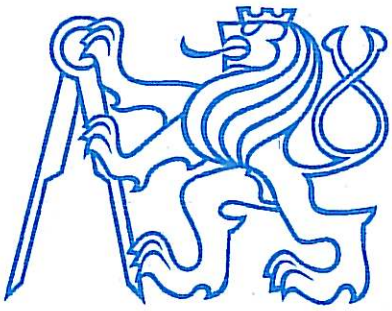

Czech Technical University
Faculty of Electrical Engineering
Department of Control Engineering

Luleå University of Technology
Kiruna Space Campus
Department of Space Science

Erasmus Mundus Master Course SpaceMaster

Input Shaping for Control of Satellites and Formation of Satellites

by

Rafael Rotter Meda

Submitted to the Department of Control Engineering, at the Czech Technical University, in partial fulfilment of the requirements for the degree of

Master of Science in Electrical Engineering and Informatics

Submitted to the Department of Space Science, at the Luleå Technical University, in partial fulfilment of the requirements for the degree of

Master of Science in Space Technology

Prague, May 2007

Abstract

Space research is currently in a trend of using smaller satellites in cooperation to achieve the same results of larger, and more expensive, single satellites. Not only that, satellites in formation enable new possibilities, like taking measurements from plasma from different points at the same time, thus creating a more complete 3D profile. Though formation flying has several advantages in terms of reducing costs and production time, it increases the requirements in control strategies not only for the spacecraft alone but also to coordinate their relative formation structure. Those less expensive and faster deployable satellites are usually less rigid than their larger counterparts; therefore they pose new challenges in spacecraft and formation flight control. This thesis investigates the effects of flexible satellites in the relative dynamics of single and formation flying satellite. It proposes a way of planning an optimal, minimum fuel, trajectory with linear programming, and design an LQR controller for the satellites to follow this trajectory. The system response to input shaped and unshaped reference signal is compared for the flexible satellite with linear and nonlinear actuators.

České vysoké učení technické v Praze
FAKULTA ELEKTROTECHNICKÁ
Katedra řídicí techniky
121 35 Praha 2, Karlovo nám.13
Tel.: 224922372, Fax: 224918646

Acknowledgements

Before risking to forget anyone, I wish to thank everybody that has participated, directly or indirectly, in those interesting years of European experience. First, the European Commission for making this possible through the Erasmus Mundus grant and also funding the SpaceMaster programme. Sven Molin, for organising the programme, and for his patience and trust when caring for all us students – not only for our academic growth but also for our well-being when going across three different countries and cultures.

Though time was short, I received some valuable help, so I would like to thank my local supervisor in Prague, Zdeněk Hurák, for his worthy advice that didn't let me drift away from the main goals in this work. Would also like to thank my supervisor at LTU, in Sweden, Thomas Gustafsson. Thanks to my friends at Way2, back in Brazil, for providing a productive environment on my last couple of weeks allowing me to finish and review this thesis.

Special thanks to my family, particularly my parents, for raising me and giving me strength and support to pursue my goals, whichever they might be.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In the early era of space research, the first satellites were considerably stiff and could be regarded as rigid bodies [1]. More recently, by using lighter and more flexible materials, it became necessary to investigate this increasing flexibility, and consequential vibration, to obtain a more faithful model of the satellite. By mitigating those vibrations, it is possible to increase the precision of the satellite's orbit and attitude, and even increase the available time for on-board instruments. There are several approaches to minimise structural vibrations [2]:

- installing dedicated hardware to isolate or dissipate the vibration;
- installing sensors and actuators to create a classic feedback control technique to attenuate the vibration;
- schedule vibration sensitive tasks to periods after all vibration has been dissipated.

Any combination of the above approaches will lead to losses in either fuel, due to the increased satellite's mass; or usage time, when vibration takes too long to settle before it is under acceptable level for certain tasks – e.g. using optical instrumentation like space telescopes, which require a perfectly still subject-to-lens (pointing) attitude.

The two most common and stronger sources of vibration are flexible appendages, like the solar panels in Figure 1.1, and liquid sloshing [1]. Figure 1.1 is an example of how flexible appendages could bend when the rockets are on.

