called **match point**. The heliocentric coordinates and velocities, the hyperbolic excess velocities and the spacecraft mass at the control nodes are used as the starting point for these propagations. The mass profile throughout the propagation is required, as it connects the thrust on each segment to the allowable size of the manoeuvre, as will be shown in Subsection 7-2-2. These initial conditions



Figure 5-1: Structure of the Sims-Flanagan formulation on a generic EME trajectory (adapted from Sims et al. [2006]).

are then propagated using the control parameters on each segment, which are the magnitude and direction of the manoeuvre along with the specific impulse. The propagation between the impulsive manoeuvres is done using a two-body model and an RK7(8)13M integrator, also known as DOPRI8 [Prince and Dormand, 1981]. This integrator has been chosen based on the discussion in the author's literature study and in Montenbruck [1992], its widespread usage, and in-house knowledge on how to write such an integrator. These forward and backward propagations have to be consistent in heliocentric coordinates, velocities and mass at the match points in order to ensure a continuous trajectory at the match points.

5-2 Structure of the state vector

Based on the Sims-Flanagan trajectory representation, the **state vector** X for a single-leg trajectory has been established. The structure of this vector will now be explained. The first 23 elements are the initial and final mass M_0 and M_f , the initial and final coordinates x_0, y_0, z_0, x_f, y_f and z_f , initial and final velocities $\dot{x}_0, \dot{y}_0, \dot{z}_0, \dot{x}_f, \dot{y}_f$ and \dot{z}_f and initial and final hyperbolic excessive velocities $V_{\infty,x_0}, V_{\infty,y_0}, V_{\infty,z_0}, V_{\infty,x_f}, V_{\infty,y_f}$ and V_{∞,z_f} of the control nodes. Furthermore, those 23 elements also include the mass of the propulsion unit M_{P_0} and the times t_0 and t_f at the initial and final control node. Those 23 elements are not always independent. For instance, in the scenario where a control node represents a planetary encounter, the control node time must be properly coupled to the coordinates and velocities of that control node. Hence, in that scenario, certain elements could be removed from the state vector. Both the coupling and removal method will be discussed in Chapter 13. For now, it has been decided to leave all 23 elements in the state vector to make the representation as generic as possible; certain optimization problems do not require the coupling between time and coordinates and velocities at the nodes. An example is the optimization of a trajectory to reach a fixed point in an orbit. Here, the coordinates and velocities would have to be kept constant while the time at the final control node would be allowed to change. Those 23 elements are followed by 4N elements representing the ΔV vectors and the specific impulse for manoeuvres 1 up to N (the forward propagation). The last 4N elements are the ΔV vectors and the specific impulse for manoeuvres N+1 up to 2N (the backward propagation). So in total, 2N manoeuvres are applied to the trajectory.

Note that the author chose to represent the ΔV vectors in a TNC coordinate frame instead of a classic Cartesian coordinate frame, as it is expected that the usage of a Cartesian coordinate system will result in convergence issues with the optimizer when one tries to optimize control node times. For instance if the initial time changes, the positions of the midpoints of the segments where the impulses are applied change. As such, the three elements of the ΔV vector also change considerably in a Cartesian reference frame. Hence, the optimizer needs several iterations to adjust the ΔV vector each time the initial or final time is shifted, resulting in an overall slow convergence. Based on the discussion of the body-centered reference frame **Tangential, Normal and Cross-track (TNC)** in Vallado [2003] - where the Tangential direction is parallel to the velocity vector, the Normal direction is in the orbital plane perpendicular to the velocity vector and the Cross-track direction is parallel with the angular momentum vector - it is expected that the three elements of the ΔV vector expressed in the TNC reference frame change less. Hence, the convergence onto an optimal solution would be sped up utilizing a TNC representation of the ΔV vectors.

5-3 Structure of the program

The structure of the program developed by the author has been visualized in Figure 5-2. The majority of the code has been written in C++. In this C++ part, the main program first initializes and sets up the optimization problem. In order to do so, it generates an initial state vector X_0 and it sets boundaries on the state vector and on constraints. These initial guesses and boundaries are passed on to the SNOPT tool. This tool calls the function usrfq. In this function, the initial guess for the state vector is propagated, from which the constraints can be calculated as well as the derivatives of each constraint with respect to each element of the state vector. These constraints and derivatives are returned to the SNOPT program after which SNOPT updates the state vector using a Sequential Quadratic Programming (SQP) algorithm [Gill, 2008]. This updated state vector is then passed to the usrfg function, which again propagates it and calculates the constraints and derivatives. This iterative method to update the state vector is repeated until certain termination criteria are met to ensure that the resulting state vector is (locally) optimal. Upon meeting the termination criteria, the SNOPT program passes the optimal state vector back to the main program. In this program, the optimal state vector is used to generate output text files. These files are then read in by a Matlab program, which creates several plots. In the next three chapters, the **propagation** module, the set-up of the constraint vector F and the calculation of the Jacobian matrix G will be explained.



Chapter 6

Propagation module

Within the written *usrfg* function, three modules can be identified in Figure 5-2. One of them is responsible for the propagation of the state vector. The structure of this propagation will be described in the next section, which will be followed by a discussion on its implementation.

6-1 Structure of the propagation

The propagation module uses the inputted state vector X to propagate the state using a forward and a backward propagation towards the match point. Before explaining the structure of the propagation, visualized in Figure 6-1, it is important to note that the used terminology has been described in Sections 5-1 and 5-2. The forward propagation starts at control node 0 and propagates towards the forward match point. The initial **mass** M_0 , the initial **coordinates** x_0 , y_0 and z_0 , the initial **velocities** \dot{x}_0 , \dot{y}_0 and \dot{z}_0 and the initial **hyperbolic excess velocities** V_{∞,x_0} , V_{∞,y_0} and V_{∞,z_0} are used as starting points for the propagation and influenced by manoeuvres 1 up to N. The sum of the initial and excess velocities form the initial spacecraft velocity, as can be seen from Equation 8-9. The same procedure is used for the backward propagation loop between control node f and the backward match point, but then with the final mass, final coordinates, velocities and excess velocities, which are also propagated to the match point and influenced by manoeuvres N+1 up to 2N.



Figure 6-1: Structure of the propagation.

The propagation itself is done **step-wise**; the coordinates, velocities and State Transition Matrix (STM) elements are propagated up to the first manoeuvre. At this manoeuvre, the velocities and the mass are updated. These updated elements are then used for the next integration step up to the next manoeuvre and finally to the match point. Before explaining how the actual propagation works, it is necessary to explain the concept of a State Transition Matrix, which will be done in the next section.

6-2 State Transition Matrix

STM's are often used to propagate the state in linear time-varying systems. From Jain and Lande [2012], it is known that such systems are of the form

$$\dot{x}(t) = A(t)x(t) + B(t)u(t) y(t) = C(t)x(t) + D(t)u(t)$$
(6-1)

where x(t) is the state vector with dimension n, u(t) is the control input vector with dimension m and y(t) is the system output vector with dimension p. A, B, C and D are four coefficient matrices of the linear system with dimensions nxn, nxm, pxn and pxm respectively [Antsaklis and Michel, 2007]. They are called state or system matrix, input matrix, output matrix and feedthrough or feedforward matrix respectively. However, systems do not always vary linearly with time. Therefore, one often needs to perform a linearization to obtain such a system.

The STM $\Phi(t, t_0)$ within a linear or linearized system satisfies the following relation [Jain and Lande, 2012].

$$\frac{\partial \Phi(t, t_0)}{\partial t} = A(t)\Phi(t, t_0) \tag{6-2}$$

If the state space system in Equation 6-1 has B = D = 0, the solution is [Jain and Lande, 2012]

$$\begin{aligned}
x(t) &= \Phi(t, t_0) x_0 \\
y(t) &= C(t) \Phi(t, t_0) x_0
\end{aligned}$$
(6-3)

As such, the STM can be defined as the linearized approximation of the change in the state at a certain time t, based on a difference in the initial state at t_0 . The STM can be used to solve the state space system in Equation 6-1. Therefore, it can be used to propagate x in time. In order to obtain an STM, some properties of STM's must be understood [Jain and Lande, 2012]

$$\Phi(t,t) = \mathbb{I}
\Phi(t_0,t_1) = \Phi^{-1}(t_1,t_0)
\Phi^{-1}(t_1,t_0) = \Phi^T(t_1,t_0)
\Phi(t_2,t_0) = \Phi(t_2,t_1)\Phi(t_1,t_0)$$
(6-4)

STM's can be found using different methods like the infinite series method, the similarity transformation method, the Cayley-Hamilton Theorem method, the Laplace transform method, etc which can all be found in Antsaklis and Michel [2007].

6-3 Propagation between manoeuvres

As explained above, the propagation of the 36 STM elements and the 6 Cartesian coordinates and velocities between two consecutive manoeuvres has been performed using a numerical integrator. Therefore, a high-order embedded Runge-Kutta formula, the **DOPRI8 integrator**, also known as **RK8(7)-13M integrator**, has been written and applied to a **two-body integration scheme**. Both will be explained in the following subsections.

6-3-1 DOPRI8 numerical integrator

In general, numerical integration solves the initial-value problem of

$$\frac{dX}{dt} = \bar{F}(t,\bar{X}) \tag{6-5}$$

$$\bar{X}(t_0) = \bar{X}_0 \tag{6-6}$$

where \bar{X}_0 is the initial state vector and F the derivative function, dependent on the time tand the state vector \bar{X} . Numerical integrators give an approximate solution of \bar{X} at certain points in time: the **mesh points**. These mesh points are separated in time by a certain value called the **step size** h_n . Here, an approximate solution for mesh point t_n will be denoted as \bar{X}_n while the exact solution is $\bar{X}(t_n)$.

The DOPRI8 integration scheme only uses the previous mesh point and is therefore called a **single-step method** where t_{n+1} is found based on t_n only. The DOPRI8 integration scheme updates the step size based on the difference between an 8th and a 7th order Runge-Kutta scheme with 13 stages, hence the name RK8(7)-13M.

For the new time t_{n+1} found from

$$t_{n+1} = t_n + h_n \tag{6-7}$$

one can find the updated state from the $7^{\rm th}\text{-}{\rm order}$ Runge-Kutta scheme

$$\bar{\hat{X}}_{n+1} = \bar{\hat{X}}_n + \sum_{i=1}^{13} \hat{b}_i \bar{K}_i$$
(6-8)

and the updated state from the 8th-order Runge-Kutta scheme

$$\bar{X}_{n+1} = \bar{\hat{X}}_n + \sum_{i=1}^{13} b_i \bar{K}_i \tag{6-9}$$

where

$$\bar{K}_1 = h_n \bar{F}(t_n, \bar{\hat{X}}_n) \tag{6-10}$$

$$\bar{K}_i = h_n \bar{F} \left(t_n + c_i \cdot h_n, \bar{X}_n + \sum_{j=1}^{i-1} a_{ij} \bar{K}_j \right)$$
 with i=2, 3, ..., 13 (6-11)

The c_i , a_{ij} , \hat{b}_i and b_i parameters are known and fixed for this specific integration scheme. They can be found in Table 6-1. The derivative function \bar{F} that is utilized for this numerical integration will be explained in the next subsection.

	b_i	$\frac{13451932}{455176623}$	0	0	0	0	$\frac{-808719846}{976000145}$	$\frac{1757004468}{5645159321}$	$\frac{656045339}{265891186}$	$\frac{-3867574721}{1518517206}$	$\frac{465885868}{322736535}$	$\frac{53011238}{667516719}$	$\frac{2}{45}$	0
].	\hat{b}_i	$\frac{14005451}{335480064}$	0	0	0	0	$\frac{-59238493}{1068277825}$	$\frac{181606767}{758867731}$	$\frac{561292985}{797845732}$	$\frac{-1041891430}{1371343529}$	$\frac{760417239}{1151165299}$	$\frac{118820643}{751138087}$	$\frac{-528747749}{2220607170}$	41
rmand, 1981													$\frac{65686358}{487910083}$	$\frac{248638103}{1413531060} 0$
nce and Do												$\frac{3065993473}{597172653}$	$\frac{3962137247}{1805957418}$	$\frac{-160528059}{685178525}$
<i>i</i> stages [Pri											$\frac{123872331}{1001029789}$	$\frac{-45442868181}{3398467696}$	$\frac{-4093664535}{808688257}$	$\frac{3936647629}{1978049680}$
ie for the 13										$\frac{800635310}{3783071287}$	$\frac{393006217}{1396673457}$	$\frac{15336726248}{1032824649}$	$\frac{5232866602}{850066563}$	$\frac{-13158990841}{6184727034}$
gration schem	a_{ij}								$\frac{-180193667}{1043307555}$	$\frac{790204164}{839813087}$	$\frac{6005943493}{2108947869}$	$\frac{-48777925059}{3047939560}$	$\frac{5731566787}{1027545527}$	$\frac{11173962825}{925320556}$
(7)-13M inte								$\frac{23124283}{180000000}$	$\frac{545815736}{2771057229}$	$\frac{100302831}{723423059}$	$\frac{-12992083}{490766935}$	$\frac{-10304129995}{1701304382}$	$\frac{-703635378}{230739211}$	$\frac{652783627}{914296604}$
of the RK8							$\frac{3}{20}$	$\frac{-28693883}{1125000000}$	$\frac{22789713}{633445777}$	$\frac{-421739975}{2616292301}$	$\frac{-309121744}{1061227803}$	$\frac{1311729495}{1432422823}$	$\frac{-477755414}{1098053517}$	$\frac{-411421997}{543043805}$
Parameters						$\frac{75}{64}$	$\frac{3}{16}$	$\frac{77736538}{692538347}$	$\frac{61564180}{158732637}$	$\frac{-433636366}{683701615}$	$-37695042795\\15268766246.0$	$\frac{8478235783}{508512852}$	$\frac{-3185094517}{667107341}$	$\frac{-5068492393}{434740067}$
: 6-1:					$\frac{3}{32}$	$\frac{-75}{64}$	0	0	0	0	0	0	0	0
able				$\frac{1}{16}$	0	0	0	0	0	0	± 0	0 6	0	0
F			$\frac{1}{18}$	$\frac{1}{48}$	$\frac{1}{32}$	$\frac{5}{16}$	<u>30</u> 80	$\frac{29443841}{614563906}$	$\frac{16016141}{946692911}$	$\frac{39632708}{573591083}$	$\frac{246121993}{1340847787}$	$\frac{-102846818}{846180014}$	$\frac{185892177}{718116043}$	$\frac{403863854}{491063109}$
	c_i	0	$\frac{1}{18}$	$\frac{1}{12}$	∞1⊢	$\frac{5}{16}$	က၊တ	$\frac{59}{400}$	$\frac{93}{200}$	$\frac{5490023248}{9719169821}$	$\frac{13}{20}$	$\frac{1201146811}{1299019798}$	1	1

24

Propagation module

If the error is larger than the tolerance, the propagated state for t_{n+1} from the 7th-order Runge-Kutta scheme is not accepted and the initial time is not updated. The same integration step is repeated with a different time step, which has been adjusted based on the error:

$$h_{n+1} = 0.9 \cdot h_n \left[\frac{\text{allowable error}}{|\bar{X}_{n+1} - \bar{X}_{n+1}|} \right]^{1/15}$$
(6-12)

If the error is larger than the tolerance, the new time step will be smaller, resulting in a smaller, acceptable error for the next integrator call. This time step update is not only performed when the error is larger than the tolerance. If the error is smaller than the tolerance, it means that the time step has been too small, as a larger time step would still result in acceptable errors. Therefore, it is more efficient to use a larger time step.

The implementation of this scheme has been validated by comparing the output of several different integration steps to the output of other validated integration schemes such as ode45 in Matlab, a Python based version of the RK8(7)-13M integrator written by Jon Herman [2012], etc. Very similar or identical results were obtained, validating the implementation of this scheme.

6-3-2 Two-body force model derivative function

In order to know the STM and coordinates and velocities as a function of time, one needs to know the derivative function \bar{F} of the 36 STM elements and the 6 Cartesian coordinates and velocities. The numerical integration can be performed solely using a two-body force model. Based on simulations in the author's literature study, it is known that gravity field, atmospheric drag, electromagnetic, third-body, radiation pressure, and relativistic perturbations do not need to be taken into account. Furthermore, the thrust does not need to be included into the numerical integration, since it is modeled as an impulsive manoeuvre at the end of each integration step. For a two-body model, the derivatives of the Cartesian coordinates and velocities can be found using:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \ddot{x} \\ \ddot{y} \\ \ddot{z} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ \frac{-\mu}{R^3} x & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{-\mu}{R^3} y & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{-\mu}{R^3} z & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix}$$
(6-13)

The derivative of the 36 STM elements can be found using the following formula

$$\dot{\Phi}_{t,t_0} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ \frac{\partial^2 \Omega}{\partial x^2} & \frac{\partial^2 \Omega}{\partial x \partial y} & \frac{\partial^2 \Omega}{\partial x \partial z} & 0 & 0 & 0 \\ \frac{\partial^2 \Omega}{\partial x \partial y} & \frac{\partial^2 \Omega}{\partial y \partial z} & \frac{\partial^2 \Omega}{\partial y \partial z} & 0 & 0 & 0 \\ \frac{\partial^2 \Omega}{\partial x \partial z} & \frac{\partial^2 \Omega}{\partial y \partial z} & \frac{\partial^2 \Omega}{\partial z^2} & 0 & 0 & 0 \end{bmatrix} \Phi_{t,t_0}$$
(6-14)

from Schaub and Junkins [2009], where

$$\Omega = \frac{\mu}{R}$$

$$\frac{\partial^2 \Omega}{\partial x^2} = 3\frac{\mu}{R^5}x^2 - \frac{\mu}{R^3}$$

$$\frac{\partial^2 \Omega}{\partial y^2} = 3\frac{\mu}{R^5}y^2 - \frac{\mu}{R^3}$$

$$\frac{\partial^2 \Omega}{\partial z^2} = 3\frac{\mu}{R^5}z^2 - \frac{\mu}{R^3}$$

$$\frac{\partial^2 \Omega}{\partial x \partial y} = 3\frac{\mu}{R^5}xy$$

$$\frac{\partial^2 \Omega}{\partial x \partial z} = 3\frac{\mu}{R^5}xz$$

$$\frac{\partial^2 \Omega}{\partial y \partial z} = 3\frac{\mu}{R^5}yz$$
(6-15)

where μ is the standard gravitational parameter of the Sun and R is the radial distance from the Sun which can be obtained from the propagated x, y and z coordinates.

6-4 Application of the manoeuvre

The propagated Cartesian velocities before the application of the manoeuvre are then updated based on the magnitude and direction of the ΔV vector. However, the ΔV vector is expressed in a TNC coordinate system. As such, one first needs to convert $[\Delta V_T, \Delta V_N, \Delta V_C]$ into $[\Delta V_x, \Delta V_y, \Delta V_z]$. This transformation procedure can be found in Appendix A. The Cartesian velocities and the mass after the manoeuvre can then be calculated. The precise implementation depends on whether it is a forward or backward propagation.

$$\dot{\varrho}_{\text{after manoeuvre}} = \dot{\varrho}_{\text{before manoeuvre}} + \Delta V_{\varrho} \text{ with } \varrho = x, y, z$$
 (6-16)

$$\dot{\varrho}_{\text{before manoeuvre}} = \dot{\varrho}_{\text{after manoeuvre}} - \Delta V_{\varrho} \text{ with } \varrho = x, y, z$$
 (6-17)

The mass can be updated using Tsiolkovsky's law for the forward and backward propagation respectively. In the backward propagation scenario, the mass after the manoeuvre is already known, and one wants to obtain the mass before the manoeuvre.

forward:	mass after manoeuvre = mass before manoeuvre $\cdot e^{-\frac{\Delta V}{I_{sp} \cdot g_0}}$	(6-18)
backward:	mass before manoeuvre = mass after manoeuvre $\cdot e^{\frac{\Delta V}{I_{sp} \cdot g_0}}$	(6-19)

Chapter 7

Constraints

To ensure realistic trajectories, constraints have to be imposed. These constraints on a trajectory for the Sims-Flanagan method depend on the type of control nodes on the legs. However, some constraints are present for all types of control nodes. Constraints are grouped in a constraint vector F, which can be calculated using the outputs of the propagation module.

7-1 Cost function

The first element of the constraint vector is the cost function, the function that needs to be minimized. So, the exact implementation depends on what the goal of the optimization is.

7-2 General constraints

Several constraints are always present, independent of the types of control nodes bounding the leg. These are the **match point** and **thrust constraints**.

7-2-1 Match point constraints

In each Sims-Flanagan problem, constraints must be imposed such that the heliocentric coordinates and velocities and the spacecraft mass from the forward and backward propagation at the match points are equal to ensure a continuous trajectory. These will be called the seven match point constraints:

$$F_{\Delta M} = M_{\text{match point, forward}} - M_{\text{match point, backward}} \in [-\varepsilon, \varepsilon]$$

$$F_{\Delta \varrho} = \varrho_{\text{match point, forward}} - \varrho_{\text{match point, backward}} \in [-\varepsilon, \varepsilon]$$

$$F_{\Delta \dot{\varrho}} = \dot{\varrho}_{\text{match point, forward}} - \dot{\varrho}_{\text{match point, backward}} \in [-\varepsilon, \varepsilon]$$
(7-1)

where ρ represents the general Cartesian coordinates x, y and z. These constraints, after proper scaling, must have values between $-\varepsilon$ and ε with ε being a small value.

7-2-2 Thrust constraints

In Section 3-3, it has been explained that solar electric propulsion is power limited. Hence, the magnitude of the manoeuvre applied at the midpoint on each segment is also limited. It must be ensured that the magnitude of this manoeuvre does not exceed the maximum that the spacecraft can provide during the duration of that specific segment with a certain power level. The thrust on a segment can be translated into an impulsive manoeuvre using

$$T = \frac{\Delta V_i M_i}{\Delta t} \tag{7-2}$$

with the duration of the segment Δt , the size of the manoeuvre ΔV_i and the mass of the system before the manoeuvre M_i .

The thrust constraints can be found by combining Equations 7-2 and 3-4 into Equation 7-3. If the power is independent of heliocentric distance, for instance using a nuclear power source, the thrust is constrained as

$$\frac{P_{\text{jet}} < \eta_{\text{jet}} P_0}{\frac{\Delta V_i M_i I_{\text{sp}} g_0}{2DT}} < \eta_{\text{jet}} P_0$$
(7-3)

where P_0 is the available electrical power. As such,

$$F_{T,i} = \frac{\Delta V_i M_i I_{\rm sp} g_0}{2P_0 \eta_{\rm jet} DT} \in [0, 1]$$

$$(7-4)$$

If the available power is dependent on heliocentric distance, which is the case for solar electric propulsion, the thrust constraints change and become

$$F_{T,i} = \frac{\Delta V_i M_i I_{\rm sp} g_0 R_i^2}{2P_0 \eta_{\rm jet} DT A U^2} \in [0, 1]$$
(7-5)

where P_0 is the available power at a heliocentric distance of 1 AU and R_i is the heliocentric distance.

7-3 Leg-specific constraints

Besides the node-independent general constraints, there are also control-node specific constraints. Those can be found in this section for three different types of control nodes: departure, flyby and return nodes.

7-3-1 Departure-node constraints

The departure at Earth adds several launcher-specific constraints.

Launch mass constraints There is a relation between the excessive velocities $V_{\infty_{x_0}}, V_{\infty_{y_0}}$ and $V_{\infty_{z_0}}$ and the maximum mass that can be launched by the launcher, $M_{L,\max}$, which can be found from Equation 9-2. Hence, it must be enforced that the initial mass cannot be higher than this maximum launch mass for a certain V_{∞} . Therefore:

$$F_{LM} = M_{L,\max} - M_0 \in [0, M_{\max,C_3=0}]$$
(7-6)

which must be between 0 and the maximum launch mass $M_{\max,C_3=0}$ for that launcher.

 C_3 constraint The launch vehicle performance coefficients in Table 9-2 used in Equation 9-2 to find the maximum launch mass are found using a polynomial fit for a certain region of C_3 values. As such, the launch vehicle performance coefficients are only valid within this region of C_3 values. Hence, it must be enforced that the C_3 value calculated by

$$F_{C_3} = V_{\infty,x_0}^2 + V_{\infty,y_0}^2 + V_{\infty,z_0}^2 \in [0, C_{3,\max}]$$
(7-7)

is smaller than the upper C_3 value for the polynomial fit for a certain launcher.

7-3-2 Flyby-node constraints

An unpowered planetary flyby adds several specific constraints. In this discussion, the inbound and outbound leg of the flyby will be indicated by the superscripts I and II respectively.

Mass-equality constraints The mass before and after the flyby must be equal in magnitude, since for an unpowered flyby, no manoeuvre is performed during the planetary flyby.

$$\mathbf{F}_{\text{mass flyby}} = M_{0^{II}} - M_{f^{I}} \in [-\varepsilon, \varepsilon]$$
(7-8)

Relative velocity equality constraint From Cornelisse et al. [1979], it is known that the incoming and outgoing relative velocities must be equal in magnitude for an unpowered flyby.

$$\mathbf{F}_{\text{relative velocities flyby}} = |V_{\infty^{II}}| - |V_{\infty^{I}}| \in [-\varepsilon, \varepsilon]$$
(7-9)

Flyby altitude A restriction on the periapse altitude of the flyby has been set to ensure that the spacecraft does not impact the planet or enters its atmosphere [Ellison et al., 2013]:

$$F_{\text{hfly}} = r_{\text{periapse}} - (r_{\text{planet}} + h_{\text{safety}}) \ge 0$$

$$F_{\text{hfly}} = \frac{\mu_{\text{planet}}}{V_{\infty}^{II^2}} \left[\frac{1}{\sin\left(\frac{\alpha}{2}\right)} - 1 \right] \ge (r_{\text{planet}} + h_{\text{safety}})$$
(7-10)

where μ_{planet} is the standard gravitational parameter of the flyby planet, h_{safety} is a safety altitude above the flyby planet surface and α is the deflection angle defined by

$$\alpha = \arccos\left[\frac{\overrightarrow{V_{\infty}^{I}} \cdot \overrightarrow{V_{\infty}^{II}}}{|V_{\infty}^{I}||V_{\infty}^{II}|}\right]$$
(7-11)

This equation has a singularity for $\alpha = 0$: the $\frac{1}{\sin(\frac{\alpha}{2})}$ term results in a division by 0. In reality, the equation still holds when α approaches 0. When the deflection angle goes to zero, it can be reasoned that no flyby has been performed. The limit for F_{hfly} for α going to 0 is found to be ∞ . It can be reasoned that a flyby at infinite distance indeed corresponds with not performing a flyby and indeed having a zero degree deflection angle.

7-3-3 Return-node constraint

Re-entry is a critical phase for any crewed mission. An important design parameter for reentry systems is the re-entry velocity. If this velocity surpasses the design value for the re-entry system, the heat load produced by the deceleration throughout the atmosphere can destroy the spacecraft and result in loss of crew. Therefore, a constraint on the re-entry velocity has to be imposed. This constraint can be translated into a constraint on the incoming hyperbolic excess velocity using the vis-viva equation:

$$\frac{V_{\infty,max}^2}{2} = \frac{V_{\text{re-entry,max}}^2}{2} - \frac{\mu_{\text{Earth}}}{r_{\text{re-entry}}}$$
(7-12)

So,

$$F_{\text{re-entry}} = V_{\infty} \in [0, V_{\infty, max}]$$
(7-13)

Chapter 8

Jacobian

SNOPT is a gradient-based optimization program. Hence, it requires the derivative of every constraint with respect to each element of the state vector. All of these derivatives were found analytically and will be briefly discussed here. A more complete explanation can be found in Appendix C.

8-1 Cost function

The derivative(s) of the cost function depend(s) on the definition of the cost function, which depends on the goal of the optimization.

8-2 Mass match point $F_{\Delta M}$

Looking at the following equations for the mass at the match points, one can see that the mass mismatch at the match point is influenced by M_0 , M_f and all the ΔV elements and specific impulses.

$$M_{\text{match point, forward}} = M_0 \cdot exp\left(-\frac{\sum_{i=1}^N \frac{\Delta V_i}{I_{\text{sp},i}}}{g_0}\right)$$
(8-1)

$$M_{\text{match point, backward}} = M_f \cdot exp\left(\frac{\sum_{i=N+1}^{2N} \frac{\Delta V_i}{I_{\text{sp},i}}}{g_0}\right)$$
(8-2)

Considering Equation 7-1, one can see that

$$\frac{\partial \Delta M}{\partial M_0} = \frac{M_{\text{match point, forward}}}{M_0}$$
(8-3)

$$\frac{\partial \Delta M}{\partial M_f} = -\frac{M_{\text{match point, backward}}}{M_f} \tag{8-4}$$

For the ΔV elements and specific impulses for the first N manoeuvres, representing the forward propagation, the derivatives are

$$\frac{\partial \Delta M}{\partial \Delta V_{\kappa,i}} = \frac{-M_{\text{match point, forward}}}{I_{\text{sp},i} \cdot g_0} \cdot \frac{\Delta V_{\kappa,i}}{\Delta V_i}$$
(8-5)

$$\frac{\partial \Delta M}{\partial \Delta I_{\rm sp,i}} = \frac{M_{\rm match point, forward} \cdot \Delta V_i}{I_{\rm sp,i}^2 \cdot g_0}$$
(8-6)

with i=1, 2, ..., N and κ =T, N or C.

For the ΔV elements and specific impulses for the last N manoeuvres, representing the backward propagation, the derivatives are

$$\frac{\partial \Delta M}{\partial \Delta V_{\kappa,i}} = \frac{-M_{\text{match point, backward}}}{I_{\text{SD,i}} \cdot g_0} \cdot \frac{\Delta V_{\kappa,i}}{\Delta V_i}$$
(8-7)

$$\frac{\partial \Delta M}{\partial \Delta I_{\rm sp,i}} = \frac{M_{\rm match \ point, backward} \cdot \Delta V_i}{I_{\rm sp,i}^2 \cdot g_0}$$
(8-8)

with i=N+1, N+2, ..., 2N and κ =T, N or C.

8-3 State match point constraints $F_{\Delta x}$, $F_{\Delta y}$, $F_{\Delta z}$, $F_{\Delta x}$, $F_{\Delta y}$, $F_{\Delta y}$

The derivatives of the state match point constraints can be subdivided into four categories: derivatives with respect to the elements making up the initial state, elements making up the final state, elements defining the forward manoeuvres and elements defining the backward manoeuvres.

8-3-1 Derivatives with respect to the initial node's coordinates, velocities and hyperbolic excess velocities

To obtain the derivatives of the match point constraints with respect to the initial node's coordinates, velocities and excess velocities, one must understand how a change in those initial conditions is propagated up to the match point. This will be explained through Figure 8-1.

First of all, one needs to realize that the initial Cartesian state, indicated by the s for "spacecraft state" subscript, is set up by a combination of the initial node's coordinates, velocities and excess velocities. These are indicated by the 0 for "initial node" subscript, according to

initial node spacecraft state =
$$\begin{bmatrix} x_{s,0} \\ y_{s,0} \\ z_{s,0} \\ \dot{y}_{s,0} \\ \dot{y}_{s,0} \\ \dot{z}_{s,0} \end{bmatrix} = \begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ \dot{x}_0 + V_{\infty,x_0} \\ \dot{y}_0 + V_{\infty,y_0} \\ \dot{y}_0 + V_{\infty,y_0} \\ \dot{z}_0 + V_{\infty,y_0} \end{bmatrix}$$
(8-9)

$$\begin{bmatrix} \Delta x_s \\ \Delta y_s \\ \Delta z_s \\ \Delta \dot{x}_s \\ \Delta \dot{$$

Figure 8-1: Propagation of a change in initial conditions to the match point.

This initial state is numerically propagated up to the state at point 1⁻ in Figure 8-1, indicating the Cartesian state before the application of the manoeuvre. From Equation 6-3, it is known that a change in the initial state can be converted into a change in the Cartesian state at point 1⁻ using the STM between point 0 and 1⁻, $\Phi_{1^-,0}$, which has been calculated and stored in the propagation module. So,

$$\begin{bmatrix} \Delta x_s \\ \Delta y_s \\ \Delta z_s \\ \Delta \dot{z}_s \\ \Delta \dot{y}_s \\ \Delta \dot{z}_s \end{bmatrix}_{1^{-}} = \Phi_{1^{-},0} \cdot \begin{bmatrix} \Delta x_s \\ \Delta y_s \\ \Delta z_s \\ \Delta \dot{z}_s \\ \Delta \dot{x}_s \\ \Delta \dot{y}_s \\ \Delta \dot{z}_s \end{bmatrix}_{0}$$
(8-10)

The directions of the manoeuvres expressed in TNC coordinates depend on the location of the manoeuvre. As such, a change in the state at point 1^- influences the directions in which the manoeuvres are applied. As such, an additional transformation matrix is required that transforms the change in the state over the manoeuvre from point 1^- to point 1^+ . Therefore, the transformation matrix $TNC_{1^+,1^-}$ explained in Appendix B can be used, which has been calculated and stored in the propagation module.

$$\begin{bmatrix} \Delta x_s \\ \Delta y_s \\ \Delta z_s \\ \Delta \dot{x}_s \\ \Delta \dot{y}_s \\ \Delta \dot{z}_s \end{bmatrix}_{1+} = TNC_{1+,1-} \cdot \begin{bmatrix} \Delta x_s \\ \Delta y_s \\ \Delta z_s \\ \Delta \dot{x}_s \\ \Delta \dot{x}_s \\ \Delta \dot{y}_s \\ \Delta \dot{z}_s \end{bmatrix}_{1-}$$
(8-11)

So, a change in the initial state can be propagated to a change of coordinates and velocities at the match point using:

$$\begin{bmatrix} \Delta x_s \\ \Delta y_s \\ \Delta z_s \\ \Delta \dot{x}_s \\ \Delta \dot{y}_s \\ \Delta \dot{z}_s \end{bmatrix}_{\text{MP,forward}} = \Phi_{MP,N^+} \cdot TNC_{N^+,N^-} \cdot \ldots \cdot \Phi_{2^-,1^+} \cdot TNC_{1^+,1^-} \cdot \Phi_{1^-,0} \cdot \begin{bmatrix} \Delta x_s \\ \Delta y_s \\ \Delta z_s \\ \Delta \dot{x}_s \\ \Delta \dot{y}_s \\ \Delta \dot{z}_s \end{bmatrix}_0$$
(8-12)

This matrix product $\Phi_{MP,N^+} \cdot TNC_{N^+,N^-} \cdot \ldots \cdot \Phi_{2^-,1^+} \cdot TNC_{1^+,1^-} \cdot \Phi_{1^-,0}$ will be called $\Psi_{MP,0}$, mapping the change from state at point θ to the state at point MP.

From the previous discussion, the derivatives of the match point constraints with respect to the initial coordinates, velocities and initial hyperbolic excess velocities can be found by realizing that

$$\Psi_{MP,0} = \begin{bmatrix} \frac{\partial x_{s,\text{MP for}}}{\partial x_{s,0}} & \frac{\partial x_{s,\text{MP for}}}{\partial y_{s,0}} & \frac{\partial x_{s,\text{MP for}}}{\partial z_{s,0}} & \frac{\partial x_{s,\text{MP for}}}{\partial \dot{x}_{s,0}} & \frac{\partial x_{s,\text{MP for}}}{\partial \dot{y}_{s,0}} & \frac{\partial x_{s,\text{MP for}}}{\partial \dot{y}_{s,0}} \\ \frac{\partial y_{s,\text{MP for}}}{\partial x_{s,0}} & \frac{\partial y_{s,\text{MP for}}}{\partial y_{s,0}} & \frac{\partial y_{s,\text{MP for}}}{\partial z_{s,0}} & \frac{\partial y_{s,\text{MP for}}}{\partial \dot{x}_{s,0}} & \frac{\partial y_{s,\text{MP for}}}{\partial \dot{y}_{s,0}} & \frac{\partial y_{s,\text{MP for}}}{\partial \dot{y}_{s,0}} \\ \frac{\partial z_{s,\text{MP for}}}{\partial x_{s,0}} & \frac{\partial z_{s,\text{MP for}}}{\partial y_{s,0}} & \frac{\partial z_{s,\text{MP for}}}{\partial z_{s,0}} & \frac{\partial z_{s,\text{MP for}}}{\partial \dot{x}_{s,0}} & \frac{\partial z_{s,\text{MP for}}}{\partial \dot{y}_{s,0}} & \frac{\partial z_{s,\text{MP for}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial \dot{y}_{s,\text{MP for}}}{\partial x_{s,0}} & \frac{\partial \dot{x}_{s,\text{MP for}}}{\partial y_{s,0}} & \frac{\partial \dot{x}_{s,\text{MP for}}}{\partial z_{s,0}} & \frac{\partial \dot{x}_{s,\text{MP for}}}{\partial \dot{x}_{s,0}} & \frac{\partial \dot{x}_{s,\text{MP for}}}{\partial \dot{y}_{s,0}} & \frac{\partial \dot{y}_{s,\text{MP for}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial \dot{y}_{s,\text{MP for}}}{\partial x_{s,0}} & \frac{\partial \dot{y}_{s,\text{MP for}}}{\partial y_{s,0}} & \frac{\partial \dot{y}_{s,\text{MP for}}}{\partial z_{s,0}} & \frac{\partial \dot{y}_{s,\text{MP for}}}{\partial \dot{x}_{s,0}} & \frac{\partial \dot{y}_{s,\text{MP for}}}{\partial \dot{y}_{s,0}} & \frac{\partial \dot{y}_{s,\text{MP for}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial \dot{y}_{s,\text{MP for}}}{\partial \dot{x}_{s,0}} & \frac{\partial \dot{y}_{s,\text{MP for}}}{\partial \dot{y}_{s,0}} & \frac{\partial \dot{y}_{s,\text{MP for}}}{\partial \dot{z}_{s,0}} & \frac{\partial \dot{y}_{s,\text{MP for}}}{\partial \dot{x}_{s,0}} & \frac{\partial \dot{y}_{s,\text{MP for}}}{\partial \dot{y}_{s,0}} & \frac{\partial \dot{y}_{s,\text{MP for}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial \dot{z}_{s,\text{MP for}}}{\partial \dot{z}_{s,0}} & \frac{\partial \dot{y}_{s,\text{MP for}}}{\partial \dot{z}_{s,0}} & \frac{\partial \dot{y}_{s,\text{MP for}}}{\partial \dot{z}_{s,0}} & \frac{\partial \dot{y}_{s,\text{MP for}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial \dot{z}_{s,\text{MP for}}}{\partial \dot{z}_{s,0}} & \frac{\partial \dot{z}_{s,\text{MP for}}}{\partial \dot{z}_{s,0}} & \frac{\partial \dot{z}_{s,\text{MP for}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial \dot{z}_{s,\text{MP for}}}{\partial \dot{z}_{s,0}} & \frac{\partial \dot{z}_{s,\text{MP for}}}{\partial \dot{z}_{s,0}} & \frac{\partial \dot{z}_{s,\text{MP for}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial \dot{z}_{s,\text{MP for}}}{\partial \dot{z}_{s,0}} & \frac{\partial \dot{z}_{s,\text{MP for}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial \dot{z}_{s,\text{MP for}}}{\partial \dot{z}_{s,0}} & \frac{\partial \dot{z}_{s,\text{MP for}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial \dot{z}_{s,\text{MP for}}}{\partial \dot{z}_{s,0}} & \frac{\partial \dot{z}_{s,\text{MP for}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial \dot{z}_{s,\text{MP for}}}{\partial \dot{z}_{s,0}} & \frac{\partial \dot{z}_{$$

The derivatives in the $\Psi_{MP,0}$ matrix are not yet the derivatives of the match point constraints with respect to the initial coordinates, velocities and excess velocities. Those can however easily be found from the derivatives in the $\Psi_{MP,0}$ matrix, as shown in Appendix C.

8-3-2 Derivatives with respect to the final node's coordinates, velocities and hyperbolic excess velocities

These derivatives can be found using a similar method as for the derivatives with respect to the initial node's coordinates, velocities and excess velocities. Therefore, they will not be discussed here but can be found in Appendix C.

8-3-3 Derivatives with respect to the forward velocity components

The derivatives with respect to the ΔV 's can be found using a similar method. An example will be given for the derivatives with respect to the first manoeuvre, which has been visualized in Figure 8-2.

Imagine a change in one of the ΔV components of the first manoeuvre. This will result in a change in the state after the first manoeuvre, still indicated by 1^+ .



Figure 8-2: Propagation of a change in manoeuvre 1 to the match point.

From Equation A-1, it is known that the change in the ΔV components of the first manoeuvre can be transformed into the change of the state after the first manoeuvre using the $\Delta TNC_{1+,1}$ matrix, which has been derived in Appendix A.

$$\begin{bmatrix} 0\\0\\\Delta\dot{x}_{s}\\\Delta\dot{y}_{s}\\\Delta\dot{z}_{s} \end{bmatrix}_{1+} = \Delta TNC_{1+,1} \cdot \begin{bmatrix} 0\\0\\\\\Delta(\Delta V_{T,1})\\\Delta(\Delta V_{N,1})\\\Delta(\Delta V_{C,1}) \end{bmatrix}_{1}$$
(8-14)

This change of the state after the first manoeuvre can then be propagated to the change of the state at the match point using matrix $\Psi_{MP,1^+}$ defined as

$$\Psi_{MP,1^+} = \Phi_{MP,N^+} \cdot TNC_{N^+,N^-} \cdot \dots TNC_{2^+,2^-} \cdot \Phi_{2^-,1^+}$$
(8-15)

Combining $\Delta TNC_{1^+,1}$ and $\Psi_{MP,1^+}$, one can see that

$$\begin{bmatrix} \Delta x_s \\ \Delta y_s \\ \Delta z_s \\ \Delta \dot{x}_s \\ \Delta \dot{y}_s \\ \Delta \dot{z}_s \end{bmatrix}_{\text{MP,forward}} = \Psi_{MP,1^+} \cdot \Delta TNC_{1^+,1} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ \Delta(\Delta V_{T,1}) \\ \Delta(\Delta V_{N,1}) \\ \Delta(\Delta V_{C,1}) \end{bmatrix}_{1}$$
(8-16)

As such, the derivatives can be found from

$$\Psi_{MP,1^{+}} \cdot \Delta TNC_{1^{+},1} = \begin{bmatrix} 0 & 0 & 0 & \frac{\partial x_{s,\text{MP for}}}{\partial \Delta V_{T,1}} & \frac{\partial x_{s,\text{MP for}}}{\partial \Delta V_{T,1}} & \frac{\partial x_{s,\text{MP for}}}{\partial \Delta V_{N,1}} & \frac{\partial x_{s,\text{MP for}}}{\partial \Delta V_{C,1}} \\ 0 & 0 & 0 & \frac{\partial y_{s,\text{MP for}}}{\partial \Delta V_{T,1}} & \frac{\partial y_{s,\text{MP for}}}{\partial \Delta V_{N,1}} & \frac{\partial y_{s,\text{MP for}}}{\partial \Delta V_{C,1}} \\ 0 & 0 & 0 & \frac{\partial z_{s,\text{MP for}}}{\partial \Delta V_{T,1}} & \frac{\partial z_{s,\text{MP for}}}{\partial \Delta V_{N,1}} & \frac{\partial z_{s,\text{MP for}}}{\partial \Delta V_{C,1}} \\ 0 & 0 & 0 & \frac{\partial y_{s,\text{MP for}}}{\partial \Delta V_{T,1}} & \frac{\partial x_{s,\text{MP for}}}{\partial \Delta V_{N,1}} & \frac{\partial x_{s,\text{MP for}}}{\partial \Delta V_{C,1}} \\ 0 & 0 & 0 & \frac{\partial y_{s,\text{MP for}}}{\partial \Delta V_{T,1}} & \frac{\partial y_{s,\text{MP for}}}{\partial \Delta V_{N,1}} & \frac{\partial y_{s,\text{MP for}}}{\partial \Delta V_{C,1}} \\ 0 & 0 & 0 & \frac{\partial y_{s,\text{MP for}}}{\partial \Delta V_{T,1}} & \frac{\partial y_{s,\text{MP for}}}{\partial \Delta V_{N,1}} & \frac{\partial y_{s,\text{MP for}}}{\partial \Delta V_{C,1}} \\ 0 & 0 & 0 & \frac{\partial z_{s,\text{MP for}}}{\partial \Delta V_{T,1}} & \frac{\partial z_{s,\text{MP for}}}{\partial \Delta V_{N,1}} & \frac{\partial z_{s,\text{MP for}}}{\partial \Delta V_{C,1}} \\ 0 & 0 & 0 & \frac{\partial z_{s,\text{MP for}}}{\partial \Delta V_{T,1}} & \frac{\partial z_{s,\text{MP for}}}{\partial \Delta V_{N,1}} & \frac{\partial z_{s,\text{MP for}}}{\partial \Delta V_{C,1}} \\ \end{bmatrix}$$
(8-17)

From the definitions of Δx_{MP} up to $\Delta \dot{z}_{MP}$ in Equation 7-1, it is known that the derivatives of $x_{s,MP}$ for up to $\dot{z}_{s,MP}$ for are equal to the derivatives of Δx_{MP} up to $\Delta \dot{z}_{MP}$. As such, $\Psi_{MP,1^+} \cdot \Delta TNC_{1^+,1}$ contains those derivatives.

8-3-4 Derivatives with respect to the backward velocity components

The derivatives with respect to the backward ΔV 's can be found using a similar method as for the forward ΔV 's. As such, they will not be discussed here but can be found in Appendix C.

8-4 Thrust constraints

The exact definition of the thrust constraints depends on whether or not the power is constant or dependent on the heliocentric distance. Depending on this definition of the thrust constraints, the derivatives change. Neither scenario impose large difficulties in the derivation of the derivatives, but are very lengthy. As such, it has been decided to skip their derivations here. The interested reader is referred to Appendix C-4.

8-5 Leg-specific constraints

The derivatives of the leg-specific constraints are trivial. As such, they will not be given here but can be found in Appendix C-5.

Chapter 9

Summary previous results

In the previous chapters, the main elements of the Sims-Flanagan method have been discussed. In this chapter, it will be explained how the Sims-Flanagan method can be applied to the optimization of low-thrust Martian flyby trajectories.

9-1 Set-up of the optimization problem and goals

To optimize low-thrust Earth-Mars-Earth trajectories, a set-up is required that includes two legs. Looking at Figure 5-1, one can see that the first and second leg are Earth-Mars and Mars-Earth respectively. The initial control node of the first leg is the departure at Earth. The constraints of such a departure node have been explained in Subsection 7-3-1. The final node of leg 1 and the departure node of leg 2 are flyby nodes and are subsequent to the constraints explained in Subsection 7-3-2. Finally, the final node of leg 2 is a re-entry node to which the constraints explained in Subsection 7-3-3 are applicable.

The goal of this research is to show that launch windows in 2018, 2019 and 2021 can be opened up or expanded using realistic assumptions on Solar Electric Propulsion (SEP) capabilities. The main research goal is to identify several launch windows in different years, for different power levels, payload mass and launchers. Typically, these are presented in the form of twodimensional grid searches with the launch date on the x-axis and the flyby date on the y-axis. This research limits itself to launch windows for a constant total time of flight of 501 days, based on Inspiration Mars [Tito et al., 2013], as the tool lacks time optimization capabilities at this stage. The addition of this third dimension would lead to an unmanageable large grid search. For each point in the grid search, the launch date, flyby date and arrival date are kept constant for the optimization of a single trajectory. These dates can then be translated into the control nodes' heliocentric coordinates and velocities using **Meeus' polynomials** [Meeus, 1991], which remain fixed throughout the optimization of this trajectory. These Meeus' polynomials are explained in Appendix G. Besides the dates, also the available power and the dry mass of the spacecraft are kept constant. This dry mass is the payload of the mission, based on the Inspiration Mars system [Inspiration Mars Foundation, 2013]. As such, the final mass at Earth return, which can be found using the following equation, is fixed:

$$M_{\text{Earth return}} = M_{\text{dry}} + M_{P_0}$$

$$= M_{\text{dry}} + P_0 \cdot k_{P_0}$$
(9-1)

where M_{P_0} is the mass of the SEP system, including the power supply as well as the propulsion system itself. k_{P_0} is the power-to-mass ratio of the SEP system and is assumed to be 30 kg per kW [Landau and Strange, 2011]. P_0 is the the available power for the SEP subsystem at a heliocentric distance of 1 AU.

The objective of the optimization procedure is to minimize the mass at Earth departure; the launch mass. The parameters which can be changed to minimize the launch mass are: the mass at Mars arrival, the mass at Mars departure (which has to be equal to the mass at Mars arrival), the departure hyperbolic excess velocity vector at the Earth's launch, the incoming and outgoing velocity vector at the Mars flyby, the incoming hyperbolic excess velocity vector at Earth arrival and the magnitude and direction of the manoeuvres. The specific impulse remains constant and has been set to 2000 s.

For simple cases, the optimization procedure does not require an initial guess of the state vector. However, since the optimization procedure does not optimize for launch, flyby and re-entry date, it has been decided to first narrow down the search space by finding feasible, chemical trajectories using Copernicus [Ocampo, 2002]. These trajectories are then used as an initial guess. This approach allows for a faster convergence towards a solution and for a quick assessment of the search space.

Based on this initial guess trajectory generated with Copernicus [Ocampo, 2002], a twodimensional grid search can be performed where the launch and flyby date vary. The re-entry date at Earth is then determined by the total mission duration length of 501 days [Inspiration Mars Foundation, 2013]. These grid searches demonstrate which combinations of launch and flyby date are feasible and as such determines several launch windows for this mission. These launch windows will be further elaborated upon in Section 9-5.

9-2 Operational considerations

Being a crewed mission, the solution must be made robust and operationally achievable. Therefore, some operational considerations need to be made.