1.1 Motivation

The original motivation for this work was [3]. It was when I first came across the problem of flexibility in spacecrafts and saw it together with the

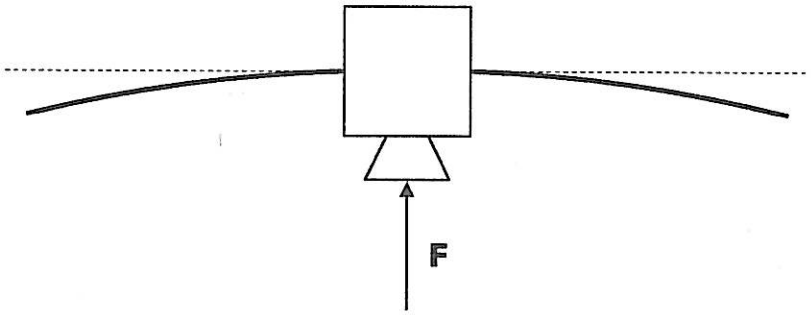

Figure 1.1: Satellite with flexible solar panels in a translational manoeuvre.

interesting topic of formation flying. However, Biediger [3] makes some over-simplified assumptions and provides some exaggerated simulations that lead to deflexions on the flexible appendage (the spacecraft used is asymmetric) of over 100°. It was the idea of my supervisor, Prof. Hurák, to try to reproduce those results and then extend that work where it was lacking.

Another major part of this work, the trajectory planning with Linear Programming, came during the initial researching process on formation flying. Tillerson made a good document [4] on the topic, and my development on it were motivated from his results.

1.2 Thesis Overview

Due to organisational matters and other issues, the total time for this thesis, between the topic definition and the delivery of this document, took no longer than three months. For this reason, many details concerning the topics addressed here had to be overlooked. I chose a fast paced approach, dealing with a few subjects that are related to formation flying, input shaping and flexible spacecrafts. The chapters are relatively short, and I believe that addressing any of them in-depth would be a complete project on its own.

Chapter 1: this introduction. Provides a brief explanation on the motivation behind this work and the organisation of this report.

Chapter 2: flexible spacecraft dynamics. Explains some concepts in orbital and attitude dynamics and then shows the modelling of the flexible spacecraft.

Chapter 3: input shaping with posicast control. Explains the theory of posicast control – the first developed input shaping.

Chapter 5: trajectory planning with Linear Programming and LQR controller design. Presets a method to determine optimal, minimum fuel,

trajectories for spacecraft manoeuvres and later develops an LQR controller to make the spacecraft track this predetermined trajectory.

Chapter 6: merges the theories presented so far. Presents the complete simulations (with flexible spacecrafts in a formation cluster) and compares the performance of the satellites using and not using input shaping.

Chapter 7: concludes this work. Briefly debates the topics covered in this thesis and proposes an extension of this work.

The more relevant m-code and Simulink© models used in the simulations are available at *http://www.kosmopolita.org/Masters/*.

Chapter 2

Spacecraft Dynamics

The spacecraft dynamics is divided in:

Orbit Dynamics: the movement of the spacecraft around a much larger and heavier body, described by the three Keplerian laws and ;

Attitude Dynamics: the rotations about the three spacecraft axis: *roll, pitch and yaw.*

The effects of vibration is more pronounced in the latter, but since this work focuses on relative orbit dynamics and formation flight, I will present the modelling of the flexible spacecraft relevant to translational (orbital) dynamics.

2.1 Spacecraft Model

Before any simulation can take place, we first need a model of the non-rigid satellite. The main reasons for using a model, like testing control strategies without the high costs of working with the actual satellites, or even prototypes, should be quite evident. Some other less pronounced reasons are worth pointing [5]:

- every good regulator of a system must include, explicitly or implicitly, a model of that system, i.e. success in the regulation of the system implies that a sufficiently similar model must first be built;
- model-based control is superior to non-model-based control.

Several techniques can be used to obtain the equations governing the dynamics of the flexible satellite. This can be a very difficult task to accomplish, but some methods allow to express this motion very efficiently and yet pursue the goal of obtaining a simplified model that adequately represents the real structure. Lagrange-Euler's is a well structure method to model dynamic systems.