First of all, it has been decided to put a higher level of restriction on the thrust level using a 90% **duty cycle**. Normally, one would expect the highest allowable thrust level to be the maximum achievable thrust level on that segment. However, it has been decided to restrict it to 90 percent of what is actually achievable. Such a margin has been shown to effectively prevent negative consequences of missed thrust [Oh et al., 2013]. Furthermore, this 90% does not only account for missed thrusts, it also accounts for tasks that may interfere with thrusting periods such as uploads & maintenance and communications & tracking [Oh et al., 2013]. Furthermore, it is unwise to have manoeuvres near planets. The reason from an operational point being to avoid additional tasks to be performed during the critical phases near the planets during the mission. From an astrodynamical point of view, it is important to avoid a dependency on manoeuvres near planets. Manoeuvres near planets have a much larger effect than heliocentric manoeuvres. If the thrusters fail during these crucial manoeuvres, the system might not be able to recover from this failure. Therefore, 3 coast arcs have been imposed on the trajectory. A 5-day coast arc has been imposed upon leaving Earth to account for the early checkout phase, a coast arc 2 weeks prior and 2 days after the Martian flyby for pre- and post-flyby operations and a final coast arc of 2 weeks prior to arrival at Earth for re-entry operations.

As mentioned in Subsection 7-3-2, there are also considerations with respect to the flyby altitude. The flyby altitude has a lower bound of 200 km since it must not impact Mars, but also avoid the Martian atmosphere. Besides this lower bound, also an upper bound has been imposed. Considering that the purpose of this mission is to be the first human visit to Mars, it is assumed to be desirable to make a close visit to Mars. Therefore, the upper bound has been set to an altitude of 2000 km. In practice however, the majority of the solutions does not approach this upper bound.

Finally, the spacecraft has to be able to withstand the re-entry heat load, which is strongly related to the re-entry velocity. Inspiration Mars sets a boundary at 14.2 km/s or lower [Inspiration Mars Foundation, 2013]. This has been translated to a hyperbolic excess velocity limit of 8.969 km/s using Equation 7-12 assuming a re-entry altitude of 200 km.

The combination of all these margins make all of the presented solutions extremely robust, in order to ensure the safe return of the crew.

9-3 Mission parameters and launcher configurations

In order to obtain realistic results, several design parameters have to be obtained or assumed. So far, several design parameters have been assumed and operational considerations have been made. In this section, other mission parameters will be listed and summarized in Table 9-1. Furthermore, the launcher performance will be explained.

This mission is based on the Inspiration Mars concept. Hence, the payload mass of this mission will be extensively used for the remainder of this thesis work. First of all, the baseline payload mass is estimated to be 13.139 tons [Inspiration Mars Foundation, 2013]. The fully margined payload mass; the payload mass including several margins such as average contingency margin, average mass growth allowance, etc is 19 tons [Inspiration Mars Foundation, 2013].

The Inspiration Mars mission assumes it will use the Space Launch System (SLS) currently under development by NASA [Donahue and Sigmon, 2013]. Hence, for this research, two different launch performance curves for the SLS rocket will be used: one for the SLS/iCPS 1xRL10B2 and another one for the SLS/LUS 4xRL10C2, which will from now on be abbreviated to 1RL and 4RL respectively. These launch curves influence the launch mass

SEP and method related parameters	
Power to mass ratio SEP system [Landau and Strange, 2011]	30 kg/kW
Jet efficiency [Jacobson et al., 2005]	60%
SEP system duty cycle [Oh et al., 2013]	90%
Specific impulse	2000 s
Number of segments entire mission	120 segments ≈ 4 days/segment
Mission related parameters	
Launch vehicles [Donahue and Sigmon, 2013]	SLS/iCPS 1xRL10B2
	SLS/LUS 4xRL10C2
Total mission duration [Inspiration Mars Foundation, 2013]	501 days
Maximum re-entry velocity [Inspiration Mars Foundation, 2013]	14.2 km/s
Pavload mass [Inspiration Mars Foundation, 2013]	13.139 - 19 tons

Table 9-1: Important design parameters and assumptions.

Parameter	SLS/iCPS 1xRL10B2	SLS/LUS 4xRL10C2
a_0	$+2.3432926311306 \ 10^{1}$	$+3.7535441522694 \ 10^{1}$
a_1	$-2.85737186333384 \ 10^{-1}$	$-5.73908125974506 \ 10^{-1}$
a_2	$-1.04274819488625 \ 10^{-3}$	$+4.1479910698052 \ 10^{-3}$
a_3	$+2.8496453601933 \ 10^{-5}$	$-1.87085855662388 \ 10^{-5}$
a_4	$-1.07472836774709 \ 10^{-7}$	$+3.7024176005762 \ 10^{-8}$

Table 9-2: Launch vehicle performance coefficients.

constraint in Equation 7-6 as $M_{L,max}$; the maximum launch mass for a certain C_3 value, is determined by the launch curve:

$$M_{\rm L,max} = a_0 + a_1 \cdot C_3 + a_2 \cdot C_3^2 + a_3 \cdot C_3^3 + a_4 \cdot C_3^4 \tag{9-2}$$

where the parameters a_0 up to a_4 have been obtained through a polynomial fit of the launch curve obtained from Figure 9-1 from Donahue and Sigmon [2013] and have been listed in Table 9-2 where the resulting launch mass is in tons for C_3 values expressed in km²/s².

9-4 Initial guess generation

The Sims-Flanagan method often performs better with an initial guess that is in the vicinity of a feasible solution. Initial guesses may be generated using a variety of methods. In some cases, one can use a Lambert solver to build approximate transfers from one planet to another. The Lambert solutions occasionally generate real interplanetary transfers, where no manoeuvres are required anywhere. In other cases, the Lambert solutions arrive at Mars with a different energy than they depart, requiring a manoeuvre.

An alternate strategy has been implemented to generate initial guesses in this study. The idea behind direct transcription problems like the Sims-Flanagan method is to define control points at each planet and then optimize their states along with the thrust profile to generate a feasible trajectory. Initial guesses may be generated by defining the same control points, but



Figure 9-1: Launch performance curves for the SLS/iCPS 1xRL10B2 and SLS/LUS 4xRL10C2 configurations [Donahue and Sigmon, 2013].

assuming ballistic transfers in every leg. The same optimizer, e.g., SNOPT, may be used to optimize the states of the control points in order to make the trajectory continuous in position - that is, the goal is to reduce the position discontinuities between each leg to zero, but permit the velocity discontinuities to be non-zero. Indeed, the velocity discontinuities between legs can be large: on the order of a kilometer per second or more. But electric propulsion has time to execute this ΔV , so even a large velocity discontinuity may be acceptable in an initial guess.

Johnson Space Center's Copernicus mission design tool [Ocampo, 2002] has been used to generate the initial guesses. Copernicus uses a variety of optimizers; SNOPT was used in this case. Copernicus modeled the motion of each planet using the DE421 Planetary and Lunar Ephemerides provided by the Jet Propulsion Laboratory. The Earth departure has been modeled as a hyperbolic departure with a periapse altitude of 185 km and an inclination of 28.5 degrees. All other aspects of the trajectory have been permitted to vary in the optimization routine. The Earth departure was propagated for about 100 days, depending on the scenario - about halfway to Mars. A control state has been defined at Mars at the closest approach of the hyperbolic flyby. The periapse altitude has been constrained to be no lower than 200 km and the state was propagated back to the time of the end of the Earth departure leg, and also forward to the approximate halfway point to the next event: the Earth return. In this way, the Mars flyby is continuous but the trajectories don't meet perfectly between Earth and Mars, and in any planetary transfer. The control point at the Earth return was placed at a periapse altitude of 60 km to simulate an Earth atmospheric entry. The optimizer was permitted to vary most aspects of each control state in order to drive the position discontinuities between each planetary encounter to zero, and to minimize the velocity discontinuities. The dates of the key events were also permitted to vary, with the constraint that the entire mission be 501 days

Using Copernicus, three nominal chemical thrust scenarios in 2018, 2019 and 2021 have been identified. Those scenarios have been summarized in Table 9-3 including the magnitude of the Deep Space Manoeuvre (DSM) representative for the velocity discontinuities, if applicable.

Launch date	Launch C_3	DSM_1	Flyby date	DSM_2	Return V_{∞}
(mm-dd-yyyy)	$(\mathrm{km}^2/\mathrm{s}^2)$	(m/s)	(mm-dd-yyyy)	(m/s)	(km/s)
01-05-2018	37.45	N.A.	08-20-2018	N.A.	8.89
12-13-2019	58.37	647.31	09-02-2020	517.28	6.10
12-10-2021	21.44	1650.41	09-18-2022	2227.85	5.17

Table 9-3: Nominal scenarios identified using Copernicus.

9-5 Results

Around each nominal chemical scenario, a low-thrust grid search has been performed using these nominal scenarios as an initial guess. An overview of all these cases has been given in Table 9-4. In the next subsections, the grid searches for these cases will be analyzed.

Case	Scenario	Power	SEP system	Payload	Fixed final	Launcher
		level	mass	mass	mass	configuration
		(kW)	(tons)	(tons)	(tons)	
1	2018	10	0.3	19	19.3	SLS/LUS 4xRL10C2
2	2018	10	0.3	13.139	13.439	SLS/LUS 4xRL10C2
3	2018	10	0.3	13.139	13.439	SLS/iCPS 1xRL10B2
4	2018	25	0.75	15	15.75	SLS/iCPS 1xRL10B2
5	2019	10	0.3	13.139	13.439	SLS/LUS 4xRL10C2
6	2021	25	0.75	13.139	13.889	SLS/LUS 4xRL10C2

Table 9-4: Investigated low-thrust grid search cases.

The results of these optimization runs have been verified using an independent propagation. This propagation tool has been written by Jonathan Herman, which he successfully used to assess the feasibility of CCAR's solution to the 6^{th} Global Trajectory Optimization Competition and which he modified for this application. This propagation was fed with the optimized output to quantify that it corresponded to a feasible trajectory.

9-5-1 Earth-Mars-Earth flyby mission in 2018

Inspiration Mars Foundation [2013] identified a 12-day launch window between the 14th of December, 2017 and the 4th of January, 2018 using classical chemical propulsion. This launch window takes into account a fully margined payload mass, explained in Section 9-3, of 19 tons [Inspiration Mars Foundation, 2013]. Based on this window, the first nominal scenario has been identified in Copernicus in Table 9-3.

Using low-thrust SEP, this launch window can be substantially increased for a payload mass of 19 tons using the SLS/LUS 4xRL10C2 launch configuration. Remember that this payload

mass does not include the mass of the propulsion subsystem. Figure 9-2 shows the grid search which has been performed around the nominal scenario. On the horizontal axis, the variation of the launch date with respect to the launch date of the nominal scenario has been plotted. On the vertical axis, the flyby date has been plotted with respect to the nominal scenario launch date. The color scale represents the minimized launch mass in tons for each combination of launch and flyby dates. The largest dot represents the combination of launch and flyby date resulting in the lowest minimized launch mass. As can be seen from Figure 9-2, this launch window would be increased from 12 to around 57 days between the 8th of December, 2017 and the 3th of February, 2018, using only 10 kW.



Figure 9-2: Launch opportunities for Case 1: 2018 - 10 kW - 19 tons - 4RL10.

The final mass at Earth arrival, which is the sum of the payload mass and the mass of the SEP system, is fixed for a certain grid search. For Case 1, the fixed final mass is 19.3 tons of which 19 tons payload mass and 300 kg SEP subsystem mass for 10 kW as can be seen in Table 9-4. Since this final mass at Earth arrival is fixed, the difference in minimized launch mass in Figure 9-2 can be attributed to the difference in required propellant mass for each launch-flyby date combination. One can see from Figure 9-2 that most data points have a launch mass quite close to the fixed final mass of 19.3 tons, indicating that there are hardly any manoeuvres and hence propellant required. This does not really comes as a surprise, since the reference trajectory from Copernicus in 2018 is a free-return trajectory. An example of such a trajectory for $t_{dep}=3$ and $t_{flyby}=233$ with hardly any manoeuvres can be seen in Figure 9-3.



Figure 9-3: Low ΔV trajectory for t_{dep} =3 and t_{flyby} =233 for Case 1: 2018 - 10 kW - 19 tons - 4RL10.

Some data points in Figure 9-2 however are further away from this 19.3 tons. Those data points represent trajectories that require more manoeuvres. An example of such a trajectory for $t_{dep}=29$ and $t_{flyby}=245$ has been plotted in Figure 9-4a. In Figure 9-4b, one can also observe the effect of the coast arc near the Martian flyby. Around 0.65 years, the coast arc is clearly visible.



Figure 9-4: High ΔV trajectory for t_{dep} =29 and t_{flyby} =245 for Case 1: 2018 - 10 kW - 19 tons - 4RL10.

The launch window could be much larger if it is assumed that the vehicle can be constructed much closer to the baseline payload mass than the fully margined payload mass. To illustrate the potential impact of this, Case 2 is identical to Case 1 but instead uses the baseline payload mass of 13139 kg [Inspiration Mars Foundation, 2013] as can be seen in Table 9-4. The launch window could then be increased to 124 days between the 10th of October, 2017 and the 11th of February, 2018, again using only 10 kW. This can be seen in Figure 9-5.



Figure 9-5: Launch opportunities for Case 2: 2018 - 10 kW - 13.139 tons - 4RL10.

The previous two cases require the more complex SLS/LUS 4xRL10C2 upper stage. It is not entirely sure that this upper stage will be finished by 2018. Therefore, it has been investigated if the baseline SLS/iCPS 1xRL10B2 upper stage configuration can be used for this mission. Using the baseline payload mass of 13139 kg and using only 10 kW (Case 3 in Table 9-4), the launch window for this configuration would become 44 days between the 25th of December, 2017 and the 7th of February, 2018, as can be seen in Figure 9-6.



Figure 9-6: Launch opportunities for Case 3: 2018 - 10 kW - 13.139 tons - 1RL10.

44 days is still a relative big launch window. Therefore, the payload mass has been gradually increased to identify what the largest mass would be with a sufficient launch window of at

least 20 days. For a payload mass of 13.5 tons, the launch window has decreased to 33 days between the 1^{st} of January, 2018 and the 3^{th} of February, 2018. For a payload mass of 13.75 tons, this decreases to a 21 day launch window between the 16^{th} of January, 2018 and the 6^{th} of February, 2018. For a payload mass of 14 tons, no launch window at all could be identified. This relation between payload mass and size of the launch window behaves non-linear.

One could further increase the payload mass if the available power for the SEP system is higher. Considering the fact that NASA states that the top technical challenge for in-space propulsion is the development of high-power electric propulsion system technologies to enable high ΔV missions with heavy payloads [Meyer et al., 2012], it is not unreasonable to assume that power levels beyond 10kW will be achievable in the near future. As an example, a payload mass of 15 tons with 25 kW of available power (Case 4 in Table 9-4), is used. As can be seen in Figure 9-7, a launch window of 36 days between the 10th of January, 2018 and the 15th of February, 2018 can be opened up. This payload mass can be increased up to 15250 kg for a launch window of 22 days between the 16th of January, 2018 and the 7th of February, 2018.



Figure 9-7: Launch opportunities for Case 4: 2018 - 25 kW - 15 tons - 1RL10.

9-5-2 Earth-Mars-Earth flyby mission in 2019

First of all, it has been investigated if the baseline SLS/iCPS 1xRL10B2 upper stage configuration can be used for this mission. However, the 1RL10 configuration could not provide enough energy to even launch the baseline payload mass of 13139 kg. In order to establish how high the payload mass could be while still having a feasible launch window of at least 20 days, several cases have been run. Using 10 kW, the maximum payload mass would be 9700 kg for a launch window of 23 days. This mass is not even close to the required 13139 kg. Therefore, the same scenario has been run with a higher power level. Still, using a power level of 25 kW, the maximum achievable payload mass is 11830 kg with a 24-day window between the 1st of November, 2019 and the 25th of November, 2019.



Figure 9-8: Launch opportunities for Case 5: 2019 - 10 kW - 13.139 tons - 4RL10.

As this shows that the 1RL10 configuration has great difficulty with the 2019 scenario, cases using the 4RL10 configuration have been run. For 13139 kg and using 10 kW, the launch window becomes 108 days between the 22^{nd} of November, 2019 and the 9th of March, 2020 as can be seen in Figure 9-8. Hence, there is quite some margin to increase the payload mass. Therefore, the payload mass has been gradually increased until a launch window of approximately 20 days was encountered. This was the case for 15750 kg, in which a 20-day launch window has been identified between the 1st of February, 2020 and the 21st of February, 2020. Note that this is still included under the umbrella "2019 launch window".



Figure 9-9: Launch opportunities for Case 6: 2021 - 25 kW - 13.139 tons - 4RL10.

9-5-3 Earth-Mars-Earth flyby mission in 2021

Much like the 2019 scenario, the 1RL10 configuration has great difficulty with the 2021 scenario. Therefore, cases using the 4RL10 configuration have been investigated. It was found to be impossible to launch the baseline payload mass of 13139 kg using 10 kW. This can be attributed to the large required DSM's, as can be seen from Table 9-3. These DSM's can be translated into a substantial ΔV requirement for the SEP system. Hence, a higher power level is required to enable these high ΔV 's. Therefore, the power level had to be increased to 25 kW to make this scenario feasible. In the end, a launch window of 33 days can be observed in Figure 9-9 between the 20th of November, 2021 and the 23th of December, 2021. If the payload mass is increased to 13.3 tons, the window becomes 22 days between the 26th of November, 2021 and the 18th of December, 2021.

9-6 Conclusion

During this research, it has been shown that SEP can be used to significantly improve crewed flyby missions of Mars. Using a modest amount of SEP, less performance is required by the upper stage of the launch vehicle, giving more margin for development of the system, and possibly allowing for the use of a smaller upper stage altogether. All of the results so far have been based on existing versions of the software.

In order to give a quick overview of the capabilities of SEP, the combination of parameters that lead to preliminary launch windows of 20 days have been listed in Table 9-5. Note that the last column for the payload mass using chemical propulsion has been calculated by translating the required C_3 for the nominal scenario's in Table 9-3 into the maximal launch mass using Equation 9-2. Using the DSM's in Table 9-3 and assuming a specific impulse of 340 s, the maximal launch mass can be translated into maximal payload mass.

Scenario	Power level	Launcher configuration	Payload mass	Payload mass
	(kW)		low-thrust (ton)	chemical (ton)
2018	10	SLS/LUS 4xRL10C2	21.5	20.95
2018	10	SLS/iCPS 1xRL10B2	13.75	12.55
2018	25	SLS/iCPS 1xRL10B2	15.25	12.55
2019	10	SLS/iCPS 1xRL10B2	9.70	5.38
2019	25	SLS/iCPS 1xRL10B2	11.83	5.38
2019	10	SLS/LUS 4xRL10C2	15.75	10.49
2021	25	SLS/LUS 4xRL10C2	13.30	8.42

Table 3-3. LIMILING CASES. 20-GAVS IAUNCH WINDOW TOP NACE TOP OF JULIA	uavs.
--	-------



Figure 9-10: Comparison maximal payload masses for chemical and low-thrust trajectories for Earth-Mars-Earth missions in 2018, 2019 and 2021 for different power levels and different launch configurations with a fixed TOF of 501 days.

The results in Table 9-5 have also been visualized in Figure 9-10. In this figure, the payload mass for the low-thrust and for the chemical reference trajectories have been plotted on the x- and y-axis respectively. A black line has been drawn that indicates where both masses are equal. If a point lies on the right side of this line, it shows that the payload mass for the low-thrust scenario is higher than the one for the chemical reference trajectory. From this figure, one can see that the trajectories utilizing SEP have a consistently higher feasible payload mass than the chemical trajectories. One can also see that the 4RL configuration results in larger payload masses than for the 1RL configuration. This does not comes as a surprise since the maximum launch mass in the launch performance curve of the 4RL configuration is substantially higher than the 1RL configuration. In this figure, one can also see that for the investigated scenarios, an increase in power level also results in a higher low-thrust payload mass. This is not always the case, as will be explained in Subsection 19-1-2.

Hence, this study has demonstrated that SEP improves the launch period and mission performance of a mission to flyby Mars and return within 501 days. Many parameters may still be adjusted, which may open up further mission concepts and improve the performance even more. This warrants further study, that would undoubtedly create even further improvements on those discussed here.

Chapter 10

Thesis research goals

Now that the status and the methods of the research performed during the author's internship have been explained, the research goals will be established based on the experience gained during this internship. During this internship, a basic version of the tool has been written and has been used to perform a first analysis.

This previously conducted research was limited to grid searches utilizing only one fixed time of flight, as the code was not capable of optimizing time. However, there may be a significant potential for improvement when the mission duration is allowed to change. So far, it has always been kept fixed at 501 days according to the rules of Inspiration Mars [Tito et al., 2013], but allowing variable flight time might create more opportunities. Therefore, the capability to optimize the epochs of important events should be added. Besides improving the current results, different mission concepts could be investigated. For instance, an additional Venus flyby could be introduced to increase the scientific return of the mission and to open up new launch windows.

During this thesis project, both potential improvements will be investigated. In order to facilitate this pursuit, three research goals can be set up. Each of them will be shortly described in the next three sections.

10-1 Research goal 1

Although the run-time of the program has always been kept into account during the development of the code during the internship, the limited duration of the internship and the time pressure of the paper deadline prevented the author from optimizing the entire code for run-time. Therefore, the first research goal is to profile the existing code, identify bottlenecks and remove them. Throughout this effort, changes will be implemented to facilitate the addition of extra legs to the trajectory. Furthermore, a promising different representation of the Sims-Flanagan transcription will be implemented and will be compared to the current representation.

10-2 Research goal 2

The second research goal is to expand the written program to include the optimization of the control nodes' times. This enables the creation of grid searches with the launch date on the x-axis and the total time of flight on the y-axis. For each grid point, the Martian flyby date will then be optimized.

10-3 Research goal 3

The changes to facilitate the addition of extra legs will help for the third research goal: the automation of the addition of extra legs. This will be tested through the optimization of a three-leg Earth-Venus-Mars-Earth mission. The main goal here is again to enable grid searches with the launch date on the x-axis and the total time of flight on the y-axis. For each grid point, the Martian and Venus flyby dates will then be optimized.
Part III

Research goal 1: Improvements to previous work

Chapter 11

Profiling

In this chapter, the basic version of the tool will be profiled. This has been done using the free program VerySleepy [Codersnotes.com]. This program identifies how much time is spent in each (sub-)function of the tool. Do note that this profiling increases the run time. As such, the resulting run times are not entirely representative. However, the ratios of the time spent in the sub-functions are, which enables the determination of so-called **bottlenecks**: inefficient points in the code that require a large amount of processing time.

First of all, such bottlenecks will be identified for a one-leg scenario. Next, those bottlenecks will be removed by implementing several adaptations to the basic version of the tool. The established adaptations will then be applied to a two-leg scenario to see if similar improvements can be observed. To avoid drawing conclusions based on only 1 one-leg and 1 two-leg scenario, a grid search will be performed around the nominal two-leg scenario.

11-1 One-leg scenario

First of all, a basic version of the tool will be profiled. It was decided to profile a oneleg rendez-vous scenario with Mars. This discussion will start by explaining and profiling a nominal test case, after which the program's function structure encountered in the profile tree will be explained. From the nominal scenario profile tree, several bottlenecks will be identified. In the next subsections, adaptations to the code are explained which remove those bottlenecks. Finally, a comparison of the actual run times for the tool using the implemented adaptations will be given and discussed.

11-1-1 Nominal one-leg scenario

The case that was investigated for this scenario is a one-leg rendez-vous scenario with Mars, inspired by the recent MAVEN mission. Both the launch and arrival date were modeled after MAVEN and are the 1^{st} of December, 2013 and the 22^{nd} of September, 2014 respectively. The



Figure 11-1: Thrust profile and trajectory for the nominal MAVEN reference mission.

hyperbolic excess velocity at the launch date was initially set to [116,-6176,-716] m/s, also based on MAVEN. The arrival hyperbolic excess velocity was set constant to [0,0,0] m/s as it is a rendez-vous mission. The power level was set to 50 kW, to ensure a feasible trajectory. The specific impulse was set constant to 2000 s. There is a one-week coasting arc after launch and a two-week coasting arc before Mars arrival for launch checkout and pre-arrival operations. Although it was not used for the MAVEN launch, the Advanced Upper Stage of the SLS was used, as this launch vehicle curve was already available in the tool. The resulting thrust profile and trajectory for this reference mission based on MAVEN can be found in Figure 11-1.

In Figure 11-2, one can see the profiling tree of this scenario. At the top of this profiling tree, functions related to the SNOPT optimization procedure such as s8iqp and s6srch can be found. The latter function is the most interesting for this profiling; this is the function in SNOPT that calls the *usrfg* function. As explained in Chapter 5, within this function, the state vector is propagated, the constraints are calculated and the derivatives of each constraint with respect to each state vector element are calculated. The most computationally



Figure 11-2: Profiling of the nominal MAVEN mission: listing of run time per (sub)function.

intensive functions within the *usrfg* function have been listed. The most time consuming function is *StateVectorPropagation*. This function gets the initial state vector elements and propagates them. As the constraint calculations hardly consume any time, they are not included in this profiling tree. The same holds for the derivatives with one major exception: the derivatives with respect to time are obtained using a finite-difference method for reasons that will be explained in Chapter 13. Therefore, a variant on the *StateVectorPropagation*, *StateVectorPropagation2* has been written.

As the StateVectorPropagation and StateVectorPropagation2 functions are the most computationally intensive, the structure within those functions will be explained. The function StateVectorPropagation calls the NumericalPropagator function twice: once for a forward propagation and once for a backward propagation. In this NumericalPropagator function, several dynamic arrays are (de-)allocated using new and free, the trajectory is propagated between the different manoeuvres using IntegratorFunction, the manoeuvres are applied and the mass is updated. A similar function NumericalPropagator2 has been written based on this function, but specifically tailored for the finite-difference method with respect to time.

As can be seen from the profiling tree, both NumericalPropagator and NumericalPropagator2 call the same IntegratorFunction. This IntegratorFunction checks if it is a forward or backward integration and if the final time for the integration has been reached or not. Depending on the answers to those questions, it calls the Dopri8Integrator, which does the actual numerical integration using the DOPRI8 scheme explained in Subsection 6-3-1. The Dopri8Integrator calls the STMPropagatorFunction, which calculates the derivatives of the STM, coordinates and velocity elements, explained in Subsection 6-3-2.

From the profiling tree in Figure 11-2, already some bottlenecks can be identified. A first one is located within the *IntegratorFunction*. There, a large amount of time is required in the *new* and *delete* functions for the allocation and de-allocation of arrays.

A second bottleneck is the large amount of time spent in the *NumericalPropagator* and *NumericalPropagator2* functions to allocate and de-allocate dynamic arrays.

A third bottleneck is the *pow* function that is called in the *DOPRI8Integrator* function. *pow* is the standard function in C++ to raise a number to a certain power. For instance, x^2 in C++ is pow(x, 2.0). Although being the standard method in C++, it is not efficient.

11-1-2 Adaptation 1: Array allocation moved up in function hierarchy

The first bottleneck was identified to be the (de-)allocation of arrays in the *IntegratorFunction*. This could be traced to the (de-)allocation of a specific array every time the *IntegratorFunction* is called. This (de-)allocation could be moved one level higher up in the function hierarchy to the *NumericalPropagator* and *NumericalPropagator2* functions.

As a result, the time spent in *new* and *free* within the *IntegratorFunction* has been removed. However, the time spent in *new* and *free* in the *NumericalPropagator* and *NumericalPropagator2* functions has been slightly increased. As the number of calls to the *NumericalPropagator(2)* function is lower than the number of calls to the *IntegratorFunction*, the arrays are (de-)allocated less often. Hence, the overall effect is a speed up of the *NumericalPropagator* and *NumericalPropagator2* functions.

11-1-3 Adaptation 2: Array allocation moved out of loop

The second bottleneck was identified to be the large amount of time spent in the *Numerical*-*Propagator* and *NumericalPropagator2* function to allocate and de-allocate dynamic arrays. This bottleneck was traced back to the allocation of several arrays within a loop that goes through the different propagation steps. Those (de-)allocations have been moved out of the loop. Hence, they only had to be allocated and de-allocated once.

The effect of this adaptation is a large reduction in the amount of time spent for the allocation and de-allocation of those arrays within the NumericalPropagator and NumericalPropagator2 functions.

11-1-4 Adaptation 3: Avoid usage of pow function

The third identified bottleneck was the time spent in the *pow* function in the *DOPRI8Integrator* function. However, the profiler is mistaken here. The *DOPRI8Integrator* does not call the *pow* function. The *STMPropagatorFunction*, in which the derivatives of the STM, coordinates and velocity elements are calculated, does.

In this function, pow has been replaced by multiplications. It was realized that removing pow was not the only improvement that could be made to the propagation function. As the

terms x^*x , y^*y , z^*z , x^*y , x^*z and y^*z are calculated several times, it was decided to create variables for each of them. This reduces the time spent for multiplication of those elements. Furthermore, it was realized that it is more efficient to replace r^*r by a separate variable. As such, r^5 can be represented as $r_2 * r_2 * r$ instead of r * r * r * r * r * r with $r_2 = r * r$.

As a result, the time required for the *pow* function is entirely removed. There is a small increase in time spent in the *STMPropagatorFunction* as the multiplications are now attributed to this function, while before adaptation 3, they were attributed to the *pow* function in the *DOPRI8IntegratorFunction*. The overall effect is a reduction of the time spent in the *DOPRI8Integrator* function.

11-1-5 Adaptation 4: Merge multiple arrays into memory pool

Although the main bottlenecks have been identified, it was thought that the (de-)allocation of the arrays in the NumericalPropagator and NumericalPropagator2 function could be made even more efficient. Within the NumericalPropagator function, 27 equally-sized arrays are allocated that are passed on to the IntegratorFunction. Since those arrays are always all passed on at the same time, they could all be combined into one single, larger array. As such, not 27 different arrays have to be allocated, passed on to sub-functions and be de-allocated, but just one array. One disadvantage of this grouping of multiple arrays into one single, larger array is that many multiplications for indexing have to be performed. For instance, if one wants to access the jth element of the 25th array called y, instead of calling y[j], one needs to call k[24*SIZE_ONE_ARRAY +j).

The time required for *new* and *free* in the *NumericalPropagator* function has been substantially reduced. However, there is a substantial increase in the *DOPRI8Integrator* function, as expected due to the multiplications performed for the indexing within this function.

11-1-6 Adaptation 5: Move step size initialization

While changing the program code required for adaptation 4, another bottleneck was identified. It was realized that every time the *IntegratorFunction* is called, the time step used for the DOPRIS integration procedure was reset to its initial value. This means that each propagation step within a forward or backward propagation starts with the resetted time step, after which this time step is updated according to the variable step size algorithm utilized in DOPRI8. Afterwards, the step size is again reset when a new propagation step is started. This is not as efficient as it could be as for consecutive propagation steps within a forward or backward propagation, a similar step size could be used. Therefore, the time step should only be reset if a new forward or backward propagation is started. This means that the resetting of the time step should be moved from the IntegratorFunction to the start of the NumericalPropagator and Numerical Propagator2 functions. The effect of this adaptation has been visualized in Figure 11-3. It is clear that after the initialization during the first 3 propagation steps, there is a difference in the number of propagation steps required for the two different methods. For the old method where the time step is reset at the beginning of each new propagation step, every single propagation step requires 7 steps. However, for the new method which remembers the propagation time step from the previous propagation step, it only requires 2



Figure 11-3: Visualization of the different time step systems.

steps per propagation step for this specific case. As such, the *DOPRI8Integrator* is not called as often as it used to be, which explains why the latter method is more efficient.

The time spent in the *IntegratorFunction* is reduced as the number of *DOPRI8Integrator* calls is drastically reduced. As such, the time spent in the *DOPRI8Integrator* is approximately halved for this specific scenario.

11-1-7 Time comparison

The run time of different versions of the tool with different adaptations active have been listed in Table 11-1. Do note that the times listed here are actual run times that were obtained without the interference of the VerySleepy profiler program.

Due to small accuracy differences between the pow and multiplication operations in C++, small differences in the propagated states occur if adaptation 3 is active. As such, the number

Adaptations	Total	Objective	Major	Calls to
active	time [s]	function	iterations	usrfg
None	22.7	0.088841865	279	510
1	19.2	0.088841865	279	510
1 and 2	6.6	0.088841865	279	510
1, 2 and 3	6.1	0.088839385	343	581
1, 2, 3 and 4	4.8	0.088839385	343	581
1, 2, 3, 4 and 5	4.1	0.088837828	399	667

Table 11-1: Comparisons of the effect of the different adaptations for a one-leg scenario.

of iterations required to perform the optimization changes. It depends on the exact scenario if more or less iterations are required. For this particular scenario, more iterations had to be done. Independent of an increase or decrease in the number of iterations, the time required per iteration or call to the usrfg function is smaller using this adaptation, which proves that the usrfg function itself is sped up due to adaptation 3.

The same reasoning can be applied to adaptation 5: small differences in the propagated states result in an increase or decrease in the calls to usrfg. Independent of an increase or decrease in the number of iterations or calls to usrfg, the time required per iteration or call to the usrfg function is smaller using this adaptation, which proves that the usrfg function itself is sped up due to adaptation 5.

11-2 Two-leg scenario

Until now, the previously mentioned improvements due to the described adaptations have only been tested on one-leg scenario's. In this section, it will be shown that the same adaptations also improve two-leg scenario's. First of all, the reference two-leg scenario will be explained, followed by a comparison of the actual run times for the tools using the implemented adaptations.

11-2-1 Nominal two-leg scenario

The case that was investigated for this scenario is a two-leg Mars flyby mission, inspired by the previously conducted Inspiration Mars based research. The scenario for this two-leg case is based on the 2018 launch window identified in De Smet et al. [2014]. The launch, flyby and arrival dates are the 2nd of February, 2018, the 7th of September, 2018 and the 21st of May, 2019 respectively. The power level was set to 10 kW and the payload mass was set to be 19 tons. The launch vehicle model is based on the SLS 4xRL10, advanced upper stage launcher. The specific impulse was set constant to 2000 s. There is a one-week coasting arc before Earth arrival for launch checkout, pre-flyby and pre-arrival operations. The resulting thrust profile and trajectory can be found in Figure 11-4.

11-2-2 Time comparison

Table 11-2 gives an overview of the time gains associated with the different adaptations. Comparing this with Table 11-1, one can see that there is less gain achieved for the two-leg case. However, this can be explained by looking at Figure 11-5, which shows the profile tree for the nominal two-leg scenario. One can see that roughly half of the time is spent in the s8iqp function instead of approximately 8% for the nominal one-leg scenario in Figure 11-2. This difference can be linked to the difference in number of infeasible iterations. For the one-leg scenario, the optimizer almost immediately finds a feasible trajectory. The two-leg scenario remains much longer on an infeasible iteration. As such, the time spent in the s8iqp function, which tries to minimize the infeasibilities, is much higher in the two-leg than for the one-leg reference case.



Figure 11-4: Thrust profile and trajectory for the nominal two-leg Inspiration Mars reference mission.

 Table 11-2:
 Comparisons of the effect of the different adaptations for a two-leg scenario.

Adaptations	Total	Objective	Major	Calls to
active	time [s]	function	iterations	usrfg
None	19.8	0.53352949709	432	675
1	19.3	0.53352949709	432	675
1 and 2	18.8	0.53352949709	432	675
1, 2 and 3	14.8	0.53352949683	410	583
1, 2, 3 and 4	14.9	0.53352949683	410	583
1, 2, 3, 4 and 5	13.0	0.53352951115	400	736



Figure 11-5: Profiling of the nominal two-leg Inspiration Mars reference mission.

As for the one-leg nominal scenario, almost 92 % is spent in the self-written function *usrfg* and only about 50 % for the two-leg nominal scenario, it is logical that the adaptations in the one-leg scenario appear to have a bigger impact than for the two-leg scenario.

One can see that much like for the one-leg scenario, every adaptation reduces the total run time, except for adaptation 4. The reason is that although the time spent in the new and free function is reduced with adaptation 4 active, the gain here is smaller than the time increase in the *DOPRI8Integrator* function due to the multiplications that have to be done for the indexing. As such, adaptation 4 is not beneficial for this specific scenario.

11-3 Grid searches

The previous two sections prove that the adaptations are advantageous for both a one-leg and a two-leg scenario. However, only 1 one-leg and 1 two-leg scenario have been investigated. It could be that the realized time gains were just a matter of luck. To eliminate this element of luck, it has been decided to apply the different versions of the tool to a two-leg scenario similar to the one described in the previous section. However, instead of looking at only one specific launch date - flyby date - arrival date combination, a grid search was performed in which the launch and flyby dates were systematically varied, from which the arrival date followed automatically as the total time of flight was kept constant at 501 days. In doing so, it was shown that the speed ups achieved from the different adaptations are not just based on mere luck, but are consistent. The results have been listed in Table 11-3 where also the optimal departure and flyby date have been listed and the corresponding minimal launch mass.

Adaptation active	Time	Minimal launch	Optimal departure	Optimal flyby
	[min]	mass [tons]	date [days]	date [days]
Nominal	31.7	19.31439	10	232
1	31.0	19.31439	10	232
1 and 2	27.2	19.31439	10	232
1, 2 and 3	20.2	19.31440	10	232
1, 2, 3 and 4	19.6	19.31440	10	232
1, 2, 3, 4 and 5	19.5	19.31443	10	232

Table 11-3: Time comparison for grid searches around two-leg scenario.

Although the time gains due to the different adaptations vary throughout the performed grid searches, it is clear that the gains identified in the previous two sections are not just mere luck. Due to variations within the grid, the percentage of time gained depends on the actual grid point. Therefore, no definite number can be attributed to these savings, only on an individual basis. However, Table 11-3 gives an overview of the time in minutes required to do the entire grid search. From this, one can conclude that the overall differences in minimized launch mass are negligible and that the effect on a two-leg scenario is larger than what Table 11-2 would suggest. The main reason is that not all points in the grid search spent a large amount of time in the s8iqp function minimizing the infeasibilities, as was the case for the specific data point used to build Table 11-2. If more time is spent in the usrfg function instead of in the s8iqp function, the effect of the adaptations is larger.

The gain from adaptation 5 is marginal in this grid, which was unexpected. Therefore, the grid was further examined to see what caused this smaller than expected gain. It was found that for one grid point, the small differences in the propagation due to adaptation 5 resulted in a numerical instability, which led the optimization tool SNOPT to iterate until the maximum allowed number of iterations was achieved. As such, this data point took 130 seconds longer with adaptation 5 active. Without it, this grid search would be 2.2 minutes faster than the case where only adaptation 1 up to 4 are active.

11-4 Conclusion

In general, the identified adaptations considerably reduce the run time of the program. In some cases however, the processing time is increased. This can be traced back to small accuracy differences in the propagated state vector for the different adaptations. These small accuracy differences result in a different number of iterations required for the SNOPT optimization tool to converge. If this required number of iterations is significantly larger, it might be that the time gain of an adaptation is canceled out by the additional run time required for the larger number of iterations. This can lead to an overall larger run time.

The general trend however is a consistent reduction in run time if the identified adaptations are applied to the program, as shown by the performed grid searches. Therefore, the implemented adaptations will remain active for the remainder of this thesis research.

Chapter 12

Throttled versus thrust representation

It must be ensured that the optimal solution does not require more thrust than the SEP system can provide. Therefore, in Subsection 7-2-2, so-called thrust constraints were introduced. Remember that these thrust constraints were dependent on the mass of the spacecraft at the midpoint of each segment. For segment j, this mass depends on the manoeuvres i < j for the forward thrust constraints and on the manoeuvres $i \ge j$ for the backward thrust constraints. As such, the thrust constraints have dense Jacobian entries corresponding to the ΔV decision variables in the previous time steps [Ellison et al., 2013].

In this chapter, a different representation of the Sims-Flanagan transcription will be evaluated. First of all, this new transcription called the **throttled representation** will be explained and its effect on the sparsity of the Jacobian will be visualized. Then, the changes in the Jacobian will be derived. This discussion will be followed by a comparison of the throttled and classical thrust representation for a single-leg trajectory.

12-1 Throttled representation

The dense entries corresponding to the ΔV decision variables in the previous time steps can be avoided by using a slightly different representation of the Sims-Flanagan transcription. Instead of representing a manoeuvre using $[\Delta V_T, \Delta V_N, \Delta V_C]$ and using thrust constraints, one could represent the manoeuvre by using a throttle vector $[u_T, u_N, u_C]$ and throttle constraints [Ellison et al., 2013]. These non-dimensional throttle parameters are scaled by the maximum ΔV that the spacecraft can apply during that segment: ΔV_{max} .

$$\Delta V_{\kappa} = u_{\kappa} \cdot \Delta V_{\text{max}}$$
 with $\kappa = \text{T}$, N, C

The maximum ΔV that can be applied at a segment depends on whether or not the power is dependent on the heliocentric distance. If it is independent of the heliocentric distance,

$$\Delta V_{\max,j} = \frac{2\eta_{\text{jet}} P_0 DT}{I_{sp,j} g_0 M_i^-} \tag{12-1}$$

If it is dependent on the heliocentric distance, the following equation applies:

$$\Delta V_{\max,j} = \frac{2\eta_{\rm jet} P_0 DT A U^2}{I_{sp,j} g_0 M_j^- R_j^2}$$
(12-2)

where R_j and M_j^- are the heliocentric distance of the spacecraft and the mass of the spacecraft before the manoeuvre is applied.

This formulation is straightforwardly applicable to the forward propagation. Using the initial node conditions, $\Delta V_{\max,j}$ can be calculated, from which the ΔV vector in the TNC coordinate system can be calculated using Equation 12-1. This can be converted into a Cartesian ΔV vector using Equation A-1, which can then be used to update the Cartesian state and mass at the manoeuvre point after application of the manoeuvre using Equations 6-16 and 6-18.

However, for the backward propagation, the mass of the spacecraft before the application of the manoeuvre is unknown, but depends on $\Delta V_{\max,j}$. In Equation 12-1 and 12-2, one can see that this $\Delta V_{\max,j}$ also depends on this mass. Rewriting Tsiolkovsky's law for the last manoeuvre 2N for the constant power case, one gets.

$$M_{2N^{-}} = M_{2N^{+}} \cdot \exp\left(\frac{\sqrt{u_{T,2N}^{2} + u_{N,2N}^{2} + u_{C,2N}^{2}}}{I_{sp,2N} \cdot g_{0}} \cdot \frac{2\eta_{\text{jet}}P_{0}DT}{I_{sp,2N} \cdot g_{0}M_{2N}^{-}}\right)$$
(12-3)

Solving for $M_{2N^{-}}$, one gets

$$M_{2N^{-}} = \frac{\sqrt{u_{T,2N}^{2} + u_{N,2N}^{2} + u_{C,2N}^{2}}}{I_{sp,2N} \cdot g_{0}} \cdot \frac{2\eta_{jet}P_{0}DT}{I_{sp,2N} \cdot g_{0}} \cdot (12-4)$$

$$W^{-1} \left(\frac{\sqrt{u_{T,2N}^{2} + u_{N,2N}^{2} + u_{C,2N}^{2}}}{I_{sp,2N} \cdot g_{0}} \cdot \frac{2\eta_{jet}P_{0}DT}{I_{sp,2N} \cdot g_{0}} \cdot \frac{1}{M_{2N^{+}}}\right)$$

where W^{-1} is the inverse of the **Lambert's function** also known as the **omega function** or **product logarithm**, which needs to be calculated using an iterative procedure [Lambert, 1758]. The result of this iterative procedure, the mass before the application of the manoeuvre, can be used to calculate $\Delta V_{\max,j}$. From this, the ΔV vector in the TNC coordinate system can be calculated using Equation 12-1. This can be converted to a Cartesian ΔV vector using Equation A-1, which can then be used to update the Cartesian state and mass at the manoeuvre point before application of the manoeuvre using Equations 6-17 and 6-19.

Instead of using thrust constraints, throttle constraints can be used for this representation of the Sims-Flanagan transcription.

$$F_{\text{throttle}} = \sqrt{u_T^2 + u_N^2 + u_C^2} \in [0, \text{duty cycle}]$$
(12-5)

12-2 Comparison of the sparsity of the Jacobian

The throttle constraints on each segment in Equation 12-5 only depend on the throttle parameters of that segment. As such, the dependency on other manoeuvres has been removed. This can also be seen from the mapping of the Jacobian in Figures 12-1 and 12-2 where the red squares indicate Jacobian entries that are always equal to zero.



Figure 12-1: Sparsity pattern of the Jacobian for a single-leg case with 20 segments, thrust constraints.



Figure 12-2: Sparsity pattern of the Jacobian for a single-leg case with 20 segments, throttle constraints.

Comparing the step structure in Figure 12-1 to the block structure in Figure 12-2, one can see that the dependency on other manoeuvres in Equations 7-4 and 7-5 has been removed. Furthermore, the initial and final control node conditions no longer affect the throttled constraints as can be seen from Equation 12-5. However, the seven match point constraints now depend on every single variable. For the thrust constraints, the specific impulses did not affect the six state match point constraints nor did the initial, final and SEP system masses. In addition, the mass match point now depends on the initial and final states and times. From the previous figures, it can be deduced that the throttle constraint representation leads to a much sparser Jacobian.

12-3 Jacobian of the throttled representation of the Sims-Flanagan transcription

Not only do the location of the Jacobian entries vary for the different representations, also the derivatives within these entries change. These differences will be explained in this section. First of all, it will be shown how a change in the initial and final state, initial and final mass of the spacecraft and the mass of the power subsystem affects the state and mass of the spacecraft at the match point. This will be followed by an explanation on how changes in the throttle vector elements and the specific impulse of each manoeuvre are propagated up to the match point. Finally, the derivatives of the throttle constraints with respect to the throttle parameters will be derived.

12-3-1 Derivatives of the match point constraints with respect to the initial and final masses, coordinates, velocities and hyperbolic excess velocities, and the mass of the SEP system

To find the derivatives of the match point constraints, first of all, it must be understood how a change in the initial state and mass and power level is propagated up to the match point for the forward propagation. Similarly, it must be understood how a change in the final state and mass and power level is propagated up to the match point for the backward propagation. This will be explained in the next two paragraphs, followed by a paragraph that shows how the results from those propagations can be used to find the derivatives.

Forward propagation If the initial state, mass and initial power level change, also the mass and state of the spacecraft before application of manoeuvre 1 change. It can be shown that

where $\Phi_{1^-,0}$ is the STM between points 0 and 1⁻. This 8x8 matrix will from now on be called $\Phi_{\text{ext}_{1^-,0}}$.

This difference in state and mass of the spacecraft before the application of manoeuvre 1, along with the change in the mass of the SEP subsystem and as such the power of the SEP system, results in a change of $\Delta V_{\max,1}$. Also, the change in the state of the spacecraft results in a change in the directions of the TNC coordinate system. As such, a variant of the $TNC_{1^+,1^-}$ matrix must be set up to account for these changes. This matrix will be called $TNC_{\text{ext}_{1^+,1^-}}$ and has been derived in Appendix D-1.

Similar to Figure 8-1, Figure 12-3 shows how changes in the initial state and mass of the spacecraft and the power of the SEP system are propagated up to the match point.

L								
					I.			
		_ 1+		2^{-}	2^{+}		MP	
Δx_s	Δx_s	Δx_s		Δx_s	$\begin{bmatrix} \Delta x_s \end{bmatrix}$		Δx_s	
Δy_s	Δy_s	Δy_s		Δy_s	Δy_s		Δy_s	
Δz_s	Δz_s	Δz_s		Δz_s	Δz_s	ъ	Δz_s	
$\Delta \dot{x}_s \Phi_{\text{ext}_{1-0}}$	$\Delta \dot{x}_s$	$\Delta \dot{x}_s$	$\Phi_{\text{ext}_{2-1+}}$	$\Delta \dot{x}_s$	$\Delta \dot{x}_{s}$	$\Psi_{\text{ext}_{MP,2^+}}$	$\Delta \dot{x}_{s}$	
$\Delta \dot{y}_s \implies \dot{y}_s$	$\Delta \dot{y}_s$	$\Delta \dot{y}_s$	\Rightarrow	$\Delta \dot{y}_{s}$	$\Delta \dot{y}_{*}$	\Rightarrow	$\Delta \dot{u}_{s}$	
$\Delta \dot{z}_s$	$\Delta \dot{z}_s$	$\Delta \dot{z}_s$		$\Delta \dot{z}_{e}$	$\Delta \dot{z}_{a}$		$\Delta \dot{z}_{a}$	
ΔM	ΔM	ΔM		ΔM	ΔM		ΔM	
ΔM_{P_0}	ΔM_{P_0}	ΔM_{P_2}		ΔM_{P}	$\Delta M_{\rm P}$		$\Delta M_{\rm P}$	
0			1+		$\sum \sum P_0 $	2+		MP
		>		=	⇒			
	TNC	$ext_{1+.1-}$		TN	$C_{\text{ext}_{2^+,2^-}}$			

Figure 12-3: Propagation of a change in initial conditions and mass of the SEP system to the match point.

So,

$$\begin{bmatrix} \Delta x_{s} \\ \Delta y_{s} \\ \Delta z_{s} \\ \Delta \dot{x}_{s} \\ \Delta \dot{x}_{s} \\ \Delta \dot{y}_{s} \\ \Delta \dot{x}_{s} \\ \Delta M_{P_{0}} \end{bmatrix}_{\mathrm{MP}}$$

$$(12-7)$$

This matrix multiplication will be called $\Psi_{\text{ext}_{MP,0}}$.

Backward propagation Similar to the forward propagation, the extended version of the STM and the TNC matrices are required. The extended version of the STM can be set up in

exactly the same manner as for the forward propagation. However, the TNC extended matrix for the backward propagation is slightly different and has been derived in Appendix D-2.

Similar to the forward propagation, the matrix $\Psi_{\text{ext}_{MP,f}}$ is the matrix product of $\Phi_{\text{ext}_{MP,N+1^-}} \cdot TNC_{\text{ext}_{N+1^-,N+1^+}} \cdot \dots \cdot \Phi_{\text{ext}_{2N-1^+,2N^-}} \cdot TNC_{\text{ext}_{2N^-,2N^+}} \cdot \Phi_{\text{ext}_{2N^+,f}}$.

Calculation of the derivatives The matrices $\Psi_{\text{ext}_{MP,0}}$ and $\Psi_{\text{ext}_{MP,f}}$ can be used to find the required derivatives. The derivation is trivial. Therefore, the derivation will not be shown here but can be found in Appendix F.

12-3-2 Derivatives of the match point constraints with respect to the throttle parameters and the specific impulses

To find the derivatives of the match point constraints with respect to the throttle parameters and the specific impulses of each manoeuvre, it must be understood how a change in a forward throttle parameter and specific impulse is propagated up to the match point in the forward propagation. Similarly, it must be understood how a change in a backward throttle parameter and specific impulse is propagated up to the match point in the backward propagation. This will be explained in the next two paragraphs, followed by a paragraph that shows how the results from those propagations can be used to find the derivatives.

Forward propagation First of all, one must understand how a change in forward throttle parameters and specific impulses influences the state and mass after the manoeuvre has been applied. Therefore,

This matrix will be further referenced as $\Delta TNC_{\text{ext}_{1+,1}}$ and has been derived in Appendix E. This change in the state and mass after the application of the manoeuvre can then be propagated up to the match point using

$$\begin{bmatrix} \Delta x_{s} \\ \Delta y_{s} \\ \Delta z_{s} \\ \Delta z_{s} \\ \Delta \dot{x}_{s} \\ \Delta \dot{y}_{s} \\ \Delta \dot{x}_{s} \\ \Delta \dot{y}_{s} \\ \Delta \dot{z}_{s} \\ \Delta M \\ \Delta M_{P_{0}} \end{bmatrix}_{\mathrm{MP}} = \Phi_{\mathrm{ext}_{MP,N^{+}}} \cdot TNC_{\mathrm{ext}_{N^{+},N^{-}}} \dots \Phi_{\mathrm{ext}_{2^{-},1^{+}}} \cdot \begin{bmatrix} \Delta x_{s} \\ \Delta y_{s} \\ \Delta x_{s} \\ \Delta \dot{x}_{s} \\ \Delta \dot{y}_{s} \\ \Delta M \\ \Delta M_{P_{0}} \end{bmatrix}_{1^{+}}$$
(12-9)

So,

$$\begin{bmatrix} \Delta x_{s} \\ \Delta y_{s} \\ \Delta z_{s} \\ \Delta \dot{x}_{s} \\ \Delta \dot{y}_{s} \\ \Delta \dot{y}_{s} \\ \Delta \dot{x}_{s} \\ \Delta \dot{y}_{s} \\ \Delta \dot{x}_{s} \\ \Delta M \\ \Delta M_{P_{0}} \end{bmatrix}_{\mathrm{MP}} = \Psi_{\mathrm{ext}_{MP,1^{+}}} \cdot \Delta TNC_{\mathrm{ext}_{1^{+},1}} \begin{bmatrix} \Delta u_{T1} \\ \Delta u_{N1} \\ \Delta u_{C1} \\ \Delta I_{sp,1} \end{bmatrix}$$
(12-10)

where $\Psi_{\operatorname{ext}_{MP,1^+}} = \Phi_{\operatorname{ext}_{MP,N^+}} \cdot TNC_{\operatorname{ext}_{N^+,N^-}} \dots \Phi_{\operatorname{ext}_{2^-,1^+}}$.

$$\Psi_{\text{ext}_{MP,1+}} \cdot \Delta TNC_{\text{ext}_{1+,1}} = \begin{bmatrix} \frac{\partial x_{s,\text{MP f}}}{\partial u_{T,1}} & \frac{\partial x_{s,\text{MP f}}}{\partial u_{N,1}} & \frac{\partial x_{s,\text{MP f}}}{\partial u_{C,1}} & \frac{\partial x_{s,\text{MP f}}}{\partial I_{sp,1}} \\ \frac{\partial y_{s,\text{MP f}}}{\partial u_{T,1}} & \frac{\partial y_{s,\text{MP f}}}{\partial u_{N,1}} & \frac{\partial y_{s,\text{MP f}}}{\partial u_{C,1}} & \frac{\partial y_{s,\text{MP f}}}{\partial I_{sp,1}} \\ \frac{\partial z_{s,\text{MP f}}}{\partial u_{T,1}} & \frac{\partial z_{s,\text{MP f}}}{\partial u_{N,1}} & \frac{\partial z_{s,\text{MP f}}}{\partial u_{C,1}} & \frac{\partial z_{s,\text{MP f}}}{\partial I_{sp,1}} \\ \frac{\partial \dot{x}_{s,\text{MP f}}}{\partial u_{T,1}} & \frac{\partial \dot{x}_{s,\text{MP f}}}{\partial u_{N,1}} & \frac{\partial \dot{x}_{s,\text{MP f}}}{\partial u_{C,1}} & \frac{\partial \dot{x}_{s,\text{MP f}}}{\partial I_{sp,1}} \\ \frac{\partial \dot{y}_{s,\text{MP f}}}{\partial u_{T,1}} & \frac{\partial \dot{y}_{s,\text{MP f}}}{\partial u_{N,1}} & \frac{\partial \dot{y}_{s,\text{MP f}}}{\partial u_{C,1}} & \frac{\partial \dot{y}_{s,\text{MP f}}}{\partial I_{sp,1}} \\ \frac{\partial \dot{y}_{s,\text{MP f}}}{\partial u_{T,1}} & \frac{\partial \dot{y}_{s,\text{MP f}}}{\partial u_{N,1}} & \frac{\partial \dot{y}_{s,\text{MP f}}}{\partial u_{C,1}} & \frac{\partial \dot{y}_{s,\text{MP f}}}{\partial I_{sp,1}} \\ \frac{\partial M_{\text{MP f}}}{\partial u_{T,1}} & \frac{\partial M_{\text{MP f}}}{\partial u_{N,1}} & \frac{\partial M_{\text{MP f}}}{\partial u_{C,1}} & \frac{\partial M_{\text{MP f}}}{\partial I_{sp,1}} \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$(12-11)$$

The last row of this matrix is composed of zeros since the mass of the power subsystem is not affected by u_T , u_N , u_C and I_{sp} .

Backward propagation First of all, one must understand how a change in backward throttle parameters and specific impulses influences the state and mass before the manoeuvre has been applied. Therefore, similarly to the forward propagation, an extended matrix must be defined: $\Delta TNC_{\text{ext}_{2N}^{-},2N}$, which will be derived in Appendix E.

This change in the state and mass before the application of the manoeuvre can then be propagated up to the match point using

$$\begin{bmatrix} \Delta x_{s} \\ \Delta y_{s} \\ \Delta z_{s} \\ \Delta z_{s} \\ \Delta \dot{x}_{s} \\ \Delta \dot{y}_{s} \\ \Delta \dot{y}_{s} \\ \Delta \dot{x}_{s} \\ \Delta \dot{x}_{s} \\ \Delta \dot{x}_{s} \\ \Delta \dot{x}_{s} \\ \Delta M_{P_{0}} \end{bmatrix}_{\mathrm{MP}} = \Phi_{\mathrm{ext}_{MP,N+1^{-}}} \cdot TNC_{\mathrm{ext}_{N+1^{-},N+1^{+}}} \cdot \ldots \cdot \Phi_{\mathrm{ext}_{2N-1^{+},2N^{-}}} \begin{bmatrix} \Delta x_{s} \\ \Delta y_{s} \\ \Delta z_{s} \\ \Delta \dot{x}_{s} \\ \Delta \dot{y}_{s} \\ \Delta \dot{z}_{s} \\ \Delta M_{L} \\ \Delta M_{P_{0}} \end{bmatrix}_{2N^{-}}$$
(12-12)

So,

$$\begin{array}{c|c} \Delta x_{s} \\ \Delta y_{s} \\ \Delta z_{s} \\ \Delta \dot{x}_{s} \\ \Delta \dot{y}_{s} \\ \Delta \dot{z}_{s} \\ \Delta \dot{x}_{s} \\ \Delta \lambda \dot{x}_{s} \\ \Delta M \\ \Delta M_{P_{0}} \end{array} = \Psi_{\text{ext}_{MP,2N^{-}}} \cdot \Delta TNC_{\text{ext}_{2N^{-},2N}} \begin{bmatrix} \Delta u_{T,2N} \\ \Delta u_{N,2N} \\ \Delta u_{C,2N} \\ \Delta u_{C,2N} \\ \Delta I_{sp,2N} \end{bmatrix}$$
(12-13)

where $\Psi_{\operatorname{ext}_{MP,2N^{-}}} = \Phi_{\operatorname{ext}_{MP,N+1^{-}}} \cdot TNC_{\operatorname{ext}_{N+1^{-},N+1^{+}}} \cdot \ldots \cdot \Phi_{\operatorname{ext}_{2N-1^{+},2N^{-}}}$.

$$\Psi_{\text{ext}_{MP,2N^{-}}} \cdot \Delta TNC_{\text{ext}_{2N^{-},2N}} = \begin{bmatrix} \frac{\partial x_{s,\text{MP b}}}{\partial u_{T,2N}} & \frac{\partial x_{s,\text{MP b}}}{\partial u_{N,2N}} & \frac{\partial x_{s,\text{MP b}}}{\partial u_{C,2N}} & \frac{\partial x_{s,\text{MP b}}}{\partial I_{sp,2N}} \\ \frac{\partial y_{s,\text{MP b}}}{\partial u_{T,2N}} & \frac{\partial y_{s,\text{MP b}}}{\partial u_{N,2N}} & \frac{\partial y_{s,\text{MP b}}}{\partial u_{C,2N}} & \frac{\partial y_{s,\text{MP b}}}{\partial I_{sp,2N}} \\ \frac{\partial z_{s,\text{MP b}}}{\partial u_{T,2N}} & \frac{\partial z_{s,\text{MP b}}}{\partial u_{N,2N}} & \frac{\partial z_{s,\text{MP b}}}{\partial u_{C,2N}} & \frac{\partial z_{s,\text{MP b}}}{\partial I_{sp,2N}} \\ \frac{\partial x_{s,\text{MP b}}}{\partial u_{T,2N}} & \frac{\partial x_{s,\text{MP b}}}{\partial u_{N,2N}} & \frac{\partial x_{s,\text{MP b}}}{\partial u_{C,2N}} & \frac{\partial x_{s,\text{MP b}}}{\partial I_{sp,2N}} \\ \frac{\partial y_{s,\text{MP b}}}{\partial u_{T,2N}} & \frac{\partial y_{s,\text{MP b}}}{\partial u_{N,2N}} & \frac{\partial x_{s,\text{MP b}}}{\partial u_{C,2N}} & \frac{\partial x_{s,\text{MP b}}}{\partial I_{sp,2N}} \\ \frac{\partial y_{s,\text{MP b}}}{\partial u_{T,2N}} & \frac{\partial y_{s,\text{MP b}}}{\partial u_{N,2N}} & \frac{\partial y_{s,\text{MP b}}}{\partial u_{C,2N}} & \frac{\partial y_{s,\text{MP b}}}{\partial I_{sp,2N}} \\ \frac{\partial z_{s,\text{MP b}}}{\partial u_{T,2N}} & \frac{\partial y_{s,\text{MP b}}}{\partial u_{N,2N}} & \frac{\partial y_{s,\text{MP b}}}{\partial u_{C,2N}} & \frac{\partial y_{s,\text{MP b}}}{\partial I_{sp,2N}} \\ \frac{\partial z_{s,\text{MP b}}}{\partial u_{T,2N}} & \frac{\partial z_{s,\text{MP b}}}{\partial u_{N,2N}} & \frac{\partial y_{s,\text{MP b}}}{\partial u_{C,2N}} & \frac{\partial y_{s,\text{MP b}}}{\partial I_{sp,2N}} \\ \frac{\partial z_{s,\text{MP b}}}{\partial u_{T,2N}} & \frac{\partial y_{s,\text{MP b}}}{\partial u_{C,2N}} & \frac{\partial y_{s,\text{MP b}}}{\partial I_{sp,2N}} \\ \frac{\partial z_{s,\text{MP b}}}{\partial u_{C,2N}} & \frac{\partial z_{s,\text{MP b}}}{\partial I_{sp,2N}} \\ \frac{\partial z_{s,\text{MP b}}}{\partial u_{T,2N}} & \frac{\partial z_{s,\text{MP b}}}{\partial u_{C,2N}} & \frac{\partial z_{s,\text{MP b}}}{\partial I_{sp,2N}} \\ \frac{\partial z_{s,\text{MP b}}}{\partial u_{T,2N}} & \frac{\partial z_{s,\text{MP b}}}{\partial u_{C,2N}} & \frac{\partial z_{s,\text{MP b}}}{\partial I_{sp,2N}} \\ \frac{\partial z_{s,\text{MP b}}}{\partial u_{C,2N}} & \frac{\partial z_{s,\text{MP b}}}{\partial I_{C,2N}} & \frac{\partial z_{s,\text{MP b}}}{\partial I_{sp,2N}} \\ \frac{\partial z_{s,\text{MP b}}}{\partial u_{T,2N}} & \frac{\partial z_{s,\text{MP b}}}{\partial u_{C,2N}} & \frac{\partial z_{s,\text{MP b}}}{\partial I_{sp,2N}} \\ \frac{\partial z_{s,\text{MP b}}}{\partial u_{T,2N}} & \frac{\partial z_{s,\text{MP b}}}{\partial u_{C,2N}} & \frac{\partial z_{s,\text{MP b}}}{\partial I_{sp,2N}} \\ \frac{\partial z_{s,\text{MP b}}}{\partial u_{C,2N}} & \frac{\partial z_{s,\text{MP b}}}{\partial u_{C,2N}} & \frac{\partial z_{s,\text{MP b}}}{\partial u_{C,2N}} \\ \frac{\partial z_{s,\text{MP b}}}{\partial u_{C,2N}} & \frac{\partial z_{s,\text{MP b}}}{\partial u_{C,2N}} & \frac{\partial z_{s,\text{MP b}}}{\partial u_{C,2N}} \\ \frac{\partial z_{s,\text{MP b}}}{\partial u_{C,2N}} & \frac{\partial z_{s,\text{MP b}}}{\partial u_{C,$$

Calculation of the derivatives Due to the definitions of the match point constraints in the form of $x_{\text{MP,for}} - x_{\text{MP,back}}$, the derivatives of the forward throttle parameters and impulses are equal to the values of the $\Psi_{\text{ext}_{MP,1^+}} \cdot \Delta TNC_{\text{ext}_{1^+,1}}$ matrix product, while the derivatives of the backward throttle parameters and impulses are equal to the negative values of the $\Psi_{\text{ext}_{MP,2N^-}} \cdot \Delta TNC_{\text{ext}_{2N^-,2N}}$ matrix product.

12-3-3 Derivatives of the throttle constraints with respect to the throttle parameters

Looking at the definition of the throttle constraint in Equation 12-5, one can see that those derivatives are trivial. The derivative of throttle constraint i with respect to throttle parameters $u_{T,j}$, $u_{N,j}$ and $u_{C,j}$ depends on the relation between i and j. If $i \neq j$, they are equal to zero. If i = j,

$$\frac{\partial F_{\text{throttle,i}}}{\partial u_{T,i}} = \frac{u_{T,i}}{\sqrt{u_{T,i}^2 + u_{N,i}^2 + u_{C,i}^2}} \\
\frac{\partial F_{\text{throttle,i}}}{\partial u_{N,i}} = \frac{u_{N,i}}{\sqrt{u_{T,i}^2 + u_{N,i}^2 + u_{C,i}^2}} \\
\frac{\partial F_{\text{throttle,i}}}{\partial u_{C,i}} = \frac{u_{C,i}}{\sqrt{u_{T,i}^2 + u_{N,i}^2 + u_{C,i}^2}}$$
(12-15)

12-4 Comparison of throttled and thrust representations

In this section, a comparison between the two different methods will be made. First of all, a nominal one-leg scenario will be explained, compared and profiled. Then, the effect of communicating the sparsity to SNOPT will be explained, after which this representation will be profiled. Finally, a grid search around the nominal scenario will be performed and compared for the different representations.

12-4-1 Nominal single-leg scenario

The investigated one-leg nominal scenario is a rendez-vous problem with Mars launching at the 1st of January 2031 with a fixed flight time of 300 days using 100 kW of power. Upon arrival at Mars, the spacecraft is injected into a 300x45000 km elliptical orbit around Mars, based on the Mars Reconnaissance Orbiter mission [Jet Propulsion Laboratory - MRO]. The size of the injection burn depends on the arrival excess velocity and the required propellant depends on the size of this burn and on the specific impulse of the orbital injection system, which was assumed to be 230 s [Benfield and Turner, 2007]. The objective of this optimization was to maximize the injected payload mass minus the mass of the SEP system.

12-4-2 Convergence issues throttled representation

It was observed that the throttled representation was more sensitive to the initial guess of the throttle vector. For the thrust representation, an initial guess of the manoeuvre could converge if the initial guess of the non-dimensional ΔV -vector is [1,1,0.1] for all manoeuvres. However, for a throttle vector initial guess of [1,0.1,0.1], the throttled representation struggled to converge and repeatedly produced SNOPT output 43: "cannot satisfy the general constraints". However, upon changing the initial guess to [1,1,1], it converged onto a solution.

This issue can be traced back to the distribution of information within the Jacobian. For the thrust-constraints representation, visualized in Figure 12-1, the derivatives of the match point constraints contain information on how a change in the size of a manoeuvre affects the state of the spacecraft and how this influences the match point. The derivatives of the thrust constraints contain information on how the size of a manoeuvre influences the allowable magnitude of the following ones.

For the throttled constraints, visualized in Figure 12-2, the latter information has been moved to the derivatives of the match point constraints. So, both sources of information are merged into one single derivative. Hence, instead of having information on the effect on all consecutive manoeuvres and on the match point, only the latter remains. It is believed that this loss of data hinders the optimizer to distinguish which part of the match point derivatives are due to the change in the manoeuvre itself and which are due to the changes in the other manoeuvres as a consequence of this change. Hence, the optimizer has more trouble identifying which manoeuvre needs to be adapted, which thwarts the converging onto a solution.

12-4-3 Validation and profiling of the nominal scenario

Upon running the nominal scenario using the thrust representation, the non-dimensional optimal objective function value is -5.7814983365E-01. Upon running the nominal scenario using the throttled representation, the optimal objective function value is -5.7814985825E-01. These results are very similar and only differ from the 8th decimal number. This corresponds to a difference of 0.9 grams and is therefore negligible. Furthermore, if plotted, the thrust profile and trajectory look exactly the same, indicating that both methods have converged onto the same solution.

Due to a coincidence, the numbers of calls to the *usrfg* function for this specific scenario are equal for the thrust and throttled representations. Due to this equality, this scenario is very suited for a comparison of the run-times. Upon profiling, it became apparent that the throttled representation requires more time in the *StateVectorPropagation* function. This does not come as a surprise, as more calculations have to be performed to establish the extended transformation matrices. Additionally, the mass before the manoeuvre has to be calculated using the inverse of the Lambert function, as can be seen in Equation 12-3, which needs to be solved using an iterative procedure. Also, the larger size of the transformation matrices increases the computational time to perform the required matrix multiplications to calculate several derivatives.

However, due to the larger sparsity in the Jacobian for the throttled representation, SNOPT requires less time in the s8getr function within the s8iqp function. This function is responsible for calculating several aspects of the reduced Hessian. Since the throttled representation leads to a much sparser Jacobian, less time is required within these functions. The overall effect is that the throttled representation is faster: the run times are 1.975 s and 1.655 s for the thrust and throttled representation respectively.

12-4-4 Sparsity

Up to this point, SNOPT does not know the structure of the **sparsity pattern**. As such, SNOPT uses all the elements of the Jacobian, including the elements that are always zero, to update the state vector. However, if the structure of the Jacobian would be communicated to SNOPT, it would only utilize the non-zero elements of the Jacobian to update the state vector, which, depending on the sparsity of the Jacobian, leads to a drastic reduction in the number of elements that are taken into account and hence the number of computations that have to be performed. Looking at Figure 12-1, one can see that the proper indexing of the Jacobian for the thrust constraints representation is rather tricky. Therefore, it had never been done before. Looking at Figure 12-2, one can see that this indexing is much more straightforward for the throttled representation. Furthermore, the sparsity of the throttled representation is much higher than for the throttled representation. As such, the gain of communicating the sparsity is higher for the throttled representation.

Despite its simpler sparsity pattern, the implementation proved to be tricky; the index of every single element in the Jacobian had to be updated.

Upon implementing the sparsity feature into the program, it had to be validated again. The optimal objective function value is -5.7814990258E-01. As only the 7th decimal number is different and as the thrust profile and trajectory look exactly the same as for the throttled case without sparsity implemented, it can be concluded that they converge onto the same solution.

Due to the inner workings of SNOPT and due to the utilization of the sparsity, the time spent in the s8iqp_ function is significantly lower, as expected. Utilizing the sparsity pattern, the run time has been decreased further from 1.655 s to 1.243 s.

12-4-5 Grid search around nominal scenario

Until now, only one specific case has been examined. To avoid drawing conclusions based on only one case where coincidence can pollute the conclusion, a grid search will be performed where the initial time is varied between 50 days before and 30 days after the reference epoch of the 1st of January 2031 with a step size of 5 days, and the time of flight is variable between 280 days and 480 days with a step size of 50 days. The results can be found in Table 12-1.

Case	Time for grid search [s]	Total calls to usrfg [-]
Thrust	124	19856
Throttle, non-sparse	117	19273
Throttle, sparse	88	19267

Table 12-1: Results of the one-leg grid search.

If one divides the total run time by the number of calls, one gets 6.24 and 6.07 ms per call to usrfg for the thrust and throttled representation respectively. So, the throttled representation requires less calls to the usrfg function, and it requires less run time to complete a call to the usrfg function. If one divides the total run time by the number of calls, one gets 6.07 and 4.57 ms per call to usrfg for the non-sparse and sparse throttled representation respectively.

From these numbers, it appears that the throttled representation is indeed much faster. However, these numbers alone do not show the entire picture. As explained, the throttled representation displayed a significant sensitivity to the initial guess. In many scenarios, the optimization had to be started multiple times with different initial guesses to bypass several convergence issues. The listed times however do only show the time it takes to perform the grid search if all throttled optimizations converge on the first effort. As such, the listed times for the grid search are in fact an utopic representation of the throttled method. If one would take into account the total time for all the optimization runs, the non-sparse throttled runs were 36 seconds slower than the thrust runs, while the sparse throttled runs were only 4 seconds faster.

12-5 Conclusion

The throttled representation has significant advantages. First of all, the Jacobian becomes much sparser, which reduces the time spent in the s8sqp function within SNOPT. This leads to a reduction of approximately 3 % for the ratio of the total run time and the number of calls. Furthermore, the sparsity pattern for the throttled representation is much simpler than for the thrust representation. As such, it is much easier to communicate the sparsity pattern to SNOPT, which further reduces the time spent within the SNOPT functions. The gain from implementing the sparsity pattern was a reduction of approximately 25 % for the ratio of the total run time and the number of calls for a one-leg trajectory.

Besides its advantages, there are also disadvantages. The throttled representation is numerically less stable and requires better scaling and a more accurate initial guess. Several options to resolve this issue have been tried. One example is the use of an automated scaling method. However, all those efforts failed to rectify the issue. Up to now, this has been bypassed by manually adjusting the initial guesses based on the output of the failed optimization. This adjustment has been mainly based on intuition and experience with the problem at hand. If this needs to be automated, a large anount of complexity needs to be added to the code. So as long as the initial-guess sensitivity issue remains, it would be more practical to use the thrust representation. Besides, the speed of this representation can still be improved by implementing its complex sparsity pattern. This will be done in Chapter 14, after an additional trade-off on the problem structure has been performed in the next chapter.

Part IV

Research goal 2: Control node's time optimization

Chapter 13

Coupling between time and ephemeris

The coordinates and velocities at the control nodes must be properly connected to the time epoch of those nodes. Otherwise, the optimization could result in a trajectory that has planetary encounters at a certain time and position when the planet is not physically there. In this chapter, two different coupling methods will be established and compared. As explained in Section 5-2, the current set-up of the state vector where both coordinates and velocities, and the control node's time are present requires proper coupling between those parameters. This can be done using ephemeris constraints, which will be explained in the next section. An alternative is to change the structure of the state vector by removing the control node's coordinates and velocities from the state vector and calculate them where needed based on the control node's time epoch. This will be explained in Section 13-2

13-1 Option 1: Ephemeris constraints

The first method that will be discussed uses **ephemeris constraints**. In the current set-up where for each node, there are six coordinates and velocities and a time epoch in the state vector, there must be six ephemeris constraints of the form:

$$-\varepsilon < \alpha_{\text{state vector}} - \alpha_{\text{ephemeris}}(t) < \varepsilon \tag{13-1}$$

where $\alpha_{\text{state vector}}$ is the coordinate or velocity in the state vector and $\alpha_{\text{ephemeris}}(t)$ is the coordinate or velocity calculated using Meeus' polynomials for time t explained in Appendix G.

13-1-1 Analytical derivatives of ephemeris constraints

The analytic derivatives of these ephemeris constraints must be determined. The derivatives with respect to the coordinates and velocities in the state vector are trivial to find; they are all equal to one. The derivatives with respect to the time require lengthy derivations. The interested reader is referred to Appendix G.

13-1-2 Analytical derivatives with respect to time

Besides the derivatives of the ephemeris constraints, also the derivatives of the time with respect to the other constraints must be found. This discussion will mainly focus on the derivatives with respect to t_0 . The derivatives with respect to t_f can be found similarly. This discussion will limit itself to the x and \dot{x} state match point constraints. Similar procedures can be used to find the derivatives of the other match point constraints.

To calculate the derivatives with respect to t_0 , one must find the difference in the constraints values when all parameters except t_0 remain constant. Since the initial node coordinates and velocities are directly inputted into the propagation module, t_0 only affects the propagation time step $\Delta t = \frac{t_f - t_0}{2N}$. If the propagation time step changes, the propagation itself changes which has an effect on the match point constraints:

$$\frac{\partial \Delta x}{\partial t_0} = \frac{\partial \Delta x}{\partial \Delta t} \cdot \frac{\partial \Delta t}{\partial t_0} = \frac{\partial \left(x_{\text{match point, forward}} - x_{\text{match point, backward}}\right)}{\partial \Delta t} \cdot \frac{-1}{2N} \quad (13-2)$$

$$\frac{\partial \Delta \dot{x}}{\partial t_0} = \frac{\partial \Delta \dot{x}}{\partial \Delta t} \cdot \frac{\partial \Delta t}{\partial t_0} = \frac{\partial \left(\dot{x}_{\text{match point, forward}} - \dot{x}_{\text{match point, backward}} \right)}{\partial \Delta t} \cdot \frac{-1}{2N} \quad (13-3)$$

This change in propagation time step has significant consequences. For instance, the STM's are not accurate anymore. STM's are set up to transform a change in coordinates and velocities before a propagation into the change in coordinates and velocities after said propagation. These relationships only apply for a specific propagation time step. To obtain accurate analytic derivatives with respect to the propagation time step, one would need to know how the STM elements change as a function of the propagation time step. Therefore, multiple methods to adjust the STM elements have been tested. The most promising will now be discussed.

A chained procedure is used to run through the propagation to find $\frac{\partial x_{\text{match point, forward}}}{\partial \Delta t}$ and $\frac{\partial \dot{x}_{\text{match point, backward}}}{\partial \Delta t}$. If the change in propagation time step is small, one can assume that

$$\frac{\partial x_1^-}{\partial \Delta t} = \frac{\dot{x}_1}{2}$$

$$\frac{\partial \dot{x}_1^-}{\partial \Delta t} = \frac{\ddot{x}_1}{2} = \frac{-\mu}{R_1^3} x_1 \cdot \frac{1}{2}$$
(13-4)

Note that the factor 2 is due to the fact that the propagation between point 0 and 1^- has been done using half a propagation time step. As explained in Subsection 8-3-1, the change before application of the manoeuvre 1^- can be converted into the change after application of the manoeuvre 1^+ using the TNC_{1+,1}- matrix explained in Appendix B.

Point 2^- where the second manoeuvre is applied does not only shift due to the change in propagation time step, it also moves as a result of the change in position and velocity at point 1^+ :

$$\frac{\partial x_2^-}{\partial \Delta t} = \dot{x}_2^- \qquad (13-5)$$

$$+ \frac{\partial x_2^-}{\partial x_1^+} \frac{\partial x_1^+}{\partial \Delta t} + \frac{\partial x_2^-}{\partial y_1^+} \frac{\partial y_1^+}{\partial \Delta t} + \frac{\partial x_2^-}{\partial z_1^+} \frac{\partial z_1^+}{\partial \Delta t}$$

$$+ \frac{\partial x_2^-}{\partial \dot{x}_1^+} \frac{\partial \dot{x}_1^+}{\partial \Delta t} + \frac{\partial x_2^-}{\partial \dot{y}_1^+} \frac{\partial \dot{y}_1^+}{\partial \Delta t} + \frac{\partial x_2^-}{\partial \dot{z}_1^+} \frac{\partial \dot{z}_1^+}{\partial \Delta t}$$

$$\frac{\partial \dot{x}_2^-}{\partial \Delta t} = \ddot{x}_2^-$$

$$+ \frac{\partial \dot{x}_2^-}{\partial x_1^+} \frac{\partial x_1^+}{\partial \Delta t} + \frac{\partial \dot{x}_2^-}{\partial y_1^+} \frac{\partial y_1^+}{\partial \Delta t} + \frac{\partial \dot{x}_2^-}{\partial z_1^+} \frac{\partial z_1^+}{\partial \Delta t}$$

$$+ \frac{\partial \dot{x}_2^-}{\partial \dot{x}_1^+} \frac{\partial \dot{x}_1^+}{\partial \Delta t} + \frac{\partial \dot{x}_2^-}{\partial \dot{y}_1^+} \frac{\partial \dot{y}_1^+}{\partial \Delta t} + \frac{\partial \dot{x}_2^-}{\partial \dot{z}_1^+} \frac{\partial \dot{z}_1^+}{\partial \Delta t}$$

$$(13-6)$$

where the derivatives of point 1^+ with respect to the propagation time step can be found combining Equation 13-4 with the TNC₁ matrix. The derivatives of point 2^- with respect to point 1^+ can be found from the STM from 1^+ to 2^- : $\Phi(2^-, 1^+)$. This procedure is iterated until the match point is reached. Note that for the final step between point N⁺ and the forward match point, the propagation has been done with half a propagation step.

A similar procedure can be set up to find $\frac{\partial x_{\text{match point, forward}}}{\partial \Delta t}$ and $\frac{\partial \dot{x}_{\text{match point, backward}}}{\partial \Delta t}$. Similar mechanisms have also been used to obtain the derivatives of the thrust constraints.

Using this first-order method where the STM is adapted based on the first derivative of the element (\dot{x} for the x elements and \ddot{x} for the \dot{x} elements), the derivatives approach the values obtained from a finite difference method. However, the remaining inaccuracy leads to numerical difficulties within the optimizer. Therefore, also higher-order methods have been evaluated. Although they provide higher accuracy, the numerical difficulties prevail. Furthermore, they require more computing power. Therefore, it was decided to obtain the derivatives with respect to t_0 and t_f using a **forward finite difference method** [Gill et al., 1981]. Using this method, the derivatives are accurate enough for the optimizer to converge onto a solution.

13-1-3 Numerical difficulties ephemeris constraints

Although the accuracy of the derivatives with respect to t_0 and t_f no longer poses a problem, there are still convergence problems using the ephemeris-constraints representation. Based on the output of the optimization runs, the problem can be traced back to the ephemeris constraints. Quite often, the optimization runs into numerical difficulties upon matching the coordinates and velocities from the state vector with those based on the ephemeris and time. Therefore, it was opted to investigate another option.

13-2 Option 2: Remove coordinates and velocities from the state vector

To avoid the unstable ephemeris constraints, one could remove the coordinates and velocities from the state vector. As such, instead of passing the control node's coordinates and velocities from the state vector to the propagator, they are calculated within the propagator based on the control node's time epoch from the state vector using Meeus' polynomials. If the time changes, not only the propagation time step changes, also the control node's coordinates and velocities change. Since those are used as the starting point of the propagation, the analytic derivatives become even more unstable. Therefore, it was again decided to use a forward finite difference method. As this method does not utilize the unstable ephemeris constraints, the convergence issues encountered for option 1 are avoided. Hence, this method succeeds to converge on a solution on almost all situations it has been applied to.

13-3 Conclusion

Option 1 needs ephemeris constraints, which introduce numerical difficulties. Furthermore, the smaller number of constraints and state variables for option 2 reduces the size of the optimization problem. As such, option 2 should theoretically be faster than option 1. It is currently impossible to prove this due to the numerical difficulties encountered in option 1. Hence, it is impossible to obtain an estimate of the time required to converge onto a solution.

However, there is one major disadvantage for option 2. If one would like to optimize a trajectory to reach a fixed point in an orbit, it would be useful to have the coordinates and velocities in the state variables. They would have to be kept constant while the time of flight would be allowed to change. Furthermore, the usage of ephemeris constraints in such situations is no longer required, which should make the problem more stable and solvable. Without the coordinates and velocities in the state vector, such a problem would still be solvable, be it more indirect. One could bypass this problem by using an ephemeris function that gives a constant position and velocity irrespective of the time.

It can be concluded that the advantages of removing the coordinates and velocities from the state vector outweigh the disadvantages. Therefore, it was decided to continue with option 2 for the remainder of this thesis work.

Chapter 14

Sparsity pattern implementation

The sparsity for the thrust representation of the Sims-Flanagan transcription has not yet been implemented. The main reason to postpone this implementation was a continuous change in the way the Sims-Flanagan representation was set up. This would always lead to a change in the sparsity pattern. Therefore, it was opted to first decide upon a fixed problem structure by trading off throttled versus thrust representations, ephemeris constraints versus removal of the coordinates and velocities from the state vector, etc. Now that those options have been compared and final decisions on the structure of the problem have been made, the sparsity pattern of the Jacobian can be implemented.



Figure 14-1: Sparsity pattern of the Jacobian for a single-leg case with 20 segments, thrust constraints, coordinates and velocities removed from the state vector.

In Figure 14-1, one can see the sparsity pattern for the final Sims-Flanagan structure using the thrust representation where the coordinates and velocities have been removed from the state vector for a single-leg, 20 segments trajectory.

The sparsity of the problem can be calculated, assuming the constraint vector only consists of standard match point and thrust constraints. For the mass match point constraint, there are $2 + 4 \cdot (2N)$ derivatives. For each of the other six match point constraints F_x up to $F_{\dot{z}}$, there are $8 + 3 \cdot (2N)$ derivatives. For the forward thrust constraints, there are $7 \cdot N$ derivatives plus the $\frac{N}{2} \cdot (N+1)$ elements in the step structure. The backward thrust constraints also have $7 \cdot N + \frac{N}{2} \cdot (N+1)$ elements. So in total, there are $8N^2+62N+50$ derivatives. The size of the Jacobian is $(7 + 2N) \cdot (11 + 8N) = 16N^2 + 78N + 77$ elements. Knowing this, the sparsity percentage can be calculated for a different number of legs. The results of these calculations can be seen in Figure 14-2.



Figure 14-2: Sparsity percentage for the removed coordinates and velocities state vector structure, using thrust constraints for different number of legs and segments.

The same grid search as in Table 12-1 for a constant-time, one-leg scenario has been performed. Due to small changes in computer settings compared to the runs in Table 12-1, the classical thrust representation case where the coordinates and velocities are still present in the state vector now requires 142 seconds of run time. If one removes the coordinates and velocities, this is reduced to 118 seconds. The difference is caused by the smaller size of the state vector and hence a smaller Jacobian size. Upon implementing the sparsity pattern for the case where the coordinates and velocities have been removed, the run time decreases to 98 seconds. One can conclude that again, the implementation of the sparsity pattern results in significant run time reductions. Note that this comparison has been performed on a one-leg scenario. From Figure 14-2, it can be deduced that the gain will be higher for higher-leg scenarios.

Chapter 15

Two-leg Earth-Mars-Earth test case

In the previous chapters, the control node's time optimization and increased speed capabilities of the software have been explained. In this chapter, the time optimization capabilities will be put to the test. It was decided to first test the optimization of a single epoch on a two-leg trajectory: a Martian flyby mission. While it would appear logical to first test a single-leg case, it was decided not to do so. Upon testing a single-leg Martian rendez-vous mission, it was observed that the higher the time of flight, the lower the launch mass became. Hence, this case is not very interesting, as the optimum lies on the imposed upper limit of time of flight. From Chapter 9, it is known that only a limited range of feasible flyby dates for a certain launch date - time of flight combination exist. Hence, it can be assessed if the tool truly converges onto the optimal flyby date.

Furthermore, in previous research, two-leg Earth-Mars-Earth launch windows for a variety of launch configurations, SEP power levels and payload masses have been established [De Smet et al., 2014]. The main recommendation from this research is to perform two-dimensional grid searches in launch date and TOF where for each grid point, the flyby date is optimized. The creation and validation of such plots is the main goal of this chapter.

Such flyby-date optimized grid searches will be produced for two different optimization representations: one where the launch mass is minimized and one where the final mass is maximized. Both options will be described in the next two sections. Finally, a comparison will be made and one representation will be selected.

15-1 Representation 1: Minimized launch mass

In this optimization problem, the final mass consisting of payload mass and the mass of the SEP subsystem as explained in Equation 9-1, is kept constant. The objective is to minimize the launch mass. In the following subsections, the performed flyby-date optimization will be validated and the results of the example case will be discussed.

15-1-1 Validation

It must be ensured that for each grid point, the optimizer converges onto the optimal flyby date. This can be checked using the grid searches established in De Smet et al. [2014]. For each departure date in the grid search, the flyby date resulting in the minimal launch mass can be determined. However, due to the limited resolution of the performed grid search, this optimal flyby date will display a discrete behavior. These optimal flyby dates will be compared to the results of a one-dimensional grid search in launch date with a constant TOF of 501 days. For each of those launch dates, the flyby time has been optimized while the launch date and return date are kept constant. The results of these optimization runs for the reference scenario in 2018 using 10 kW of SEP on a payload mass of 13.139 tons using the SLS-4RL launch configuration can be found in Figure 15-1.



Figure 15-1: Flyby date optimization validation of the 2018 - 10 kW - 13 tons - 4RL scenario, representation 1 where the launch mass is minimized.

Analysis of the grid search Before analyzing the performance of the flyby-date optimization, an analysis of the grid search will be performed. First of all, one can estimate the maximum propellant mass for a time of flight of 501 days. The mass flow rate of the SEP system can be found by rewriting Equation 3-4.

$$\dot{m} = \frac{2P\eta_{\text{jet}}}{(I_{sp}g_0)^2} \tag{15-1}$$

If one multiplies this mass flow by the time of flight of 501 days and the duty cycle of 90% and if one assumes that the power is constant at all time to 10 kW, one gets an estimate of the upper boundary of the total propellant mass of 1215 kg. If one adds the mass of the SEP system, 300 kg for a 10 kW system, an upper limit on the launch mass of 14.6 tons is obtained. In reality, this number is lower, since the power is not always equal to 10 kW but decreases with the square of heliocentric distance. Enforced coast arcs further lower this number.



Figure 15-2: C_3 profile of the 2018 - 10 kW - 13 tons - 4RL scenario , representation 1 where the launch mass is minimized.

The propellant mass is not the only limiting factor on the maximum launch mass; the required launch C_3 also imposes a boundary. If one looks at Figure 15-2, one can see a trend: the earlier the departure date, the higher the required C_3 becomes. This can be explained by looking at the problem geometry in Figure 15-3. Note that those trajectory figures have been stripped down for clarity reasons. The utilized legend and axis are the same as for previous figures such as Figure 9-3a. In Figure 15-3a, one can see that due to the geometry of the problem, a sudden and large deviation from the Earth's orbit is required to make the transfer to Mars. Hence, a large C_3 is required. On the other hand, for a later departure date, the geometry allows a transfer where the spacecraft stays closer to Earth's orbit for a longer time. Hence, it does not need the large instantaneous change and as such only needs a smaller C_3 .



Figure 15-3: Trajectories for the 2018 - 10 kW - 13 tons - 4RL scenario, representation 1 where the launch mass is minimized.

The lower limit of the departure date at -88 days can be explained through the large C_3 requirement at early departure dates. At -88 days, the required C_3 is 60 km²/s². Based on the launch curves in Figure ??, it is known that for a C_3 of 60 km²/s², the maximum launch mass is 14.4 tons, the launch mass at the -88 departure date. At departure date -90, a slightly higher C_3 value is required. However, at that point, the launcher is no longer capable of launching the maximum mass of 14.4 tons required to fulfill the mission.

The observed C_3 profile explains the lower departure date cut-off of the launch window. However, it does not explain the upper departure date cut-off; the required C_3 here is approximately 40 km²/s², which means that the theoretical upper launch mass limit would be about 21 tons. The limiting factor here is the allowable hyperbolic return velocity of 8.969 km/s. In Figure 15-4, one can see that the hyperbolic return velocity increases with increasing departure date. The reason for this can again be found from the problem geometry in Figures 15-3a and 15-3b. One can observe that for the earlier departure date, the geometry is such that the return leg is more or less parallel with the orbit of the Earth. For later departure dates, the return leg makes a much larger angle with the Earth's orbit and has a higher relative velocity. In order to keep the relative velocity within the allowable 8.969 km/s, the SEP system has to perform more and more work for increasing departure dates. In the end, at departure date 38, the SEP system is continuously active, hence the maximal launch mass of 14.4 tons is reached there. For a later departure date, the SEP system does not have sufficient time to comply with the relative velocity constraint.



Figure 15-4: Return hyperbolic excess velocity profile of the 2018 - 10 kW - 13 tons - 4RL scenario, representation 1 where the launch mass is minimized.

Besides explaining the boundaries on the departure date, also the trend within the same departure date needs to be explained. As an example, departure date 0 will be discussed. From Figure 9-3, it is known that the trajectory for [3,233] is almost a free return trajectory. However, if the flyby date is shifted by a few days, the geometry does not allow for a free return trajectory. Instead, more thrusting is required on both legs. This explains why for both earlier and later flyby dates than 233 days, the required propellant mass and hence the launch mass is larger.
Also the shape needs to be explained. One can observe an upward slope. This is partially because of the set-up of the figure. For a later departure date, the optimal flyby date becomes later for a similar time of flight of the first leg. However, this does not capture the entire picture; the later flyby date does not correlate exactly with the later departure date. One can see that the time of flight of the first leg becomes shorter. This can be traced back to the geometry of the problem. On the earlier departure date, a transfer of approximately 300 degrees is required, while for the later departure date, the geometry is more favorable and allows for transfers of approximately 180 degrees. Logically, the latter transfers are shorter.

Flyby date optimization validation In Figure 15-1, the optimal flyby dates from the validated two-dimensional grid search are displayed as a cyan square. The optimal flyby dates from the to be validated one-dimensional grid search in departure dates where the flyby dates are optimized are displayed as green diamonds. One can see that they are almost identical, validating the results from the one-dimensional grid search.

To show this is not just mere luck, the same will be done for another scenario in 2018 using 10 kW of SEP on a payload mass of 19 tons using the SLS-4RL launch configuration. The results can be found in Figure 15-5. Again, the flyby date optimization performs very well.



Figure 15-5: Flyby date optimization validation of the 2018 - 10 kW - 19 tons - 4RL scenario, representation 1 where the launch mass is minimized.

This grid search shows similar characteristics as for the 13 ton scenario. Hence, the specific analysis will be skipped here, but a comparison will be made instead. First of all, for both the 13.139 and the 19 ton scenario, the absolute minimum launch mass is found at a departure and flyby date of respectively 8 and 233 days from the 5th of January, 2018. The reason for this will be explained in Subsection 19-1-2. An interesting difference is the fact that the launch window shrinks about 60 days on the left-hand side of the launch window, while it only shrinks 10 days on the right-hand side. This can be explained by looking at the C_3 and the return velocity in Figures 15-2 and 15-4. The maximum C_3 for a launch mass of 20 tons is $40 \text{ km}^2/\text{s}^2$, which is much lower than the $60 \text{ km}^2/\text{s}^2$ for the 14 ton scenario. Hence, a large range of departure dates becomes infeasible for this higher mass. On the right-hand side, there is less manoeuvrability on the second leg because of the higher mass. However, the re-entry velocity constraint stays the same, which explains why the launch window is only marginally decreased here.

15-1-2 Two-dimensional grid search in departure date and TOF

Now that the flyby date optimization has been validated, the main results of this chapter - the two-dimensional grid searches in launch date and TOF - can be created. For each of these grid points, the flyby date has been optimized. The result can be seen in Figures 15-6a and 15-6b. The analysis of these figures will be done in Chapter 19.



Figure 15-6: Launch date - TOF grid search with flyby date optimization for 2018 - 10 kW - 4RL, representation 1 where the launch mass is minimized.

15-2 Representation 2: Maximized final mass

In this representation of the optimization problem, the final mass, known as dry mass, is maximized. The launch mass, coupled to the launch curve, is free. The minimum allowable final mass is set to 10.3 ton: 10 tons of payload mass and 0.3 tons of SEP system mass for the 10 kW scenarios.

The main advantage of the second optimization formulation is that only one run is required to determine the launch window for different payload masses. If one wants to know the launch window for a payload mass of 13 tons, one filters out the launch date - TOF combinations resulting in a payload mass of at least 13 tons out of the bigger 10 ton launch window. For the first optimization formulation, for each different payload mass, every point had to be optimized again. On the other hand, the disadvantage of the second formulation is the extra degree of freedom. For the first representation, the final mass is fixed and the other masses are allowed to change. For the second representation, also the final mass is allowed to change. As such, the convergence on an optimal solution will be slightly slower.

15-2-1 Validation

The same validation method as for representation 1 has been applied on the 2018 - 10 kW - SLS 4RL scenario for a minimum payload mass of 10 tons. From Figure 15-7, one can see that the flyby date optimization performs well again. The processes within this grid are similar to the ones explained in the previous subsections and will therefore be skipped here.



Figure 15-7: Flyby date optimization validation of the 2018 - 10 kW - minimal 10 tons - 4RL scenario, representation 2 where the final mass is maximized.

15-2-2 Two-dimensional grid search in departure date and TOF

Upon validating the flyby-date optimization, the main results of this chapter - the twodimensional grid searches in launch date and TOF - can be created. For each of these grid points, the flyby date has been optimized. The filtered results for a minimum payload mass of 13.139 tons and the subset thereof for 19 tons can be seen in Figures 15-8a and 15-8b. If one compares those to Figures 15-6a and 15-6b, one can see that both representations result in identical launch windows. The actual analysis of these figures will be done in Chapter 19.



Figure 15-8: Launch date - TOF grid search with flyby date optimization for 2018 - 10 kW - 4RL, representation 2 where the final mass is maximized.

15-3 Conclusion

The flyby-date optimization has been validated and produces similar results for both representations. While representation 1 requires a computational run for each different payload mass, representation 2 only needs one, slightly more computational expensive run. Therefore, the second representation will be used for the remainder of this thesis work.

Part V

Research goal 3: Addition of dynamic legs

Chapter 16

Structure of the automated leg addition

Research goal 3 is to automate the addition of extra legs. Therefore, several changes to the code had to be made. These will now be briefly explained.

First of all, it was decided to automate the number of legs based on an input file where for each node, the type and planet number of the control node are specified, along with initial guesses for the time at the control node, the incoming hyperbolic excess velocity and the outgoing hyperbolic excess velocities. Also, several boundaries are listed per node, dependent on the control node type. Examples are the allowable flyby altitudes, the maximum return hyperbolic excess velocities, etc. Also, based on the number of control nodes, the number of legs can be set up; if there are x control nodes, there are x-1 legs. An example of such an input file can be found in Table 16-1.

For now, it was decided to only implement three different **types of control nodes**: type 1 is a planetary departure bounded by launcher constraints explained in Subsection 7-3-1. Type 2 is a planetary flyby bounded by the flyby constraints explained in Subsection 7-3-2. Type 3 is a planetary re-entry bounded by the constraint explained in Subsection 7-3-3. If wanted, additional control node types can be added. However, the explained input file needs to be altered in order to do so.

In conclusion, the program reads in the input file, determines the number and type of nodes and the number of legs. Then, it sets up the constraint vector using the following structure: specific constraints node 1 - general constraints leg 1 - specific constraints node 2 - general constraints leg 2 - specific constraints node 3 - ... general constraints last leg - specific constraints final node. Based on the same structure, the sparsity structure is set up and the Jacobian is calculated. For this entire process, all the propagation, sparsity, constraints and derivative functions had to be automated.

 Table 16-1:
 Example input file for an Earth-Venus-Mars-Earth mission.

Node type	Planet number	Julian date planetary encounter
1	3	2459540.42042924
2	2	2459616.68271822
2	4	2459834.40504626
3	3	2460040.42042924
Vinf,in,x	Vinf,in,y	Vinf,in,z
0	0	0
3593.21685	-4350.64041	-1886.23513
2315.91614	-4187.35764	-2413.50897
-1452.52563	-3438.10721	-856.30422
Vinf,out,x	Vinf,out,y	Vinf,out,z
538.03779	-3319.69797	-1521.13727
3686.10963	-4308.69370	-1801.36522
-847.85336	-4462.35926	-2844.45980
0	0	0
hfly low	hfly upper	Vreturn limit
0	0	0
200	2000	0
400	2000	0
0	0	8969

Chapter 17

Three-leg Earth-Venus-Mars-Earth test case

Based on the automation explained in the previous chapter, a three-leg Earth-Venus-Mars-Earth mission could be set up. Similar to the two-leg Earth-Mars-Earth scenarios, the ultimate goal of this chapter is to create grid searches for launch date and TOF where for each grid point, the Venus and Mars flyby dates are optimized. Based on the discussion in Chapter 15, it is decided to only utilize the second optimization representation where the final mass is maximized.