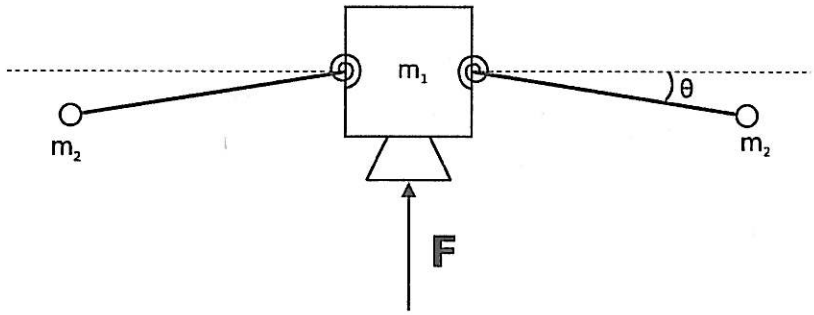

Figure 2.1: Simplified satellite model with flexible appendages.

2.1.1 Lagrange-Euler Formulation

Lagrange-Euler formulation is based on the energy flow in a system. It subtracts the total potential energy $\mathcal{P}$ from the total kinetic energy $\mathcal{K}_e$ in the system, the *Lagrangian* function

$$\mathcal{L} = \mathcal{K}_e - \mathcal{P}. \tag{2.1}$$

From this we can derive the forces and torques ($\mathcal{T}$) by using the *Lagrange equation*

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}}\right) - \frac{\partial \mathcal{L}}{\partial \mathbf{q}} = \mathcal{T}, \tag{2.2}$$

where $\mathbf{q} = [q_1\ q_2\ ...\ q_n]^T$ is the set of generalised coordinates for a system with n degrees of freedom.

Consider now the model in Figure 2.1, a discretised version of the model in Figure 1.1, where a force $\mathbf{F}$, from the thrusters, is applied to change the translational velocity of the satellite. The development here is made for a pair of symmetric appendages – commonly the case for solar panels. In this model, both panels are modelled as cantilevers with a discrete mass m_2 attached with a distance l from the satellite's body (mass m_1).

The point masses m_2 are interconnected by torsion springs of torsion coefficient k, which represents the rigidity of the solar panels, expressed in terms of its natural frequency as [6]

$$\omega_n = \sqrt{\frac{k}{m_2}}. \tag{2.3}$$

The dissipation of this vibration is due to internal frictions of the appendages, when they are moving, and can be related to the damping coefficient ζ as [6]

$$\zeta = \frac{k_d}{2}\sqrt{\frac{1}{km_2}}, \tag{2.4}$$

where k_d is the mechanical dissipative constant. According to [1], all structural materials have some inherent damping, so the open-loop resonant poles are stable, not just critically stable.

For the satellite in Figure 2.1the total kinetic energy is

$$\begin{aligned} \mathcal{K}_e &= \mathcal{K}_{e1} + 2\mathcal{K}_{e2} \\ &= \frac{1}{2} m_1 \dot{x}_1^2 + m_2 \dot{x}_2^2. \end{aligned} \tag{2.5}$$

But the appendage constrains m_2 to rotate around its fixation point, so

$$x_2 = x_1 - l \sin\theta, \tag{2.6a}$$

$$\dot{x}_2 = \dot{x}_1 - \dot{\theta} l \cos\theta, \tag{2.6b}$$

and (2.5) can be rewritten as

$$\mathcal{K}_e = \frac{1}{2} m_1 \dot{x}_1^2 + m_2 (\dot{x}_1 - \dot{\theta} l \cos\theta)^2. \tag{2.7}$$

For this approximation, the kinetic energy on the axis perpendicular to $\mathbf{F}$ and parallel to the panels is nil, what would happen in reality if the solar panels are perfectly symmetric.

The total potential energy will be solely the energy stored in the torsion spring with a torsion coefficient k, as

$$\begin{aligned} \mathcal{P} = 2\mathcal{P}_2 &= 2\left(\frac{1}{2} k\theta^2\right) \\ &= k\theta^2. \end{aligned} \tag{2.8}$$

The damping force is proportional to the deflection rate:

$$D = k_d \dot{\theta}. \tag{2.9}$$

The generalised coordinates being selected as (x_1, θ), so the Lagrange equations for this system are

$$\begin{aligned} \frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{x}_1}\right) - \frac{\partial \mathcal{L}}{\partial x_1} &= F, \\ \frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\theta}}\right) - \frac{\partial \mathcal{L}}{\partial \theta} &= -2D \end{aligned} \tag{2.10}$$

with the Lagrangian

$$\mathcal{L} = \frac{1}{2} m_1 \dot{x}_1^2 + m_2 (\dot{x}_1 - \dot{\theta} l \cos\theta)^2 - k\theta^2. \tag{2.11}$$