17-1 Initial trajectory issues

Using Copernicus and a method similar to the one explained in Section 9-4, a nominal chemical thrust scenario in 2021 has been identified. This scenario has been summarized in Table 17-1. From initial runs for constant times, it was observed that about 100 kW of SEP power was required for a payload mass of 15 tons for the reference scenario launching on the 21^{st} of November, 2021, flying by Venus 77 days later, followed by a Martian flyby day 217 days later and finally returning to Earth 206 days later for a total time of flight of 501 days. It was however realized that this was due to the enforced upper boundary of the flyby altitude at Venus and Mars of 2000 km. While this does not introduce any problems for the Martian flyby, it is too low for the Venus flyby, since the reference trajectory from Copernicus has a flyby altitude of 800 000 km. If this 2000 km upper boundary is removed, only 41.3 kW is required for the same trajectory. For 20 kW, this means that for this launch date - flyby dates - return date combination, the maximum payload mass is about 7 tons. The trajectory and thrust profile for this mission can be seen in Figures 17-1a and 17-1b. If on the other hand, the flyby dates are optimized, about 10.5 tons of payload can be launched. By shifting the Venus flyby date 16 days, from 77 days after launch for the fixed case to 61 days after launch, a large difference in required thrust is realized, which can be seen in Figures 17-2a and 17-2b. By shifting the Venus flyby date, the first leg becomes completely ballistic. The

difference in required thrust also shows in the required propellant: only 957 kg of propellant is required for the time optimized case versus 1859 kg of propellant for the time fixed case.



Figure 17-1: 20 kW, 7 ton payload Earth-Venus-Mars-Earth mission launching on the 21st of November, 2021 with a total time of flight of 501 days where the Venus and Martian flyby date are kept fixed.

Launch	Launch	DSM_1	Flyby	DSM_2	Flyby	DSM_3	Return
date	C_3		date 1		date 2		V_{∞}
(m-d-y)	$(\mathrm{km}^2/\mathrm{s}^2)$	(m/s)	(m-d-y)	(m/s)	(m-d-y)	(m/s)	$(\rm km/s)$
11-21-2021	13.62	2823	02-5-2022	2931	09-11-2022	1846	3.83

Table 17-1: Nominal Earth-Venus-Mars-Earth 2021 scenario identified using Copernicus.



Figure 17-2: 20 kW, 7 ton payload Earth-Venus-Mars-Earth mission launching on the 21st of November, 2021 with a total time of flight of 501 days where the Venus and Martian flyby date are optimized.

17-2 Local optima issues

Based on the initial guess obtained from Copernicus for the reference trajectory launching on the 21st of November, 2021, a grid search in launch date and total TOF where both flyby dates are optimized has been performed. The results for departure dates -20 and +15 can be seen in Figure 17-3.



Figure 17-3: Earth-Venus-Mars-Earth grid search in departure date and TOF where both flyby dates are optimized: 2021 - 20 kW - 4RL, no information passed on.

This grid search has been performed **without passing on any information** from one point to the other. It can be observed that this method does not provide good results. There are large jumps in final mass and there are discontinuities where the trajectory becomes infeasible from one point to the other. It is believed that the additional flyby creates multiple local optima, making the optimization difficult. To overcome these issues, several methods have been tried out where different sorts of data have been passed from one point to the next in different directions. Those efforts have been done on a more extensive grid search to increase the number of data points and hence the understanding of the problem.

First of all, a method where information from a point with a **lower TOF is passed to a point with a higher TOF** will be explained. Here, one starts at a low time of flight, uses the initial guess on this first point, passes on all the state variables of the solution to the next point, increases the time of flight, does the optimization, passes on to the next point, etc. At each new launch date, the initial guess for the state variables is reset to the initial guess from Copernicus. The results from these runs can be seen in Figure 17-4.

Although the mass profile in this figure has been cleared of most discontinuities and is changing more smoothly, there are still large sudden changes in mass. If one looks at the times of flight

for the individual legs in Figure 17-4b, one can see that the sudden changes in mass are closely related to sudden changes in this time of flight.



Figure 17-4: Earth-Venus-Mars-Earth grid search in departure date and TOF where both flyby dates are optimized: 2021 - 20 kW - 4RL, information passed from low TOF to higher TOF.

Similarly, a method where information from a point with a **higher TOF** is passed to a **point with a lower TOF** has been tried. One starts at a high time of flight, uses the initial guess for this first point, passes on all the state variables of the solution to the next point, decreases the time of flight, does the optimization, passes on to the next point, etc. At each



new launch date, the initial guess for the state variables is reset to the initial guess from Copernicus. The results from these runs can be seen in Figure 17-5.

Figure 17-5: Earth-Venus-Mars-Earth grid search in departure date and TOF where both flyby dates are optimized: 2021 - 20 kW - 4RL, information passed from high TOF to lower TOF.

If one compares Figures 17-4a and 17-5a, one can see that both figures are very different. However, there are some regions where both procedures result in the same final mass. Looking at Figures 17-4b and 17-5b, one can see that where the final masses are the same on the same grid point, also the time of flights are the same. To find out what is happening at grid points where sudden changes in mass and time of flight of the first leg occur, it was decided to zoom in on one of those points: the point that will be examined is located at departure date -10 days and total time of flight of 601 days. The solutions where both grid searches converge on have been listed in Table 17-2.

Case	Final mass [tons]	TOF_1 [days]	TOF_2 [days]	TOF ₃ [days]
Low-to-high TOF	27.7	140.8	213.6	246.6
High-to-low TOF	22.9	70.3	188.8	341.9

Table 17-2: Local optima solutions for $[t_0, TOF] = [-10, 601]$.

To understand the processes active at this grid point, a grid search has been run. On the x-axis, one can find the TOF of the first leg between Earth and Venus. On the y-axis, one can find the TOF of the second leg between Venus and Mars. The results of this grid search can be found in Figure 17-6.



Figure 17-6: Broad grid search on point [t₀,TOF]=[-10,601].

It is interesting to note that there are apparently two different feasible windows for the first time of flight: a region between 60 and 80 days and a second region between 120 and 150 days. Figures 17-7a and 17-7b zoom in on both regions.

The 56-84 day region depicted in Figure 17-7a has a maximum final mass of 22.9 tons at a TOF_1 of 70 days and a TOF_2 of 189 days. This almost perfectly corresponds with the optimal solution found by the high-to-low TOF passed-on case in Table 17-2.

The 116-148 day region depicted in Figure 17-7b is slightly more complex, as there are three band structures visible. The maximum final mass for this region is 27.7 tons at a TOF_1 of



140 days and a TOF_2 of 214 days. Again, this almost perfectly corresponds with the optimal solution found by the low-to-high TOF passed-on case in Table 17-2.

Figure 17-7: Zoom in on window 1 and 2 of point $[t_0, TOF] = [-10, 601]$.

As such, it can be reasoned that at the $[t_0, TOF] = [-10, 601]$ point, the sudden change in payload mass for the low-to-high TOF passed case is closely related to the sudden change in time of flight of the different legs. The latter can be attributed to the fact that at this point, a jump occurs from one **local optimum** to another. Based on this discussion, it is known that the optimization gets stuck in local optima. In the next section, a multi-start method to overcome this problem will be developed and evaluated.

17-3 Multi-start method

The main idea of a multi-start method is to optimize the same launch date - total time of flight combination multiple times using different initial guesses. First of all, the working principles will be explained, followed by an assessment of the performance.

17-3-1 Working principles

For each new departure date, one starts at a low TOF. At each of those first points, four optimization runs are performed. From the previous optimization runs, it is known that the time of flight of the first leg varies between 50 and 150 days. The time of flight of the second leg varies between 140 and 260 days. Therefore, the four different initial guesses for TOF_1 and TOF_2 are 50 and 140 days, 50 and 260 days, 150 and 140 days and 150 and 260 days. For each point, these four optimization runs are performed. Upon finishing these optimization runs, it is checked if at least one of those runs produces a feasible trajectory. If this is not the case, the optimizer moves on to the next point and tries those four combinations of initial guesses again. If one or more feasible trajectories are found, the state vector resulting in the best trajectory is stored and passed on to the next point. There, this state vector is used as initial guess. Also, the four initial guess cases are run at this point.

As such, at each point, the problem is fed by four initial guesses at four different boundaries of the $\text{TOF}_1\text{-}\text{TOF}_2$ design space. This increases the chance of finding the different local optima and identifying the best local optimum. Furthermore, the utilization of multiple runs allows the method to recover from numerical instabilities. Also, by passing on the best found solution to the next point, the smoothness of the mass profile is ensured. The results for a launch window in 2021 utilizing 20 kW with a maximum TOF of 621 days has been depicted in Figure 17-8. For each grid point, the departure date and TOF are fixed while the flyby dates are optimized.



Figure 17-8: Results multi-start method Earth-Venus-Mars-Earth mission 2021 - 20 kW - 4RL.

17-3-2 Analysis

One can see that the utilized multi-start method gives smoother results than the singlestart method. Furthermore, the detected final masses are higher. In this subsection, a more broad discussion on the performance of the utilized multi-start method will be given and the processes active within the grid will be explained.

First of all, the time of flight of the first leg and the departure C_3 have been plotted in Figures 17-9 and 17-10.



Figure 17-9: TOF₁ Earth-Venus-Mars-Earth mission 2021 - 20 kW - 4RL.



Figure 17-10: C₃ Earth-Venus-Mars-Earth mission 2021 - 20 kW - 4RL.

Figure 17-11 lists all the solutions from all the different initial guesses. The leftmost line for each departure date gives the solution based on the previous best. The second, third, fourth and fifth line are the results of the first, second, third and fourth initial guess respectively.



Figure 17-11: Results of the different runs Earth-Venus-Mars-Earth mission 2021 - 20 kW - 4RL.

On all these figures, four different area's A, B, C and D have been identified in which trajectories showcase similar characteristics. Furthermore, Point 1 up to 5 have been indicated. These are examples of points where a transitioning occurs between different areas or an indication of other interesting behavior. This labeling facilitates the upcoming discussions where the different area's and the transitionings between them will be discussed.

Area A Looking at Figure 17-11, one can see that the first optimal solution within the A area is found using the second initial guess: $[TOF_1, TOF_2] = [50, 260]$. This corresponds well with the low TOF_1 's that are found in this region. An example of such a first optimal solution for $[t_0, TOF] = [-40, 541]$ can be seen in Figure 17-12a. Note that this trajectory figure has been stripped down for clarity reasons. The legend and axis can be found in previous figures such as Figure 17-1b.

Looking at Figures 17-9 and 17-10, one can see that the TOF₁ drops and the C_3 rises for higher departure dates. This can be understood from Figures 17-12a up to 17-12c showing the trend from $[t_0, \text{TOF}] = [-40, 541]$ to $[t_0, \text{TOF}] = [0, 497]$ and $[t_0, \text{TOF}] = [20, 513]$. Comparing Figure 17-12b with the trajectory for $[t_0, \text{TOF}] = [-40, 541]$ in Figure 17-12a, one can see that the second and third leg hardly differ as the times of the Venus and Mars flyby are kept constant. However, the departure date is different requiring a different first leg. This explains why the TOF of the first leg decreases and why the C_3 needs to increase to enable this shorter leg. At one point, the TOF of the first leg cannot be further decreased to keep the Venus flyby constant due to limits on the achievable C_3 . Hence, the Venus flyby needs to be moved. Due to this resulting change in geometry, the trajectory cannot go as deep into Venus' orbit as before, as can be seen in Figure 17-12c. This decreases the deliverable thrust by the SEP system, which reduces the feasible final mass for the higher departure dates, as can be seen in Figure 17-8.



Figure 17-12: Trajectories within the A area.

Area A transitioning into area D The aforementioned trend in area A for increasing departure dates is unsustainable for two reasons. One is the ever-increasing C_3 requirement caused by the decrease in TOF of the first leg. The second is the reduction of SEP capabilities resulting from the increasing minimum heliocentric distance. Hence, trajectories of the same family as those in area A become infeasible between departure date 20 and 30. Here, a transitioning occurs into a new area D. One can see from Figure 17-9 that in this area D, the TOF of the first leg is larger than in the A area. This increase in TOF_1 alleviates the large C_3 requirement.

Area C: transitioning from area A through Point 1 From Figure 17-9, one can see that there is a sudden jump in the TOF of the first leg occuring at Point 1. In order to understand the processes active here, one needs to look at a zoom in on Figure 17-11: Figure 17-13.

Final mass [ton]



Departure date 0

Figure 17-13: Visualization of the multi-start method around Point 1.

From this figure, one can see that the first feasible solution for departure date 0 is found using the second initial guess at a TOF of 497 days. This solution is passed on throughout the A area, until it reaches the boundary at a TOF of 535 days. Meanwhile, initial guess 3 and 4 start producing different results at a TOF of 529 days. These solutions are very different from the previous found solutions and are characterized by a higher TOF of the first leg. However, those solutions have a lower final mass. Hence, the multi-start method correctly identifies the first solution family as the optimal. Upon increasing the TOF to 533 days, the second solution family converges on a higher final mass, but still smaller than the result of the first solution family. At 537 days, the second family improves further and converges onto a better solution than the first family. As such, the multi-start method makes the jump from one solution family to the next, explaining the sudden jump in TOF₁. Without this multi-start method, the first solution family would be passed on to higher TOF's. As can be seen from Figure 17-4a, this results in significantly lower final masses.

Area B: transitioning from area A through Point 2 For departure date -30, a first solution is found in area A at [-30,533]. This solution is passed on throughout area A until it reaches the boundary with area B between TOF 553 and 557 days. Here, a new family becomes optimal. This family of solutions in area B is characterized by a significantly higher C_3 . To understand what is happening, Figures 17-14a and 17-14b picture the trajectories for both families for a TOF of 557 days.



Figure 17-14: Difference in trajectories between first and second family at the boundary between area A and B through Point 2 where $[t_0, TOF] = [-30, 557]$.

As mentioned, the second family requires a higher C_3 . This higher C_3 is only possible since the first family has a considerable margin between the actual launch mass and the maximal possible launch mass; the first family trajectories have a low required C_3 and a low final mass, resulting in a margin of up to 14 tons. One would expect that this margin needs to go to zero to achieve optimal trajectories. The reason why this does not happen is two-fold.

First of all, the margin of 14 tons for the first family could theoretically be used to increase the final mass and hence the initial mass while maintaining the same low C_3 in the A area. If this would be feasible, it would mean that this identified A area leading up to the transitioning would be sub-optimal. However, increasing the final mass is impossible since significant thrusting is required between the Venus and Mars flyby as seen from Figure 17-14a. Using a higher final mass, the SEP system cannot provide enough thrust to connect the Venus and Mars flyby.

Secondly, the margin of 14 tons for the first family could theoretically be used to increase the C_3 on the lower TOF's trajectories in the A area. However, the required Venus flyby geometry for those second family trajectories causes the Martian flyby geometry to change as well. Figure 17-14b show the consequent large thrust requirements on the final leg. For the low TOF trajectories in area A, there is not enough time available to perform this thrusting.

The previous discussion explains why the second family only appears in area B and shows that area A is indeed optimal. However, one last peculiarity needs some explanation. From Figure 17-11, one can see that as soon as a member of this second family appears at 557 days using initial guess 4, it is better than the other family. One would expect a transitioning period where both families co-exist, as is the case for Point 1. Although it appears absent,

this transitioning period is actually there. At 549 days, one can see that initial condition 3 converges onto a solution. Upon checking this solution, it could be attributed to this second family. However, at 553 days, no solution is visible due to a numerical instability occurring at initial guess 3 and 4. Therefore, the solution at 557 days was fed back to 553 days and as expected, a solution of the second family appeared. This shows that there is indeed a transitioning period leading up to Point 2.

Area C: transitioning from area B through Point 3 Throughout the B area, the margin between feasible launch mass and actual launch mass gradually goes down to zero by maintaining a more or less constant C_3 while enabling a higher final mass. This causes another unsustainable trend: at one point, the required C_3 needs to be reduced to enable further growth in final mass. This happens on the boundary with area C, which is characterized by a lower C_3 . In order to understand what is happening, Figure 17-15 shows an example of the C-family. One can see that the TOF of the first leg becomes larger. Hence, there is more thrusting time on the first leg allowing a reduction in the required C_3 .



Figure 17-15: C-family trajectory at [t₀,TOF]=[-30, 577].

Again, some remark needs to be made about the transitioning period: the identified transitioning between 573 and 577 days is not entirely correct. For the 573-day scenario and initial guess 4, the optimizer does not converge onto the better C-type family with a TOF₁ of 155 days. Instead, it converges onto the sub-optimal B-type family with a TOF₁ of 128 days. This can be explained by looking at the set-up of this initial guess: the initial TOF₁ is 150 days. Probably, the optimizer senses an increase in final mass in both directions from this initial guess of 150 days. Instead of increasing the TOF₁ to 155 days, the optimizer ends up going in the wrong direction and gets stuck in a local optimum. This also explains why there is a considerable jump in final mass at this transitioning. In reality, this jump is smaller, as the transitioning actually occurs 4 days earlier than detected by the multi-start method. **Conclusion on transitioning from area A to B to C** So in conclusion, in area A, the trajectories have a large margin between feasible launch mass and actual mass. This margin cannot be reduced to zero by increasing the final mass, since the second leg requires significant thrusting. Then, in area B, this margin is reduced by utilizing a higher C_3 , which allows for a higher final mass, but requires more thrusting on the final leg. This made this trajectory type impossible or sub-optimal in area A. As the margin decreases towards zero and the final mass increases over the B area, the C_3 must be lowered, which happens in area C. However, this can only happen upon increasing the length of the first leg, which made this trajectory type impossible or sub-optimal in area B.

Area C: transitioning from area D through Point 4 The processes in this transitioning are very similar to the ones explained in the previous paragraphs. Therefore, they will not be discussed here.

Point 5 From Figure 17-11, the role of passing on the previous best solution becomes apparent. One can see that at the TOF's leading up to Point 5, the results from initial guess 4 and the results from the previous best solution are always very similar. However, at point 5, the optimizer converges onto a sub-optimal solution using initial guess 4. But, the run based on the previous best result does converge onto the optimal solution. Hence, the utilized multi-start method is able to recover from such issues and ensure a continuous mass profile.

17-3-3 Conclusion

In the previous subsection, the complex processes active throughout the grid have been explained. Furthermore, it has been explained why there are discrete jumps in TOF's of the different legs, C_3 , etc. It can be concluded that the multi-start method performs well. Using the multi-start method, different trajectory families are identified. Additionally, transitioning periods where different families co-exist but differ in final mass are visible in Figure 17-11. The points where a transitioning occurs from one family to another are also visible, explaining the location of the discrete jumps. This all results in a smooth mass distribution throughout the grid.

Part VI

Validation, results, conclusions and recommendations

Chapter 18

Validation

Throughout the author's internship and thesis work, continuous verification and validation efforts have been made. It is impossible to list all of those efforts. However, this chapter will summarize the main validation work performed.

18-1 Basic version code developed during author's internship

Propagation module The propagation module has been validated on many different levels. First of all, it has been checked that the state vector elements have been properly translated into the state of the spacecraft at the initial and final control node. Furthermore, the numerical integration of the STM elements and Cartesian coordinates and velocities has been validated comparing the output of several different integration steps to the output of other validated integration schemes such as ode45 in Matlab, a Python based version of the RK8(7)-13M integrator written by Jon Herman [2012], etc. Results identical to numerical precision were obtained, validating the implementation of this integration scheme. Finally, the mass update step has been validated by comparing the outputted mass profile throughout a trajectory to what would be expected based on the manoeuvre sizes from the state vector.

Constraints module The constraints module has been validated by manually calculating the constraint function values based on the validated output of the propagation module. Those function values corresponded to the values obtained from the constraints module within numerical precision.

Jacobian module The Jacobian module has been validated using a finite-difference check build in into SNOPT. Furthermore, a random sample survey has been performed where the selected derivatives have been checked by hand based on the validated propagation output.

Optimization program set-up and coupling with SNOPT Besides validating the main modules, also the coupling with SNOPT and the program set-up needed validation. Therefore,

the implementation of a simple optimization problem for which the answer is known has been evaluated: the constrained Rosen-Suzuki function [Hock and Schittkowski, 1981].

One-leg scenario The validated modules and SNOPT coupling have been combined into a single program. This basic version of the code has been validated by comparing its results for simple scenario's with the results from Jon Herman's code [Herman, 2012], which output has been validated by the Jet Propulsion Laboratory in the context of GTOC 6.

Two-leg EME scenario's, fixed TOF 501 days For the two-leg EME trajectories, first of all, the feasibility of the trajectories has been assessed using an independent propagation tool. This tool has been written by Jon Herman, which he successfully used to assess the feasibility of CCAR's solution to the 6th Global Trajectory Optimization Competition and which he modified for this application. This propagation was fed with the optimized output to quantify that it corresponded to a feasible trajectory. However, this feasibility check does not validate the global optimality of the found solution. Due to the large number of state vector elements, it is impossible to perform a grid search in every single state vector element to prove the global optimality of the solution. To increase the confidence in the found solution, optimization runs with a large number of different initial guesses have been performed. The runs that converged onto a solution all converged onto the same solution.

18-2 Time optimization capabilities

The results from the flyby-date optimized Earth-Mars-Earth have been validated by comparing them to the validated fixed-time grid searches. Furthermore, the time optimization capabilities have been validated through an Earth-Asteroid-Earth rendez-vous mission design study conducted for Lockheed Martin, shown in Appendix I. The results and launch windows have been compared to results obtained using EMTG, a low-thrust optimization tool developed by NASA's Goddard Space Flight Center, which uses evolutionary algorithms to increase its odds of identifying the global optimum without relying on an initial guess [Englander et al., 2014]. The identified launch windows from EMTG and the developed code are almost identical. Furthermore, comparing individual grid points, the obtained trajectories and thrust profile for both methods are nearly identical, including the optimized time epochs. Although both methods could potentially be wrong, the fact that they both converge onto an almost identical solution increases the confidence in the obtained results.

18-3 Additional legs/EVME

The automation of the addition of extra legs has been developed and extensively used during GTOC 7. The optimized trajectories have been assessed by an external jury, and have been proven to be feasible, as will be explained in Appendix H.

For the EVME trajectories, optimization runs with a large number of different initial guesses have been performed. During this process, it was realized that they did not converge onto the same solution due to local optima issues. This problem was circumvented by developing a multi-start method. This method has been validated by comparing its identified solution with the best solution found from the large number of runs with different initial guesses.

Chapter 19

Results

In this chapter, the final results will be produced using the developed methods described in the previous chapters i.e. multi-start. Each of the three Earth-Mars-Earth launch windows identified in Chapter 9 will be discussed in the next sections. This will be followed by a section in which an Earth-Venus-Mars-Earth launch window will be assessed. Finally, a summary on the obtained launch windows will be given.

19-1 Earth-Mars-Earth flyby mission in 2018

This scenario has a nominal launch date on the 5^{th} of January 2018, followed by a Martian flyby 227 days later and Earth return 274 days later. From Chapter 9, it is known that launch windows exist for both the 1RL and 4RL configuration of the SLS for missions with a total time of flight of 501 days using 10 kW of SEP. Launch windows for both configurations will be discussed in the next two subsections. For these optimization runs, the final mass, composed of the payload mass and the mass of the SEP system, is maximized.

19-1-1 SLS 4RL launch configuration

In this subsection, the Earth-Mars-Earth flyby mission in 2018 using the SLS 4RL configuration will be discussed. First of all, low-resolution results have been produced and are depicted in Figure 19-1. To establish this grid, at each departure date, the initial guess obtained from Copernicus for the reference 501 days scenario is used. If a feasible solution is found, it is passed on to a higher TOF. In this figure, several anomalies can be observed. These will be discussed in the next paragraphs.



Figure 19-1: Grid search EME - 2018 - 10 kW - 4RL, low resolution.

Maximum TOF drop-off at -80 The maximum TOF rises smoothly from departure date -120 up to -90. Around departure date -80, this trend suddenly stops; at -90, the maximum TOF is 553 days, while for -80, the maximum is 549 days. The explanation for this behavior can be deduced from Figures 19-2a and 19-2b, showing the feasible TOF_1 's for different TOF's. For the -80 departure date, there are two diverging windows. Among others, these are different in flyby altitude. For the lower TOF_1 window, the required flyby occurs at a slightly higher altitude than for the higher TOF_1 window. The lower TOF_1 window is however not present for the -90 departure date. This can be attributed to the 2000 km flyby altitude

upper limit. Due to the planetary geometry, less orbital change from the flyby is required at departure date -90. Hence, the optimal flyby occurs slightly higher. This phenomenon in combination with the higher flyby altitude requirement for the lower TOF_1 window causes the lower TOF_1 window to disappear.



Figure 19-2: Time of flight leg 1 vs. total time of flight at departure date -90 and -80.

The presence of this lower TOF_1 window at departure date -80 causes the sudden drop-off in maximum TOF. By passing on the solution from lower TOF's to higher TOF's, the optimizer gets stuck in the local optima at those lower TOF_1 's. When this low TOF_1 window stops

around 549 days, the optimizer does not go to the higher TOF_1 window, but returns an infeasibility. Figure 19-2b shows another issue with the performed grid search. The cyan squares are the results as obtained from Figure 19-1a. The green diamonds are the actual optimal flyby dates as obtained from the TOF_1 vs. TOF grid search. So, the optimizer did not return the actual optimal solutions, but got stuck in local optima.



Figure 19-3: Grid search EME - 2018 - 10 kW - 4RL, low resolution, increased launch window 1.

The inconvergence issue was tested by performing runs that start at a TOF of 555 days with an initial guess for TOF_1 of 340 days and passing on feasible solutions to lower TOF's for

departure date -80 and -70. The results of those runs have been added to the previous results in Figures 19-3a and 19-3b. One can see that at departure date -80, there are indeed launch opportunities for a TOF of 551 and 553 days for a TOF₁ of 340 days. At departure date -70, launch opportunities between 551 and 555 days TOF are found for a TOF₁ of around 280 days. Interestingly, the launch mass increases again, which is an indication that there might be a separate launch window above this window.



Figure 19-4: Grid search EME - 2018 - 10 kW - 4RL, low resolution, increased launch window 2.

Second launch window The existence of a second launch window is assessed by performing an additional run starting at at TOF of 621 days and passing on information from high to low TOF's. The results can be seen in Figures 19-4a and 19-4b. These figures show that there is indeed a second launch window that interferes with the first launch window.

To understand the dynamics behind the second launch window, a grid search in Figure 19-5 has been performed on departure date -50, where both launch windows meet.



Figure 19-5: Time of flight leg 1 vs. total time of flight at departure date -50.

One can see that there are two distinct windows. For a TOF between 500 and 549 days, the TOF₁ is more or less constant at 270 days. At the low TOF's, a substantial flyby effect is visible in Figure 19-6a along with a small thrust requirement on the second leg. For higher TOF's, the thruster requirement on the second leg becomes smaller by increasing the flyby altitude from 200 km to 2000 km while keeping the flyby date more or less constant. Ultimately, at a TOF of 523 days, a free return trajectory becomes feasible, as shown in Figure 19-6b. In order to continue this trend, the flyby altitude would need to keep increasing. However, this altitude is bounded at 2000 km. Hence, more thrust is required on the second leg to compensate for the sub-optimal flyby geometry. This can be seen comparing Figure 19-6b with 19-6c. The latter figure depicts the situation at the boundary of the first window at a TOF of 549 days. Here, the second leg needs continuous thrusting.

While the low TOF window ends at 549 days, a new higher TOF window starts here as well. Since the low TOF region ends, and the higher TOF region starts, with a final mass of 10 tons, there is only a jump in TOf and not in final mass. By increasing the time of flight of the first leg, the geometry changes, facilitating the return to Earth. At the boundary, the feasible TOF_1 space is limited. Upon further increasing the TOF, a larger coast arc becomes feasible, widening the feasible TOF_1 space again. An example of such a trajectory with a coast arc can be seen in Figure 19-6d. While in the first launch window, the TOF_1 remains more or less constant, the TOF_1 changes linear with the increase in TOF to continuously adjust to the change in the return geometry.



Figure 19-6: Trend in trajectories for increasing time of flight for departure date -50.

Maximum TOF drop-off at -30 The maximum TOF rises smoothly from 40 to -20. However, at -30, the upper limit on the TOF suddenly drops. The explanation for this behavior can be deduced from Figures 19-7a and 19-7b.

Similar to the drop-off at -80, the reason for the drop at -30 is due to the sudden stop in one of two diverging windows. By passing on the solution from lower TOF's to higher TOF's, the optimizer gets stuck in local optima at the high TOF_1 's. When this high TOF_1 window stops around 547 days, the optimizer does not switch over to the lower TOF_1 window, but outputs an infeasibility.



Figure 19-7: Time of flight leg 1 vs. total time of flight at departure date -30 and -20.

In order to find the feasible points past a TOF of 547 days, additional runs have been performed starting at a TOF of 561 days where the information is passed on from higher to lower TOF's with an initial guess of 220 days. The results can be seen in Figures 19-8a and 19-8b.

One can see that the expected upper limit for the TOF is indeed increased for departure date -50 and -40 with a TOF₁ of around 220 days. However, for departure date -70 and -60, the additional launch window at higher TOF's identified in Figures 19-4a and 19-4b become apparent again.


Figure 19-8: Grid search EME - 2018 - 10 kW - 4RL, low resolution, increased launch window 3.

Figure 19-7a shows another issue with the performed grid search: the optimizer did not return the actual optimal solutions for TOF 537 up to 553 days. The same thing happens at departure date -20 for TOF's between 535 and 537 days, explaining the peculiar jump in final mass in Figure 19-1a.

High-resolution run Now that it is known what is happening at the high TOF's, a high-resolution run will be performed. The result for 38000 feasible trajectories can be seen in Figure 19-9. Similar features as in Figure 19-1a can be observed with sudden drops around departure date -80 and -20. Also, interference from the second launch window can be observed.



Figure 19-9: Grid search EME - 2018 - 10 kW - 4RL, high resolution.

Based on the knowledge obtained from the low resolution runs, additional runs have been performed to complete the figure and to find a better local optimum wherever it is known that the initial run returns an inferior local optimum. Note that it has been decided to filter out any results from the additional launch window since this window is characterized by a lower payload mass and is therefore of no interest. The results can be seen in Figure 19-10.



Figure 19-10: Grid search EME - 2018 - 10 kW - 4RL, high resolution, cleaned.

Important to note is that instead of plotting the final mass, it was opted to display the payload mass, which is equal to the final mass minus 300 kg of SEP system mass. This facilitates

reading the size of a launch window for a certain lower bound of payload mass. For a minimal payload mass of 10 tons, this launch window is 169 days long between the 6th of September, 2017 and the 22nd of February, 2018. However, 10 tons is insufficient. Remembering from Chapter 9 that the baseline mass of the Inspiration Mars mission is 13.139 tons, the launch window becomes 136 days long between the 1st of October, 2017 and the 14th of February, 2018. If on the other hand the fully margined mass of 19 tons is used, the launch window is even further narrowed down to 65 days between the 13th of November, 2017 and the 6th of February, 2018. A launch window of 21 days between the 3th of January, 2018 and the 24th of January, 2018 can be found for a payload mass of 21.8 tons.

Analysis In Section 15-1, an extensive analysis of the results throughout the departure dates for a TOF of 501 days has been given. In this paragraph, the validity of this analysis throughout a larger range of TOF's will be assessed. For this discussion, Figures 19-11, 19-12, 19-13 and 19-14 plotting the TOF_1 , C_3 , hyperbolic excess return velocity and flyby altitude are required.

The observed trends for 501 days re-occur throughout the entire grid. For reasons explained in Section 15-1, for increasing departure date, the time of flight of the first leg decreases, the C_3 decreases and the return hyperbolic excess velocity increases.

Besides trends for increasing departure dates, some trends on a constant departure date can be observed. For the majority of the departure dates, the time of flight of the first legs remains more or less constant throughout the feasible range of time of flights. This trend has already been explained for departure date -50 in Figure 19-5 by explaining how the flyby altitude increases for higher TOF's. This trend can be seen for other departure dates in Figure 19-14.



Figure 19-11: Grid search EME - 2018 - 10 kW - 4RL, high resolution, TOF1.



Figure 19-12: Grid search EME - 2018 - 10 kW - 4RL, high resolution, C₃.





However, regions with sudden changes in the time of flight of the first leg also exist. These are due to the artificial 'multi-start' method utilized to produce the figure. From Figure 19-2b, it is known that at departure date -80, the 17^{th} of October, 2017, at a TOF of 543, the higher TOF₁ window is a better local optima. This point is exactly where the transitioning occurs in Figure 19-11. Similarly, from Figure 19-7a, it is known that at departure date -30, the 6^{th} of December, 2017, the transitioning from the high TOF₁ window to the low TOF₁

window occurs at a TOF of approximately 535 days. Indeed, in Figure 19-11, the transitioning occurs at this TOF. This shows that the artificially established multi-start, high-resolution run, resolves the problems with local convergence explained in the maximum TOF drop-off at -80 and -30 paragraphs discussed earlier in this section.



Figure 19-14: Grid search EME - 2018 - 10 kW - 4RL, high resolution, h_{fly}.

Conclusion For this specific problem set-up, there are some issues with local optima. However, using an artificial multi-start method on the regions known to suffer from local optima, those issues can be resolved. This selective multi-start method requires previous knowledge on where those problems arise and how the initial guess needs to be adjusted to solve this problem. It can be reasoned that this is not the most robust set-up. A more generic multistart method could be written. However, due to physical time constraints, it has been decided not to do so. Furthermore, the problematic regions are furthest from the centre where the payload masses are highest. Hence for this scenario, the implementation of the more generic multi-start method would only provide improvements to uninteresting regions.

19-1-2 SLS 1RL launch configuration

Based on the large launch window established in the previous subsection for the 2018 - 10 kW - 4RL scenario, it has been realized that the more complex 4RL configuration of the SLS might not be required to fulfill this mission. Therefore, in this subsection, a similar analysis will be performed, but this time with the 1RL configuration of the SLS. The results of this analysis can be seen in Figure 19-15.

In this figure, one can recognize similar problems as for the 4RL scenario such as an abrupt drop in maximum TOF at a departure date around the 15th of December, 2017. Again, these

problems occur in the least interesting area where the maximum possible payload masses are low. Therefore, it has been decided to ignore the numerical instabilities at the high TOF's, but zoom in on the region with a minimal payload mass of 13.139 ton, the baseline mass of the Inspiration Mars mission. The result of this zoom can be seen in Figure 19-16.



Figure 19-15: Grid search EME - 2018 - 10 kW - 1RL.



Figure 19-16: Grid search EME - 2018 - 10 kW - 1RL, payload mass \ge 13.139 ton.

One can see that the baseline mass of the Inspiration Mars window, 13.139 tons, can be launched during a 60-day long launch window between the 9^{th} of December, 2017 and the 7^{th}

of February, 2018. The maximum attainable payload mass is about 14.25 tons at the 17th of January, 2018. This is the same date as the optimal date for the 4RL configuration. This does not come as a surprise, looking at the launch curves in Figure ??. The optimal trajectory for the 4RL configuration requires a C_3 of around 30 km²/s². Using the 1RL, around 15 tons can be launched for this C_3 value. Hence, the same optimal geometry can be flown, resulting in the same optimal departure date. A launch window of 22 days between the 5th of January, 2018 and the 27th of January, 2018 can be found for a payload mass of 14.1 tons. Hence, the fully margined mass of 19 tons is not attainable using the 1RL configuration and 10 kW.

One potential way to increase the maximum attainable payload mass to the fully margined mass of 19 tons could be to increase the power level. However, for the 1RL configuration, this does not work. The reason for this is that for a C_3 of $0 \text{ km}^2/\text{s}^2$, the 1RL configuration is only capable of launching 23.43 tons. Assuming a payload mass of 19 tons and 300 kg for a 10 kW SEP system, the maximum attainable C_3 would be around 14 km²/s². This is much lower than the C_3 of 37.45 km²/s² identified for the nominal scenario in Table 9-3. Furthermore, this assumes that no SEP propellant is being used at all. If a 10 kW SEP system would be constantly active, assuming a specific impulse of 2000 seconds and a TOF of 501 days with a 90% duty cycle, the maximum propellant it could expell during that time would be approximately 1200 kg, based on Equation 15-1. That means that the maximum launch mass would be 20.5 tons, which limits the maximum launch C_3 even further to 10 km²/s². For a 20 kW SEP system, the maximum SEP propellant would be approximately 2400 kg. This results in a maximum launch mass of approximately 22 tons, which means the maximum launch C_3 becomes 5 km²/s². Although the 20 kW SEP system is capabable of providing a larger ΔV , the low maximum attainable launch C_3 makes the launch of 19 tons of payload impossible for the 1RL configuration, even with a higher power level.

19-2 Earth-Mars-Earth flyby mission in 2019

From Chapter 9, it is known that this launch window only exists for the 4RL configuration. Hence, only this launcher configuration will be discussed here. Based on the lessons learned in the previous section, it was decided to start with a broad grid search to identify the presence of one or more potential launch windows. The results of this grid search can be found in Figure 19-17.

In this figure, one can see that again, there are two launch windows; the main launch window at lower TOF's where the final masses are highest and a second launch window at higher TOF's with lower maximal final masses. Based on this broad grid search, it has been decided to zoom in onto the main launch window at lower TOF's, since the maximized final masses are the highest in this region. The results of this zoom-in can be seen in Figure 19-18.

Similar to the 2018 - 10kW - 4RL scenario, one can observe a notch at the high end of the TOF spectrum. This is an indication of a problem at high TOF's: the launch window at the left of the notch is cut off too early. This is probably caused by a similar local optima issue as in the "Maximum TOF drop-off at -30" paragraph on page 123. Furthermore, the high TOF in combination with a low final payload mass does not make for a very interesting region. Therefore, these issues can be neglected.



Figure 19-17: Grid search EME - 2019 - 10 kW - 4RL, broad search.



Figure 19-18: Grid search EME - 2019 - 10 kW - 4RL, zoom in on main window.

One can see that the baseline mass of the Inspiration Mars window, 13.139 tons, can be launched during a 119 day long launch window between the 19^{th} of November, 2019 and the 17^{th} of March, 2020. The maximum attainable payload mass is about 17 tons. A launch window of 24 days between the 7^{th} of February, 2020 and the 2^{nd} of March, 2020 can be found for a payload mass of 17.2 tons.

As the fully margined mass of 19 tons is not attainable using the 4RL configuration and 10 kW, it has been checked if the fully margined mass would be attainable upon increasing the power level. Therefore, another run for 25 kW has been performed, which can be found in Figure 19-19.



Figure 19-19: Grid search EME - 2019 - 25 kW - 4RL, zoom in on main window.

Increasing the power level to 25 kW results in a feasible launch window for the fully margined payload mass of 19 tons. The launch window is approximately 74 days long between the 27th of December, 2019 and the 10th of March, 2020. The maximum attainable payload mass is about 19.35 tons. A launch window of 25 days between the 26th of January, 2020 and the 20th of February, 2020 can be found for a payload mass of 19.3 tons.

One can further conclude that increasing the power level has more effects than just increasing the final mass. It also has consequences on the shape of the contour plot and on the location of the maximum final mass. If there would be a large margin between the maximal feasible launch mass and the actual launch mass at the optimal location, the optimal geometry could be flown for the scenario with a higher power level and higher final mass. However, in this scenario, this margin is small at the location of the maximum final mass. Hence, upon increasing the power level and final mass, the C_3 needs to be reduced. This means that partially, the increased SEP capabilities from the higher power level are used to compensate for this loss in C_3 . The other part can be used to increase the final mass. For the 10 kW scenario, the optimal scenario requires a C_3 of 45 km²/s². However, for the 25 kW scenario, this is reduced to 34 km²/s². Because of this reduction in C_3 , the time of flight of the first leg needs to increase from 210 to 230 days. This change in flyby geometry causes the difference in shape of the contour plot and the change in location of the maximum final mass.

19-3 Earth-Mars-Earth flyby mission in 2021

First of all, runs have been performed using 10 kW. However, just like observed in De Smet et al. [2014], no launch windows could be found. Therefore, the power level has been increased to 20 kW. The results of this run can be seen in Figure 19-20.



Figure 19-20: Grid search EME - 2021 - 20 kW - 4RL, broad search.

In this figure, one can identify two launch windows: the main launch window at higher TOF's and an interesting region with low TOF's. In order to understand what happens at these low TOF's, two reference trajectories at the 25th of November, 2021 have been plotted in Figures



(a) Main launch window: TOF=501 days.

(b) Second launch window: TOF=391 days.

Figure 19-21: Trajectories of two different Earth-Mars-Earth scenario's launching on the 25th of November, 2021 - 20kW - 4RL.

19-21a and 19-21b. One is situated in the main launch window with a total TOF of 501 days. The other one is located in the second window at a total TOF of 391 days. From these figures, the presence of the unexpected launch window at low total TOF's can be understood. While the first legs between Earth and Mars are almost identical for both situations, there is a significant difference in the return legs. The much shorter return leg for the shorter TOF mission is a free return trajectory, while the longer return leg geometry for the main launch window scenario cannot be performed on a free return trajectory.

Unfortunately, this second, low TOF, launch window does not achieve the minimal baseline payload mass of 13.139 tons. Therefore, a zoom in on this launch window for a scenario with 25 kW of power has been run which can be seen in Figure 19-22. Using 25 kW, a 28 day long launch window can be identified between the 6th of November, 2021 and the 4th of December, 2021 for low TOF's ranging between 380 and 445 days. An obvious feature is a very abrupt change in payload mass at the lowest TOF for each departure date. This is due to an abrupt required change in flyby altitude and required C_3 value, which limits the payload mass. The transitioning appears to be very abrupt due to the utilized resolution. In reality, it is more smooth.



Figure 19-22: Grid search EME - 2021 - 25 kW - 4RL, second launch window.

This low TOF launch window however does not have the highest payload mass. For this 25 kW case, the main launch window has a 25 day window between the 30^{th} of December, 2021 and the 25^{th} of January, 2022 for a payload mass of 14.5 tons. Similarly for the 20 kW scenario, Figure 19-23 shows a zoom in on this main launch window region. One can see that the baseline mass of the Inspiration Mars window, 13.139 tons, can be launched during a 63 day long launch window between the 23^{th} of December, 2021 and the 24^{th} of February, 2022. The maximum attainable payload mass is about 13.5 tons. A launch window of 21 days between the 15^{th} of January, 2022 and the 5^{th} of February, 2022 can be found for a payload mass of 13.47 tons.



Figure 19-23: Grid search EME - 2021 - 20 kW - 4RL, main launch window.

19-4 Earth-Venus-Mars-Earth flyby mission in 2021

The results of a multi-start grid search for an Earth-Venus-Mars-Earth flyby mission in 2021 can be seen in Figure 19-24. This figure is an expanded, higher resolution version of Figure 17-8. Again, the TOF has been cut off at 621 days, as this is a human mission. The trends within this figure for departure dates between -50, the 2nd of October, 2021 and 40, the 31th of December, 2021, have already been extensively discussed in Section 17-3. However, some interesting features in this figure still need some consideration.

First of all, from Figure 19-25, it can be seen that area A and C do no longer appear for departure dates lower than -55, the 27^{th} of September, 2021. This could be expected from the trends observed in Figure 17-9. At lower departure dates, the TOF₁'s of area A increase and equal the TOF₁'s from area B. Furthermore, the geometry at departure date -55 requires a higher C_3 . Hence, the final mass remains lower and the jump to the C area can no longer be made. As such, for departure dates lower than -55, all solutions are part of area B.

Another interesting feature is the gap at departure date 25, the 16th of December, 2021. Looking back to Figure 17-9, this gap appears exactly where the boundary between area A and D occurs. This confirms the transitioning from area A into D at this location.

Another striking feature is the sudden cut-off at departure date 50, the 10^{th} of January, 2022 where the maximum TOF is 605 days. The reasons for this behavior are the very high required C_3 values at this departure date. The optimal TOF here is around 580 days. For both higher and lower TOF's, the required propellant mass increase. Due to the very high C_3 requirements here, there is no margin between the maximum feasible launch mass and the actual launch mass. Therefore, to increase the propellant mass and enable the mission, the final mass has to drop. Finally, at a TOF of 557 and 609 days, the final mass drops below 10 tons. Hence, the trajectory becomes infeasible there.



Figure 19-25: Grid search EVME - 2021 - 20kW - 4RL, TOF₁.

A launch window of 20 days between the 17th of October, 2021 and the 6th of November, 2021 can be found for a payload mass of 27.7 tons. However, this launch window has TOF's between 573 and 609 days. Considering this thesis work is performed for crewed missions, it will be investigated if feasible launch windows for lower total TOF's exist. Therefore, in Figure 19-26, one can find the launch window for a minimal payload mass of 19 tons. In this figure, one can see that with a minimal time of flight of 543 days, 20 day long launch periods open up between the 21th of November, 2021 and the 11th of December, 2021. Similarly, for

13.139 tons, a 25-day period opens up between the $16^{\rm th}$ of November, 2021 and the $11^{\rm th}$ of December, 2021 for a TOF of 517 days.



Figure 19-26: Grid search EVME - 2021 - 20 kW - 4RL, payload mass \ge 19 ton.

19-5 Conclusion

In order to give a quick overview of the different scenarios, the combination of parameters that lead to preliminary launch windows of approximately 20 days have been listed in Table 19-1. The low-thrust, 501 days TOF payload masses have been obtained from Table 9-5 while the chemical payload masses have been obtained using a similar method as in Section 9-6.

Scenario	Power	Launcher	Payload mass	Payload mass	Payload mass	TOF
	level	configur-	low-thrust	low-thrust	chemical	range
		ation	free TOF	501 days TOF	501 days TOF	
	(kW)		(tons)	(tons)	(tons)	(days)
Earth-Mars-Earth scenarios						
2018	10	4RL	21.8	21.2	20.95	500-511
2018	10	1RL	14.1	13.75	12.55	501 - 513
2019	10	4RL	17.2	15.75	10.49	509-519
2019	25	4RL	19.3	N.A.	10.49	516-524
2021	20	4RL	13.4	N.A.	8.42	538-547
2021	25	4RL	14.6	13.3	8.42	535 - 545
2021	25	4RL	13.3	13.3	8.42	379-433
Earth-Venus-Mars-Earth scenarios						
2021	20	4RL	27.7	N.A.	3.12	573-609
2021	20	4RL	19.0	N.A.	3.12	543
2021	20	4RL	13.1	N.A.	3.12	517

Table 19-1: Limiting cases: 20 days launch window for variable TOF's.



Figure 19-27: Comparison maximal payload masses for chemical and low-thrust trajectories for Earth-Mars-Earth missions in 2018, 2019 and 2021 and for an Earth-Venus-Mars-Earth mission in 2021 for different power levels and different launch configurations for free TOF's.

Comparison with chemical trajectories The results in Table 19-1 have also been visualized in Figure 19-27. In this figure, the payload mass for the low-thrust, free TOF cases and for the chemical reference trajectories have been plotted on the x-and y-axis respectively. From this figure, one can see that the trajectories utilizing SEP have a consistently higher feasible payload mass than the chemical trajectories. One can also see that the 4RL configuration results in larger payload masses than for the 1RL configuration. This does not comes as a surprise since the maximum launch mass in the launch performance curve of the 4RL configuration is substantially higher than the 1RL configuration. In this figure, one can also see that for the investigated scenarios, an increase in power level also results in a higher low-thrust payload mass. This is not always the case, as explained in Subsection 19-1-2.

From this Figure, one can see that the free TOF, low-thrust scenarios have substantial higher payload masses than their chemical, 501 days TOF counterparts. For instance, for the EVME-2021-20kW-4RL scenario, the chemical payload mass is maximal 3.12 ton, while with lowthrust, this could be increased to 27.7 tons. This large difference can be partially attributed to the very high ΔV requirement for the chemical DSM's in Table 17-1. This results in a very high chemical propellant mass. This can be drastically reduced using the more mass-efficient, low-thrust propulsion system. The made comparison is not entirely fair, since the low-thrust scenario has a TOF of upto 609 days. It could well be that the chemical payload mass could be substantially higher if the TOF would be allowed to increase from 501 to 609 days.

Comparison with low-thrust, fixed 501 days TOF scenarios Table 19-1 and Figure 19-28 show that opening up the TOF for low-thrust trajectories increases the payload mass compared to low-thrust, 501 days fixed TOF trajectories. However, it must be noted that the TOF required for those higher payload masses are in general higher than 501 days. There is one exception though: for the EME-2021-25kW-4RL scenario, a launch window has been identified that has a similar payload mass as for the 501 days fixed TOF case, but at a substantially lower TOF between 379 and 433 days.



Figure 19-28: Comparison maximal payload masses for low-thrust, fixed 501 days TOF and variable-TOF, low-thrust trajectories for Earth-Mars-Earth missions in 2018, 2019 and 2021 for different power levels and different launch configurations.

Besides increasing the payload mass, also new launch windows are opened. An example is the EME-2021-20kW-4RL scenario. For a fixed TOF of 501 days, the maximal payload mass was found to be below the Inspiration Mars baseline payload mass of 13.139 tons. Hence, this window had been assessed infeasible. If the TOF is allowed to change, a launch window for a payload mass of 13.4 tons becomes feasible for a TOF between 538 and 547 days. Note that this payload mass is even slightly higher than the payload mass for the 501 days TOF fixed EME-2021-25kW-4RL scenario.

Besides increasing the payload mass and opening up new launch windows, also new mission concepts have been identified. The highest payload mass has been found for an EVME-2021-20kW-4RL. However, this requires a fairly high TOF of upto 609 days. For lower TOF's, the achievable payload mass drops considerably. For instance, for a TOF of 517 days, the maximal payload mass is decreased to 13.1 tons. Compared to the EME-2018-10kW-4RL scenario, which can achieve 21.8 tons for a lower TOF and only using half of the power, this might not be desirable.

Conclusion The large differences in payload mass, required power, required flight time, etc. make it difficult to make comparisons between the different scenarios and to draw conclusions. Depending on the exact mission requirements, different scenario's are more attractive. The following discussion should facilitate the scenario selection for potential future mission design. If high TOF's are acceptable, the EVME scenario results in the highest payload masses and has a higher scientific return, as it flybys an additional planet. If such high TOF's are not acceptable, the EME-2018-10kW-4RL scenario is capable of a payload of 21.8 tons while maintaining a fairly low TOF and power level. If the mission can be build close to the baseline payload mass of 13.139 tons, the EME-2021-25kW-4RL scenario is attractive due to its low TOF but requires a power level of 25 kW, potentially rendering it impossible.

Chapter 20

Conclusions

The conclusions have been categorized based on the different research goals and launch windows.

20-1 Research goal 1

- 1. Profiling is crucial for an efficient code. The most common bottleneck is the (de-) allocation of arrays. If possible, this needs to be done as high up in the function hierarchy. Furthermore, the (de-)allocation must be avoided within iterations and loops.
- 2. A throttled representation is sparser and faster. However, the re-distribution of information within the Jacobian introduces convergence problems. This could be solved by creating a better initial guess or by writing a smart multi-start method, which offsets the time gains. Hence, the more stable thrust representation is preferred.

20-2 Research goal 2

- 1. To connect planetary ephemeris with time, ephemeris constraints can be used. However, those are unstable. Therefore, the coordinates and velocities at the control nodes have been removed from the state vector. This avoids the usage of those unstable constraints and reduces the size of the optimization problem, making it faster.
- 2. The analytical derivatives of the constraints with respect to time are difficult to find. The main reason is the inapplicability of STM's to scenario's with different propagation time steps. Therefore, a forward-finite difference method has been written to calculate those derivatives with respect to time.
- 3. Communicating the sparsity pattern to SNOPT increases the speed considerably. The exact gain depends on the sparsity of the problem, largely determined by the number

of legs and the number of segments per leg, and on the number of infeasible iterations encountered during the optimization.

4. Maximizing the payload mass is slightly slower than minimizing the launch mass. However, the latter requires a computational run to determine the launch window for every different payload mass. When one maximizes the payload mass, the launch windows for different payload masses can be filtered out of the results of one single computational run.

20-3 Research goal 3

- 1. Time epochs have a very large effect on trajectories. Hence, the optimization of those time epochs introduces local optima problems. Those local optima problems can be overcome by using a multi-start method. Such a method uses initial guesses on the boundaries of the expected design space of the time of flights of the different legs.
- 2. The developed low-thrust optimization tool is versatile. It is easy to adapt the code to include different type of control nodes and to increase the number of legs using an automated approach based on an input file. Besides the main thesis goal of optimizing flyby missions, the tool has been used on asteroid rendez-vous missions, asteroid rendez-vous and return missions, multiple asteroid tour missions for GTOC7, etc.

20-4 Launch windows

- 1. A variable flight time increases the feasible payload mass compared to a fixed flight time of 501 days.
- 2. A variable flight time opens up previously impossible launch windows.
- 3. Increasing the power level does not always lead to an increase in maximum payload mass, as expected from Figure 3-3.
- 4. Multiple launch windows can be found above each other, grouped by flight time. The window around the initial guess is optimal for all considered cases.
- 5. Local optima issues arise for high flight times for E-M-E scenarios. Those could be resolved using a multi-start method. However, those regions are not interesting, as they have a high time of flight and a low maximum payload mass.
- 6. New mission concepts such as an additional Venus flyby open up new launch opportunities with higher payload masses. The highest feasible payload mass has been found for such an E-V-M-E mission. This launch window however relies on a high flight time of around 570-610 days.

Chapter 21

Recommendations

No work is truly ever finished. In this chapter, several interesting regions that could be further investigated will be listed, as well as some recommendations to facilitate these investigations.

21-1 Earth departure

In this study, Earth departure considerations have been neglected. Instead of depending entirely on the launch C_3 to leave the Earth system, other options could be considered that could potentially open up new opportunities. A few options are spiral escape trajectories, lunar flybys, low-energy trajectories to raise the orbit's apogee, etc. While it is expected that such options will result in a lower propellant mass than using a chemical upper stage, the time of flight will increase. This could potentially conflict with the time of flight limitations for a crewed mission. Other combinations such as launching a small crew vehicle to a larger cargo module near the Sphere of Influence could also be considered.

21-2 Flyby altitude limit

For the fixed 501 days transfers, the flyby altitude is close to the minimum flyby altitude. From this, it was incorrectly assumed that the upper flyby altitude limit does not have a considerable effect on the launch windows. However, for several scenarios, the maximum Martian flyby altitude is reached at higher TOF's. The enforced sub-optimal flyby geometry for higher TOF's increases the thrust requirements. This decreases the maximum payload mass or makes the trajectory even infeasible. Therefore, it is suggested that the 2000 km Martian flyby limit is re-evaluated. Additional research could be performed to determine the changes in the launch windows using different upper limits. The resulting changes could then be traded off against the difference in science value of the mission.

21-3 Higher fidelity of the trajectory design

The fidelity of the trajectory design could be increased. Among others, the force model could be expanded to include certain perturbations such as third-body effects, solar radiation pressure, relativistic effects, etc. In addition, existing engines could be modeled into the optimization. In this research, it was assumed that for all power levels, the specific impulse of the thruster is constant. However, in reality, the specific impulse achievable for a specific engine is dependent on the power level.

21-4 Global optimization

For a higher number of flybys, it is expected that the number of local optima will increase. This would make it difficult for a multi-start method at the boundaries of the solution space to find the global optimum. Furthermore, the required number of starts for a multi-start method at the boundaries of the solution space would increase exponentially with the number of legs. Therefore, it is suggested to investigate different global optimization algorithms such as genetic algorithms, particle swarm optimization, differential evolution, simulated annealing, monotonic basin hopping, ant colony optimization, etc.

21-5 Programming platform

At the time of writing, the developed tool only works on a Windows platform within a Visual Studio environment as the tool needs to communicate with the optimization package SNOPT. For now, all the libraries and projects have been compiled in Visual Studio and are dependent on a specific Visual Studio compiler. For future investigations, it would be beneficial to port the code to a Linux platform and compile the SNOPT libraries and projects using a standard compiler. Then, more powerful computers than the author's laptop could be used to run the code, improving the run time.

21-6 Optimization toolbox

During this research, extensive use has been made of the optimization package SNOPT. The usage of other optimization packages has never been considered. It is possible that different optimization packages are more suitable for this problem and are faster. One package that could be considered is IPOPT [Wächter and Biegler, 2006]. This package is open source, has excellent documentation and community support, support for multiple CPU core optimization and parallel solving on the GPU, can be tuned to specific problems to improve the performance, etc. Furthermore, it is written in C++, facilitating the connection to the developed tool written in C++.

Part VII

Appendices

Appendix A

TNC transformation

The TNC function has to convert $[\Delta V_T, \Delta V_N, \Delta V_C]$ into $[\Delta V_x, \Delta V_y, \Delta V_z]$. In order to do so, one first needs to know how a Tangential, Normal and Cross-track (TNC) reference frame is defined.

The TNC reference frame is centered at the satellite's Center of Mass (COM). The **tan-gential**, also known as **in-track direction** and **along-track direction** is parallel to the velocity vector. The **normal** direction lies in the orbital plane, perpendicular to the velocity vector. The **cross-track** direction is normal to the plane defined by the tangential and normal directions. Hence, it is parallel with the angular momentum vector [Vallado, 2003].

Using the propagated Cartesian coordinates and velocities, the TNC function calculates the unit vectors \hat{T} , \hat{N} and \hat{C} that determine the directions of the TNC reference system at that specific point.

In Vallado [2003], a method to convert the orientations from TNC to their Cartesian orientations can be found.

$$\begin{bmatrix} \Delta V_x \\ \Delta V_y \\ \Delta V_z \end{bmatrix} = \begin{bmatrix} T_i & N_i & C_i \\ T_j & N_j & C_j \\ T_k & N_k & C_k \end{bmatrix} \cdot \begin{bmatrix} \Delta V_T \\ \Delta V_N \\ \Delta V_T \end{bmatrix}$$
(A-1)

with

$$\hat{T} = \frac{\bar{r}}{|\bar{r}|}$$

$$\hat{C} = \frac{\bar{r} \times \bar{r}}{|\bar{r} \times \bar{r}|}$$

$$\hat{N} = \hat{T} \times \hat{C}$$
(A-2)

As such,

$$T_{i} = \frac{\dot{x}}{\sqrt{\dot{x}^{2} + \dot{y}^{2} + \dot{z}^{2}}}$$

$$T_{j} = \frac{\dot{y}}{\sqrt{\dot{x}^{2} + \dot{y}^{2} + \dot{z}^{2}}}$$

$$T_{k} = \frac{\dot{z}}{\sqrt{\dot{x}^{2} + \dot{y}^{2} + \dot{z}^{2}}}$$
(A-3)

$$N_{i} = \frac{x\dot{y}^{2} - y\dot{y}\dot{x} - \dot{x}\dot{z}z + x\dot{z}^{2}}{\sqrt{\dot{x}^{2} + \dot{y}^{2} + \dot{z}^{2}} \cdot \sqrt{\left(y\dot{z} - \dot{y}z\right)^{2} + \left(z\dot{x} - \dot{x}z\right)^{2} + \left(x\dot{y} - \dot{x}y\right)^{2}}}{y\dot{z}^{2} - z\dot{z}\dot{y} - \dot{x}\dot{y}x + y\dot{x}^{2}}}$$

$$N_{j} = \frac{y\dot{z}^{2} - z\dot{z}\dot{y} - \dot{x}\dot{y}x + y\dot{x}^{2}}{\sqrt{\dot{x}^{2} + \dot{y}^{2} + \dot{z}^{2}} \cdot \sqrt{\left(y\dot{z} - \dot{y}z\right)^{2} + \left(z\dot{x} - \dot{x}z\right)^{2} + \left(x\dot{y} - \dot{x}y\right)^{2}}}$$

$$N_{k} = \frac{z\dot{x}^{2} - x\dot{x}\dot{z} - \dot{y}\dot{z}y + z\dot{y}^{2}}{\sqrt{\dot{x}^{2} + \dot{y}^{2} + \dot{z}^{2}} \cdot \sqrt{\left(y\dot{z} - \dot{y}z\right)^{2} + \left(z\dot{x} - \dot{x}z\right)^{2} + \left(x\dot{y} - \dot{x}y\right)^{2}}}$$
(A-4)

$$C_{i} = \frac{y\dot{z} - \dot{y}z}{\sqrt{(y\dot{z} - \dot{y}z)^{2} + (z\dot{x} - \dot{x}z)^{2} + (x\dot{y} - \dot{x}y)^{2}}}}{\sqrt{(y\dot{z} - \dot{y}z)^{2} + (z\dot{x} - \dot{x}z)^{2} + (x\dot{y} - \dot{x}y)^{2}}}}{\sqrt{(y\dot{z} - \dot{y}z)^{2} + (z\dot{x} - \dot{x}z)^{2} + (x\dot{y} - \dot{x}y)^{2}}}}{\sqrt{(y\dot{z} - \dot{y}z)^{2} + (z\dot{x} - \dot{x}z)^{2} + (x\dot{y} - \dot{x}y)^{2}}}}$$
(A-5)

The elements T_i up to C_k can then be used to convert $[\Delta V_T, \Delta V_N, \Delta V_C]$ into $[\Delta V_x, \Delta V_y, \Delta V_z]$ using a simple matrix multiplication:

$$\begin{bmatrix} \Delta V_T \\ \Delta V_N \\ \Delta V_C \end{bmatrix} = \begin{bmatrix} T_i & T_j & T_k \\ N_i & N_j & N_k \\ C_i & C_j & C_k \end{bmatrix} \cdot \begin{bmatrix} \Delta V_x \\ \Delta V_y \\ \Delta V_z \end{bmatrix}$$
(A-6)

Appendix B

TNC transformation matrix

For the calculation of the Jacobian, a transformation matrix is required mapping how changes in Cartesian coordinates and velocities before a manoeuvre influence the change in Cartesian coordinates and velocities after the manoeuvre.

For an impulsive manoeuvre, the Cartesian coordinates do not change during a manoeuvre. So, a change in Cartesian coordinates before the manoeuvre has a one-on-one relation with the change in the Cartesian coordinates after the manoeuvre. This can be seen looking at the following equations where $^+$ indicates the state after a manoeuvre and $^-$ indicates the state before a manoeuvre.

$$\Delta x^{+} = \Delta x^{-}$$

$$\Delta y^{+} = \Delta y^{-}$$

$$\Delta z^{+} = \Delta z^{-}$$

(B-1)

A change in Cartesian coordinates and velocities before the manoeuvre changes the elements T_i up to C_k within Equations A-3, A-4 and A-5. Hence, they influence the change in Cartesian velocities after the manoeuvre.

$$\begin{array}{rcl} \Delta \dot{z}^{+} & = & \displaystyle \frac{\partial \dot{z}^{+}}{\partial x^{-}} \cdot \Delta x^{-} + \displaystyle \frac{\partial \dot{z}^{+}}{\partial y^{-}} \cdot \Delta y^{-} + \displaystyle \frac{\partial \dot{z}^{+}}{\partial z^{-}} \cdot \Delta z^{-} + \\ & & \displaystyle \frac{\partial \dot{z}^{+}}{\partial \dot{x}^{-}} \cdot \Delta \dot{x}^{-} + \displaystyle \frac{\partial \dot{z}^{+}}{\partial \dot{y}^{-}} \cdot \Delta \dot{y}^{-} + \displaystyle \frac{\partial \dot{z}^{+}}{\partial \dot{z}^{-}} \cdot \Delta \dot{z}^{-} \end{array}$$

The established equations can be combined into a single transformation matrix:

$$\begin{bmatrix} \Delta x^{+} \\ \Delta y^{+} \\ \Delta z^{+} \\ \Delta \dot{x}^{+} \\ \Delta \dot{y}^{+} \\ \Delta \dot{z}^{+} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ a & b & c & d & e & f \\ g & h & i & j & k & l \\ m & n & o & p & q & r \end{bmatrix} \begin{bmatrix} \Delta x^{-} \\ \Delta y^{-} \\ \Delta \dot{z}^{-} \\ \Delta \dot{x}^{-} \\ \Delta \dot{z}^{-} \end{bmatrix}$$
(B-3)

Considering that

$$\dot{x}^{+} = \dot{x}^{-} + T_{i}^{-} \Delta V_{T} + N_{i}^{-} \Delta V_{N} + C_{i}^{-} \Delta V_{C}
\dot{y}^{+} = \dot{y}^{-} + T_{j}^{-} \Delta V_{T} + N_{j}^{-} \Delta V_{N} + C_{j}^{-} \Delta V_{C}
\dot{z}^{+} = \dot{z}^{-} + T_{k}^{-} \Delta V_{T} + N_{k}^{-} \Delta V_{N} + C_{k}^{-} \Delta V_{C}$$
(B-4)

One can see that

$$\begin{aligned} a &= \frac{\partial \dot{x}^{+}}{\partial x^{-}} = \frac{\partial T_{i}^{-}}{\partial x^{-}} \Delta V_{T} + \frac{\partial N_{i}^{-}}{\partial x^{-}} \Delta V_{N} + \frac{\partial C_{i}^{-}}{\partial x^{-}} \Delta V_{C} \\ b &= \frac{\partial \dot{x}^{+}}{\partial y^{-}} = \frac{\partial T_{i}^{-}}{\partial y^{-}} \Delta V_{T} + \frac{\partial N_{i}^{-}}{\partial y^{-}} \Delta V_{N} + \frac{\partial C_{i}^{-}}{\partial z^{-}} \Delta V_{C} \\ c &= \frac{\partial \dot{x}^{+}}{\partial z^{-}} = \frac{\partial T_{i}^{-}}{\partial z^{-}} \Delta V_{T} + \frac{\partial N_{i}^{-}}{\partial z^{-}} \Delta V_{N} + \frac{\partial C_{i}^{-}}{\partial z^{-}} \Delta V_{C} \\ d &= \frac{\partial \dot{x}^{+}}{\partial \dot{x}^{-}} = \frac{\partial T_{i}^{-}}{\partial \dot{x}^{-}} \Delta V_{T} + \frac{\partial N_{i}^{-}}{\partial \dot{x}^{-}} \Delta V_{N} + \frac{\partial C_{i}^{-}}{\partial z^{-}} \Delta V_{C} \\ t &= \frac{\partial \dot{x}^{+}}{\partial \dot{y}^{-}} = \frac{\partial T_{i}^{-}}{\partial \dot{y}^{-}} \Delta V_{T} + \frac{\partial N_{i}^{-}}{\partial \dot{y}^{-}} \Delta V_{N} + \frac{\partial C_{i}^{-}}{\partial \dot{y}^{-}} \Delta V_{C} \\ f &= \frac{\partial \dot{x}^{+}}{\partial \dot{z}^{-}} = \frac{\partial T_{i}^{-}}{\partial \dot{z}^{-}} \Delta V_{T} + \frac{\partial N_{i}^{-}}{\partial \dot{z}^{-}} \Delta V_{N} + \frac{\partial C_{i}^{-}}{\partial \dot{z}^{-}} \Delta V_{C} \\ g &= \frac{\partial \dot{y}^{+}}{\partial x^{-}} = \frac{\partial T_{j}^{-}}{\partial x^{-}} \Delta V_{T} + \frac{\partial N_{j}^{-}}{\partial x^{-}} \Delta V_{N} + \frac{\partial C_{j}^{-}}{\partial x^{-}} \Delta V_{C} \\ h &= \frac{\partial \dot{y}^{+}}{\partial y^{-}} = \frac{\partial T_{j}^{-}}{\partial y^{-}} \Delta V_{T} + \frac{\partial N_{j}^{-}}{\partial y^{-}} \Delta V_{N} + \frac{\partial C_{j}^{-}}{\partial y^{-}} \Delta V_{C} \\ i &= \frac{\partial \dot{y}^{+}}{\partial z^{-}} = \frac{\partial T_{j}^{-}}{\partial z^{-}} \Delta V_{T} + \frac{\partial N_{j}^{-}}{\partial z^{-}} \Delta V_{N} + \frac{\partial C_{j}^{-}}{\partial z^{-}} \Delta V_{C} \\ k &= \frac{\partial \dot{y}^{+}}{\partial \dot{y}^{-}} = \frac{\partial T_{j}^{-}}{\partial \dot{x}^{-}} \Delta V_{T} + \frac{\partial N_{j}^{-}}{\partial \dot{x}^{-}} \Delta V_{N} + \frac{\partial C_{j}^{-}}{\partial \dot{x}^{-}} \Delta V_{C} \\ k &= \frac{\partial \dot{y}^{+}}{\partial \dot{y}^{-}} = \frac{\partial T_{j}^{-}}{\partial \dot{y}^{-}} \Delta V_{T} + \frac{\partial N_{j}^{-}}{\partial \dot{x}^{-}} \Delta V_{N} + \frac{\partial C_{j}^{-}}{\partial \dot{x}^{-}} \Delta V_{C} + 1 \\ l &= \frac{\partial \dot{y}^{+}}{\partial \dot{y}^{-}} = \frac{\partial T_{j}^{-}}{\partial \dot{y}^{-}} \Delta V_{T} + \frac{\partial N_{j}^{-}}{\partial \dot{y}^{-}} \Delta V_{N} + \frac{\partial C_{j}^{-}}{\partial \dot{y}^{-}} \Delta V_{C} \\ \end{array}$$

$$m = \frac{\partial \dot{z}^{+}}{\partial x^{-}} = \frac{\partial T_{k}^{-}}{\partial x^{-}} \Delta V_{T} + \frac{\partial N_{k}^{-}}{\partial x^{-}} \Delta V_{N} + \frac{\partial C_{k}^{-}}{\partial x^{-}} \Delta V_{C}$$

$$n = \frac{\partial \dot{z}^{+}}{\partial y^{-}} = \frac{\partial T_{k}^{-}}{\partial y^{-}} \Delta V_{T} + \frac{\partial N_{k}^{-}}{\partial y^{-}} \Delta V_{N} + \frac{\partial C_{k}^{-}}{\partial y^{-}} \Delta V_{C}$$

$$o = \frac{\partial \dot{z}^{+}}{\partial z^{-}} = \frac{\partial T_{k}^{-}}{\partial z^{-}} \Delta V_{T} + \frac{\partial N_{k}^{-}}{\partial z^{-}} \Delta V_{N} + \frac{\partial C_{k}^{-}}{\partial z^{-}} \Delta V_{C}$$

$$p = \frac{\partial \dot{z}^{+}}{\partial \dot{x}^{-}} = \frac{\partial T_{k}^{-}}{\partial \dot{x}^{-}} \Delta V_{T} + \frac{\partial N_{k}^{-}}{\partial \dot{x}^{-}} \Delta V_{N} + \frac{\partial C_{k}^{-}}{\partial \dot{x}^{-}} \Delta V_{C}$$

$$q = \frac{\partial \dot{z}^{+}}{\partial \dot{y}^{-}} = \frac{\partial T_{k}^{-}}{\partial \dot{y}^{-}} \Delta V_{T} + \frac{\partial N_{k}^{-}}{\partial \dot{y}^{-}} \Delta V_{N} + \frac{\partial C_{k}^{-}}{\partial \dot{y}^{-}} \Delta V_{C}$$

$$r = \frac{\partial \dot{z}^{+}}{\partial \dot{z}^{-}} = \frac{\partial T_{k}^{-}}{\partial \dot{z}^{-}} \Delta V_{T} + \frac{\partial N_{k}^{-}}{\partial \dot{z}^{-}} \Delta V_{N} + \frac{\partial C_{k}^{-}}{\partial \dot{z}^{-}} \Delta V_{C} + 1$$

So, in order to find the analytical expressions for a up to r, one needs the partial derivatives of T_i^- up to C_k^- with respect to \mathbf{x}^- , \mathbf{y}^- , \mathbf{z}^- , $\dot{\mathbf{x}}^-$, $\dot{\mathbf{y}}^-$ and $\dot{\mathbf{z}}^-$. In the next Equations, these partial derivatives will be listed. Do note that the - subscript has been omitted for clarity reasons. Furthermore, the following definitions will be used:

$$term_{1} = \dot{x}^{2} + \dot{y}^{2} + \dot{z}^{2} term_{2} = \left(y\dot{z} - \dot{y}z\right)^{2} + \left(z\dot{x} - \dot{x}z\right)^{2} + \left(x\dot{y} - \dot{x}y\right)^{2} term_{3} = x\dot{y}^{2} - y\dot{y}\dot{x} - \dot{x}\dot{z}z + x\dot{z}^{2} term_{4} = y\dot{z}^{2} - z\dot{z}\dot{y} - \dot{x}\dot{y}x + y\dot{x}^{2} term_{5} = z\dot{x}^{2} - x\dot{x}\dot{z} - \dot{y}\dot{z}y + z\dot{y}^{2}$$
 (B-8)

In Equations A-3 up to A-5, the definitions of T_i up to C_k can be found. Using elementary algebra, the following analytical derivatives could be derived:

$$\frac{\partial T_i}{\partial x} = \frac{\partial T_i}{\partial y} = \frac{\partial T_i}{\partial z} = 0$$
(B-9)

$$\frac{\partial T_i}{\partial \dot{x}} = \frac{\dot{y}^2 + \dot{z}^2}{\sqrt{\text{term}_1^3}} \tag{B-10}$$

$$\frac{\partial T_i}{\partial \dot{y}} = \frac{-\dot{x} \cdot \dot{y}}{\sqrt{\text{term}_1^3}} \tag{B-11}$$

$$\frac{\partial T_i}{\partial \dot{z}} = \frac{-\dot{x} \cdot \dot{z}}{\sqrt{\text{term}_1^3}} \tag{B-12}$$

$$\frac{\partial T_j}{\partial x} = \frac{\partial T_j}{\partial y} = \frac{\partial T_j}{\partial z} = 0$$
(B-13)

$$\frac{\partial T_j}{\partial \dot{x}} = \frac{-\dot{x} \cdot \dot{y}}{\sqrt{\text{term}_1^3}} \tag{B-14}$$

$$\frac{\partial T_j}{\partial \dot{y}} = \frac{\dot{x}^2 + \dot{z}^2}{\sqrt{\text{term}_1^3}} \tag{B-15}$$

$$\frac{\partial T_j}{\partial \dot{z}} = \frac{-\dot{y} \cdot \dot{z}}{\sqrt{\text{term}_1^3}} \tag{B-16}$$

$$\frac{\partial T_k}{\partial x} = \frac{\partial T_k}{\partial y} = \frac{\partial T_k}{\partial z} = 0$$
(B-17)

$$\frac{\partial T_k}{\partial \dot{x}} = \frac{-\dot{x} \cdot \dot{z}}{\sqrt{\text{term}_1^3}} \tag{B-18}$$

$$\frac{\partial T_k}{\partial \dot{y}} = \frac{-\dot{y} \cdot \dot{z}}{\sqrt{\text{term}_1^3}} \tag{B-19}$$

$$\frac{\partial T_k}{\partial \dot{z}} = \frac{\dot{x}^2 + \dot{y}^2}{\sqrt{\text{term}_1^3}} \tag{B-20}$$

$$\begin{aligned} \frac{\partial N_i}{\partial x} &= \frac{1}{\operatorname{term}_1 \cdot \operatorname{term}_2} \left[\sqrt{\operatorname{term}_1 \cdot \operatorname{term}_2} \cdot \left(\dot{y}^2 + \dot{z}^2 \right) - & (B-21) \right. \\ & \left. \operatorname{term}_3 \cdot \left(\sqrt{\frac{\operatorname{term}_1}{\operatorname{term}_2}} \cdot \left((z\dot{x} - x\dot{z}) \cdot (-\dot{z}) + (x\dot{y} - \dot{x}y)\dot{y} \right) \right) \right] \\ \frac{\partial N_i}{\partial y} &= \frac{1}{\operatorname{term}_1 \cdot \operatorname{term}_2} \left[\sqrt{\operatorname{term}_1 \cdot \operatorname{term}_2} \cdot \left(- \dot{y}\dot{x} \right) - & (B-22) \right. \\ & \left. \operatorname{term}_3 \cdot \left(\sqrt{\frac{\operatorname{term}_1}{\operatorname{term}_2}} \cdot \left((y\dot{z} - z\dot{y})\dot{z} + (x\dot{y} - \dot{x}y) \cdot (-\dot{x}) \right) \right) \right] \\ \frac{\partial N_i}{\partial z} &= \frac{1}{\operatorname{term}_1 \cdot \operatorname{term}_2} \left[\sqrt{\operatorname{term}_1 \cdot \operatorname{term}_2} \cdot \left(- \dot{x}\dot{z} \right) - & (B-23) \right. \\ & \left. \operatorname{term}_3 \cdot \left(\sqrt{\frac{\operatorname{term}_1}{\operatorname{term}_2}} \cdot \left((y\dot{z} - z\dot{y}) \cdot (-\dot{y}) + (z\dot{x} - \dot{z}x) \cdot \dot{x} \right) \right) \right] \\ \frac{\partial N_i}{\partial \dot{x}} &= \frac{1}{\operatorname{term}_1 \cdot \operatorname{term}_2} \left[\sqrt{\operatorname{term}_1 \cdot \operatorname{term}_2} \cdot \left(- y\dot{y} - z\dot{z} \right) - & (B-24) \right. \\ & \left. \operatorname{term}_3 \cdot \left(\sqrt{\frac{\operatorname{term}_2}{\operatorname{term}_1}} \cdot \dot{x} + \sqrt{\frac{\operatorname{term}_1}{\operatorname{term}_2}} \cdot \left((z\dot{x} - x\dot{z})z + (x\dot{y} - \dot{x}y) \cdot (-y) \right) \right) \right] \\ \frac{\partial N_i}{\partial \dot{y}} &= \frac{1}{\operatorname{term}_1 \cdot \operatorname{term}_2} \left[\sqrt{\operatorname{term}_1 \cdot \operatorname{term}_2} \cdot \left(- y\dot{x} + 2x\dot{y} \right) - & (B-25) \right. \\ & \left. \operatorname{term}_3 \cdot \left(\sqrt{\frac{\operatorname{term}_2}{\operatorname{term}_1}} \cdot \dot{y} + \sqrt{\frac{\operatorname{term}_1}{\operatorname{term}_2}} \cdot \left((y\dot{z} - z\dot{y}) \cdot (-z) + (x\dot{y} - \dot{x}y)x \right) \right) \right] \\ \frac{\partial N_i}{\partial \dot{z}} &= \frac{1}{\operatorname{term}_1 \cdot \operatorname{term}_2} \left[\sqrt{\operatorname{term}_1 \cdot \operatorname{term}_2} \cdot \left(- x\dot{z} + 2x\dot{z} \right) - & (B-26) \right. \end{aligned}$$

$$\operatorname{term}_{3} \cdot \left(\sqrt{\frac{\operatorname{term}_{2}}{\operatorname{term}_{1}}} \cdot \dot{z} + \sqrt{\frac{\operatorname{term}_{1}}{\operatorname{term}_{2}}} \cdot \left((y\dot{z} - z\dot{y})y + (z\dot{x} - \dot{z}x) \cdot (-x) \right) \right) \right]$$

$$\begin{aligned} \frac{\partial N_j}{\partial x} &= \frac{1}{\operatorname{term}_1 \cdot \operatorname{term}_2} \left[\sqrt{\operatorname{term}_1 \cdot \operatorname{term}_2} \cdot \left(-\dot{x}\dot{y} \right) - \right. & (B-27) \\ & \operatorname{term}_4 \cdot \left(\sqrt{\frac{\operatorname{term}_1}{\operatorname{term}_2}} \cdot \left((z\dot{x} - x\dot{z}) \cdot (-\dot{z}) + (x\dot{y} - \dot{x}y)\dot{y} \right) \right) \right] \\ \frac{\partial N_j}{\partial y} &= \frac{1}{\operatorname{term}_1 \cdot \operatorname{term}_2} \left[\sqrt{\operatorname{term}_1 \cdot \operatorname{term}_2} \cdot \left(\dot{x}^2 + \dot{z}^2 \right) - \right. & (B-28) \\ & \operatorname{term}_4 \cdot \left(\sqrt{\frac{\operatorname{term}_1}{\operatorname{term}_2}} \cdot \left((y\dot{z} - z\dot{y})\dot{z} + (x\dot{y} - \dot{x}y) \cdot (-\dot{x}) \right) \right) \right] \\ \frac{\partial N_j}{\partial z} &= \frac{1}{\operatorname{term}_1 \cdot \operatorname{term}_2} \left[\sqrt{\operatorname{term}_1 \cdot \operatorname{term}_2} \cdot \left(-\dot{y}\dot{z} \right) - \right. & (B-29) \\ & \operatorname{term}_4 \cdot \left(\sqrt{\frac{\operatorname{term}_1}{\operatorname{term}_2}} \cdot \left((y\dot{z} - z\dot{y}) \cdot (-\dot{y}) + (z\dot{x} - x\dot{z})\dot{x} \right) \right) \right] \\ \frac{\partial N_j}{\partial \dot{x}} &= \frac{1}{\operatorname{term}_1 \cdot \operatorname{term}_2} \left[\sqrt{\operatorname{term}_1 \cdot \operatorname{term}_2} \cdot \left(-x\dot{y} + 2y\dot{x} \right) - \right. & (B-30) \\ & \operatorname{term}_3 \cdot \left(\sqrt{\frac{\operatorname{term}_2}{\operatorname{term}_1}} \cdot \dot{x} + \sqrt{\frac{\operatorname{term}_1}{\operatorname{term}_2}} \cdot \left((z\dot{x} - x\dot{z})z + (x\dot{y} - \dot{x}y) \cdot (-y) \right) \right) \right] \\ \frac{\partial N_j}{\partial \dot{y}} &= \frac{1}{\operatorname{term}_1 \cdot \operatorname{term}_2} \left[\sqrt{\operatorname{term}_1 \cdot \operatorname{term}_2} \cdot \left(-z\dot{z} - x\dot{x} \right) - \right. & (B-31) \\ & \operatorname{term}_3 \cdot \left(\sqrt{\frac{\operatorname{term}_2}{\operatorname{term}_1}} \cdot \dot{y} + \sqrt{\frac{\operatorname{term}_1}{\operatorname{term}_2}} \cdot \left((y\dot{z} - z\dot{y}) \cdot (-z) + (x\dot{y} - \dot{x}y) x \right) \right) \right] \\ \frac{\partial N_j}{\partial \dot{z}} &= \frac{1}{\operatorname{term}_1 \cdot \operatorname{term}_2} \left[\sqrt{\operatorname{term}_1 \cdot \operatorname{term}_2} \cdot \left(-z\dot{y} + 2y\dot{z} \right) - \right. & (B-32) \\ & \operatorname{term}_3 \cdot \left(\sqrt{\frac{\operatorname{term}_2}{\operatorname{term}_1}} \cdot \dot{z} + \sqrt{\frac{\operatorname{term}_1}{\operatorname{term}_2}} \cdot \left((y\dot{z} - z\dot{y}) y + (z\dot{x} - \dot{z}x) \cdot (-x) \right) \right) \right) \right] \end{aligned}$$

$$\frac{\partial N_k}{\partial x} = \frac{1}{\operatorname{term}_1 \cdot \operatorname{term}_2} \left[\sqrt{\operatorname{term}_1 \cdot \operatorname{term}_2} \cdot \left(-\dot{x}\dot{z} \right) - \left(\operatorname{B-33} \right) \right]$$

$$\operatorname{term}_5 \cdot \left(\sqrt{\frac{\operatorname{term}_1}{\operatorname{term}_2}} \cdot \left((z\dot{x} - x\dot{z}) \cdot (-\dot{z}) + (x\dot{y} - \dot{x}y)\dot{y} \right) \right) \right]$$

$$\frac{\partial N_k}{\partial y} = \frac{1}{\operatorname{term}_1 \cdot \operatorname{term}_2} \left[\sqrt{\operatorname{term}_1 \cdot \operatorname{term}_2} \cdot \left(-\dot{y}\dot{z} \right) - \left(\operatorname{B-34} \right) \right]$$

$$\operatorname{B-34} \left(\operatorname{B-34} \right) = \frac{1}{\operatorname{term}_1 \cdot \operatorname{term}_2} \left[\sqrt{\operatorname{term}_1 \cdot \operatorname{term}_2} \cdot \left(-\dot{y}\dot{z} \right) - \left(\operatorname{B-34} \right) \right]$$

$$\operatorname{term}_{5} \cdot \left(\sqrt{\frac{\operatorname{term}_{1}}{\operatorname{term}_{2}}} \cdot \left((y\dot{z} - z\dot{y})\dot{z} + (x\dot{y} - \dot{x}y) \cdot (-\dot{x}) \right) \right) \right]$$

$$\frac{\partial N_{k}}{\partial z} = \frac{1}{\operatorname{term}_{1} \cdot \operatorname{term}_{2}} \left[\sqrt{\operatorname{term}_{1} \cdot \operatorname{term}_{2}} \cdot \left(\dot{x}^{2} + \dot{y}^{2} \right) - (B-35) \right]$$

$$\operatorname{term}_{5} \cdot \left(\sqrt{\frac{\operatorname{term}_{1}}{\operatorname{term}_{2}}} \cdot \left((y\dot{z} - z\dot{y}) \cdot (-\dot{y}) + (z\dot{x} - x\dot{z})\dot{x} \right) \right) \right]$$

$$\frac{\partial N_{k}}{\partial \dot{x}} = \frac{1}{\operatorname{term}_{1} \cdot \operatorname{term}_{2}} \left[\sqrt{\operatorname{term}_{1} \cdot \operatorname{term}_{2}} \cdot \left(-x\dot{z} + 2z\dot{x} \right) - (B-36) \right]$$

$$\operatorname{term}_{5} \cdot \left(\sqrt{\frac{\operatorname{term}_{2}}{\operatorname{term}_{1}}} \cdot \dot{x} + \sqrt{\frac{\operatorname{term}_{1}}{\operatorname{term}_{2}}} \cdot \left((z\dot{x} - x\dot{z})z + (x\dot{y} - \dot{x}y) \cdot (-y) \right) \right) \right]$$

$$\frac{\partial N_{k}}{\partial \dot{y}} = \frac{1}{\operatorname{term}_{1} \cdot \operatorname{term}_{2}} \left[\sqrt{\operatorname{term}_{1} \cdot \operatorname{term}_{2}} \cdot \left(-y\dot{z} + 2z\dot{y} \right) - (B-37) \right]$$

$$\operatorname{term}_{5} \cdot \left(\sqrt{\frac{\operatorname{term}_{2}}{\operatorname{term}_{1}}} \cdot \dot{y} + \sqrt{\frac{\operatorname{term}_{1}}{\operatorname{term}_{2}}} \cdot \left((y\dot{z} - z\dot{y}) \cdot (-z) + (x\dot{y} - \dot{x}y)x \right) \right) \right]$$

$$\frac{\partial N_{k}}{\partial \dot{z}} = \frac{1}{\operatorname{term}_{1} \cdot \operatorname{term}_{2}} \left[\sqrt{\operatorname{term}_{1} \cdot \operatorname{term}_{2}} \cdot \left(-x\dot{x} - y\dot{y} \right) - (B-38) \right]$$

$$\operatorname{term}_{5} \cdot \left(\sqrt{\frac{\operatorname{term}_{2}}{\operatorname{term}_{1}}} \cdot \dot{z} + \sqrt{\frac{\operatorname{term}_{1}}{\operatorname{term}_{2}}} \cdot \left((y\dot{z} - z\dot{y})y + (z\dot{x} - \dot{z}x) \cdot (-x) \right) \right) \right]$$

$$\frac{\partial C_i}{\partial x} = \frac{-\left(y\dot{z} - \dot{y}z\right)}{\sqrt{\text{term}_2^3}} \left[\left(z\dot{x} - \dot{z}x\right) \cdot \left(-\dot{z}\right) + \left(x\dot{y} - \dot{x}y\right)\dot{y} \right]$$
(B-39)

$$\frac{\partial C_i}{\partial y} = \frac{1}{\text{term}_2} \left[\sqrt{\text{term}_2} \cdot \dot{z} - \frac{y\dot{z} - \dot{y}z}{\sqrt{\text{term}_2}} \cdot \left(\left(y\dot{z} - \dot{y}z \right) \dot{z} + \left(x\dot{y} - \dot{x}y \right) \cdot (-\dot{x}) \right) \right]$$
(B-40)

$$\frac{\partial C_i}{\partial z} = \frac{1}{\text{term}_2} \left[\sqrt{\text{term}_2} \cdot (-\dot{y}) - \frac{y\dot{z} - \dot{y}z}{\sqrt{\text{term}_2}} \cdot \left(\left(y\dot{z} - \dot{y}z \right) \cdot (-\dot{y}) + \left(z\dot{x} - \dot{z}x \right) \dot{x} \right) \right] (\text{B-41})$$

$$\frac{\partial C_i}{\partial \dot{x}} = \frac{-(y\dot{z} - \dot{y}z)}{\sqrt{\text{term}_2^3}} \left[\left(z\dot{x} - \dot{z}x \right) \cdot z + \left(x\dot{y} - \dot{x}y \right) \cdot (-y) \right]$$
(B-42)

$$\frac{\partial C_i}{\partial \dot{y}} = \frac{1}{\text{term}_2} \left[\sqrt{\text{term}_2} \cdot (-z) - \frac{y\dot{z} - \dot{y}z}{\sqrt{\text{term}_2}} \cdot \left(\left(y\dot{z} - \dot{y}z \right) \cdot (-z) + \left(x\dot{y} - \dot{x}y \right) x \right) \right) \right] \text{B-43}$$

$$\frac{\partial C_i}{\partial \dot{z}} = \frac{1}{\text{term}_2} \left[\sqrt{\text{term}_2} \cdot y - \frac{y\dot{z} - \dot{y}z}{\sqrt{\text{term}_2}} \cdot \left(\left(y\dot{z} - \dot{y}z \right) y + \left(z\dot{x} - \dot{z}x \right) \cdot (-x) \right) \right] \quad (\text{B-44})$$

$$\frac{\partial C_j}{\partial x} = \frac{1}{\text{term}_2} \left[\sqrt{\text{term}_2} \cdot (-\dot{z}) - \frac{z\dot{x} - \dot{z}x}{\sqrt{\text{term}_2}} \cdot \left(\left(z\dot{x} - \dot{z}x \right) \cdot (-\dot{z}) + \left(x\dot{y} - \dot{x}y \right) \dot{y} \right) \right] (\text{B-45})$$

$$\frac{\partial C_j}{\partial y} = \frac{-\left(z\dot{x} - \dot{z}x\right)}{\sqrt{\text{term}_2^3}} \left[\left(y\dot{z} - \dot{y}z\right)\dot{z} + \left(x\dot{y} - \dot{x}y\right) \cdot (-\dot{x}) \right]$$
(B-46)

$$\frac{\partial C_j}{\partial z} = \frac{1}{\text{term}_2} \left[\sqrt{\text{term}_2} \cdot \dot{x} - \frac{z\dot{x} - \dot{z}x}{\sqrt{\text{term}_2}} \cdot \left(\left(y\dot{z} - \dot{y}z \right) \cdot \left(-\dot{y} \right) + \left(z\dot{x} - \dot{z}x \right)\dot{x} \right) \right] \quad (B-47)$$

$$\frac{\partial C_j}{\partial \dot{x}} = \frac{1}{\text{term}_2} \left[\sqrt{\text{term}_2} \cdot z - \frac{z\dot{x} - \dot{z}x}{\sqrt{\text{term}_2}} \cdot \left(\left(z\dot{x} - \dot{z}x \right) z + \left(x\dot{y} - \dot{x}y \right) \cdot (-y) \right) \right] \quad (B-48)$$

$$\frac{\partial C_j}{\partial \dot{y}} = \frac{-\left(z\dot{x} - \dot{z}x\right)}{\sqrt{\text{term}_2^3}} \left[\left(y\dot{z} - \dot{y}z\right) \cdot (-z) + \left(x\dot{y} - \dot{x}y\right)x) \right]$$
(B-49)

$$\frac{\partial C_j}{\partial \dot{z}} = \frac{1}{\text{term}_2} \left[\sqrt{\text{term}_2} \cdot (-x) - \frac{z\dot{x} - \dot{z}x}{\sqrt{\text{term}_2}} \cdot \left(\left(y\dot{z} - \dot{y}z \right)y + \left(z\dot{x} - \dot{z}x \right) \cdot (-x) \right) \right] (\text{B-50})$$

$$\frac{\partial C_k}{\partial x} = \frac{1}{\text{term}_2} \left[\sqrt{\text{term}_2} \cdot \dot{y} - \frac{x\dot{y} - \dot{x}y}{\sqrt{\text{term}_2}} \cdot \left(\left(z\dot{x} - \dot{z}x \right) \cdot (-\dot{z}) + \left(x\dot{y} - \dot{x}y \right) \dot{y} \right) \right] \quad (B-51)$$

$$\frac{\partial C_k}{\partial y} = \frac{1}{\text{term}_2} \left[\sqrt{\text{term}_2} \cdot (-\dot{x}) - \frac{x\dot{y} - \dot{x}y}{\sqrt{\text{term}_2}} \cdot \left(\left(y\dot{z} - \dot{y}z \right) \dot{z} + \left(x\dot{y} - \dot{x}y \right) \cdot (-\dot{y}) \right) \right] (B-52)$$

$$\frac{\partial C_k}{\partial z} = \frac{-\left(x\dot{y} - \dot{x}y\right)}{\sqrt{\text{term}_2^3}} \left[\left(y\dot{z} - \dot{y}z\right) \cdot \left(-\dot{y}\right) + \left(z\dot{x} - \dot{z}x\right)\dot{x} \right]$$
(B-53)

$$\frac{\partial C_k}{\partial \dot{x}} = \frac{1}{\text{term}_2} \left[\sqrt{\text{term}_2} \cdot (-y) - \frac{x\dot{y} - \dot{x}y}{\sqrt{\text{term}_2}} \cdot \left(\left(z\dot{x} - \dot{z}x \right) z + \left(x\dot{y} - \dot{x}y \right) \cdot (-y) \right) \right] \text{(B-54)}$$

$$\frac{\partial C_k}{\partial \dot{x}} = \frac{1}{\sqrt{\text{term}_2}} \left[\sqrt{\text{term}_2} \cdot x - \frac{x\dot{y} - \dot{x}y}{\sqrt{\text{term}_2}} \cdot \left(\left(y\dot{z} - \dot{y}z \right) \cdot (-z) + \left(x\dot{y} - \dot{x}y \right) y \right) \right] \text{(B-55)}$$

$$\overline{\partial \dot{y}} = \frac{1}{\operatorname{term}_2} \left[\sqrt{\operatorname{term}_2} \cdot x - \frac{1}{\sqrt{\operatorname{term}_2}} \cdot \left(\left(yz - yz \right) \cdot (-z) + \left(xy - xy \right) y \right) \right] \quad (B-55)$$

$$\partial C_i = -\left(x\dot{y} - \dot{x}y \right) \left[\zeta_i = -\left(x\dot{y} - \dot{x}y \right) \right] \quad (B-55)$$

$$\frac{\partial C_k}{\partial \dot{z}} = \frac{-(xy - xy)}{\sqrt{\text{term}_2^3}} \left[\left(y\dot{z} - \dot{y}z \right) y + \left(z\dot{x} - \dot{z}x \right) \cdot (-x) \right]$$
(B-56)

Appendix C

Jacobian

SNOPT is a gradient-based optimization program. Hence, it requires the derivative of every constraint with respect to each element of the state vector. Some derivatives have already been explained in Chapter 8. Here, an extensive overview of all derivatives will be given.

C-1 Cost function

The derivative(s) of the cost function depend(s) on the definition of the cost function, which depends on the goal of the optimization.

C-2 Mass match point $F_{\Delta M}$

Looking at the following equations for the mass at the match points, one can see that the mass mismatch at the match point is influenced by M_0 , M_f and all the ΔV elements and specific impulses.

$$M_{\text{match point, forward}} = M_0 \cdot exp\left(-\frac{\sum_{i=1}^N \frac{\Delta V_i}{I_{\text{sp},i}}}{g_0}\right)$$
 (C-1)

$$M_{\text{match point, backward}} = M_f \cdot exp\left(\frac{\sum_{i=N+1}^{2N} \frac{\Delta V_i}{I_{\text{sp},i}}}{g_0}\right)$$
(C-2)

Considering Equation 7-1, one can see that

$$\frac{\partial \Delta M}{\partial M_0} = \frac{M_{\text{match point, forward}}}{M_0} \tag{C-3}$$

$$\frac{\partial \Delta M}{\partial M_f} = -\frac{M_{\text{match point, backward}}}{M_f} \tag{C-4}$$

For the ΔV elements and specific impulses for the first N manoeuvres, representing the forward propagation, the derivatives are

$$\frac{\partial \Delta M}{\partial \Delta V_{\kappa,i}} = \frac{-M_{\text{match point, forward}}}{I_{\text{sp},i} \cdot g_0} \cdot \frac{\Delta V_{\kappa,i}}{\Delta V_i}$$
(C-5)

$$\frac{\partial \Delta M}{\partial \Delta I_{\rm sp,i}} = \frac{M_{\rm match point, forward} \cdot \Delta V_i}{I_{\rm sp,i}^2 \cdot g_0} \tag{C-6}$$

with i=1, 2, ..., N and κ =T, N or C.

For the ΔV elements and specific impulses for the last N manoeuvres, representing the backward propagation, the derivatives are

$$\frac{\partial \Delta M}{\partial \Delta V_{\kappa,i}} = \frac{-M_{\text{match point, backward}}}{I_{\text{sp},i} \cdot g_0} \cdot \frac{\Delta V_{\kappa,i}}{\Delta V_i}$$
(C-7)

$$\frac{\partial \Delta M}{\partial \Delta I_{\rm sp,i}} = \frac{M_{\rm match \ point, backward} \cdot \Delta V_i}{I_{\rm sp,i}^2 \cdot g_0}$$
(C-8)

with i=N+1, N+2, ..., 2N and κ =T, N or C.

C-3 State match point constraints $F_{\Delta x}$, $F_{\Delta y}$, $F_{\Delta z}$, $F_{\Delta x}$, $F_{\Delta y}$, $F_{\Delta y}$

The derivatives of the state match point constraints can be subdivided into four categories: derivatives with respect to the elements making up the initial state, elements making up the final state, elements defining the forward manoeuvres and elements defining the backward manoeuvres.

C-3-1 Derivatives with respect to the initial node's coordinates, velocities and hyperbolic excess velocities

To obtain the derivatives of the match point constraints with respect to the initial node's coordinates, velocities and excess velocities, one must understand how a change in those initial conditions is propagated up to the match point. This will be done through Figure C-1.