For small deformations on the appendages:

$$\sin\theta \approx \theta, \tag{2.12a}$$

$$\cos\theta \approx 1, \tag{2.12b}$$

and (2.11) can be simplified as

$$\mathcal{L} = \frac{1}{2}m_1\dot{x}_1^2 + m_2(\dot{x}_1 - \dot{\theta}l)^2 - k\theta^2. \tag{2.13}$$

Thus,

$$\frac{\partial \mathcal{L}}{\partial \dot{x}_1} = m_1\dot{x}_1 + 2m_2\dot{x}_1 - 2m_2\dot{\theta}l, \tag{2.14a}$$

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{x}_1}\right) = m_1\ddot{x}_1 + 2m_2\ddot{x}_1 - 2m_2\ddot{\theta}l, \tag{2.14b}$$

$$\frac{\partial \mathcal{L}}{\partial x_1} = 0, \tag{2.14c}$$

$$\frac{\partial \mathcal{L}}{\partial \dot{\theta}} = -2m_2\dot{x}_1 l + 2m_2\dot{\theta}l^2, \tag{2.14d}$$

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\theta}}\right) = -2m_2\ddot{x}_1 l + 2m_2\ddot{\theta}l^2, \tag{2.14e}$$

$$\frac{\partial \mathcal{L}}{\partial \theta} = -2k\theta, \tag{2.14f}$$

substituting in (2.10) yields

$$\begin{aligned} \ddot{x}_1(m_1 + 2m_2) - 2m_2\dot{\theta}l &= F, \\ -2m_2\ddot{x}_1 l + 2m_2\ddot{\theta}l^2 + 2k_d\dot{\theta} + 2k\theta &= 0 \end{aligned} \tag{2.15}$$

Finally, solving (2.15) for $\ddot{x}$ and $\ddot{\theta}$ yields the equations of motion for this system:

$$\ddot{x} = \frac{1}{m_1}\left(F + \frac{2k_d\dot{\theta} + 2k\theta}{l}\right) \tag{2.16}$$

and

$$\ddot{\theta} = -\frac{F}{lm_1} - (k_d\dot{\theta} + k\theta)\left(\frac{2}{m_1 l^2} + \frac{1}{m_2 l^2}\right) \tag{2.17}$$

with x the position of the satellite's body in the same direction of the propulsion force $\mathbf{F}$, and θ the deflection angle of the solar panels.

2.2 Propulsion System

Though some degree of approximations does not invalidate the spacecraft model it is important to have a general knowledge of the workings of a propulsion system when designing a control strategy that uses it as an actuator. The force of an individual rocket engine comes from the propellant, which is ejected at high speeds. This force can be calculated from the velocity of the expelled mass and the rate of expelled mass in the relation

$$F = V_e\frac{dm}{dt} + A_e(P_e - P_a) = V_{ef}\frac{dm}{dt}, \tag{2.18}$$

T_s	15 ms
T_{sd}	10 ms
MIB	30-40 mN-s

Table 2.1: Typical characteristics for a 10-N bipropellant thruster

where V_e is the exhaust velocity, A_e is the area of the nozzle, P_e and P_a are exhaust and ambient pressures (respectively), V_{ef} is the effective exhaust velocity and dm/dt is the rate of propellant mass being expelled [6].

The propulsion system can change both translatory and angular velocities of a spacecraft. Its principle is the same when producing force or torque to the spacecraft, but generally different actuators are used for each situation since different characteristics are required for each task. When changing the linear velocity of the satellite larger periods of propulsion are needed than when changing the angular velocity in attitude control manoeuvres. Also, proportional gas jets are much more difficult to build than on-off thrust engines [1], and the latter is more frequently used. This nonlinearity must be kept in mind when developing a control strategy for the spacecraft that uses on-off rockets.

From the control perspective, the actuator characteristics of major concern are:

- the thrust level F;
- the specific impulse I_{sp};
- the minimum impulse bit MIB: the minimum amount of impulse it can deliver in an on-off cycle;
- the maximum number of activations;
- the starting time T_s: time it takes the thruster to reach 90% of its nominal force;
- the shutdown time T_{sd}: time it takes the thruster to have zero output after an *off* command; and
- the total impulse $\int_{t=0}^{\infty} F dt$.