First of all, one needs to realize that the initial Cartesian state, indicated by the s for "spacecraft state" subscript, is set up by a combination of the initial node's coordinates, velocities and excess velocities. These are indicated by the 0 for "initial node" subscript, according to

initial node spacecraft state =
$$\begin{bmatrix} x_{s,0} \\ y_{s,0} \\ z_{s,0} \\ \dot{y}_{s,0} \\ \dot{y}_{s,0} \\ \dot{z}_{s,0} \end{bmatrix} = \begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ \dot{x}_0 + V_{\infty,x_0} \\ \dot{y}_0 + V_{\infty,y_0} \\ \dot{y}_0 + V_{\infty,y_0} \\ \dot{z}_0 + V_{\infty,y_0} \end{bmatrix}$$
(C-9)

$$\begin{bmatrix} \Delta x_s \\ \Delta y_s \\ \Delta z_s \\ \Delta \dot{x}_s \\ \Delta \dot{$$

Figure C-1: Propagation of a change in initial conditions to the match point.

This initial state is numerically propagated up to the state at point 1^- in Figure ??, indicating the Cartesian state before the application of the manoeuvre. From Equation 6-3, it is known that a change in the initial state can be converted into a change in the Cartesian state at point 1^- using the STM between point 0 and 1^- , $\Phi_{1^-,0}$, which has been calculated and stored in the propagation module. So,

$$\begin{bmatrix} \Delta x_s \\ \Delta y_s \\ \Delta z_s \\ \Delta z_s \\ \Delta \dot{x}_s \\ \Delta \dot{x}_s \end{bmatrix}_{1^{-}} = \Phi_{1^{-},0} \cdot \begin{bmatrix} \Delta x_s \\ \Delta y_s \\ \Delta z_s \\ \Delta \dot{x}_s \\ \Delta \dot{x}_s \\ \Delta \dot{y}_s \\ \Delta \dot{z}_s \end{bmatrix}_{0}$$
(C-10)

The directions of the manoeuvres expressed in TNC coordinates depend on the location of the manoeuvre. As such, a change in the state at point 1^- influences the directions in which the manoeuvres are applied. As such, an additional transformation matrix is required that transforms the change in the state over the manoeuvre from point 1^- to point 1^+ . Therefore, the transformation matrix $TNC_{1^+,1^-}$ explained in Appendix B can be used, which has been calculated and stored in the propagation module.

$$\begin{bmatrix} \Delta x_s \\ \Delta y_s \\ \Delta z_s \\ \Delta \dot{x}_s \\ \Delta \dot{y}_s \\ \Delta \dot{z}_s \end{bmatrix}_{1+} = TNC_{1+,1-} \cdot \begin{bmatrix} \Delta x_s \\ \Delta y_s \\ \Delta z_s \\ \Delta \dot{x}_s \\ \Delta \dot{x}_s \\ \Delta \dot{y}_s \\ \Delta \dot{z}_s \end{bmatrix}_{1-}$$
(C-11)

So, a change in the initial state can be propagated to a change of coordinates and velocities at the match point using:

$$\begin{bmatrix} \Delta x_s \\ \Delta y_s \\ \Delta z_s \\ \Delta \dot{x}_s \\ \Delta \dot{y}_s \\ \Delta \dot{z}_s \end{bmatrix}_{\text{MP,forward}} = \Phi_{MP,N^+} \cdot TNC_{N^+,N^-} \cdot \ldots \cdot \Phi_{2^-,1^+} \cdot TNC_{1^+,1^-} \cdot \Phi_{1^-,0} \cdot \begin{bmatrix} \Delta x_s \\ \Delta y_s \\ \Delta z_s \\ \Delta \dot{x}_s \\ \Delta \dot{y}_s \\ \Delta \dot{z}_s \end{bmatrix}_0 \quad (C-12)$$

This matrix product $\Phi_{MP,N^+} \cdot TNC_{N^+,N^-} \cdot \ldots \cdot \Phi_{2^-,1^+} \cdot TNC_{1^+,1^-} \cdot \Phi_{1^-,0}$ will be called $\Psi_{MP,0}$, mapping the change from state at point θ to the state at point MP.

From the previous discussion, the derivatives of the match point constraints with respect to the initial coordinates, velocities and initial hyperbolic excess velocities can be found by realizing that

$$\Psi_{MP,0} = \begin{bmatrix} \frac{\partial x_{s,MP \text{ for}}}{\partial x_{s,0}} & \frac{\partial x_{s,MP \text{ for}}}{\partial y_{s,0}} & \frac{\partial x_{s,MP \text{ for}}}{\partial z_{s,0}} & \frac{\partial x_{s,MP \text{ for}}}{\partial \dot{x}_{s,0}} & \frac{\partial x_{s,MP \text{ for}}}{\partial \dot{y}_{s,0}} & \frac{\partial x_{s,MP \text{ for}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial y_{s,MP \text{ for}}}{\partial x_{s,0}} & \frac{\partial y_{s,MP \text{ for}}}{\partial y_{s,0}} & \frac{\partial y_{s,MP \text{ for}}}{\partial z_{s,0}} & \frac{\partial y_{s,MP \text{ for}}}{\partial \dot{x}_{s,0}} & \frac{\partial y_{s,MP \text{ for}}}{\partial \dot{y}_{s,0}} & \frac{\partial y_{s,MP \text{ for}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial z_{s,MP \text{ for}}}{\partial x_{s,0}} & \frac{\partial z_{s,MP \text{ for}}}{\partial y_{s,0}} & \frac{\partial z_{s,MP \text{ for}}}{\partial z_{s,0}} & \frac{\partial z_{s,MP \text{ for}}}{\partial \dot{x}_{s,0}} & \frac{\partial z_{s,MP \text{ for}}}{\partial \dot{y}_{s,0}} & \frac{\partial z_{s,MP \text{ for}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial \dot{x}_{s,MP \text{ for}}}{\partial x_{s,0}} & \frac{\partial \dot{x}_{s,MP \text{ for}}}{\partial y_{s,0}} & \frac{\partial \dot{x}_{s,MP \text{ for}}}{\partial z_{s,0}} & \frac{\partial \dot{x}_{s,MP \text{ for}}}{\partial \dot{x}_{s,0}} & \frac{\partial \dot{x}_{s,MP \text{ for}}}{\partial \dot{y}_{s,0}} & \frac{\partial \dot{x}_{s,MP \text{ for}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial \dot{y}_{s,MP \text{ for}}}{\partial x_{s,0}} & \frac{\partial \dot{y}_{s,MP \text{ for}}}{\partial y_{s,0}} & \frac{\partial \dot{y}_{s,MP \text{ for}}}{\partial z_{s,0}} & \frac{\partial \dot{y}_{s,MP \text{ for}}}{\partial \dot{x}_{s,0}} & \frac{\partial \dot{y}_{s,MP \text{ for}}}{\partial \dot{y}_{s,0}} & \frac{\partial \dot{y}_{s,MP \text{ for}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial \dot{z}_{s,MP \text{ for}}}{\partial x_{s,0}} & \frac{\partial \dot{y}_{s,MP \text{ for}}}{\partial \dot{y}_{s,0}} & \frac{\partial \dot{y}_{s,MP \text{ for}}}{\partial \dot{z}_{s,0}} & \frac{\partial \dot{y}_{s,MP \text{ for}}}{\partial \dot{y}_{s,0}} & \frac{\partial \dot{y}_{s,MP \text{ for}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial \dot{z}_{s,MP \text{ for}}}{\partial \dot{y}_{s,0}} & \frac{\partial \dot{y}_{s,MP \text{ for}}}{\partial \dot{z}_{s,0}} & \frac{\partial \dot{y}_{s,MP \text{ for}}}{\partial \dot{y}_{s,0}} & \frac{\partial \dot{y}_{s,MP \text{ for}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial \dot{z}_{s,MP \text{ for}}}{\partial \dot{y}_{s,0}} & \frac{\partial \dot{z}_{s,MP \text{ for}}}{\partial \dot{z}_{s,0}} & \frac{\partial \dot{z}_{s,MP \text{ for}}}{\partial \dot{y}_{s,0}} & \frac{\partial \dot{z}_{s,MP \text{ for}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial \dot{z}_{s,MP \text{ for}}}{\partial \dot{y}_{s,0}} & \frac{\partial \dot{z}_{s,MP \text{ for}}}{\partial \dot{z}_{s,0}} \\ \end{bmatrix} \end{bmatrix}$$

The derivatives in the $\Psi_{MP,0}$ matrix are not yet the derivatives of the match point constraints with respect to the initial coordinates, velocities and excess velocities. Those can however easily be found from the derivatives in the $\Psi_{MP,0}$ matrix. As an example, the derivatives of the x-coordinate match point constraint will be given with respect to initial coordinate x_0 , initial velocity \dot{x}_0 and initial hyperbolic excess velocity V_{∞,x_0} .

$$\frac{\partial \Delta x_{MP}}{\partial x_0} = \frac{\partial \left(x_{MP \text{ forward}} - x_{MP \text{ backward}}\right)}{\partial x_0} \\
= \frac{\partial x_{MP \text{ forward}}}{\partial x_0} - 0 \quad (C-14) \\
= \frac{\partial x_{MP \text{ forward}}}{\partial x_{s,0}} \cdot \frac{\partial x_{s,0}}{\partial x_0} \\
= \frac{\partial x_{MP \text{ forward}}}{\partial x_{s,0}} \cdot 1 \\
= \Psi_{MP,0}[1, 1]$$

where the definitions of Δx_{MP} and $x_{s,0}$ from respectively Equations 7-1 and C-9 have been used.
Similarly,

$$\frac{\partial \Delta x_{MP}}{\partial \dot{x}_0} = \frac{\partial \left(x_{MP \text{ forward}} - x_{MP \text{ backward}} \right)}{\partial \dot{x}_0} \\
= \frac{\partial x_{MP \text{ forward}}}{\partial \dot{x}_0} - 0 \quad (C-15) \\
= \frac{\partial x_{MP \text{ forward}}}{\partial \dot{x}_{s,0}} \cdot \frac{\partial \dot{x}_{s,0}}{\partial \dot{x}_0} \\
= \frac{\partial x_{MP \text{ forward}}}{\partial \dot{x}_{s,0}} \cdot 1 \\
= \Psi_{MP,0}[1, 4]$$

where the definitions of Δx_{MP} and $\dot{x}_{s,0}$ from respectively Equations 7-1 and C-9 have been used.

Similarly,

$$\frac{\partial \Delta x_{MP}}{\partial V_{\infty,x_0}} = \frac{\partial \left(x_{MP \text{ forward}} - x_{MP \text{ backward}} \right)}{\partial V_{\infty,x_0}} \\
= \frac{\partial x_{MP \text{ forward}}}{\partial V_{\infty,x_0}} - 0 \quad (C-16) \\
= \frac{\partial x_{MP \text{ forward}}}{\partial \dot{x}_{s,0}} \cdot \frac{\partial \dot{x}_{s,0}}{\partial V_{\infty,x_0}} \\
= \frac{\partial x_{MP \text{ forward}}}{\partial \dot{x}_{s,0}} \cdot 1 \\
= \Psi_{MP,0}[1, 4]$$

where the definitions of Δx_{MP} and $\dot{x}_{s,0}$ from respectively Equations 7-1 and C-9 have been used.

The derivatives with respect to y_0 , z_0 , \dot{y}_0, \dot{z}_0 , V_{∞,y_0} and V_{∞,z_0} can be found in a similar way to be the second, third, fifth, sixth, fifth and sixth element of the first row of $\Psi_{MP,0}$. The derivatives of the other match point constraints Δy_{MP} , Δz_{MP} , $\Delta \dot{x}_{MP}$, $\Delta \dot{y}_{MP}$ and $\Delta \dot{z}_{MP}$ can be found in a similar manner to be the second, third, fourth, fifth and sixth row of $\Psi_{MP,0}$.

C-3-2 Derivatives with respect to the final node's coordinates, velocities and hyperbolic excess velocities

First of all, one needs to realize that the final Cartesian state, indicated by the s for "spacecraft state" subscript, is set up by a combination of the final node's coordinates, velocities and excess velocities. These are indicated by the f for "final node" subscript, according to

final node spacecraft state =
$$\begin{bmatrix} x_{s,f} \\ y_{s,f} \\ z_{s,f} \\ \dot{y}_{s,f} \\ \dot{z}_{s,f} \end{bmatrix} = \begin{bmatrix} x_f \\ y_f \\ z_f \\ \dot{x}_f + V_{\infty,x_f} \\ \dot{y}_f + V_{\infty,y_f} \\ \dot{z}_f + V_{\infty,z_f} \end{bmatrix}$$
(C-17)

Based on a similar reasoning as for the derivatives with respect to the initial coordinates, velocities and excess velocities, the $\Psi_{MP,f}$ matrix can be defined as

$$\Psi_{MP,f} = \Phi_{MP,N+1^{-}} \cdot TNC_{N+1^{-},N+1^{+}} \cdot \ldots \cdot TNC_{2N^{-},2N^{+}} \cdot \Phi_{2N^{+},f}$$
(C-18)

Based on a similar reasoning as for the derivatives with respect to the initial coordinates, velocities and excess velocities, it can be shown that

$$\Psi_{MP,f} = \begin{bmatrix} \frac{\partial x_{s,MP \text{ back}}}{\partial x_{s,f}} & \frac{\partial x_{s,MP \text{ back}}}{\partial y_{s,f}} & \frac{\partial x_{s,MP \text{ back}}}{\partial z_{s,f}} & \frac{\partial x_{s,MP \text{ back}}}{\partial \dot{x}_{s,f}} & \frac{\partial x_{s,MP \text{ back}}}{\partial \dot{y}_{s,f}} & \frac{\partial x_{s,MP \text{ back}}}{\partial \dot{y}_{s,f}} & \frac{\partial x_{s,MP \text{ back}}}{\partial \dot{z}_{s,f}} \\ \frac{\partial y_{s,MP \text{ back}}}{\partial x_{s,f}} & \frac{\partial y_{s,MP \text{ back}}}{\partial y_{s,f}} & \frac{\partial y_{s,MP \text{ back}}}{\partial z_{s,f}} & \frac{\partial y_{s,MP \text{ back}}}{\partial \dot{z}_{s,f}} & \frac{\partial y_{s,MP \text{ back}}}{\partial \dot{y}_{s,f}} & \frac{\partial y_{s,MP \text{ back}}}{\partial \dot{z}_{s,f}} \\ \frac{\partial z_{s,MP \text{ back}}}{\partial x_{s,f}} & \frac{\partial z_{s,MP \text{ back}}}{\partial y_{s,f}} & \frac{\partial z_{s,MP \text{ back}}}{\partial y_{s,f}} & \frac{\partial z_{s,MP \text{ back}}}{\partial z_{s,f}} & \frac{\partial z_{s,MP \text{ back}}}{\partial \dot{z}_{s,f}} & \frac{\partial z_{s,MP \text{ back}}}{\partial \dot{x}_{s,f}} & \frac{\partial z_{s,MP \text{ back}}}{\partial \dot{y}_{s,f}} & \frac{\partial z_{s,MP \text{ back}}}{\partial \dot{z}_{s,f}} \\ \frac{\partial \dot{y}_{s,MP \text{ back}}}{\partial x_{s,f}} & \frac{\partial \dot{y}_{s,MP \text{ back}}}{\partial y_{s,f}} & \frac{\partial \dot{y}_{s,MP \text{ back}}}{\partial z_{s,f}} & \frac{\partial \dot{y}_{s,MP \text{ back}}}{\partial \dot{z}_{s,f}} & \frac{\partial \dot{y}_{s,MP \text{ back}}}{\partial \dot{x}_{s,f}} & \frac{\partial \dot{y}_{s,MP \text{ back}}}{\partial \dot{y}_{s,f}} & \frac{\partial \dot{y}_{s,MP \text{ back}}}{\partial \dot{z}_{s,f}} \\ \frac{\partial \dot{y}_{s,MP \text{ back}}}{\partial x_{s,f}} & \frac{\partial \dot{y}_{s,MP \text{ back}}}{\partial y_{s,f}} & \frac{\partial \dot{y}_{s,MP \text{ back}}}{\partial z_{s,f}} & \frac{\partial \dot{y}_{s,MP \text{ back}}}{\partial \dot{x}_{s,f}} & \frac{\partial \dot{y}_{s,MP \text{ back}}}{\partial \dot{y}_{s,f}} & \frac{\partial \dot{y}_{s,MP \text{ back}}}{\partial \dot{z}_{s,f}} \\ \frac{\partial \dot{z}_{s,MP \text{ back}}}{\partial \dot{x}_{s,f}} & \frac{\partial \dot{y}_{s,MP \text{ back}}}{\partial \dot{y}_{s,f}} & \frac{\partial \dot{y}_{s,MP \text{ back}}}{\partial \dot{z}_{s,f}} \\ \frac{\partial \dot{z}_{s,MP \text{ back}}}{\partial \dot{x}_{s,f}} & \frac{\partial \dot{y}_{s,MP \text{ back}}}{\partial \dot{z}_{s,f}} & \frac{\partial \dot{z}_{s,MP \text{ back}}}{\partial \dot{z}_{s,f}} \\ \frac{\partial \dot{z}_{s,MP \text{ back}}}{\partial \dot{z}_{s,f}} & \frac{\partial \dot{z}_{s,MP \text{ back}}}{\partial \dot{z}_{s,f}} & \frac{\partial \dot{z}_{s,MP \text{ back}}}{\partial \dot{z}_{s,f}} \\ \frac{\partial \dot{z}_{s,MP \text{ back}}}{\partial \dot{z}_{s,f}} & \frac{\partial \dot{z}_{s,MP \text{ back}}}{\partial \dot{z}_{s,f}} \\ \frac{\partial \dot{z}_{s,MP \text{ back}}}{\partial \dot{z}_{s,f}} & \frac{\partial \dot{z}_{s,MP \text{ back}}}{\partial \dot{z}_{s,f}} \\ \frac{\partial \dot{z}_{s,MP \text{ back}}}{\partial \dot{z}_{s,f}} & \frac{\partial \dot{z}_{s,MP \text{ back}}}{\partial \dot{z}_{s,f}} \\ \frac{\partial \dot{z}_{s,MP \text{ back}}}{\partial \dot{z}_{s,f}} & \frac{\partial \dot{z}_{s,MP \text{ back}}}{\partial \dot{z}_{s,f}} \\$$

The derivatives in the $\Psi_{MP,f}$ matrix are not yet the derivatives of the match point constraints with respect to the final coordinates, velocities and excess velocities. Those can however easily be found from the derivatives in the $\Psi_{MP,f}$ matrix. As an example, the derivatives of the x-coordinate match point constraint will be given with respect to final coordinate x_f , final velocity \dot{x}_f and final hyperbolic excess velocity V_{∞,x_f} .

$$\frac{\partial \Delta x_{MP}}{\partial x_f} = \frac{\partial \left(x_{MP \text{ forward}} - x_{MP \text{ backward}} \right)}{\partial x_f}$$

$$= 0 - \frac{\partial x_{MP \text{ backward}}}{\partial x_f}$$

$$= -\frac{\partial x_{MP \text{ backward}}}{\partial x_{s,f}} \cdot \frac{\partial x_{s,f}}{\partial x_f}$$

$$= -\frac{\partial x_{MP \text{ backward}}}{\partial x_{s,f}} \cdot 1$$

$$= -\Psi_{MP,f}[1, 1]$$
(C-20)

where the definitions of Δx_{MP} and $x_{s,f}$ from respectively Equations 7-1 and C-17 have been used.

Similarly,

$$\frac{\partial \Delta x_{MP}}{\partial \dot{x}_{f}} = \frac{\partial \left(x_{MP \text{ forward}} - x_{MP \text{ backward}}\right)}{\partial \dot{x}_{f}} \\
= 0 - \frac{\partial x_{MP \text{ backward}}}{\partial \dot{x}_{f}} \qquad (C-21) \\
= -\frac{\partial x_{MP \text{ backward}}}{\partial \dot{x}_{s,f}} \cdot \frac{\partial \dot{x}_{s,f}}{\partial \dot{x}_{f}} \\
= -\frac{\partial x_{MP \text{ backward}}}{\partial \dot{x}_{s,f}} \cdot 1 \\
= -\Psi_{MP,f}[1, 4]$$

where the definitions of Δx_{MP} and $\dot{x}_{s,f}$ from respectively Equations 7-1 and C-17 have been used.

Similarly,

$$\frac{\partial \Delta x_{MP}}{\partial V_{\infty,x_f}} = \frac{\partial \left(x_{MP \text{ forward}} - x_{MP \text{ backward}} \right)}{\partial V_{\infty,x_f}} \\
= 0 - \frac{\partial x_{MP \text{ backward}}}{\partial V_{\infty,x_f}} \qquad (C-22) \\
= -\frac{\partial x_{MP \text{ backward}}}{\partial \dot{x}_{s,f}} \cdot \frac{\partial \dot{x}_{s,f}}{\partial V_{\infty,x_f}} \\
= -\frac{\partial x_{MP \text{ backward}}}{\partial \dot{x}_{s,f}} \cdot 1 \\
= -\Psi_{MP,f}[1, 4]$$

where the definitions of Δx_{MP} and $\dot{x}_{s,f}$ from respectively Equations 7-1 and C-17 have been used.

The derivatives with respect to y_f , z_f , \dot{y}_f , \dot{z}_f , V_{∞,y_f} and V_{∞,z_f} can be found in a similar way to be the negative of the second, third, fifth, sixth, fifth and sixth element of the first row of $\Psi_{MP,f}$. The derivatives of the other match point constraints Δy_{MP} , Δz_{MP} , $\Delta \dot{x}_{MP}$, $\Delta \dot{y}_{MP}$ and $\Delta \dot{z}_{MP}$ can be found in a similar manner to be the negative of the second, third, fourth, fifth and sixth row of $\Psi_{MP,f}$.

C-3-3 Derivatives with respect to the forward velocity components

The derivatives with respect to the ΔV 's can be found using a similar method. An example will be given for the derivatives with respect to the first manoeuvre, which has been visualized in Figure C-2.

Imagine a change in one of the ΔV components of the first manoeuvre. This will result in a change in the state after the first manoeuvre, still indicated by 1⁺.



Figure C-2: Propagation of a change in manoeuvre 1 to the match point.

From Equation A-1, it is known that the change in the ΔV components of the first manoeuvre can be transformed into the change of the state after the first manoeuvre using the $\Delta TNC_{1+,1}$ matrix, which has been derived in Appendix A.

$$\begin{bmatrix} 0\\0\\\Delta\dot{x}_{s}\\\Delta\dot{y}_{s}\\\Delta\dot{z}_{s} \end{bmatrix}_{1+} = \Delta TNC_{1+,1} \cdot \begin{bmatrix} 0\\0\\\\\Delta(\Delta V_{T,1})\\\Delta(\Delta V_{N,1})\\\Delta(\Delta V_{C,1}) \end{bmatrix}_{1}$$
(C-23)

This change of the state after the first manoeuvre can then be propagated to the change of the state at the match point using matrix $\Psi_{MP,1^+}$ defined as

$$\Psi_{MP,1^+} = \Phi_{MP,N^+} \cdot TNC_{N^+,N^-} \cdot \dots TNC_{2^+,2^-} \cdot \Phi_{2^-,1^+} \tag{C-24}$$

Combining $\Delta TNC_{1^+,1}$ and $\Psi_{MP,1^+}$, one can see that

$$\begin{bmatrix} \Delta x_s \\ \Delta y_s \\ \Delta z_s \\ \Delta \dot{x}_s \\ \Delta \dot{y}_s \\ \Delta \dot{z}_s \end{bmatrix}_{\text{MP,forward}} = \Psi_{MP,1^+} \cdot \Delta TNC_{1^+,1} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ \Delta(\Delta V_{T,1}) \\ \Delta(\Delta V_{N,1}) \\ \Delta(\Delta V_{C,1}) \end{bmatrix}_{1}$$
(C-25)

As such, the derivatives can be found from

$$\Psi_{MP,1^{+}} \cdot \Delta TNC_{1^{+},1} = \begin{bmatrix} 0 & 0 & 0 & \frac{\partial x_{s,\text{MP for}}}{\partial \Delta V_{r,1}} & \frac{\partial x_{s,\text{MP for}}}{\partial \Delta V_{r,1}} & \frac{\partial x_{s,\text{MP for}}}{\partial \Delta V_{c,1}} \\ 0 & 0 & 0 & \frac{\partial y_{s,\text{MP for}}}{\partial \Delta V_{r,1}} & \frac{\partial y_{s,\text{MP for}}}{\partial \Delta V_{c,1}} \\ 0 & 0 & 0 & \frac{\partial z_{s,\text{MP for}}}{\partial \Delta V_{r,1}} & \frac{\partial z_{s,\text{MP for}}}{\partial \Delta V_{c,1}} \\ 0 & 0 & 0 & \frac{\partial z_{s,\text{MP for}}}{\partial \Delta V_{r,1}} & \frac{\partial z_{s,\text{MP for}}}{\partial \Delta V_{c,1}} \\ 0 & 0 & 0 & \frac{\partial y_{s,\text{MP for}}}{\partial \Delta V_{r,1}} & \frac{\partial x_{s,\text{MP for}}}{\partial \Delta V_{c,1}} \\ 0 & 0 & 0 & \frac{\partial y_{s,\text{MP for}}}{\partial \Delta V_{r,1}} & \frac{\partial x_{s,\text{MP for}}}{\partial \Delta V_{c,1}} \\ 0 & 0 & 0 & \frac{\partial y_{s,\text{MP for}}}{\partial \Delta V_{r,1}} & \frac{\partial y_{s,\text{MP for}}}{\partial \Delta V_{c,1}} \\ 0 & 0 & 0 & \frac{\partial y_{s,\text{MP for}}}{\partial \Delta V_{r,1}} & \frac{\partial y_{s,\text{MP for}}}{\partial \Delta V_{c,1}} \\ 0 & 0 & 0 & \frac{\partial z_{s,\text{MP for}}}{\partial \Delta V_{r,1}} & \frac{\partial z_{s,\text{MP for}}}{\partial \Delta V_{c,1}} \\ 0 & 0 & 0 & \frac{\partial z_{s,\text{MP for}}}{\partial \Delta V_{r,1}} & \frac{\partial z_{s,\text{MP for}}}{\partial \Delta V_{c,1}} \\ 0 & 0 & 0 & \frac{\partial z_{s,\text{MP for}}}{\partial \Delta V_{r,1}} & \frac{\partial z_{s,\text{MP for}}}{\partial \Delta V_{c,1}} \\ 0 & 0 & 0 & \frac{\partial z_{s,\text{MP for}}}{\partial \Delta V_{r,1}} & \frac{\partial z_{s,\text{MP for}}}{\partial \Delta V_{c,1}} \\ \end{array} \right]$$
(C-26)

From the definitions of Δx_{MP} up to $\Delta \dot{z}_{MP}$ in Equation 7-1, it is known that the derivatives of $x_{s,MP}$ for up to $\dot{z}_{s,MP}$ for are equal to the derivatives of Δx_{MP} up to $\Delta \dot{z}_{MP}$. As such, $\Psi_{MP,1^+} \cdot \Delta TNC_{1^+,1}$ contains those derivatives.

C-3-4 Derivatives with respect to the backward velocity components

The derivatives with respect to the backward ΔV 's can be found similarly. An example will be given for the derivatives with respect to the 2Nth manoeuvre.

Imagine a change in one of the ΔV components of the $2N^{th}$ manoeuvre. This will result in a change in the state before the $2N^{th}$ manoeuvre, still indicated by $2N^{-}$.

From Equation A-1, it is known that the change in the ΔV components of the 2Nth manoeuvre can be transformed into the change of the state before the 2Nth manoeuvre using the ΔTNC matrix, which has been explained in Appendix A. However, because the change in the ΔV components of the 2Nth manoeuvre has to be transformed into the change of the state before the 2Nth manoeuvre and not after, according to Equation 6-17, the transformation needs to be performed using $\Delta TNC_{2N^-,2N} = -\Delta TNC_{2N^+,2N}$.

$$\begin{bmatrix} 0\\0\\ \\\Delta \dot{x}_{s}\\\Delta \dot{y}_{s}\\\Delta \dot{z}_{s} \end{bmatrix}_{2N^{-}} = \Delta TNC_{2N^{-},2N} \cdot \begin{bmatrix} 0\\0\\ \\0\\\Delta (\Delta V_{T,2N})\\\Delta (\Delta V_{N,2N})\\\Delta (\Delta V_{C,2N}) \end{bmatrix}$$
(C-27)

This change of the state before the 2Nth manoeuvre can then be propagated to the change of the state at the match point using matrix $\Psi_{MP,2N^-}$ defined as

$$\Psi_{MP,2N^{-}} = \Phi_{MP,N+1^{-}} \cdot TNC_{N+1^{-},N+1^{+}} \cdot \dots TNC_{2N-1^{-},2N-1^{+}} \cdot \Phi_{2N-1^{+},2N^{-}}$$
(C-28)

Combining $\Delta TNC_{2N^-,2N}$ and $\Psi_{MP,2N^-}$, one can see that

Jacobian

$$\begin{bmatrix} \Delta x_s \\ \Delta y_s \\ \Delta z_s \\ \Delta \dot{x}_s \\ \Delta \dot{y}_s \\ \Delta \dot{z}_s \end{bmatrix}_{\text{MP,backward}} = \Psi_{MP,2N^-} \cdot \Delta TNC_{2N^-,2N} \cdot \begin{bmatrix} 0 \\ 0 \\ 0 \\ \Delta(\Delta V_{T,2N}) \\ \Delta(\Delta V_{N,2N}) \\ \Delta(\Delta V_{C,2N}) \end{bmatrix}$$
(C-29)

As such, the derivatives can be found from

$$\Psi_{MP,2N^{-}} \cdot \Delta TNC_{2N^{-},2N} = \begin{bmatrix} 0 & 0 & 0 & \frac{\partial x_{s,MP \text{ back}}}{\partial \Delta V_{T,2N}} & \frac{\partial x_{s,MP \text{ back}}}{\partial \Delta V_{D,2N}} & \frac{\partial x_{s,MP \text{ back}}}{\partial \Delta V_{C,2N}} \\ 0 & 0 & 0 & \frac{\partial y_{s,MP \text{ back}}}{\partial \Delta V_{T,2N}} & \frac{\partial y_{s,MP \text{ back}}}{\partial \Delta V_{D,2N}} & \frac{\partial y_{s,MP \text{ back}}}{\partial \Delta V_{C,2N}} \\ 0 & 0 & 0 & \frac{\partial z_{s,MP \text{ back}}}{\partial \Delta V_{T,2N}} & \frac{\partial z_{s,MP \text{ back}}}{\partial \Delta V_{D,2N}} & \frac{\partial z_{s,MP \text{ back}}}{\partial \Delta V_{C,2N}} \\ 0 & 0 & 0 & \frac{\partial x_{s,MP \text{ back}}}{\partial \Delta V_{T,2N}} & \frac{\partial z_{s,MP \text{ back}}}{\partial \Delta V_{D,2N}} & \frac{\partial z_{s,MP \text{ back}}}{\partial \Delta V_{C,2N}} \\ 0 & 0 & 0 & \frac{\partial y_{s,MP \text{ back}}}{\partial \Delta V_{T,2N}} & \frac{\partial y_{s,MP \text{ back}}}{\partial \Delta V_{D,2N}} & \frac{\partial y_{s,MP \text{ back}}}{\partial \Delta V_{C,2N}} \\ 0 & 0 & 0 & \frac{\partial y_{s,MP \text{ back}}}{\partial \Delta V_{T,2N}} & \frac{\partial y_{s,MP \text{ back}}}{\partial \Delta V_{D,2N}} & \frac{\partial y_{s,MP \text{ back}}}{\partial \Delta V_{C,2N}} \\ 0 & 0 & 0 & \frac{\partial z_{s,MP \text{ back}}}{\partial \Delta V_{T,2N}} & \frac{\partial z_{s,MP \text{ back}}}{\partial \Delta V_{D,2N}} & \frac{\partial z_{s,MP \text{ back}}}{\partial \Delta V_{C,2N}} \\ 0 & 0 & 0 & \frac{\partial z_{s,MP \text{ back}}}{\partial \Delta V_{T,2N}} & \frac{\partial z_{s,MP \text{ back}}}{\partial \Delta V_{C,2N}} & \frac{\partial z_{s,MP \text{ back}}}{\partial \Delta V_{C,2N}} \\ 0 & 0 & 0 & \frac{\partial z_{s,MP \text{ back}}}{\partial \Delta V_{T,2N}} & \frac{\partial z_{s,MP \text{ back}}}{\partial \Delta V_{C,2N}} & \frac{\partial z_{s,MP \text{ back}}}{\partial \Delta V_{C,2N}} \\ 0 & 0 & 0 & \frac{\partial z_{s,MP \text{ back}}}{\partial \Delta V_{T,2N}} & \frac{\partial z_{s,MP \text{ back}}}{\partial \Delta V_{C,2N}} & \frac{\partial z_{s,MP \text{ back}}}{\partial \Delta V_{C,2N}} \\ \end{array} \right]$$

From the definitions of Δx_{MP} up to $\Delta \dot{z}_{MP}$ in Equation 7-1, it is known that the derivatives of $x_{s,MP}$ for up to $\dot{z}_{s,MP}$ for are equal to the negative of the derivatives of Δx_{MP} up to $\Delta \dot{z}_{MP}$. As such, $-\Psi_{MP,2N} \cdot \Delta TNC_{2N}$, contains those derivatives.

C-4 Thrust constraints

The exact definition of the thrust constraints depends on whether or not the power is constant or dependent on the heliocentric distance. Depending on this definition of the thrust constraints, the derivatives change. Both scenarios will be explained in the next subsections.

C-4-1 Constant power

In the constant power case, the thrust constraint on segment j is defined as

$$F_{T_j} = \frac{\Delta V_j M_j}{P_0} \tag{C-31}$$

where M_j is the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j, which can be found from:

$$M_j = M_0 \cdot exp\left(\frac{-\Delta V_1}{I_{\text{sp},1} \cdot g_0}\right) \cdot \dots \cdot exp\left(\frac{-\Delta V_{j-1}}{I_{\text{sp},j-1} \cdot g_0}\right)$$
(C-32)

for the forward thrust constraints for which j is between 1 and N and

$$M_j = M_f \cdot exp\left(\frac{\Delta V_j}{I_{\rm sp,j} \cdot g_0}\right) \cdot \dots \cdot exp\left(\frac{\Delta V_{2N}}{I_{\rm sp,2N} \cdot g_0}\right) \tag{C-33}$$

for the backward thrust constraints for which j is between N+1 and 2N.

C-4-1-1 Derivatives with respect to M_0 , M_f and M_{P_0}

The derivatives with respect to M_0 and M_f are straightforward. From Equations C-31, C-32 and C-33, one gets

$$\frac{\partial F_{T_j}}{\partial M_0} = \frac{\Delta V_j}{P_0} \frac{M_j}{M_0} \tag{C-34}$$

for the forward thrust constraints and zero for the backward thrust constraints.

$$\frac{\partial F_{T_j}}{\partial M_f} = \frac{\Delta V_j}{P_0} \frac{M_j}{M_f} \tag{C-35}$$

for the backward thrust constraints and zero for the forward thrust constraints.

The influence of M_{P_0} on the thrust constraint is not visible at first. One needs to rewrite Equation C-31 into

$$F_{T_j} = \frac{\Delta V_j M_j k_{P_0}}{M_{P_0}} \tag{C-36}$$

where k_{P_0} is the power to mass ratio of the SEP system. Using this formulation of the thrust constraint, one gets

$$\frac{\partial F_{T_j}}{\partial M_{P_0}} = \frac{-\Delta V_j M_j k_{P_0}}{M_{P_0}^2} \tag{C-37}$$

for both the forward and backward thrust constraints.

C-4-1-2 Derivatives with respect to $\Delta V_{T,i}$, $\Delta V_{N,i}$ and $\Delta V_{C,i}$

Taking the derivative of Equation C-31, one gets:

$$\frac{\partial F_{T_j}}{\partial \Delta V_i} = \frac{M_j}{P_0} \frac{\partial \Delta V_j}{\partial \Delta V_i} + \frac{\Delta V_j}{P_0} \frac{\partial M_j}{\partial \Delta V_i}$$
(C-38)

There are 3 possible combinations of i and j.

i < j The magnitude of manoeuvre j is not affected by the magnitude of manoeuvre i. As such, $\frac{\partial \Delta V_j}{\partial \Delta V_i} = 0$ for both the forward and backward thrust constraints.

Looking at Equation C-32, one can see that the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j is affected by the magnitude of manoeuvre i. Therefore, the derivatives for the forward thrust constraints are

$$\frac{\partial F_{T_j}}{\partial \Delta V_i} = \frac{\Delta V_j}{P_0} \cdot \frac{-M_j}{I_{\text{sp},i} \cdot g_0} \tag{C-39}$$

Looking at Equation C-33, one can see that the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j is not affected by the magnitude of manoeuvre i. Therefore, the derivatives for the backward thrust constraints are

$$\frac{\partial F_{T_j}}{\partial \Delta V_i} = 0 \tag{C-40}$$

i=j Manoeuvres j and i are the same. As such, $\frac{\partial \Delta V_j}{\partial \Delta V_i} = 1$ in this scenario, for both the forward and backward thrust constraints.

Looking at Equation C-32, one can see that the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j is not influenced by the magnitude of that manoeuvre. Therefore, the derivatives for the forward thrust constraints are:

$$\frac{\partial F_{T_j}}{\partial \Delta V_i} = \frac{M_j}{P_0} \tag{C-41}$$

Looking at Equation C-33, one can see that the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j is affected by the magnitude of that manoeuvre. Therefore, the derivatives for the backward thrust constraints are:

$$\frac{\partial F_{T_j}}{\partial \Delta V_i} = \frac{M_j}{P_0} + \frac{\Delta V_j}{P_0} \frac{M_j}{I_{\rm sp,j} \cdot g_0} \tag{C-42}$$

i > j The magnitude of manoeuvre j is not affected by the magnitude of manoeuvre i. As such, $\frac{\partial \Delta V_j}{\partial \Delta V_i} = 0$ for both the forward and backward thrust constraints.

Looking at Equation C-32, one can see that the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j is not affected by the magnitude of manoeuvre i. Therefore, the derivatives for the forward thrust constraints are

$$\frac{\partial F_{T_j}}{\partial \Delta V_i} = 0 \tag{C-43}$$

Looking at Equation C-33, one can see that the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j is affected by the magnitude of manoeuvre i. Therefore, the derivatives for the backward thrust constraints are

$$\frac{\partial F_{T_j}}{\partial \Delta V_i} = \frac{\Delta V_j}{P_0} \cdot \frac{M_j}{I_{\text{sp},i} \cdot g_0} \tag{C-44}$$

Chain rule From these derivatives with respect to ΔV_i , it is straightforward to find the derivatives with respect to $\Delta V_{T,i}$, $\Delta V_{N,i}$ and $\Delta V_{C,i}$ using the chain rule:

$$\frac{\partial F_{T_j}}{\partial \Delta V_{T,i}} = \frac{\partial F_{T_j}}{\partial \Delta V_i} \cdot \frac{\partial \Delta V_i}{\partial \Delta V_{T,i}}$$
(C-45)

$$= \frac{\partial F_{T_j}}{\partial \Delta V_i} \cdot \frac{\Delta V_{T,i}}{\Delta V_i} \tag{C-46}$$

C-4-1-3 Derivatives with respect to I_{sp}

Upon looking at Equation C-31, it seems that the specific impulse does not have an influence on F_{T_j} . However, looking at the entire inequality constraint, the influence of I_{sp} becomes apparent:

$$F_{T_j} = \frac{\Delta V_j m_j}{P_0} \leqslant \frac{2\eta_{\text{jet}} DT}{I_{\text{sp},j} \cdot g_0} \tag{C-47}$$

This can be rewritten into:

$$F_{T_j} = \frac{\Delta V_j m_j I_{\text{sp},j}}{P_0} \leqslant \frac{2\eta_{\text{jet}} DT}{g_0}$$
(C-48)

Taking the derivative of Equation C-48, one gets

$$\frac{\partial F_{T_j}}{\partial I_{\rm sp,i}} = \frac{\Delta V_j m_j}{P_0} \cdot \frac{\partial I_{\rm sp,j}}{\partial I_{\rm sp,i}} + \frac{\Delta V_j I_{\rm sp,j}}{P_0} \cdot \frac{\partial m_j}{\partial I_{\rm sp,i}}$$
(C-49)

Again, there are 3 possible combinations of i and j.

 $\mathbf{i} < \mathbf{j}$ The specific impulse of manoeuvre j is not affected by the specific impulse of manoeuvre i. As such, $\frac{\partial \Delta I_{\text{sp,j}}}{\partial \Delta I_{\text{sp,i}}} = 0$ for both the forward and backward thrust constraints.

Looking at Equation C-32, one can see that the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j is affected by the specific impulse of manoeuvre i. Therefore, the derivatives for the forward thrust constraints are

$$\frac{\partial F_{T_j}}{\partial I_{\rm sp,i}} = \frac{\Delta V_j \Delta V_i I_{\rm sp,j} M_j}{I_{\rm sp,i}^2 \cdot g_0 \cdot P_0} \tag{C-50}$$

Looking at Equation C-33, one can see that the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j is not affected by the specific impulse of manoeuvre i. Therefore, the derivatives for the backward thrust constraints are

$$\frac{\partial F_{T_j}}{\partial \Delta I_{\rm sp,i}} = 0 \tag{C-51}$$

i=j Manoeuvres j and i are the same. As such, $\frac{\partial \Delta I_{\text{sp},j}}{\partial \Delta I_{\text{sp},i}} = 1$ in this scenario, for both the forward and backward thrust constraints.

Looking at Equation C-32, one can see that the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j is not influenced by the specific impulse of that manoeuvre. Therefore, the derivatives for the forward thrust constraints are:

$$\frac{\partial F_{T_j}}{\partial \Delta I_{\rm sp,i}} = \frac{\Delta V_j \cdot M_j}{P_0} \tag{C-52}$$

Looking at Equation C-33, one can see that the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j is affected by the specific impulse of that manoeuvre. Therefore, the derivatives for the backward thrust constraints are:

$$\frac{\partial F_{T_j}}{\partial \Delta I_{\rm sp,i}} = \frac{\Delta V_j \cdot M_j}{P_0} - \frac{M_j \cdot \Delta V_j^2}{P_0 \cdot I_{\rm sp,j} \cdot g_0} \tag{C-53}$$

i>**j** The specific impulse of manoeuvre j is not affected by the specific impulse of manoeuvre i. As such, $\frac{\partial \Delta I_{\text{sp,j}}}{\partial \Delta I_{\text{sp,i}}} = 0$ for both the forward and backward thrust constraints.

Looking at Equation C-32, one can see that the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j is not affected by the specific impulse of manoeuvre i. Therefore, the derivatives for the forward thrust constraints are

$$\frac{\partial F_{T_j}}{\partial \Delta I_{\rm sp,i}} = 0 \tag{C-54}$$

Looking at Equation C-33, one can see that the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j is affected by the specific impulse of manoeuvre i. Therefore, the derivatives for the backward thrust constraints are

$$\frac{\partial F_{T_j}}{\partial \Delta I_{\rm sp,i}} = -\frac{\Delta V_j \Delta V_i I_{\rm sp,j} M_j}{I_{\rm sp,i}^2 \cdot g_0 \cdot P_0} \tag{C-55}$$

C-4-2 Power dependent on heliocentric distance

For this case, the thrust constraint on segment j is defined as

$$F_{T_j} = \frac{\Delta V_j M_j R_j^2}{P_0} \tag{C-56}$$

where M_j is the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j.

C-4-2-1 Derivatives with respect to M_0 , M_f and M_{P_0}

The derivatives with respect to M_0 and M_f are straightforward. From Equations C-56, C-32 and C-33, one gets

$$\frac{\partial F_{T_j}}{\partial M_0} = \frac{\Delta V_j R_j^2}{P_0} \frac{M_j}{M_0} \tag{C-57}$$

for the forward thrust constraints and zero for the backward thrust constraints.

$$\frac{\partial F_{T_j}}{\partial M_f} = \frac{\Delta V_j R_j^2}{P_0} \frac{M_j}{M_f} \tag{C-58}$$

for the backward thrust constraints and zero for the forward thrust constraints.

The influence of M_{P_0} on the thrust constraint is not visible at first. One needs to rewrite Equation C-56 into

$$F_{T_j} = \frac{\Delta V_j M_j R_j^2 k_{P_0}}{M_{P_0}}$$
(C-59)

where k_{P_0} is the power to mass ratio of the SEP system. Using this formulation of the thrust constraint, one gets

$$\frac{\partial F_{T_j}}{\partial M_{P_0}} = \frac{-\Delta V_j M_j R_j^2 k_{P_0}}{M_{P_0}^2}$$
(C-60)

for both the forward and backward thrust constraints.

C-4-2-2 Derivatives with respect to $\Delta V_{T,i}$, $\Delta V_{N,i}$ and $\Delta V_{C,i}$

Taking the derivative of Equation C-56, one gets:

$$\frac{\partial F_{T_j}}{\partial \Delta V_i} = \frac{M_j R_j^2}{P_0} \frac{\partial \Delta V_j}{\partial \Delta V_i} + \frac{\Delta V_j R_j^2}{P_0} \frac{\partial M_j}{\partial \Delta V_i} + \frac{\Delta V_j M_j 2R_j}{P_0} \frac{\partial R_j}{\partial \Delta V_i}$$
(C-61)

where

$$\frac{\partial R_j}{\partial \Delta V_i} = \frac{\partial \sqrt{x_j^2 + y_j^2 + z_j^2}}{\partial \Delta V_i}$$
(C-62)
$$= \frac{1}{2\sqrt{x_j^2 + y_j^2 + z_j^2}} \cdot \left(2x_j \frac{\partial x_j}{\partial \Delta V_i} + 2y_j \frac{\partial y_j}{\partial \Delta V_i} + 2z_j \frac{\partial z_j}{\partial \Delta V_i} \right)$$

$$= \frac{x_j}{R_j} \frac{\partial x_j}{\partial \Delta V_i} + \frac{y_j}{R_j} \frac{\partial y_j}{\partial \Delta V_i} + \frac{z_j}{R_j} \frac{\partial z_j}{\partial \Delta V_i}$$

Again, there are 3 possible combinations for i and j.

i < j The magnitude of manoeuvre j is not affected by the magnitude of manoeuvre i. As such, $\frac{\partial \Delta V_j}{\partial \Delta V_i} = 0$ for both the forward and backward thrust constraints.

Looking at Equation C-32, one can see that the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j is affected by the magnitude of manoeuvre i. The location of the midpoint of segment j is affected by the magnitude of manoeuvre i. As such, the partial derivatives of x_j , y_j and z_j with respect to ΔV_i have to be found, which can be done by looking at Figure C-2. For instance for the partial derivatives of x_2 , y_2 and z_2 with respect to ΔV_1 , one can see that a change in ΔV_1 results in a change in the state of 1⁺ through transformation matrix ΔTNC_1 , which is being propagated up to 2⁻ through $\Phi_{2,1}$. So in general, a change in ΔV_i is propagated up to a change in state at the location of manoeuvre j through $\Psi_{j,i^+} \cdot \Delta TNC_i$.

Therefore, the derivatives for the forward thrust constraints are

$$\begin{aligned} \frac{\partial F_{T_j}}{\partial \Delta V_{T,i}} &= \frac{m_j R_j \Delta V_j}{P_0} \left[\frac{-R_j}{I_{\text{sp},i} \cdot g_0} \cdot \frac{\Delta V_{T,i}}{\Delta V_i} + 2 \left(\frac{x_j}{R_j} \frac{\partial x_j}{\partial \Delta V_{T,i}} + \frac{y_j}{R_j} \frac{\partial y_j}{\partial \Delta V_{T,i}} + \frac{z_j}{R_j} \frac{\partial z_j}{\partial \Delta V_{T,i}} \right) \right] \\ \frac{\partial F_{T_j}}{\partial \Delta V_{N,i}} &= \frac{m_j R_j \Delta V_j}{P_0} \left[\frac{-R_j}{I_{\text{sp},i} \cdot g_0} \cdot \frac{\Delta V_{N,i}}{\Delta V_i} + 2 \left(\frac{x_j}{R_j} \frac{\partial x_j}{\partial \Delta V_{N,i}} + \frac{y_j}{R_j} \frac{\partial y_j}{\partial \Delta V_{N,i}} + \frac{z_j}{R_j} \frac{\partial z_j}{\partial \Delta V_{N,i}} \right) \right] \\ \frac{\partial F_{T_j}}{\partial \Delta V_{C,i}} &= \frac{m_j R_j \Delta V_j}{P_0} \left[\frac{-R_j}{I_{\text{sp},i} \cdot g_0} \cdot \frac{\Delta V_{C,i}}{\Delta V_i} + 2 \left(\frac{x_j}{R_j} \frac{\partial x_j}{\partial \Delta V_{C,i}} + \frac{y_j}{R_j} \frac{\partial y_j}{\partial \Delta V_{C,i}} + \frac{z_j}{R_j} \frac{\partial z_j}{\partial \Delta V_{C,i}} \right) \right] \end{aligned}$$

where the partial derivatives of x_j , y_j and z_j with respect to ΔV_T , ΔV_N and ΔV_C can be found in the $\Psi_{j,i^+} \cdot \Delta TNC_i$ matrix.

Looking at Equation C-33, one can see that the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j is not affected by the magnitude of manoeuvre i. Furthermore, the location of the midpoint of segment j is not affected by the magnitude of manoeuvre i. Therefore, the derivatives for the backward thrust constraints are

$$\frac{\partial F_{T_j}}{\partial \Delta V_{T,i}} = \frac{\partial F_{T_j}}{\partial \Delta V_{N,i}} = \frac{\partial F_{T_j}}{\partial \Delta V_{C,i}} = 0$$
(C-64)

i=j Manoeuvres j and i are the same. As such, $\frac{\partial \Delta V_j}{\partial \Delta V_i} = 1$ in this scenario, for both the forward as backward thrust constraints.

Looking at Equation C-32, one can see that the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j is not influenced by the magnitude of that manoeuvre. Furthermore, the location of the midpoint of segment j is not affected by the magnitude of the manoeuvre at that midpoint. Therefore, the derivatives for the forward thrust constraints are:

$$\frac{\partial F_{T_j}}{\partial \Delta V_{T,i}} = \frac{M_j R_j^2}{P_0} \cdot \frac{\Delta V_{T,i}}{\Delta V_i}$$

$$\frac{\partial F_{T_j}}{\partial \Delta V_{N,i}} = \frac{M_j R_j^2}{P_0} \cdot \frac{\Delta V_{N,i}}{\Delta V_i}$$

$$\frac{\partial F_{T_j}}{\partial \Delta V_{C,i}} = \frac{M_j R_j^2}{P_0} \cdot \frac{\Delta V_{C,i}}{\Delta V_i}$$
(C-65)

Looking at Equation C-33, one can see that the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j is affected by the magnitude of that manoeuvre. Therefore, the derivatives for the backward thrust constraints are

$$\frac{\partial F_{T_j}}{\partial \Delta V_{T,i}} = \left(\frac{M_j R_j^2}{P_0} + \frac{\Delta V_j R_j^2 M_j}{P_0 I_{\rm sp,j} g_0}\right) \cdot \frac{\Delta V_{T,i}}{\Delta V_i}$$
$$\frac{\partial F_{T_j}}{\partial \Delta V_{N,i}} = \left(\frac{M_j R_j^2}{P_0} + \frac{\Delta V_j R_j^2 M_j}{P_0 I_{\rm sp,j} g_0}\right) \cdot \frac{\Delta V_{N,i}}{\Delta V_i}$$
$$\frac{\partial F_{T_j}}{\partial \Delta V_{C,i}} = \left(\frac{M_j R_j^2}{P_0} + \frac{\Delta V_j R_j^2 M_j}{P_0 I_{\rm sp,j} g_0}\right) \cdot \frac{\Delta V_{C,i}}{\Delta V_i}$$

i > j The magnitude of manoeuvre j is not affected by the magnitude of manoeuvre i. As such, $\frac{\partial \Delta V_j}{\partial \Delta V_i} = 0$ for both the forward and backward thrust constraints.

Looking at Equation C-32, one can see that the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j is not affected by the magnitude of manoeuvre i. Furthermore, the location of the midpoint of segment j is not affected by the magnitude of manoeuvre i. Therefore, the derivatives for the forward thrust constraints are

$$\frac{\partial F_{T_j}}{\partial \Delta V_{T,i}} = \frac{\partial F_{T_j}}{\partial \Delta V_{N,i}} = \frac{\partial F_{T_j}}{\partial \Delta V_{C,i}} = 0$$
(C-66)

Looking at Equation C-33, one can see that the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j is affected by the magnitude of manoeuvre i. The location of the midpoint of segment j is affected by the magnitude of manoeuvre i. As such, the partial derivatives of x_j , y_j and z_j with respect to ΔV_i have to be found. For instance for the partial derivatives of x_{2N-1} , y_{2N-1} and z_{2N-1} with respect to ΔV_{2N} , one can see that a change in ΔV_{2N} results in a change in the state of 1⁻ through transformation matrix $-\Delta TNC_{2N}$, which is being propagated up to 2⁺ through $\Phi_{2,1}$. So in general, a change in ΔV_i is propagated up to a change in state at the location of manoeuvre j through $\Psi_{j,i} - \Delta TNC_i$.

Therefore, the derivatives for the backward thrust constraints are

$$\frac{\partial F_{T_j}}{\partial \Delta V_{T,i}} = \frac{m_j R_j \Delta V_j}{P_0} \left[\frac{R_j}{I_{\text{sp},i} \cdot g_0} \cdot \frac{\Delta V_{T,i}}{\Delta V_i} + 2 \left(\frac{x_j}{R_j} \frac{\partial x_j}{\partial \Delta V_{T,i}} + \frac{y_j}{R_j} \frac{\partial y_j}{\partial \Delta V_{T,i}} + \frac{z_j}{R_j} \frac{\partial z_j}{\partial \Delta V_{T,i}} \right) \right]$$

$$\frac{\partial F_{T_j}}{\partial \Delta V_{N,i}} = \frac{m_j R_j \Delta V_j}{P_0} \left[\frac{R_j}{I_{\text{sp},i} \cdot g_0} \cdot \frac{\Delta V_{N,i}}{\Delta V_i} + 2 \left(\frac{x_j}{R_j} \frac{\partial x_j}{\partial \Delta V_{N,i}} + \frac{y_j}{R_j} \frac{\partial y_j}{\partial \Delta V_{N,i}} + \frac{z_j}{R_j} \frac{\partial z_j}{\partial \Delta V_{N,i}} \right) \right]$$

$$\frac{\partial F_{T_j}}{\partial \Delta V_{C,i}} = \frac{m_j R_j \Delta V_j}{P_0} \left[\frac{R_j}{I_{\text{sp},i} \cdot g_0} \cdot \frac{\Delta V_{C,i}}{\Delta V_i} + 2 \left(\frac{x_j}{R_j} \frac{\partial x_j}{\partial \Delta V_{C,i}} + \frac{y_j}{R_j} \frac{\partial y_j}{\partial \Delta V_{C,i}} + \frac{z_j}{R_j} \frac{\partial z_j}{\partial \Delta V_{C,i}} \right) \right]$$

where the partial derivatives of x_j , y_j and z_j with respect to ΔV_T , ΔV_N and ΔV_C are the negatives of the elements within the $\Psi_{j,i} - \Delta TNC_i$ matrix.

C-4-2-3 Derivatives with respect to I_{sp}

Upon looking at Equation C-56, it seems that the specific impulse does not have an influence on F_{T_j} . However, looking at the entire inequality constraint, the influence of I_{sp} becomes apparent:

$$F_{T_j} = \frac{\Delta V_j m_j R_j^2}{P_0} \leqslant \frac{2\eta_{jet} DT A U^2}{I_{\rm sp,j} \cdot g_0}$$
(C-68)

This can be rewritten into:

$$F_{T_j} = \frac{\Delta V_j m_j I_{\text{sp},j}}{P_0} \leqslant \frac{2\eta_{jet} DT A U^2}{g_0} \tag{C-69}$$

Taking the derivative of Equation C-69, one gets

$$\frac{\partial F_{T_j}}{\partial I_{\rm sp,i}} = \frac{\Delta V_j m_j R_j^2}{P_0} \cdot \frac{\partial I_{\rm sp,j}}{\partial I_{\rm sp,i}} + \frac{\Delta V_j R_j^2 I_{\rm sp,j}}{P_0} \cdot \frac{\partial m_j}{\partial I_{\rm sp,i}}$$
(C-70)

Again, there are 3 possible combinations of i and j.

 $\mathbf{i} < \mathbf{j}$ The specific impulse of manoeuvre j is not affected by the specific impulse of manoeuvre i. As such, $\frac{\partial \Delta I_{\text{sp,j}}}{\partial \Delta I_{\text{sp,i}}} = 0$ for both the forward and backward thrust constraints.

Looking at Equation C-32, one can see that the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j is affected by the specific impulse of manoeuvre i. Therefore, the derivatives for the forward thrust constraints are

$$\frac{\partial F_{T_j}}{\partial I_{\rm sp,i}} = \frac{\Delta V_j \Delta V_i I_{\rm sp,j} m_j R_j^2}{I_{\rm sp,i}^2 \cdot g_0 \cdot P_0} \tag{C-71}$$

Looking at Equation C-33, one can see that the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j is not affected by the specific impulse of manoeuvre i. Therefore, the derivatives for the backward thrust constraints are

$$\frac{\partial F_{T_j}}{\partial \Delta I_{\rm sp,i}} = 0 \tag{C-72}$$

i=j Manoeuvres j and i are the same. As such, $\frac{\partial \Delta I_{\text{sp},j}}{\partial \Delta I_{\text{sp},i}} = 1$ in this scenario, for both the forward and backward thrust constraints.

Looking at Equation C-32, one can see that the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j is not influenced by the specific impulse of that manoeuvre. Therefore, the derivatives for the forward thrust constraints are:

$$\frac{\partial F_{T_j}}{\partial I_{\rm sp,i}} = \frac{\Delta V_j M_j R_j^2}{P_0} \tag{C-73}$$

Looking at Equation C-33, one can see that the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j is affected by the specific impulse of that manoeuvre. Therefore, the derivatives for the backward thrust constraints are:

$$\frac{\partial F_{T_j}}{\partial I_{\rm sp,i}} = \frac{\Delta V_j M_j R_j^2}{P_0} - \frac{\Delta V_j^2 R_j^2 M_j}{P_0 I_{\rm sp,j} g_0} \tag{C-74}$$

i > j The specific impulse of manoeuvre j is not affected by the specific impulse of manoeuvre i. As such, $\frac{\partial \Delta I_{sp,j}}{\partial \Delta I_{sp,i}} = 0$ for both the forward and backward thrust constraints.

Looking at Equation C-32, one can see that the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j is not affected by the specific impulse of manoeuvre i. Therefore, the derivatives for the forward thrust constraints are:

$$\frac{\partial F_{T_j}}{\partial \Delta I_{\rm sp,i}} = 0 \tag{C-75}$$

Looking at Equation C-33, one can see that the mass of the spacecraft before the manoeuvre applied at the midpoint of segment j is affected by the specific impulse of manoeuvre i. Therefore, the derivatives for the backward thrust constraints are:

$$\frac{\partial F_{T_j}}{\partial \Delta I_{\rm sp,i}} = -\frac{\Delta V_j \Delta V_i I_{\rm sp,j} M_j R_j^2}{I_{\rm sp,i}^2 \cdot g_0 \cdot P_0} \tag{C-76}$$

C-4-2-4 Derivatives with respect to the initial coordinates x_0, y_0 and z_0 , the initial velocities \dot{x}_0, \dot{y}_0 and \dot{z}_0 and the initial excess velocities $V_{\infty,x_0}, V_{\infty,y_0}$ and V_{∞,z_0}

In this discussion, the derivatives with respect to x_0 , \dot{x}_0 and V_{∞,x_0} will be shown. Using a similar method, the other derivatives can be found as well.

Using Equation C-56:

$$\frac{\partial F_{T_j}}{\partial x_0} = \frac{2M_j \Delta V_j R_j}{P_0} \frac{\partial R_j}{\partial x_0}
\frac{\partial F_{T_j}}{\partial \dot{x}_0} = \frac{2M_j \Delta V_j R_j}{P_0} \frac{\partial R_j}{\partial \dot{x}_0}
\frac{\partial F_{T_j}}{\partial V_{\infty,x_0}} = \frac{2M_j \Delta V_j R_j}{P_0} \frac{\partial R_j}{\partial V_{\infty,x_0}}$$
(C-77)

$$\frac{\partial R_{j}}{\partial x_{0}} = \frac{x_{j}}{R_{j}} \frac{\partial x_{j}}{\partial x_{0}} + \frac{y_{j}}{R_{j}} \frac{\partial y_{j}}{\partial x_{0}} + \frac{z_{j}}{R_{j}} \frac{\partial z_{j}}{\partial x_{0}}$$

$$\frac{\partial R_{j}}{\partial \dot{x}_{0}} = \frac{x_{j}}{R_{j}} \frac{\partial x_{j}}{\partial \dot{x}_{0}} + \frac{y_{j}}{R_{j}} \frac{\partial y_{j}}{\partial \dot{x}_{0}} + \frac{z_{j}}{R_{j}} \frac{\partial z_{j}}{\partial \dot{x}_{0}}$$

$$\frac{\partial R_{j}}{\partial V_{\infty,x_{0}}} = \frac{x_{j}}{R_{j}} \frac{\partial x_{j}}{\partial V_{\infty,x_{0}}} + \frac{y_{j}}{R_{j}} \frac{\partial y_{j}}{\partial V_{\infty,x_{0}}} + \frac{z_{j}}{R_{j}} \frac{\partial z_{j}}{\partial V_{\infty,x_{0}}}$$
(C-78)

Imagine a change in one of the initial coordinates, velocities or excess velocities. As such, the initial state defined by Equation C-9 changes. Looking at Figure C-1, this change can be propagated towards the midpoint of segment j using $\Psi_{j^-,0} = \Phi_{j,j-1} \cdot TNC_{j-1} \cdot \ldots \cdot TNC_1 \cdot \Phi_{1,0}$. So,

$$\Psi_{j^-,0} = \begin{bmatrix} \frac{\partial x_j}{\partial x_{s,0}} & \frac{\partial x_j}{\partial y_{s,0}} & \frac{\partial x_j}{\partial z_{s,0}} & \frac{\partial x_j}{\partial x_{s,0}} & \frac{\partial x_j}{\partial y_{s,0}} & \frac{\partial x_j}{\partial z_{s,0}} & \frac{\partial x_j}{\partial z_{s,0}} \\ \frac{\partial y_j}{\partial x_{s,0}} & \frac{\partial y_j}{\partial y_{s,0}} & \frac{\partial z_j}{\partial z_{s,0}} & \frac{\partial z_j}{\partial x_{s,0}} & \frac{\partial y_j}{\partial y_{s,0}} & \frac{\partial y_j}{\partial z_{s,0}} \\ \frac{\partial z_j}{\partial x_{s,0}} & \frac{\partial z_j}{\partial y_{s,0}} & \frac{\partial z_j}{\partial z_{s,0}} & \frac{\partial x_j}{\partial x_{s,0}} & \frac{\partial x_j}{\partial y_{s,0}} & \frac{\partial z_j}{\partial z_{s,0}} \\ \frac{\partial z_j}{\partial x_{s,0}} & \frac{\partial z_j}{\partial y_{s,0}} & \frac{\partial z_j}{\partial z_{s,0}} & \frac{\partial x_j}{\partial x_{s,0}} & \frac{\partial x_j}{\partial x_{s,0}} & \frac{\partial x_j}{\partial x_{s,0}} \\ \frac{\partial y_j}{\partial x_{s,0}} & \frac{\partial y_j}{\partial y_{s,0}} & \frac{\partial z_j}{\partial z_{s,0}} & \frac{\partial z_j}{\partial x_{s,0}} & \frac{\partial y_j}{\partial y_{s,0}} & \frac{\partial z_j}{\partial z_{s,0}} \\ \frac{\partial z_j}{\partial x_{s,0}} & \frac{\partial y_j}{\partial y_{s,0}} & \frac{\partial z_j}{\partial z_{s,0}} & \frac{\partial z_j}{\partial x_{s,0}} & \frac{\partial z_j}{\partial y_{s,0}} & \frac{\partial z_j}{\partial z_{s,0}} \\ \frac{\partial z_j}{\partial x_{s,0}} & \frac{\partial y_{s,0}}{\partial y_{s,0}} & \frac{\partial z_j}{\partial z_{s,0}} & \frac{\partial z_j}{\partial x_{s,0}} & \frac{\partial z_j}{\partial y_{s,0}} & \frac{\partial z_j}{\partial z_{s,0}} \\ \end{array} \right]$$
(C-79)

From this matrix, the required derivatives can be found by remembering the definitions of $x_{s,0}$ up to $\dot{z}_{s,0}$ in Equation C-9.

$$\begin{array}{rcl} \frac{\partial x_j}{\partial x_0} &=& \frac{\partial x_j}{\partial x_{s,0}} \cdot \frac{\partial x_{s,0}}{\partial x_0} \\ &=& \frac{\partial x_j}{\partial x_{s,0}} \cdot 1 \\ \\ \frac{\partial x_j}{\partial \dot{x}_0} &=& \frac{\partial x_j}{\partial \dot{x}_{s,0}} \cdot \frac{\partial \dot{x}_{s,0}}{\partial \dot{x}_0} \\ &=& \frac{\partial x_j}{\partial \dot{x}_{s,0}} \cdot 1 \\ \\ \frac{\partial x_j}{\partial V_{\infty,x_0}} &=& \frac{\partial x_j}{\partial \dot{x}_{s,0}} \cdot \frac{\partial \dot{x}_{s,0}}{\partial V_{\infty,x_0}} \\ &=& \frac{\partial x_j}{\partial \dot{x}_{s,0}} \cdot 1 \end{array}$$

C-4-2-5 Derivatives with respect to the final coordinates x_f, y_f and z_f , the final velocities \dot{x}_f, \dot{y}_f and \dot{z}_f and the final excess velocities $V_{\infty,x_f}, V_{\infty,y_f}$ and V_{∞,z_f}

In this discussion, the derivatives with respect to x_f , \dot{x}_f and V_{∞,x_f} will be shown. Using a similar method, the other derivatives can be found as well.

Using Equation C-56:

$$\frac{\partial F_{T_j}}{\partial x_f} = \frac{2M_j \Delta V_j R_j}{P_0} \frac{\partial R_j}{\partial x_f}
\frac{\partial F_{T_j}}{\partial \dot{x}_f} = \frac{2M_j \Delta V_j R_j}{P_0} \frac{\partial R_j}{\partial \dot{x}_f}
\frac{\partial F_{T_j}}{\partial V_{\infty,x_f}} = \frac{2M_j \Delta V_j R_j}{P_0} \frac{\partial R_j}{\partial V_{\infty,x_f}}$$
(C-80)

$$\frac{\partial R_{j}}{\partial x_{f}} = \frac{x_{j}}{R_{j}} \frac{\partial x_{j}}{\partial x_{f}} + \frac{y_{j}}{R_{j}} \frac{\partial y_{j}}{\partial x_{f}} + \frac{z_{j}}{R_{j}} \frac{\partial z_{j}}{\partial x_{f}}$$

$$\frac{\partial R_{j}}{\partial \dot{x}_{f}} = \frac{x_{j}}{R_{j}} \frac{\partial x_{j}}{\partial \dot{x}_{f}} + \frac{y_{j}}{R_{j}} \frac{\partial y_{j}}{\partial \dot{x}_{f}} + \frac{z_{j}}{R_{j}} \frac{\partial z_{j}}{\partial \dot{x}_{f}}$$

$$\frac{\partial R_{j}}{\partial V_{\infty,x_{f}}} = \frac{x_{j}}{R_{j}} \frac{\partial x_{j}}{\partial V_{\infty,x_{f}}} + \frac{y_{j}}{R_{j}} \frac{\partial y_{j}}{\partial V_{\infty,x_{f}}} + \frac{z_{j}}{R_{j}} \frac{\partial z_{j}}{\partial V_{\infty,x_{f}}}$$
(C-81)

Imagine a change in one of the final coordinates, velocities or excess velocities. As such, the final state defined by Equation C-17 changes. This change can be propagated towards the midpoint of segment j using $\Psi_{j+,f} = \Phi_{j,j+1} \cdot TNC_{j+1} \cdot \ldots \cdot TNC_{2N} \cdot \Phi_{2N,f}$. So,

$$\Psi_{j+,f} = \begin{bmatrix} \frac{\partial x_j}{\partial x_{s,f}} & \frac{\partial x_j}{\partial y_{s,f}} & \frac{\partial x_j}{\partial z_{s,f}} & \frac{\partial x_j}{\partial \dot{x}_{s,f}} & \frac{\partial x_j}{\partial \dot{y}_{s,f}} & \frac{\partial x_j}{\partial \dot{z}_{s,f}} \\ \frac{\partial y_j}{\partial x_{s,f}} & \frac{\partial y_j}{\partial y_{s,f}} & \frac{\partial y_j}{\partial z_{s,f}} & \frac{\partial y_j}{\partial \dot{x}_{s,f}} & \frac{\partial y_j}{\partial \dot{y}_{s,f}} & \frac{\partial y_j}{\partial \dot{z}_{s,f}} \\ \frac{\partial z_j}{\partial x_{s,f}} & \frac{\partial z_j}{\partial y_{s,f}} & \frac{\partial z_j}{\partial z_{s,f}} & \frac{\partial z_j}{\partial \dot{x}_{s,f}} & \frac{\partial z_j}{\partial \dot{x}_{s,f}} & \frac{\partial z_j}{\partial \dot{x}_{s,f}} \\ \frac{\partial \dot{y}_j}{\partial x_{s,f}} & \frac{\partial \dot{y}_j}{\partial y_{s,f}} & \frac{\partial \dot{x}_j}{\partial z_{s,f}} & \frac{\partial \dot{x}_j}{\partial \dot{x}_{s,f}} & \frac{\partial \dot{x}_j}{\partial \dot{y}_{s,f}} & \frac{\partial \dot{x}_j}{\partial \dot{z}_{s,f}} \\ \frac{\partial \dot{y}_j}{\partial x_{s,f}} & \frac{\partial \dot{y}_j}{\partial y_{s,f}} & \frac{\partial \dot{y}_j}{\partial z_{s,f}} & \frac{\partial \dot{y}_j}{\partial \dot{x}_{s,f}} & \frac{\partial \dot{y}_j}{\partial \dot{y}_{s,f}} & \frac{\partial \dot{y}_j}{\partial \dot{z}_{s,f}} \\ \frac{\partial \dot{z}_j}{\partial x_{s,f}} & \frac{\partial \dot{z}_j}{\partial y_{s,f}} & \frac{\partial \dot{z}_j}{\partial z_{s,f}} & \frac{\partial \dot{z}_j}{\partial \dot{x}_{s,f}} & \frac{\partial \dot{z}_j}{\partial \dot{y}_{s,f}} & \frac{\partial \dot{z}_j}{\partial \dot{z}_{s,f}} \\ \frac{\partial \dot{z}_j}{\partial x_{s,f}} & \frac{\partial \dot{z}_j}{\partial y_{s,f}} & \frac{\partial \dot{z}_j}{\partial z_{s,f}} & \frac{\partial \dot{z}_j}{\partial \dot{x}_{s,f}} & \frac{\partial \dot{z}_j}{\partial \dot{y}_{s,f}} & \frac{\partial \dot{z}_j}{\partial \dot{z}_{s,f}} \\ \end{array} \right]$$
(C-82)

From this matrix, the required derivatives can be found by remembering the definitions of $x_{s,f}$ up to $\dot{z}_{s,f}$ in Equation C-17.

$$\begin{array}{rcl} \displaystyle \frac{\partial x_j}{\partial x_f} &=& \displaystyle \frac{\partial x_j}{\partial x_{s,f}} \cdot \frac{\partial x_{s,f}}{\partial x_f} \\ &=& \displaystyle \frac{\partial x_j}{\partial x_{s,f}} \cdot 1 \\ \\ \displaystyle \frac{\partial x_j}{\partial \dot{x}_f} &=& \displaystyle \frac{\partial x_j}{\partial \dot{x}_{s,f}} \cdot \frac{\partial \dot{x}_{s,f}}{\partial \dot{x}_f} \\ &=& \displaystyle \frac{\partial x_j}{\partial \dot{x}_{s,f}} \cdot 1 \\ \\ \displaystyle \frac{\partial x_j}{\partial V_{\infty,x_f}} &=& \displaystyle \frac{\partial x_j}{\partial \dot{x}_{s,f}} \cdot \frac{\partial \dot{x}_{s,f}}{\partial V_{\infty,x_f}} \\ &=& \displaystyle \frac{\partial x_j}{\partial \dot{x}_{s,f}} \cdot 1 \end{array}$$

C-5 Leg-specific constraints

C-5-1 Departure-node constraints

Launch mass constraints The derivative with respect to the initial mass is trivial:

$$\frac{\partial F_{LM}}{\partial M_0} = -1 \tag{C-83}$$

The maximum launch mass for a certain excessive velocity can be written as a polynomial:

$$M_{L,max} = a_0 + a_1 C_3 + a_2 C_3^2 + a_3 C_3^3 + a_4 C_4^4$$
(C-84)

with

$$C_3 = V_{\infty,x_0}^2 + V_{\infty,y_0}^2 + V_{\infty,z_0}^2$$
(C-85)

Combining these equations, the following derivatives are obtained:

$$\frac{\partial F_{LM}}{\partial V_{\infty,x_0}} = \left(a_1 + 2a_2C_3 + 3a_3C_3^2 + 4a_4C_3^3\right)\frac{\partial C_3}{\partial V_{\infty,x_0}}$$

$$= \left(a_1 + 2a_2C_3 + 3a_3C_3^2 + 4a_4C_3^3\right)2V_{\infty,x_0}$$
(C-86)

$$\frac{\partial F_{LM}}{\partial V_{\infty,y_0}} = \left(a_1 + 2a_2C_3 + 3a_3C_3^2 + 4a_4C_3^3\right)\frac{\partial C_3}{\partial V_{\infty,y_0}}$$
(C-87)

$$= \left(a_1 + 2a_2C_3 + 3a_3C_3^2 + 4a_4C_3^3\right)2V_{\infty,y_0}$$

$$\frac{\partial C_3}{\partial V_{\infty,z_0}} = \left(a_1 + 2a_2C_3 + 3a_3C_3^2 + 4a_4C_3^3\right)\frac{\partial C_3}{\partial V_{\infty,z_0}}$$
(C-88)

$$V_{\infty,z_0} = \left(a_1 + 2a_2C_3 + 3a_3C_3^2 + 4a_4C_3^3\right)\partial V_{\infty,z_0}$$

= $\left(a_1 + 2a_2C_3 + 3a_3C_3^2 + 4a_4C_3^3\right)2V_{\infty,z_0}$

 C_3 constraint The derivatives for this constraint are trivial:

$$\frac{\partial F_{C_3}}{\partial V_{\infty,x_0}} = 2V_{\infty,x_0} \tag{C-89}$$

$$\frac{\partial F_{C_3}}{\partial V_{\infty,y_0}} = 2V_{\infty,y_0} \tag{C-90}$$

$$\frac{\partial F_{C_3}}{\partial V_{\infty,z_0}} = 2V_{\infty,z_0} \tag{C-91}$$

C-5-2 Flyby-node constraints

Mass equality constraint The mass before and after the flyby must be equal in magnitude. The derivatives of this constraint are trivial:

$$\frac{\partial F_{\text{mass flyby}}}{M_{f \log 1}} = -1 \tag{C-92}$$
$$\frac{\partial F_{\text{mass flyby}}}{M_{0 \log 2}} = 1$$

Relative velocity equality constraint The incoming and outgoing relative velocities must be equal in magnitude. In this discussion, the hyperbolic excess velocity of the inbound and outbound leg of the flyby will be indicated by the I and II subscripts respectively. The derivatives of this constraint are trivial:

$$\frac{\partial F_{\text{relative velocities flyby}}}{V_{\infty, x_{\epsilon}^{I}}} = -V_{\infty, x_{f}^{I}} \tag{C-93}$$

$$\frac{\partial F_{\text{relative velocities flyby}}}{V_{\infty,y_f^I}} = -V_{\infty,y_f^I} \tag{C-94}$$

$$\frac{\partial F_{\text{relative velocities flyby}}}{V_{\infty,z_f^I}} = -V_{\infty,z_f^I} \tag{C-95}$$

$$\frac{\partial F_{\text{relative velocities flyby}}}{V_{\infty,x_0^{II}}} = V_{\infty,x_0^{II}} \tag{C-96}$$

$$\frac{\partial F_{\text{relative velocities flyby}}}{V_{\infty,y_0^{II}}} = V_{\infty,y_0^{II}} \tag{C-97}$$

$$\frac{\partial F_{\text{relative velocities flyby}}}{V_{\infty, z_0^{II}}} = V_{\infty, z_0^{II}} \tag{C-98}$$

Pericenter altitude Before listing the derivatives, some intermediate variables will be defined

$$\gamma = |V_{\infty I}|^2 = V_{\infty x}^2 + V_{\infty y}^2 + V_{\infty z}^2$$
(C-99)

$$\beta = |V_{\infty^{II}}|^2 = V_{\infty^{II}_x}^2 + V_{\infty^{II}_y}^2 + V_{\infty^{II}_z}^2$$
(C-100)

The following derivatives have been obtained from Ellison et al. [2013].

$$\frac{\partial F_{\text{hfly}}}{\partial V_{\varpi_x^I}} = \frac{-\mu_{\text{p}}\cos\left(\frac{\arccos\alpha}{2}\right)\left(V_{\varpi_x^{II}}V_{\varpi_y^I}^2 - V_{\varpi_x^I}V_{\varpi_y^{II}}V_{\varpi_y^I} + V_{\varpi_x^{II}}V_{\varpi_z^I}^2 - V_{\varpi_x^I}V_{\varpi_z^{II}}V_{\varpi_z^I}\right)}{(\alpha - 1)\sqrt{1 - \alpha^2}\sqrt{\beta^3}\sqrt{\gamma^3}} \\
\frac{\partial F_{\text{hfly}}}{\partial V_{\varpi_y^I}} = \frac{-\mu_{\text{p}}\cos\left(\frac{\arccos\alpha}{2}\right)\left(V_{\varpi_y^{II}}V_{\varpi_x^I}^2 - V_{\varpi_y^I}V_{\varpi_x^{II}}V_{\varpi_x^I} + V_{\varpi_y^{II}}V_{\varpi_z^I}^2 - V_{\varpi_y^I}V_{\varpi_z^{II}}V_{\varpi_z^I}\right)}{(\alpha - 1)\sqrt{1 - \alpha^2}\sqrt{\beta^3}\sqrt{\gamma^3}} \\
\frac{\partial F_{\text{hfly}}}{\partial V_{\varpi_z^I}} = \frac{-\mu_{\text{p}}\cos\left(\frac{\arccos\alpha}{2}\right)\left(V_{\varpi_z^{II}}V_{\varpi_x^I}^2 - V_{\varpi_z^I}V_{\varpi_x^{II}}V_{\varpi_z^I} + V_{\varpi_z^{II}}V_{\varpi_z^I}^2 - V_{\varpi_z^I}V_{\varpi_y^{II}}V_{\varpi_y^I}\right)}{(\alpha - 1)\sqrt{1 - \alpha^2}\sqrt{\beta^3}\sqrt{\gamma^3}}$$

$$\begin{split} \frac{\partial F_{\text{hfly}}}{\partial V_{\infty_x^{II}}} &= \frac{\mu_{\text{p}}}{\beta^2} \Bigg[2V_{\infty_x^{II}} \left(1 - \frac{1}{\sin(\frac{\arccos\alpha}{2})} \right) - \frac{\cos(\frac{\arccos\alpha}{2})}{(\alpha - 1)\sqrt{1 - \alpha^2}\sqrt{\beta}\sqrt{\gamma}} \cdot \\ & \left(V_{\infty_x^{I}} V_{\infty_y^{II}}^{2II} - V_{\infty_y^{I}} V_{\infty_x^{II}} V_{\infty_y^{II}} + V_{\infty_x^{I}} V_{\infty_z^{II}}^{2II} - V_{\infty_z^{I}} V_{\infty_x^{II}} V_{\infty_z^{II}} \right) \Bigg] \\ \frac{\partial F_{\text{hfly}}}{\partial V_{\infty_y^{II}}} &= \frac{\mu_{\text{p}}}{\beta^2} \Bigg[2V_{\infty_y^{II}} \left(1 - \frac{1}{\sin(\frac{\arccos\alpha}{2})} \right) - \frac{\cos(\frac{\arccos\alpha}{2})}{(\alpha - 1)\sqrt{1 - \alpha^2}\sqrt{\beta}\sqrt{\gamma}} \cdot \\ & \left(V_{\infty_y^{I}} V_{\infty_x^{II}}^{2II} - V_{\infty_x^{I}} V_{\infty_y^{II}} V_{\infty_x^{II}} + V_{\infty_y^{I}} V_{\infty_z^{II}}^{2II} - V_{\infty_z^{I}} V_{\infty_z^{II}} \right) \Bigg] \\ \frac{\partial F_{\text{hfly}}}{\partial V_{\infty_z^{II}}} &= \frac{\mu_{\text{p}}}{\beta^2} \Bigg[2V_{\infty_x^{II}} \left(1 - \frac{1}{\sin(\frac{\arccos\alpha}{2})} \right) - \frac{\cos(\frac{\arccos\alpha}{2})}{(\alpha - 1)\sqrt{1 - \alpha^2}\sqrt{\beta}\sqrt{\gamma}} \cdot \\ & \left(V_{\infty_z^{I}} V_{\infty_x^{II}}^{2II} - V_{\infty_x^{I}} V_{\infty_z^{II}} V_{\infty_z^{II}} + V_{\infty_z^{I}} V_{\infty_y^{II}}^{2II} - V_{\infty_y^{I}} V_{\infty_z^{II}} V_{\infty_z^{II}} \right) \Bigg] \end{split}$$

C-5-3 Return-node constraint

The derivatives of the re-entry velocity constraint are trivial:

$$\frac{\partial \text{Re-entry velocity constraint}}{V_{\infty,x_f}} = \frac{V_{\infty,x_f}}{\sqrt{V_{\infty x_f}^2 + V_{\infty,y_f}^2 + V_{\infty,z_f}^2}}$$
(C-101)

$$\frac{\partial \text{Re-entry velocity constraint}}{V_{\infty,y_f}} = \frac{V_{\infty,y_f}}{\sqrt{V_{\infty,x_f}^2 + V_{\infty,y_f}^2 + V_{\infty,z_f}^2}}$$
(C-102)

$$\frac{\partial \text{Re-entry velocity constraint}}{V_{\infty, z_f \log 2}} = \frac{V_{\infty, z_f}}{\sqrt{V_{\infty, x_f}^2 + V_{\infty, y_f}^2 + V_{\infty, z_f}^2}}$$
(C-103)

$$(C-104)$$

·

Appendix D

Extended TNC transformation matrix

D-1 Forward propagation

The change in the state and mass of the spacecraft and in the mass of the power subsystem before the manoeuvre must be mapped to a change in those elements after the manoeuvre. Therefore,

$$\begin{bmatrix} \Delta x_s \\ \Delta y_s \\ \Delta z_s \\ \Delta \dot{x}_s \\ \Delta M_{AP_0} \end{bmatrix}_{1^+} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ a & b & c & d & e & f & f_M & f_P \\ g & h & i & j & k & l & l_M & l_P \\ g & h & i & j & k & l & l_M & l_P \\ m & n & o & p & q & r & r_M & r_P \\ s & t & u & 0 & 0 & 0 & v_M & v_P \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \Delta x_s \\ \Delta y_s \\ \Delta \dot{x}_s \\ \Delta \dot{x}_s \\ \Delta \dot{x}_s \\ \Delta M_{AP_0} \end{bmatrix}_{1^-}$$
(D-1)

First of all, a, d, f_M and f_P will be calculated. Using a similar method, elements a up to r_P can be found. Afterwards, s, v_M and v_P will be calculated. Similarly, t and u can be calculated.

Considering that

$$\dot{x}^{+} = \dot{x}^{-} + \Delta V_{\max} \left(u_T T_i + u_N N_i + u_C C_i \right)$$
 (D-2)

One can see that

$$a = \frac{\partial \dot{x}^{+}}{\partial x^{-}} = \Delta V_{\max} \left(u_{T} \frac{\partial T_{i}^{-}}{\partial x^{-}} + u_{N} \frac{\partial N_{i}^{-}}{\partial x^{-}} + u_{C} \frac{\partial C_{i}^{-}}{\partial x^{-}} \right) + \frac{\partial \Delta V_{\max}}{\partial x^{-}} \cdot \left(u_{T} T_{i} + u_{N} N_{i} + u_{C} C_{i} \right)$$

$$d = \frac{\partial \dot{x}^{+}}{\partial \dot{x}^{-}} = 1 + \Delta V_{\max} \left(u_{T} \frac{\partial T_{i}^{-}}{\partial \dot{x}^{-}} + u_{N} \frac{\partial N_{i}^{-}}{\partial \dot{x}^{-}} + u_{C} \frac{\partial C_{i}^{-}}{\partial \dot{x}^{-}} \right) + \frac{\partial \Delta V_{\max}}{\partial \dot{x}^{-}} \cdot \left(u_{T} T_{i} + u_{N} N_{i} + u_{C} C_{i} \right)$$

$$f_{M} = \frac{\partial \dot{x}^{+}}{\partial M^{-}} = \frac{\partial \Delta V_{\max}}{\partial M^{-}} \cdot \left(u_{T} T_{i} + u_{N} N_{i} + u_{C} C_{i} \right)$$

$$f_{P} = \frac{\partial \dot{x}^{+}}{\partial M_{P0}} = \frac{\partial \Delta V_{\max}}{\partial M_{P0}} \cdot \left(u_{T} T_{i} + u_{N} N_{i} + u_{C} C_{i} \right)$$

$$(D-3)$$

Considering that

$$M^{+} = M^{-} \exp\left(\frac{-\sqrt{u_{T}^{2} + u_{N}^{2} + u_{C}^{2}}}{I_{sp}g_{0}} \cdot \Delta V_{\max}\right)$$
(D-4)

One can see that

$$s = \frac{\partial M^{+}}{\partial x^{-}} = -M^{+} \frac{\sqrt{u_{T}^{2} + u_{N}^{2} + u_{C}^{2}}}{I_{sp}g_{0}} \cdot \frac{\partial \Delta V_{\max}}{\partial x^{-}}$$

$$v_{M} = \frac{\partial M^{+}}{\partial M^{-}} = -M^{+} \frac{\sqrt{u_{T}^{2} + u_{N}^{2} + u_{C}^{2}}}{I_{sp}g_{0}} \cdot \frac{\partial \Delta V_{\max}}{\partial M^{-}}$$

$$v_{P} = \frac{\partial M^{+}}{\partial M_{P0}^{-}} = -M^{+} \frac{\sqrt{u_{T}^{2} + u_{N}^{2} + u_{C}^{2}}}{I_{sp}g_{0}} \cdot \frac{\partial \Delta V_{\max}}{\partial M_{P0}}$$
(D-5)

The partial derivatives for the case where power is independent of the heliocentric distance are defined as:

$$\frac{\partial \Delta V_{\text{max}}}{\partial x^{-}} = 0$$

$$\frac{\partial \Delta V_{\text{max}}}{\partial \dot{x}^{-}} = 0$$

$$\frac{\partial \Delta V_{\text{max}}}{\partial M^{-}} = \frac{-\Delta V_{\text{max}}}{M^{-}}$$

$$\frac{\partial \Delta V_{\text{max}}}{\partial M_{P0}} = \frac{\Delta V_{\text{max}}}{M_{P0}}$$
(D-6)

The partial derivatives for the case where the power is dependent of the heliocentric distance are defined as:

$$\frac{\partial \Delta V_{\text{max}}}{\partial x^{-}} = \frac{-2\Delta V_{\text{max}}}{R^{-}} \cdot \frac{x^{-}}{R^{-}}$$

$$\frac{\partial \Delta V_{\text{max}}}{\partial \dot{x}^{-}} = 0$$

$$\frac{\partial \Delta V_{\text{max}}}{\partial M^{-}} = \frac{-\Delta V_{\text{max}}}{M^{-}}$$

$$\frac{\partial \Delta V_{\text{max}}}{\partial M_{P0}} = \frac{\Delta V_{\text{max}}}{M_{P0}}$$
(D-7)

D-2 Backward propagation

Again,

First of all, a, d, f_M and f_P will be calculated. Using a similar method, elements a up to r_P can be found. Afterwards, s, v_M and v_P will be calculated. Similarly, t and u can be calculated.

Considering that

$$\dot{x}^{-} = \dot{x}^{+} - \Delta V_{\max} \left(u_T T_i + u_N N_i + u_C C_i \right)$$
 (D-9)

One can see that

$$a = \frac{\partial \dot{x}^{-}}{\partial x^{+}} = -\Delta V_{\max} \left(u_{T} \frac{\partial T_{i}^{-}}{\partial x^{-}} + u_{N} \frac{\partial N_{i}^{-}}{\partial x^{-}} + u_{C} \frac{\partial C_{i}^{-}}{\partial x^{+}} \right) - \frac{\partial \Delta V_{\max}}{\partial x^{-}} \cdot \left(u_{T} T_{i} + u_{N} N_{i} + u_{C} C_{i} \right)$$

$$d = \frac{\partial \dot{x}^{-}}{\partial \dot{x}^{+}} = 1 - \Delta V_{\max} \left(u_{T} \frac{\partial T_{i}^{-}}{\partial \dot{x}^{-}} + u_{N} \frac{\partial N_{i}^{-}}{\partial \dot{x}^{-}} + u_{C} \frac{\partial C_{i}^{-}}{\partial \dot{x}^{+}} \right) - \frac{\partial \Delta V_{\max}}{\partial \dot{x}^{-}} \cdot \left(u_{T} T_{i} + u_{N} N_{i} + u_{C} C_{i} \right)$$

$$f_{M} = \frac{\partial \dot{x}^{-}}{\partial M^{+}} = -\frac{\partial \Delta V_{\max}}{\partial M^{+}} \cdot \left(u_{T} T_{i} + u_{N} N_{i} + u_{C} C_{i} \right)$$

$$(D-10)$$

$$f_{P} = \frac{\partial \dot{x}^{-}}{\partial M_{P0}} = -\frac{\partial \Delta V_{\max}}{\partial M_{P0}} \cdot \left(u_{T} T_{i} + u_{N} N_{i} + u_{C} C_{i} \right)$$

Considering that

•

$$M^{-} = M^{+} \exp\left(\frac{\sqrt{u_{T}^{2} + u_{N}^{2} + u_{C}^{2}}}{I_{sp}g_{0}} \cdot \Delta V_{\max}\right)$$
(D-11)

One can see that

$$s = \frac{\partial M^{-}}{\partial x^{+}} = M^{-} \frac{\sqrt{u_{T}^{2} + u_{N}^{2} + u_{C}^{2}}}{I_{sp}g_{0}} \cdot \frac{\partial \Delta V_{\max}}{\partial x^{+}}$$

$$v_{M} = \frac{\partial M^{-}}{\partial M^{+}} = M^{-} \frac{\sqrt{u_{T}^{2} + u_{N}^{2} + u_{C}^{2}}}{I_{sp}g_{0}} \cdot \frac{\partial \Delta V_{\max}}{\partial M^{+}}$$

$$v_{P} = \frac{\partial M^{-}}{\partial M_{P0}^{+}} = M^{-} \frac{\sqrt{u_{T}^{2} + u_{N}^{2} + u_{C}^{2}}}{I_{sp}g_{0}} \cdot \frac{\partial \Delta V_{\max}}{\partial M_{P0}}$$
(D-12)

The partial derivatives for the power independent of the heliocentric distance are defined as:

$$\frac{\partial \Delta V_{\max}}{\partial x^{+}} = 0$$

$$\frac{\partial \Delta V_{\max}}{\partial \dot{x}^{+}} = 0$$

$$\frac{\partial \Delta V_{\max}}{\partial M^{+}} = \frac{-\Delta V_{\max}}{M^{+}} \cdot \frac{1}{1 + V_{\max} \frac{\sqrt{u_{T}^{2} + u_{N}^{2} + u_{C}^{2}}}{I_{sp}g_{0}}}$$

$$\frac{\partial \Delta V_{\max}}{\partial M_{P0}} = \frac{\Delta V_{\max}}{M_{P0}} \cdot \frac{1}{1 + V_{\max} \frac{\sqrt{u_{T}^{2} + u_{N}^{2} + u_{C}^{2}}}{I_{sp}g_{0}}}$$
(D-13)

The partial derivatives for the case where the power is dependent of the heliocentric distance are defined as:

$$\frac{\partial \Delta V_{\max}}{\partial x^{+}} = \frac{-2\Delta V_{\max}}{R^{-}} \cdot \frac{x^{+}}{R^{-}} \cdot \frac{1}{1 + V_{\max} \sqrt{\frac{u_{T}^{2} + u_{N}^{2} + u_{C}^{2}}{I_{sp}g_{0}}}}$$

$$\frac{\partial \Delta V_{\max}}{\partial \dot{x}^{+}} = 0$$

$$\frac{\partial \Delta V_{\max}}{\partial M^{+}} = \frac{-\Delta V_{\max}}{M^{+}} \cdot \frac{1}{1 + V_{\max} \sqrt{\frac{u_{T}^{2} + u_{N}^{2} + u_{C}^{2}}{I_{sp}g_{0}}}}$$

$$\frac{\partial \Delta V_{\max}}{\partial M_{P0}} = \frac{\Delta V_{\max}}{M_{P0}} \cdot \frac{1}{1 + V_{\max} \sqrt{\frac{u_{T}^{2} + u_{N}^{2} + u_{C}^{2}}{I_{sp}g_{0}}}}$$
(D-14)

Appendix E

Extended TNC matrix

E-1 Forward propagation

The matrix $\Delta VNC_{\rm ext,1}$ introduced in Subsection 12-3-2 will now be derived.

Considering that

$$\dot{x}^{+} = \dot{x}^{-} + \Delta V_{\max} \left(u_T T_i + u_N N_i + u_C C_i \right)$$
 (E-2)

One can see that

$$a = \frac{\partial \dot{x}^{+}}{\partial u_{T}} = \Delta V_{\max} T_{i}$$

$$d = \frac{\partial \dot{x}^{+}}{\partial I_{sp}} = \frac{-\Delta V_{\max}}{I_{sp}} \cdot \left(u_{T} T_{i} + u_{N} N_{i} + u_{C} C_{i} \right)$$
(E-3)

Considering that

$$M^{+} = M^{-} \exp\left(\frac{-\sqrt{u_{T}^{2} + u_{N}^{2} + u_{C}^{2}}}{I_{\rm sp}g_{0}} \cdot \Delta V_{\rm max}\right)$$
(E-4)

One can see that

$$m = \frac{\partial M^{+}}{\partial u_{T}} = -M^{+} \frac{\Delta V_{\max}}{I_{sp}g_{0}} \cdot \frac{u_{T}}{\sqrt{u_{T}^{2} + u_{N}^{2} + u_{C}^{2}}}$$
$$p = \frac{\partial M^{+}}{\partial u_{T}} = M^{+} \frac{2\sqrt{u_{T}^{2} + u_{N}^{2} + u_{C}^{2}} \Delta V_{\max}}{I_{sp}^{2}g_{0}}$$
(E-5)

E-2 Backward propagation

The matrix $\Delta VNC_{\mathrm{ext},2N}$ introduced in Subsection 12-3-2 will now be derived.

$$\begin{bmatrix} \Delta x_s \\ \Delta y_s \\ \Delta y_s \\ \Delta z_s \\ \Delta \dot{x}_s \\ \Delta \dot{y}_s \\ \Delta \dot{z}_s \\ \Delta M \\ \Delta$$

Considering that

$$\dot{x}^{-} = \dot{x}^{+} - \Delta V_{\max} \left(u_T T_i + u_N N_i + u_C C_i \right)$$
(E-7)

One can see that

$$a = \frac{\partial \dot{x}^{-}}{\partial u_{T}} = -\Delta V_{\max} T_{i} - \left(u_{T} T_{i} + u_{N} N_{i} + u_{C} C_{i} \right) \frac{\partial \Delta V_{\max}}{\partial u_{T}}$$

$$d = \frac{\partial \dot{x}^{-}}{\partial I_{\rm sp}} = -\left(u_{T} T_{i} + u_{N} N_{i} + u_{C} C_{i} \right) \frac{\partial \Delta V_{\max}}{\partial I_{\rm sp}}$$
(E-8)

Considering that

$$M^{-} = M^{+} \exp\left(\frac{\sqrt{u_{T}^{2} + u_{N}^{2} + u_{C}^{2}}}{I_{sp}g_{0}} \cdot \Delta V_{\max}\right)$$
(E-9)

One can see that

$$m = \frac{\partial M^{-}}{\partial u_{T}} = \frac{M^{-}}{I_{\rm sp}g_{0}} \cdot \left(\frac{\Delta V_{\rm max}u_{T}}{\sqrt{u_{T}^{2} + u_{N}^{2} + u_{C}^{2}}} + \sqrt{u_{T}^{2} + u_{N}^{2} + u_{C}^{2}}\frac{\partial\Delta V_{\rm max}}{\partial u_{T}}\right)$$
$$p = \frac{\partial M^{-}}{\partial u_{T}} = \frac{M^{-}}{g_{0}}\sqrt{u_{T}^{2} + u_{N}^{2} + u_{C}^{2}} \cdot \left(\frac{-\Delta V_{\rm max}}{I_{\rm sp}^{2}} + \frac{1}{I_{sp}}\frac{\partial\Delta V_{\rm max}}{\partial I_{\rm sp}}\right)$$
(E-10)

The partial derivatives are calculated to be:

$$\frac{\partial \Delta V_{\max}}{\partial u_T} = \frac{-\Delta V_{\max}^2 u_T}{I_{sp} g_0 \sqrt{u_T^2 + u_N^2 + u_C^2}} \cdot \frac{1}{1 + V_{\max} \frac{\sqrt{u_T^2 + u_N^2 + u_C^2}}{I_{sp} g_0}}$$
(E-11)
$$\frac{\partial \Delta V_{\max}}{\partial I_{sp}} = \frac{-\Delta V_{\max}}{I_{sp}} \left(1 - \frac{\Delta V_{\max} \sqrt{u_T^2 + u_N^2 + u_C^2}}{I_{sp} g_0} \right) \cdot \frac{1}{1 + V_{\max} \frac{\sqrt{u_T^2 + u_N^2 + u_C^2}}{I_{sp} g_0}}$$

Appendix F

Changes to the Jacobian for the throttled representation

In this chapter, it will be shown how the throttled representation influences the Jacobian. The derivatives of the match point constraints with respect to the initial and final coordinates, the initial and final velocities, the initial and final excess velocities, the initial and final mass M_0 and M_f and mass of the SEP system M_{P0} will be given here.