The total impulse often dictates the life expectancy of a satellite, so saving fuel on the manoeuvres is of paramount importance. The other characteristics are treated as nonlinear constraints in the simulations. As an example, Table 2.1 [6] lists typical values for a 10 N bipropellant thruster.

m_1	50 kg
m_2	1.5 kg
l	1 m
ω_n	5 rad/s
ζ	0.003

Table 2.2: Spacecraft parameters

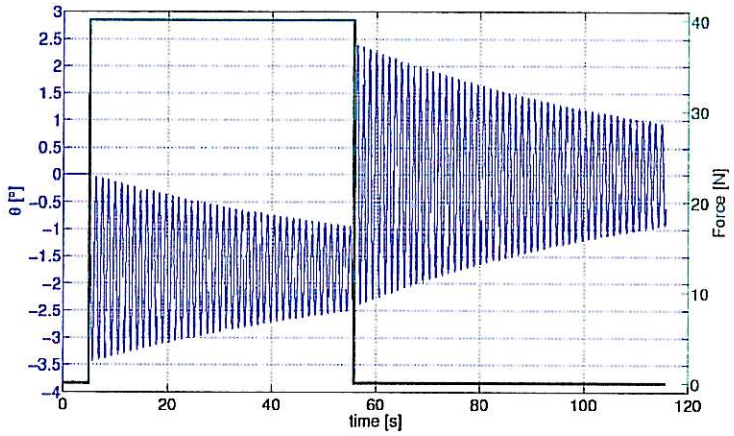

Figure 2.2: Panels deflection during a translation manoeuvre.

2.3 Vibration Effects on Orbital Manoeuvres

Table 2.2 lists the spacecraft parameters considered throughout this work. The selected ω_n and ζ values are typical for solar panels.

Figure 2.2 shows the deflection of the panels when a thrust of 40 N is applied to the satellite for 50 s. It is visible from the graph that if the thruster was on long enough the deflection of the panels would converge to a small negative angle. Depending on the moment the thruster is shut off the oscillations can be even more aggravated. Since ζ is small, the panels take a long time before they stop oscillating.

Figure 2.3 shows the difference between the desired position (considering a point of mass system, $m_1 + 2m_2$) and the position of the flexible satellite. This difference is relatively small, sitting below 2.5 mm. Though this can be disregarded in most situations, for more precision critical applications, like the Laser Interferometer Space Antenna (LISA) satellite formation, where the relative positions between satellites should have accuracies under 1 mm, those oscillations can considerably reduce the available time for this satellite cluster to perform its data measurements.

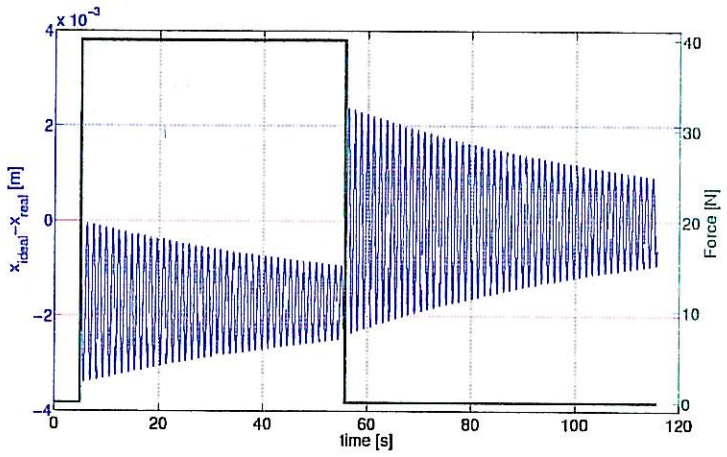

Figure 2.3: Relative position between a fictional rigid body model and the flexible satellite.

2.4 Command Modulation

Simply by modulating the thruster's command it is possible to drastically reduce the effects of the vibration mode in the appendages. This command modulation can be treated as an input shaper – in the broad sense, anything that modifies (shapes) a signal can be regarded as an input shaper, though a more specific kind will be treated from Chapter 3 and onwards.

Figure 2.4 is a simple example of how the vibration can be attenuated. It compares shaped and unshaped responses of the rate of angular deviation $\dot{\theta}$. The values for the modulated pulses width were found numerically.

The drawbacks of such approach can be seen in Figure 2.5. Input shaping usually delays the system response – in this case, a change in the satellite's velocity. The length of this delay is exactly the addition of the initial and the final pulse, a total of 406 ms, a minor drawback for most applications, specially when regarding the sooner availability of the satellite for more precise operations since its position has virtually no oscillation after a manoeuvre using shaped thrusts. A closer inspection of this is presented in Figure 2.6, that shows the difference between the unshaped and shaped velocity's output of the same orbital manoeuvre. It is worth noting that there is no increase in fuel consumption for equivalent manoeuvres with and without command shaping.

Though the values used for this simulation are theoretical, they fall in the range of typical values for real satellites. When considering the feasibility of such approach, mainly the thruster's limitations should be considered, so the question is whether it can perform pulses with the required width. In a situation where four of the thrusters presented in Table 2.1 could be

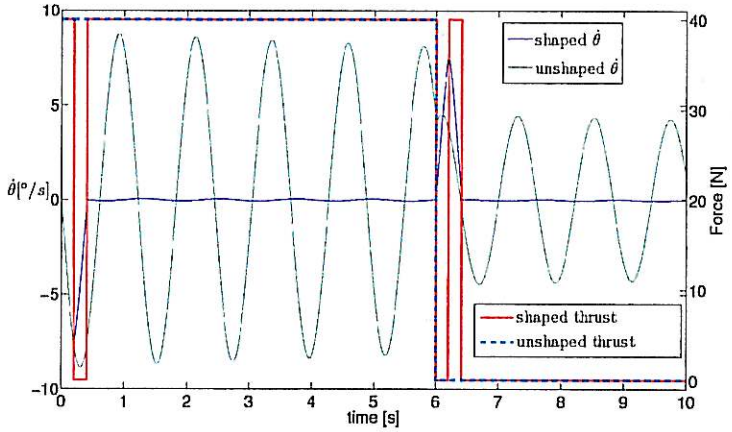

Figure 2.4: Angular deflection rate of the solar panels.

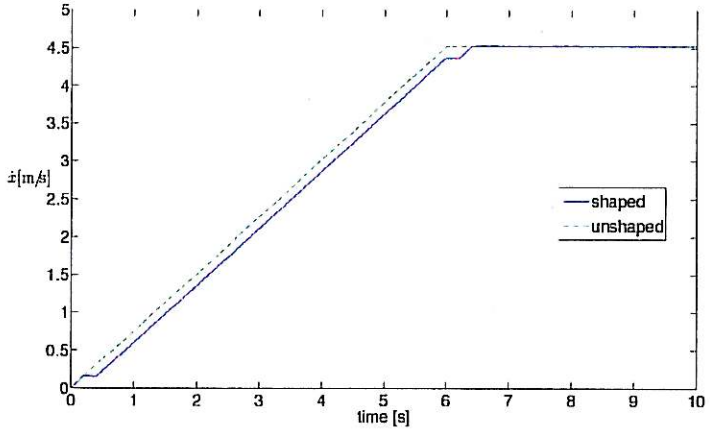

Figure 2.5: Velocity change after an orbit manoeuvre.

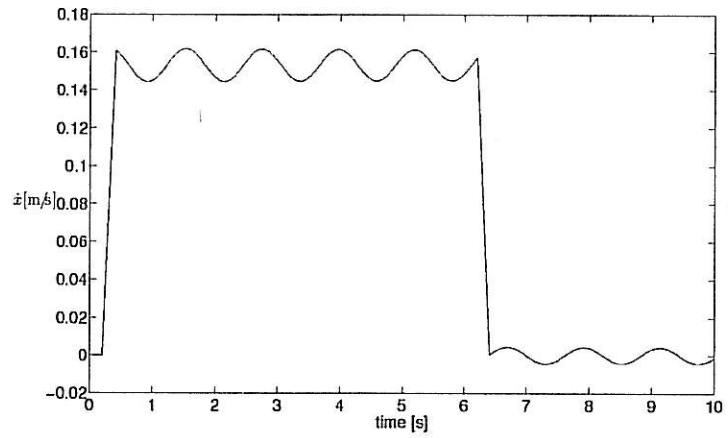

Figure 2.6: Difference between velocities during an unshaped and a shaped orbit manoeuvre.

used (one in each corner of one of the satellite's side, a somewhat common configuration that can also use the same thrusters for attitude manoeuvres), for a total of 40 N as in the simulations performed here, the limitations of the thruster would not pose any major problems for the shaped commands, where the smallest pulse was of 203 ms.