Based on the discussion in Subsection 12-3-1, it is known that

$\Psi_{\text{ext},MP,0} =$	$\begin{bmatrix} \frac{\partial x_{s,\mathrm{MP f}}}{\partial x_{s,0}} \\ \frac{\partial y_{s,\mathrm{MP f}}}{\partial x_{s,0}} \\ \frac{\partial z_{s,\mathrm{MP f}}}{\partial x_{s,0}} \\ \frac{\partial \dot{x}_{s,\mathrm{MP f}}}{\partial x_{s,0}} \\ \frac{\partial \dot{y}_{s,\mathrm{MP f}}}{\partial x_{s,0}} \\ \frac{\partial \dot{z}_{s,\mathrm{MP f}}}{\partial x_{s,0}} \\ \frac{\partial M_{\mathrm{MP f}}}{\partial x_{s,0}} \\ \frac{\partial M_{\mathrm{MP f}}}{\partial x_{s,0}} \\ 0 \end{bmatrix}$	$\begin{array}{c} \frac{\partial x_{s,\mathrm{MP f}}}{\partial y_{s,0}} \\ \frac{\partial y_{s,\mathrm{MP f}}}{\partial y_{s,0}} \\ \frac{\partial z_{s,\mathrm{MP f}}}{\partial y_{s,0}} \\ \frac{\partial \dot{z}_{s,\mathrm{MP f}}}{\partial y_{s,0}} \\ \frac{\partial \dot{x}_{s,\mathrm{MP f}}}{\partial y_{s,0}} \\ \frac{\partial \dot{y}_{s,\mathrm{MP f}}}{\partial y_{s,0}} \\ \frac{\partial \dot{z}_{s,\mathrm{MP f}}}{\partial y_{s,0}} \\ \frac{\partial M_{\mathrm{MP f}}}{\partial y_{s,0}} \\ \frac{\partial M_{\mathrm{MP f}}}{\partial y_{s,0}} \\ 0 \end{array}$	$\begin{array}{c} \frac{\partial x_{s,\mathrm{MP \ f}}}{\partial z_{s,0}} \\ \frac{\partial y_{s,\mathrm{MP \ f}}}{\partial z_{s,0}} \\ \frac{\partial z_{s,\mathrm{MP \ f}}}{\partial z_{s,0}} \\ \frac{\partial \dot{z}_{s,\mathrm{MP \ f}}}{\partial z_{s,0}} \\ \frac{\partial \dot{y}_{s,\mathrm{MP \ f}}}{\partial z_{s,0}} \\ \frac{\partial \dot{y}_{s,\mathrm{MP \ f}}}{\partial z_{s,0}} \\ \frac{\partial \dot{z}_{s,0}}{\partial z_{s,0}} \\ \frac{\partial M_{\mathrm{MP \ f}}}{\partial z_{s,0}} \\ \frac{\partial M_{\mathrm{MP \ f}}}{\partial z_{s,0}} \\ \end{array}$	$\begin{array}{c} \frac{\partial x_{s,\mathrm{MP \ f}}}{\partial \dot{x}_{s,0}} \\ \frac{\partial y_{s,\mathrm{MP \ f}}}{\partial \dot{x}_{s,0}} \\ \frac{\partial z_{s,\mathrm{MP \ f}}}{\partial \dot{x}_{s,0}} \\ \frac{\partial \dot{z}_{s,\mathrm{MP \ f}}}{\partial \dot{x}_{s,0}} \\ \frac{\partial \dot{x}_{s,0}}{\partial \dot{x}_{s,0}} \\ \frac{\partial \dot{y}_{s,\mathrm{MP \ f}}}{\partial \dot{x}_{s,0}} \\ \frac{\partial \dot{z}_{s,\mathrm{MP \ f}}}{\partial \dot{x}_{s,0}} \\ \frac{\partial M_{\mathrm{MP \ f}}}{\partial \dot{x}_{s,0}} \\ \frac{\partial M_{\mathrm{MP \ f}}}{\partial \dot{x}_{s,0}} \\ \end{array}$	$\begin{array}{c} \frac{\partial x_{s,\mathrm{MP \ f}}}{\partial \dot{y}_{s,0}} \\ \frac{\partial y_{s,\mathrm{MP \ f}}}{\partial \dot{y}_{s,0}} \\ \frac{\partial z_{s,\mathrm{MP \ f}}}{\partial \dot{y}_{s,0}} \\ \frac{\partial \dot{z}_{s,\mathrm{MP \ f}}}{\partial \dot{y}_{s,0}} \\ \frac{\partial \dot{x}_{s,\mathrm{MP \ f}}}{\partial \dot{y}_{s,0}} \\ \frac{\partial \dot{y}_{s,\mathrm{MP \ f}}}{\partial \dot{y}_{s,0}} \\ \frac{\partial \dot{z}_{s,\mathrm{MP \ f}}}{\partial \dot{y}_{s,0}} \\ \frac{\partial M_{\mathrm{MP \ f}}}{\partial \dot{y}_{s,0}} \\ \frac{\partial M_{\mathrm{MP \ f}}}{\partial \dot{y}_{s,0}} \\ 0 \end{array}$	$\begin{array}{c} \frac{\partial x_{s,\mathrm{MP \ f}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial y_{s,\mathrm{MP \ f}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial z_{s,\mathrm{MP \ f}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial \dot{z}_{s,\mathrm{MP \ f}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial \dot{x}_{s,\mathrm{MP \ f}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial \dot{y}_{s,\mathrm{MP \ f}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial \dot{z}_{s,0}}{\partial \dot{z}_{s,0}} \\ \frac{\partial M_{\mathrm{MP \ f}}}{\partial \dot{z}_{s,0}} \\ \frac{\partial M_{\mathrm{MP \ f}}}{\partial \dot{z}_{s,0}} \\ \end{array}$	$ \begin{array}{c} \frac{\partial x_{s,\mathrm{MP f}}}{\partial M_0} \\ \frac{\partial y_{s,\mathrm{MP f}}}{\partial M_0} \\ \frac{\partial z_{s,\mathrm{MP f}}}{\partial M_0} \\ \frac{\partial \dot{x}_{s,\mathrm{MP f}}}{\partial M_0} \\ \frac{\partial \dot{y}_{s,\mathrm{MP f}}}{\partial M_0} \\ \frac{\partial \dot{y}_{s,\mathrm{MP f}}}{\partial M_0} \\ \frac{\partial \dot{M}_{\mathrm{MP f}}}{\partial M_0} \\ \frac{\partial M_{\mathrm{MP f}}}{\partial M_0} \\ 0 \end{array} $	$ \begin{array}{c} \frac{\partial x_{s,\mathrm{MP f}}}{\partial M_{P0}} \\ \frac{\partial y_{s,\mathrm{MP f}}}{\partial M_{P0}} \\ \frac{\partial z_{s,\mathrm{MP f}}}{\partial M_{P0}} \\ \frac{\partial z_{s,\mathrm{MP f}}}{\partial M_{P0}} \\ \frac{\partial \dot{x}_{s,\mathrm{MP f}}}{\partial M_{P0}} \\ \frac{\partial \dot{y}_{s,\mathrm{MP f}}}{\partial M_{P0}} \\ \frac{\partial dx_{s,\mathrm{MP f}}}{\partial M_{P0}} \\ \frac{\partial M_{\mathrm{MP f}}}{\partial M_{P0}} \\ \frac{\partial M_{\mathrm{MP f}}}{\partial M_{P0}} \\ 1 \end{array} \right) $	
	L	9	5	5	5	5	9	- 」 (F	-1)

$$\Psi_{\text{ext},MP,f} = \begin{bmatrix} \frac{\partial x_{s,\text{MP b}}}{\partial x_{s,f}} & \frac{\partial x_{s,\text{MP b}}}{\partial y_{s,f}} & \frac{\partial x_{s,\text{MP b}}}{\partial z_{s,f}} & \frac{\partial x_{s,\text{MP b}}}{\partial \dot{x}_{s,f}} & \frac{\partial x_{s,\text{MP b}}}{\partial \dot{y}_{s,f}} & \frac{\partial x_{s,\text{MP b}}}{\partial \dot{z}_{s,f}} & \frac{\partial y_{s,\text{MP b}}}{\partial \dot{z}_{s,f}} & \frac{\partial y_{s,\text{MP b}}}{\partial \dot{z}_{s,f}} & \frac{\partial z_{s,\text{MP b}}}{\partial z_{s,f}} & \frac{\partial z_{s,\text{MP b}}}{\partial \dot{z}_{s,f}} & \frac{\partial z_{s,\text{MP b}}}{\partial z_{s,f}} & \frac{\partial z_{s,\text{MP b}}}{\partial d_{f}} & \frac{\partial z_{s,\text{MP b}}}{\partial M_{f}} & \frac{\partial z_{s,\text{MP b}}}{\partial M_{P0}} \\ & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \right]$$

The derivatives within the matrices $\Psi_{\text{ext},MP,0}$ and $\Psi_{\text{ext},MP,f}$ in Equations F-1 and F-2 can be used to find the required derivatives. As an example, the derivatives of the x-coordinate match point constraint will be given with respect to the initial and final coordinate x_0 and x_f , initial and final velocity \dot{x}_0 and \dot{x}_f , initial and final hyperbolic excess velocity V_{∞,x_0} and V_{∞,x_f} , initial and final mass M_0 and M_f and the mass of the SEP system. The derivatives of the other match point constraints Δy_{MP} , Δz_{MP} , $\Delta \dot{x}_{MP}$, $\Delta \dot{y}_{MP}$, $\Delta \dot{z}_{MP}$ and ΔM_{MP} with respect to the other parameters y_0 , z_0 , \dot{y}_0 , \dot{z}_0 , V_{∞,y_0} , V_{∞,z_0} , y_f , z_f , \dot{y}_f , \dot{z}_f , V_{∞,y_f} and V_{∞,z_f} can then be found in a similar way by utilizing different elements of the $\Psi_{\text{ext},MP,0}$ and $\Psi_{\text{ext},MP,f}$ matrices.

$$\frac{\partial \Delta x_{MP}}{\partial x_0} = \frac{\partial \left(x_{MP \text{ forward}} - x_{MP \text{ backward}}\right)}{\partial x_0} \\
= \frac{\partial x_{MP \text{ forward}}}{\partial x_0} - 0 \quad (F-3) \\
= \frac{\partial x_{MP \text{ forward}}}{\partial x_{s,0}} \cdot \frac{\partial x_{s,0}}{\partial x_0} \\
= \frac{\partial x_{MP \text{ forward}}}{\partial x_{s,0}} \cdot 1 \\
= \Psi_{\text{ext},MP,0}[1,1]$$

$$\frac{\partial \Delta x_{MP}}{\partial x_{f}} = \frac{\partial \left(x_{MP \text{ forward}} - x_{MP \text{ backward}} \right)}{\partial x_{f}} \\
= 0 - \frac{\partial x_{MP \text{ backward}}}{\partial x_{f}} \qquad (F-4) \\
= -\frac{\partial x_{MP \text{ backward}}}{\partial x_{s,f}} \cdot \frac{\partial x_{s,f}}{\partial x_{f}} \\
= -\frac{\partial x_{MP \text{ backward}}}{\partial x_{s,f}} \cdot 1 \\
= -\Psi_{\text{ext},MP,f}[1,1]$$

Similarly,

$$\frac{\partial \Delta x_{MP}}{\partial \dot{x}_{0}} = \frac{\partial \left(x_{MP \text{ forward}} - x_{MP \text{ backward}}\right)}{\partial \dot{x}_{0}} \\
= \frac{\partial x_{MP \text{ forward}}}{\partial \dot{x}_{0}} - 0 \quad (F-5) \\
= \frac{\partial x_{MP \text{ forward}}}{\partial \dot{x}_{s,0}} \cdot \frac{\partial \dot{x}_{s,0}}{\partial \dot{x}_{0}} \\
= \frac{\partial x_{MP \text{ forward}}}{\partial \dot{x}_{s,0}} \cdot 1 \\
= \Psi_{\text{ext},MP,0}[1, 4]$$

$$\frac{\partial \Delta x_{MP}}{\partial \dot{x}_{f}} = \frac{\partial \left(x_{MP \text{ forward}} - x_{MP \text{ backward}} \right)}{\partial \dot{x}_{f}} \\
= 0 - \frac{\partial x_{MP \text{ backward}}}{\partial \dot{x}_{f}} \qquad (F-6) \\
= -\frac{\partial x_{MP \text{ backward}}}{\partial \dot{x}_{s,f}} \cdot \frac{\partial \dot{x}_{s,f}}{\partial \dot{x}_{f}} \\
= -\frac{\partial x_{MP \text{ backward}}}{\partial \dot{x}_{s,f}} \cdot 1 \\
= -\Psi_{\text{ext},MP,f}[1, 4]$$

Similarly,

$$\frac{\partial \Delta x_{MP}}{\partial V_{\infty,x_0}} = \frac{\partial \left(x_{MP \text{ forward}} - x_{MP \text{ backward}} \right)}{\partial V_{\infty,x_0}} \\
= \frac{\partial x_{MP \text{ forward}}}{\partial V_{\infty,x_0}} - 0 \quad (F-7) \\
= \frac{\partial x_{MP \text{ forward}}}{\partial \dot{x}_{s,0}} \cdot \frac{\partial \dot{x}_{s,0}}{\partial V_{\infty,x_0}} \\
= \frac{\partial x_{MP \text{ forward}}}{\partial \dot{x}_{s,0}} \cdot 1 \\
= \Psi_{\text{ext},MP,0}[1, 4]$$

$$\frac{\partial \Delta x_{MP}}{\partial V_{\infty,x_f}} = \frac{\partial \left(x_{MP \text{ forward}} - x_{MP \text{ backward}} \right)}{\partial V_{\infty,x_f}} \\
= 0 - \frac{\partial x_{MP \text{ backward}}}{\partial V_{\infty,x_f}} \\
= -\frac{\partial x_{MP \text{ backward}}}{\partial \dot{x}_{s,f}} \cdot \frac{\partial \dot{x}_{s,f}}{\partial V_{\infty,x_f}}$$
(F-8)

$$= -\frac{\partial x_{\text{MP backward}}}{\partial \dot{x}_{s,f}} \cdot 1$$
$$= -\Psi_{\text{ext},MP,f}[1,4]$$

Similarly,

$$\frac{\partial \Delta x_{MP}}{\partial M_0} = \frac{\partial \left(x_{MP \text{ forward}} - x_{MP \text{ backward}} \right)}{\partial M_0}$$
$$= \frac{\partial x_{MP \text{ forward}}}{\partial M_0} - 0$$
$$= \Psi_{\text{ext},MP,0}[1,7]$$
(F-9)

$$\frac{\partial \Delta x_{MP}}{\partial M_f} = \frac{\partial \left(x_{MP \text{ forward}} - x_{MP \text{ backward}} \right)}{\partial M_f}$$

= $0 - \frac{\partial x_{MP \text{ backward}}}{\partial M_f}$ (F-10)
= $-\Psi_{\text{ext},MP,f}[1,7]$

Similarly,

$$\frac{\partial \Delta x_{MP}}{\partial M_{P0}} = \frac{\partial \left(x_{MP \text{ forward}} - x_{MP \text{ backward}} \right)}{\partial M_{P0}} \\
= \frac{\partial x_{MP \text{ forward}}}{\partial M_{P0}} - \frac{\partial x_{MP \text{ forward}}}{\partial M_{P0}} \\
= \Psi_{\text{ext}, MP, 0}[1, 8] - \Psi_{\text{ext}, MP, f}[1, 8]$$
(F-11)

Appendix G

Analytical derivatives of ephemeris constraints with respect to time

For the optimization of realistic scenarios, the departure and arrival time of a trajectory have to be consistent with the heliocentric coordinates and velocities of the initial and final control nodes of a trajectory. Therefore, Meeus' method to calculate the ephemeris at a certain epoch for a certain planet have been used. This method uses Meeus' polynomials [Meeus, 1991] to model the orbital motion of the planets as function of the time T measured in Julian centuries.

$$a = a_0 + a_1 T + a_2 T^2 + a_3 T^3$$

$$e = e_0 + e_1 T + e_2 T^2 + e_3 T^3$$

$$i = i_0 + i_1 T + i_2 T^2 + i_3 T^3$$

$$L = L_0 + L_1 T + L_2 T^2 + L_3 T^3$$

$$\Omega = \Omega_0 + \Omega_1 T + \Omega_2 T^2 + \Omega_3 T^3$$

$$\Pi = \Pi_0 + \Pi_1 T + \Pi_2 T^2 + \Pi_3 T^3$$

(G-1)

where a is the semi-major axis, e the eccentricity, i the inclination, L the mean longitude of the planet, Ω the longitude of the ascending node and Π the longitude of the perihelion. The argument of perihelion ω can be found from

$$\omega = \Pi - \Omega \tag{G-2}$$

The true anomaly θ can be found from

$$\theta = M + C_{\text{cen}}$$

$$= L - \Pi + C_{\text{cen}}$$

$$= L - \Pi + \left(2e - \frac{e^3}{4} + \frac{5}{96}e^5\right)\sin\left(M_{\text{rad}}\right) + \left(\frac{5}{4}e^2 - \frac{11}{24}e^4\right)\sin\left(2M_{\text{rad}}\right) \quad (\text{G-3})$$

$$+ \left(\frac{13}{12}e^3 - \frac{43}{64}e^5\right)\sin\left(3M_{\text{rad}}\right) + \frac{103}{96}e^4\sin\left(4M_{\text{rad}}\right) + \frac{1097}{960}e^5\sin\left(5M_{\text{rad}}\right)$$

where M is the mean anomaly. Note that the results of the equations defining angles are in degrees. However, in the definition of the true anomaly, the sines of M, 2M, ..., 5M are taken with respect to the radial value of M. The result of the equation defining the semi-major axis is in AU.

If one needs to know how the Cartesian coordinates and velocities resulting from this ephemeris changes with time, several derivatives have to be calculated. First of all, the derivatives of the Kepler elements have to be found:

$$\frac{\partial a}{\partial T} = a_1 T + 2a_2 T + 3a_3 T^2 \tag{G-4}$$

$$\frac{\partial e}{\partial T} = e_1 T + 2e_2 T + 3e_3 T^2$$
 (G-5)

$$\frac{\partial i}{\partial T} = i_1 T + 2i_2 T + 3i_3 T^2 \tag{G-6}$$

$$\frac{\partial\Omega}{\partial T} = \Omega_1 T + 2\Omega_2 T + 3\Omega_3 T^2 \tag{G-7}$$

$$\frac{\partial \Pi}{\partial T} = \Pi_1 T + 2\Pi_2 T + 3\Pi_3 T^2 \tag{G-8}$$

The derivative of the argument of perihelion ω can be found from

$$\frac{\partial \omega}{\partial T} = \frac{\partial \Pi}{\partial T} - \frac{\partial \Omega}{\partial T}$$
(G-9)

The derivative of the true anomaly θ can be found from

$$\frac{\partial \theta}{\partial T} = \frac{\partial M}{\partial T} + \frac{\partial C_{cen}}{\partial T} \\
= \frac{\partial L}{\partial T} - \frac{\partial \Pi}{\partial T} + \frac{\partial C_{cen}}{\partial T} \\
= \frac{\partial L}{\partial T} - \frac{\partial \Pi}{\partial T} \\
+ \left(2 - \frac{3e^2}{4} + \frac{25}{96}e^4\right) \sin(M)\frac{\partial e}{\partial T} + \left(2e - \frac{e^3}{4} + \frac{5}{96}e^5\right) \cos(M) \\
+ \left(\frac{5}{2}e - \frac{11}{6}e^3\right) \sin(2M)\frac{\partial e}{\partial T} + \left(\frac{5}{4}e^2 - \frac{11}{24}e^4\right) 2\cos(2M) \\
+ \left(\frac{13}{4}e^2 - \frac{215}{64}e^4\right) \sin(3M)\frac{\partial e}{\partial T} + \left(\frac{13}{12}e^3 - \frac{43}{64}e^5\right) 3\cos(3M) \\
+ \frac{103}{24}e^3 \sin(4M)\frac{\partial e}{\partial T} + \frac{103}{24}e^4 \cos(4M) \\
+ \frac{1097}{192}e^4 \sin(5M)\frac{\partial e}{\partial T} + \frac{1097}{960}e^5 \cos(5M)$$
(G-10)

Note that the derivative of the semi-major axis has unit [AU per Julian century]. All the other derivatives have units [degrees per Julian century].

These derivatives can then be used to find the derivatives of the initial Cartesian coordinates and velocities if one knows how to convert from Kepler elements to Cartesian elements. This conversion can be done using the following equations [Wertz, 2009]:

$$\begin{bmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} l_1 & l_2 & 0 & 0 \\ m_1 & m_2 & 0 & 0 \\ n_1 & n_2 & 0 & 0 \\ 0 & 0 & l_1 & l_2 \\ 0 & 0 & m_1 & m_2 \\ 0 & 0 & n_1 & n_2 \end{bmatrix} \begin{bmatrix} r \cos \theta \\ r \sin \theta \\ -\frac{\mu}{H} \sin \theta \\ \frac{\mu}{H} (e + \cos \theta) \end{bmatrix}$$
(G-11)

where

$$l_{1} = \cos(\Omega)\cos(\omega) - \sin(\Omega)\sin(\omega)\cos(i)$$

$$l_{2} = -\cos(\Omega)\sin(\omega) - \sin(\Omega)\cos(\omega)\cos(i)$$

$$m_{1} = \sin(\Omega)\cos(\omega) + \cos(\Omega)\sin(\omega)\cos(i)$$

$$m_{2} = -\sin(\Omega)\sin(\omega) + \cos(\Omega)\cos(\omega)\cos(i)$$

$$n_{1} = \sin(\omega)\sin(i)$$

$$n_{2} = \cos(\omega)\sin(i)$$

$$H = \sqrt{\mu a(1 - e^{2})}$$
(G-12)

However, this method only works with angles in radians and the semi-major axis in meters. Therefore, the derivatives in equations G-4 up to G-10 have been converted into units of [m per Julian century] and [rad per Julian century] for the semi-major axis and angles respectively. In order to find the derivatives of the initial coordinates and velocities, one can use the following equations:

$$\frac{\partial x}{\partial T} = l_1 \frac{\partial r \cos \theta}{\partial T} + \frac{\partial l_1}{\partial T} r \cos \theta + l_2 \frac{\partial r \sin \theta}{\partial T} + \frac{\partial l_2}{\partial T} r \sin \theta$$
(G-13)

$$\frac{\partial y}{\partial T} = m_1 \frac{\partial r \cos \theta}{\partial T} + \frac{\partial m_1}{\partial T} r \cos \theta + m_2 \frac{\partial r \sin \theta}{\partial T} + \frac{\partial m_2}{\partial T} r \sin \theta$$
(G-14)

$$\frac{\partial z}{\partial T} = n_1 \frac{\partial r \cos \theta}{\partial T} + \frac{\partial n_1}{\partial T} r \cos \theta + n_2 \frac{\partial r \sin \theta}{\partial T} + \frac{\partial n_2}{\partial T} r \sin \theta$$
(G-15)

$$\frac{\partial \dot{x}}{\partial T} = \frac{-\mu}{H^2} \frac{\partial H}{\partial T} \left(-l_1 \sin \theta + l_2 (e + \cos \theta) \right)$$
$$\frac{\mu}{H} \left(-\frac{\partial l_1}{\partial T} \sin \theta - l_1 \cos \theta \frac{\partial \theta}{\partial T} + \frac{\partial l_2}{\partial T} (e + \cos \theta) + l_2 \frac{\partial e}{\partial T} - \sin \theta l_2 \frac{\partial \theta}{\partial T} \right)$$
(G-16)

$$\frac{\partial \dot{y}}{\partial T} = \frac{-\mu}{H^2} \frac{\partial H}{\partial T} \left(-m_1 \sin \theta + m_2 (e + \cos \theta) \right)$$
(G-17)

$$\frac{\mu}{H} \left(-\frac{\partial m_1}{\partial T} \sin \theta - m_1 \cos \theta \frac{\partial \theta}{\partial T} + \frac{\partial m_2}{\partial T} (e + \cos \theta) + m_2 \frac{\partial e}{\partial T} - \sin \theta m_2 \frac{\partial \theta}{\partial T} \right)$$
$$\frac{\partial \dot{z}}{\partial T} = \frac{-\mu}{H^2} \frac{\partial H}{\partial T} \left(-n_1 \sin \theta + n_2 (e + \cos \theta) \right)$$
$$\frac{\mu}{H} \left(-\frac{\partial n_1}{\partial T} \sin \theta - n_1 \cos \theta \frac{\partial \theta}{\partial T} + \frac{\partial n_2}{\partial T} (e + \cos \theta) + n_2 \frac{\partial e}{\partial T} - \sin \theta n_2 \frac{\partial \theta}{\partial T} \right)$$
(G-18)

where

$$\frac{\partial r \cos \theta}{\partial T} = \frac{1}{1 + e \cos \theta} \left[\frac{\partial a}{\partial T} \cos \theta (1 - e^2) - a \sin \theta \frac{\partial \theta}{\partial T} (1 - e^2) - 2ae \cos \theta \frac{\partial e}{\partial T} \right] -a(1 - e^2) \cos \theta \left[\frac{\partial e}{\partial T \cos \theta - e \sin \theta \frac{\partial \theta}{\partial T}} \right]$$
(G-19)

$$\frac{\partial r \sin \theta}{\partial T} = \frac{1}{1 + e \cos \theta} \left[\frac{\partial a}{\partial T} \sin \theta (1 - e^2) + a \cos \theta \frac{\partial \theta}{\partial T} (1 - e^2) - 2ae \sin \theta \frac{\partial e}{\partial T} \right] -a(1 - e^2) \sin \theta \left[\frac{\partial e}{\partial T \cos \theta - e \sin \theta \frac{\partial \theta}{\partial T}} \right]$$
(G-20)

$$\frac{\partial l_1}{\partial T} = -\sin\Omega\cos\omega\frac{\partial\Omega}{\partial T} - \cos\Omega\sin\omega\frac{\partial\omega}{\partial T} -\cos\Omega\sin\omega\cos i\frac{\partial\Omega}{\partial T} - \sin\Omega\cos\omega\cos i\frac{\partial\omega}{\partial T} + \sin\Omega\sin\omega\sin i\frac{\partial i}{\partial T}$$
(G-21)

$$\frac{\partial l_2}{\partial T} = \sin \Omega \sin \omega \frac{\partial \Omega}{\partial T} - \cos \Omega \cos \omega \frac{\partial \omega}{\partial T} - \cos \Omega \cos \omega \cos i \frac{\partial \Omega}{\partial T} + \sin \Omega \sin \omega \cos i \frac{\partial \omega}{\partial T} + \sin \Omega \cos \omega \sin i \frac{\partial i}{\partial T}$$
(G-22)

$$\frac{\partial m_1}{\partial T} = \cos\Omega\cos\omega\frac{\partial\Omega}{\partial T} - \sin\Omega\sin\omega\frac{\partial\omega}{\partial T} -\sin\Omega\sin\omega\cos i\frac{\partial\Omega}{\partial T} + \cos\Omega\cos\omega\cos i\frac{\partial\omega}{\partial T} - \cos\Omega\sin\omega\cos i\frac{\partial i}{\partial t}$$
(G-23)

$$-\sin\Omega\sin\omega\cos\imath\frac{\partial}{\partial T} + \cos\Omega\cos\omega\cos\imath\frac{\partial}{\partial T} - \cos\Omega\sin\omega\cos\imath\frac{\partial}{\partial T} \quad (G-23)$$
$$\frac{\partial m_2}{\partial T} = -\cos\Omega\sin\omega\frac{\partial\Omega}{\partial T} - \sin\Omega\cos\omega\frac{\partial\omega}{\partial T}$$

$$-\sin\Omega\cos\omega\cos i\frac{\partial\Omega}{\partial T} - \cos\Omega\sin\omega\cos i\frac{\partial\omega}{\partial T} - \cos\Omega\sin\omega\sin i\frac{\partial\omega}{\partial T} - \cos\Omega\cos\omega\sin i\frac{\partial i}{\partial T} \quad (G-24)$$

$$\frac{\partial n_1}{\partial T} = \cos\omega \sin i \frac{\partial \omega}{\partial T} + \sin\omega \cos i \frac{\partial i}{\partial T}$$
(G-25)

$$\frac{\partial n_2}{\partial T} = -\sin\omega\sin i\frac{\partial\omega}{\partial T} + \cos\omega\cos i\frac{\partial i}{\partial T} \tag{G-26}$$

$$\frac{\partial H}{\partial T} = \frac{1}{2\sqrt{\mu a(1-e^2)}} \frac{\partial \left(\mu a(1-e^2)\right)}{\partial T}$$
$$= \frac{1}{2H} \cdot \left[\mu \frac{\partial a}{\partial T}(1-e^2) - 2\mu a e \frac{\partial e}{\partial T}\right]$$
(G-27)

Note that all these derivatives are with respect to the time in Julian centuries units. In order to find them with respect to time in seconds, one can use

$$\frac{\partial x}{\partial t} = \frac{\partial x}{\partial T} \frac{\partial T}{\partial JDE} \frac{\partial JDE}{\partial t}
= \frac{\partial x}{\partial T} \cdot \frac{1}{36525} \cdot \frac{1}{86400}$$
(G-28)

Appendix H

Global Trajectory Optimization Competition

During the author's stay at the University of Colorado, a student team led by dr. Jeffrey S. Parker participated in the seventh edition of the Global Trajectory Optimization Competition (GTOC). Throughout this competition, extensive use has been made of the tool developed for this thesis work. In this appendix, the way the code contributed will be shortly described.

H-1 Problem statement

The global optimization problem that had to be solved for GTOC7 is a multiple-ship mission to main belt asteroids. A mother ship containing three exploration probes is launched from Earth anytime between the 1st of January, 2021 and the 31st of December, 2030. Anytime and anywhere after launch, exploration probes can be released. Upon the release of a probe, each probe has 6 years to go and rendez-vous with asteroids and then rendez-vous again with the mother ship. Exactly 12 years after the launch, the last probe has to be back at the mother ship. For each different asteroid the probe can rendez-vous with and stay with it for 30 days, one point is earned, if the probe makes it back in time to the mother ship. The second objective and tiebreaker is the sum of the remaining probe masses.

The mother ship can be launched anytime between the 1st of January, 2021 and the 31^{st} of December, 2030 with an hyperbolic excess velocity between 0 and 6 km/s in any direction, without a penalty for a higher used hyperbolic excess velocity. Of its 24 tons of mass, 6 is allocated for structure, 6 to the exploration probes and 12 to propellant. This propellant is used for the mother ship's high-thrust nuclear thermal propulsion with an I_{sp} of 900 s.

Each exploration probe weighs 2 tons; 0.8 tons of structure and 1.2 tons of SEP propellant. The propulsion system for the exploration probes achieve a specific impulse of 3000 s. The maximum thrust for the probes is limited to 0.3 N. Based on these numbers, they each have a ΔV capability of about 27 km/s!

H-2 Strategy

Although the speed with which the developed code can produce feasible trajectories is relatively fast, the sheer size of the problem makes it impossible to calculate every possible trajectory between two asteroids: a total window of 22 years, 16256 main-belt asteroids to choose from, countless combinations of asteroid tours, ... Therefore, it was decided to develop a fast method to estimate the flight time between two asteroids. This method could then be used to quickly develop entire asteroid tours. The most promising ones could then be assessed for feasibility using the developed low-thrust code and optimized for final probe mass. To achieve a feasible trajectory, the optimization tool optimized the flight time between the asteroids, the stay time at each asteroid and the trajectory to leave the mother ship and return to it. Finally, for the best found solution, the optimization tool has been run to generate a continuous low-thrust profile that meets the stringent accuracy requirements of GTOC. Each of the different usages of the code will be shortly described in the next sections.

H-3 Validation of single-leg approximation tools

Independently, multiple approximation tools have been written such as linearized low-thrust dynamics approximations, Edelbaum approximations, modified Clohessy-Wiltshire equations of motion, chemical Lambert solution approximations, ... Those different approximations have been traded off based on accuracy and speed. In order to know the accuracy, the written low-thrust optimization tool has been used to create a database of more than 5000 feasible trajectories between asteroids, of which one can be seen in Figure H-1. Based on close approximity analysis performed in parallel by other students, the optimization tool got a combination of two asteroid indexes and a close approximity epoch. Based on this input, the tool found the shortest transfer possible in terms of time of flight. The different approximation tools were then fed with the departure date of the transfer and the asteroid indexes and the estimated time of flight.



Figure H-1: Example of a single asteroid-to-asteroid transfer.
Based on this comparison, the chemical Lambert solution approximations turned out to be the most accurate method. However, it was observed that there was a relationship between the estimated time of flight and the error in ΔV . As such, a curve fit could be developed to adjust the ΔV from Lambert's problem as a function of TOF. Upon tweaking the Lambert's solutions, all asteroid-asteroid transfers with a transfer time smaller than 180 days for the complete time window of 22 years with a 10 day resolution were calculated using GPU parallel computation. This took about 2 days of run-time for approximately 4 trillion transfers. As an example, the transfers between asteroids 13733 and 15938 for a period of 10 years are shown in Figure H-2. Based on the thrust level, the specific impulse, the mass of the probe and the time of flight, one can estimate which ΔV would be feasible for the low-thrust system of the probe. Those feasible points have been plotted using red stars. For this asteroid, several scenario's have been determined feasible using the written low-thrust optimization tool. These points are plotted as cyan circles. The tweaked Lambert's solutions with the feasibility assessment performs very well, at a fraction of the run time of the written low-thrust optimization tool.



Figure H-2: Example of a single asteroid-to-asteroid GLambert grid search.

H-4 Validation of asteroid tours

Using the database of 4 trillion transfers, several search algorithms have been developed to search through and explore the entire design space for potential asteroid tours. Some of the considered strategies are populated greedy searches and the traveling salesman approach. These algorithms needed verification to ensure that they created feasible asteroid tours. Therefore, the low-thrust optimization tool checked the most promising asteroid tours for feasibility. This proved to be an excellent test for the developed software: the dynamic leg addition capability of the tool was put to the test due to the high number of asteroids visited in a single asteroid tour. Tours with up to 14 different asteroid visits have been encountered and successfully optimized. Furthermore, for each leg, the flight and stay time had to be optimized, which was also an excellent test for the TOF optimization capability of the code.



Figure H-3: Trajectories for an asteroid tour visiting 7 asteroids.

As an example, Figures H-3a and H-3b show the effect of the optimization of the flight and stay time. In Figure H-3a, the trajectory is shown for a 7 point asteroid tour. In this scenario, the dates fed from the search algorithm have been kept constant. This leads to an infeasible

trajectory, as there is a discontinuity in the z-coordinate of the match point on the leg between asteroids 3 and 4. However, if the optimizer is allowed to vary the flight and stay times of the asteroid, while keeping the probe release and retrieval dates constant, the trajectory becomes feasible, as can be seen in Figure H-3b. Besides making some infeasible asteroid tours feasible, the optimization of the flight and stay times result in significant propellant savings, compared to feasible, non-time optimized asteroid tours.

H-5 Building entire probe solutions

Up to this point, no considerations with respect to the mother ship or on how to connect different probe tours had been made. In order to save fuel of the mother ship and to reduce the difficulty level of the problem, it was decided to let all three probes start at the same asteroid. As such, the mother ship remained in the same orbit and the three probes also had to return to that orbit.

Using the confidence gained in building asteroid tours, many different three probe tours have been created using global search algorithms. The most promising were fed back to yet another version of the low-thrust optimization tool. This version was slightly adapted to include the probe release and retrieval legs from and to the mother ship. This tool got the deployment and retrieval time and deployment orbit, along with an estimate of the asteroid rendez-vous times and departure dates. The code then tried to generate feasible trajectories for all three probes. For each probe, it would at first try to rendez-vous with all its designated asteroids. However, this resulted almost always in trajectories for which the probe could not return to the mother ship in time. As such, an iterative procedure has been set up where systematically, one of the asteroids was removed from the tour, giving the probe more time to make it back to the mother ship. At that point, the entire tour became feasible. This proces was repeated for all three probes, after which the score of the full three probe solution could be determined. This whole process has been repeated on many different three probe solutions, ultimately resulting in a 28 point solution. For this best 28 point solution, a high accuracy run had to be performed to build a submittable solution file. This solution file had to include a continuous low-thrust profile with at least one output line per day. The required accuracy at each special event such as deployment, asteroid rendez-vous and departure, and mother ship return had to be within strict boundaries. The Euclidean norm of the vector differences had to be below 1000 km and 1 m/s for the position and velocity respectively. These accuracy requirements were checked by the competition's jury. The code developed for this thesis and adapted for the competition met all of these accuracy requirements, which is an additional validation of the accuracy of the written code.

H-6 Conclusion

The developed tool was extensively used throughout this competition in many different setups, indicating the flexibility of the tool. Furthermore, the submitted solution was checked by an external jury, validating the feasibility and accuracy of the developed tool. In the end, our final submitted solution, of which the trajectories can be found in Figure H-4, landed us the 10th place.



Appendix I

Earth-Asteroid-Earth rendez-vous mission

In this chapter, it will be shown how the results for an Earth-Asteroid-Earth rendez-vous mission from the developed code compare to the results from the Evolutionary Mission Trajectory Generator code [Englander et al., 2014], developed by Jacob Englander at NASA's Goddard Space Flight center. This Sims-Flanagan based low-thrust optimization tool uses Monotonic Basin Hopping to circumvent issues with local optima and to avoid the need for an initial guess.

This mission launches from Earth, flies to an asteroid, rendez-vous with it, stays with it for at least 30 days and then flies back to Earth. Much like for the EME and EVME scenario's, the Earth departure and return date are fixed, while the asteroid rendez-vous and departure dates are optimized. The actual mission design parameters will intentionally not be disclosed here, as these are still part of an ungoing investigation.

First of all, an individual point has been investigated. The optimized trajectory for the selfdeveloped code and the trajectory from EMTG are shown in Figures I-1 and I-2 respectively. Even though both methods have been started with different initial guesses, they both converge on almost the same solution. As one can see, they both converge on the same optimal asteroid rendez-vous and departure dates, the departure C_3 is identical, the return hyperbolic excess velocities are also identical, the size and direction of the applied manoeuvres on each segment are also very similar. Furthermore, the launch mass, which has been minimized for these optimizations, only differs by 8 kg, or 0.015 %. These differences are negligibly small and hence, one can conclude that both tools have converged on the same solution.



Figure I-1: Earth-Asteroid-Earth trajectory: own code.



Figure I-2: Earth-Asteroid-Earth trajectory: EMTG.

Besides comparing individual points, also entire launch windows have been compared. An example can be found in Figures I-3 and I-4. Note that due to a plotting bug, the infeasible trajectories in Figure I-4 have been accidentally displayed as dark red and can hence be ignored. One can see that both identified launch windows are nearly identical. They both start at the 25th of November, 2027 and both end at the 6th of June, 2028. Furthermore, they both display a sudden jump in TOF around the 17th of April, 2028. Also, the same band structures where more or less propellant mass is required are visible and also the minimal possible TOF's for each departure date are the same everywhere, as is the downward slope between the 25th of November, 2027 and the 17th of April, 2028.







Figure I-4: Earth-Asteroid-Earth launch window: EMTG.

Part VIII

Bibliography and index

Bibliography

- P.J. Antsaklis and A.N. Michel. A Linear Systems Primer. Birkhauser Boston, 2007.
- M.P.J. Benfield and M.W. Turner. Spacecraft Propulsion System Impacts When Incorporating Advanced Chemical Propulsion System Technologies. 2007.
- L. Casalino and G. Colasurdo. GTOC 7: Multi-spacecraft exploration of the asteroid belt. http://sophia.estec.esa.int/gtoc_portal/?page_id=515, June 2014. - Last accessed on 22 October 2014.
- Codersnotes.com. Very Sleepy. http://www.codersnotes.com/sleepy Last accessed on 25-03-2014.
- J.W. Cornelisse, H.F.R. Schöyer, and K.F. Wakker. Rocket Propulsion and Spaceflight Dynamics. Pitman, 1979.
- S. De Smet, J.S. Parker, J.F.C. Herman, and R. Noomen. Preliminary Design of a Crewed Mars Flyby Mission using Solar Electric Propulsion. In AAS/AIAA Space Flight Mechanics meeting, Santa Fe, New Mexico, 26-30 January, 2014.
- B. Donahue and S. Sigmon. The Space Launch System Capabilities with a New Large Upper Stage. In AIAA SPACE 2013 Conference and Exposition, San Diego, California, 10-12 September, 2013.
- D.H. Ellison, J.A. Englander, and B.A. Conway. Robust Global Optimization of Low-Thrust Multiple-Flyby Trajectories. In *Collection of Technical Papers - AIAA/AAS Astrodynamics* Specialist Conference, Hilton Head, South Carolina, 11-15 August, 2013.
- S. Elvik. Optimization of a low-energy transfer to Mars using Dynamical Systems Theory and Low-Thrust Propulsion. MSc thesis, Delft University of Technology, 2004.
- J.A. Englander, D.H. Ellison, and B.A. Conway. Global Optimization of Low-Thrust, Multiple-Flyby Trajectories at Medium and Medium-High Fidelity. In AAS/AIAA Space Flight Mechanics meeting, Santa Fe, New Mexico, 26-30 January, 2014.

- P. Fortescue, J. Stark, and G. Swinerd, editors. Spacecraft Systems Engineering. Wiley, 2003.
- R. Frisbee. Advanced space propulsion for the XXI century. *Journal of Propulsion and Power*, 19(6):129–154, 2003.
- P.E. Gill. Users Guide for SNOPT Version 7: Software for Large-Scale Nonlinear Programming. Technical report, University of California, San Diego, 2008.
- P.E. Gill, W. Murray, and M.H. Wright. *Practical Optimization*. Elsevier Academic Press, 1981.
- J.F.C. Herman. Preliminary Mission Analysis for a Far-Side, Highly Inclined Solar Observatory using Electric Propulsion. MSc thesis, Delft University of Technology, 2012.
- W. Hock and K. Schittkowski. Test Examples for Nonlinear Programming Codes, volume 187 of Lecture Notes in Economics and Mathematical Systems. Berlin-Heidelberg-New York: Springer-Verlag, 1981.
- Inspiration Mars Foundation. Architecture study report summary. Technical report, Inspiration Mars Foundation, 2013. URL http://www.inspirationmars.org/IM% 20Architecture%20Study%20Report%20Summary.pdf. Last accessed on 21-04-2014.
- D.T. Jacobson, D.H. Manzellam, and J.W. John. The Performance and Wear Characterization of a High-Power High-Isp NASA Hall Thruster. In *Collection of Technical Papers 41st* AIAA/ASME/SAE/ASEE Joint propulsion conference and exhibit, Tucson, Arizona, 11-13 July, 2005.
- V. Jain and B.K. Lande. Computation of the State Transition Matrix for General Linear Time-Varying Systems. International Journal of Engineering Research & Technology, 1(6), August 2012.
- Jet Propulsion Laboratory MRO. Mars Reconaissance Orbiter Propulsion. http://mars.jpl.nasa.gov/mro/mission/spacecraft/parts/propulsion/. Last accessed on 28-09-2014.
- D.J. Korsmeyer, R.R. Landis, R.G. Merrill, D.D. Mazanek, R.D. Falck, and R.B. Adams. A Flexible Path for Human and Robotic Space Exploration. In *SpaceOps 2010 Conference*, Huntsville, Montana, 25-30 April 2010.
- J.H. Lambert. Observationes Variae in Mathesin Puram. Acta Helveticae Physico-Mathematico-Anatomico-Botanico-Medica, pages 128–168, 1758.
- D. Landau and N. Strange. Human Exploration of Near-Earth Asteroids via Solar Electric Propulsion. In AAS/AIAA Space Flight Mechanics Meeting Proceedings, New Orleans, Lousiana, 13-17 February, 2011.
- liveScience. Why we're obsessed with Mars. http://www.livescience.com/ 22119-mars-cultural-fascination.html, August 2012. - Last accessed on 10 March 2014.
- J.H. Meeus. Astronomical Algorithms. Willmann-Bell, Incorporated, 1991.

- M. Meyer, L. Johnson, B. Palaszewksi, D. Goebel, H. White, and D. Coote. In-Space Propulsion Systems Roadmap - Technology Area 2. Technical report, NASA, April 2012. Available at http://www.nasa.gov/sites/default/files/501329main_TA02-ID_ rev3-NRC-wTASR.pdf - Last accessed on 25-03-2014.
- O. Montenbruck. Numerical integration methods for orbital motion. Celestial Mechanics and Dynamical Astronomy, 53:59–69, 1992.
- C. A. Ocampo. COPERNICUS: A trajectory design and optimization system. In First International Conference on Libration Points and Libration Point Missions, Parador d'Aiguablava, Girona, Spain, 10 - 14 June 2002. Institut d'Estudis Espacials de Catalunya.
- D.Y. Oh, J.S. Snydert, D.M. Goebel, R.R. Hofer, D.F. Landau, and T.M. Randolph. Solar Electric Propulsion for Discovery Class Missions. In 33rd International Electric Propulsion Conference, Washington D.C., 6-10 October, 2013.
- M.R. Patel, J.M. Longuski, and J.A. Sims. Mars Free Return Trajectories. *Journal of Space-craft and Rockets*, 35(3):350–354, May-June 1998.
- T. Polsgrove, R. Hopkins, L. Kos, and T. Crane. Comparison of Performance Predictions for New Low-Thrust Trajectory Tools. In AIAA/AAS Astrodynamics Specialist Conference and Exhibit, Keystone, Colorado, 21-24 August, 2006.
- P.J. Prince and J.R. Dormand. High order embedded Runge-Kutta formulae. Journal of Computational and Applied Mathematics, 7:67 75, 1981.
- M.D. Rayman and S.N. Williams. Design of the First Interplanetary Solar Electric Propulsion Mission. Journal of Spacecraft and Rockets, 39:589–595, 2002.
- M.D. Rayman, P.A. Chadbourne, J.S. Culwell, and S.N. Williams. Mission design for Deep Space 1: A low-thrust technology validation mission. Acta Astronautica, 45:381 – 388, 1999.
- M.D. Rayman, T.C. Fraschetti, C.A. Raymond, and C.T. Russell. Dawn: A mission in development for exploration of main belt asteroids Vesta and Ceres. *Acta Astronautica*, 58: 605–616, June 2006.
- C. Sauer. VARITOP/SEPTOP/SAIL users manual. Technical report, Jet Propulsion Laboratory, 2003.
- H. Schaub and J.L. Junkins. *Analytical Mechanics of Space Systems*. AIAA Education Series, Reston, Virginia, 2nd edition, October 2009.
- G. Schiaparelli. La vita sul pianeta Marte. Natura ed Arte, 2:83–95, June 1 1895.
- J.A. Sims. Jupiter Icy Moons Orbiter Mission Design Overview. Technical report, Jet Propulsion Laboratory - NASA, 2006.
- J.A. Sims, P.A. Finlayson, E.A. Rinderle, M.A. Vavrina, and T.D. Kowalkowski. Implementation of a low-thrust trajectory optimization algorithm for preliminary design. Technical report, JPL, 2006.

- D.A. Tito, G. Anderson, J.P. Carrico, J. Clark, B. Finger, G.A. Lantz, M. E. Loucks, T. Mac-Callum, J. Poynter, T.H. Squire, and S.P. Worden. Feasibility analysis for a manned Mars free-return mission in 2018. In *Aerospace Conference*, 2013 IEEE, Big Sky, Montana, 2-9 March, 2013.
- D.A. Vallado. Covariance transformations for satellite flight dynamics operations. In AAS/A-IAA Astrodynamics Specialist Conference, Big Sky, Montana, 3-7 August, 2003.
- K.F. Wakker. Astrodynamics-II Lecture Notes AE4-847II. TU Delft, 2010.
- A. Wächter and L.T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1):25–57, 2006.
- J.R. Wertz. Orbit and Constellation Design and Management. Microcosm Press, second edition, 2009.
- G.J. Whiffen. An investigation of a Jupiter Galilean Moon Orbiter trajectory. Technical report, JPL, 2003.
- G.J. Whiffen. Mystic: Implementation of the Static Dynamic Optimal Control Algorithm for High-Fidelity, Low-Thrust Trajectory Design. In Collection of Technical Papers -AIAA/AAS Astrodynamics Specialist Conference and Exhibit, Keystone, Colorado, 21-24 August 2006.

All internet sites were accessed between 25 March 2014 and 14 October 2014, unless stated otherwise.

Index

Along-track direction, 141

Bottleneck, 53

Constraint vector, 19 Control input vector, 22 Control node, 17 Control node type, 93 Cost function, 27 Cross-track direction, 141

Dawn, 8 Deep Space 1, 7 Departure node constraints, 29 Discretized thrust profile, 17 DOPRI8, 23 Duty cycle, 38

Electromagnetic SEP, 10 Electrostatic SEP, 10 Electrothermal SEP, 10 Ephemeris constraints, 77 Exhaust velocity, 11

Feedforward matrix, 22 Feedthrough matrix, 22 Flexible path scenario, v, 5 Flyby constraints, 29 Forward finite difference method, 79

In-track direction, 141 Input matrix, 22 Inspiration Mars, 6 Integration stage, 23

Jacobian matrix, 19

Jet efficiency, 11 Jet power, 11 JIMO, 8

Lambert's function, 64 Leg, 17 Local optimum, 102

MALTO, 15 Mass-to-power ratio SEP system, 11 Match point, 17 Match point constraints, 27 Maximized final mass representation, 89 Meeus' polynomials, 37 Mesh points, 23 Minimized launch mass representation, 83 Multi-start method, 103 Mystic, 8, 15

Normal direction, 141 Numerical integrator, 23

Omega function, 64 Output matrix, 22

Power conversion efficiency, 11 Product logarithm, 64 Profiling, 53

Re-entry constraint, 30 RKF8(7)-13M, 23

Segment, 17 SEPTOP, 7, 15 Sequential quadratic programming, 19 Sims-Flanagan method, 17 Single-step method, 23 SLS/iCPS 1xRL10B2, 39 SLS/LUS 4xRL10C2, 39 Sparsity, 72 Specific impulse, 11 State matrix, 22 State Transition Matrix, 22 State vector, 18, 22 Step size, 23 System matrix, 22 System output vector, 22 Tangential direction, 141 Throttled Sims-Flanagan transcription, 63 Thrust, 10 Thrust constraint, 28 TNC: Tangential, Normal and Cross-track direction, 19 Two-body force model, 25

VARITOP, 15