2.5 Optimal Control

It is quite natural to assume that an area that implements high-end technologies, such as space research, would not rely on something as simple as the open-loop design presented so far. Among several other possibilities, it is quite common to use optimal control for spacecraft manoeuvres.

In optimal control the goal is to minimise some cost function

$$J(t_0) = \phi(x(T), T) + \int_{t_0}^{T} L(x(t), u(t), t)dt \tag{2.19}$$

in the interval $[t_0, T]$. Since the problem consists in also minimising the states a redefinition of the state variables is usually required such that

$$\psi(x(T), T) = 0. \tag{2.20}$$

Therefore, the optimal control problem is to find some optimal input $u^*(t)$ that minimises (2.19).

2.5.1 Minimum-time

This section deals briefly with minimum-time manoeuvres, as an introduction to the more usual minimum-fuel optimal control problem. As an exam-

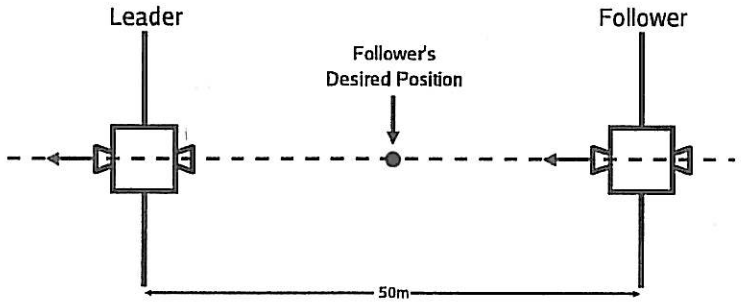

Figure 2.7: Same-orbit leader-follower trajectory.

ple, consider a satellite that has to position itself in a reference to another spacecraft. We wish to keep both satellites apart in a fixed distance with the same speed. In a leader-follower configuration, the follower would be solely responsible for this task, performing all the control effort. In this example, a satellite approaching another satellite is required to maintain a 10 m distance from the leader, and the control set to start when they are 50 m apart, like illustrated in Figure 2.7. For now, I will disregard the movements in the other axes. The complete, 3-D relative dynamics, is presented in Chapter 4.

Using the leader as the reference to define the control variable, (2.20) can be written as

$$\psi(x(T),T) = \begin{bmatrix} x_l(T) - x_f(T) - 20 \\ v_l(T) - v_f(T) \end{bmatrix} = 0. \tag{2.21}$$

Since the reference is the leader spacecraft, its velocity v_l and position x_l are, obviously, zero. So it is a question of bringing the follower's velocity v_f to zero, and position it 20 m behind the leader, when $x_f + 10$ would be also zero.

In minimum-time problems, using an on-off actuator requires a *bang-bang* control – when the controller switches between its extremes, always working with the actuators at its maximum output, or off. Though this makes the problem more challenging, it is more realistic and also when vibration is more accentuated due to the step-shaped input, since it excites more the vibrational modes in the spacecraft. If possible to use proportional control, the vibrations could be attenuated in a variety of ways. This, of course, would not necessarily produce minimum time and/or fuel consumption optimal solutions. In minimum time, the cost function (2.19) is reduced to

$$J(t_0) = \int_{t_0}^{T} dt, \tag{2.22}$$

with T not fixed (it should be as small as possible).

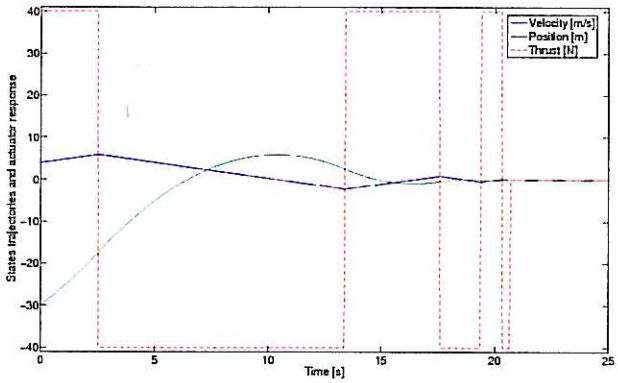

Figure 2.8: States trajectories in minimum time bang-bang unshaped control.

Figure 2.8 presents the solution for this problem with

$$x_0 = \begin{bmatrix} -30 \\ 4 \end{bmatrix},$$

which means that the spacecraft is 4 m/s faster than the leader and 30 m distant from reaching the desired position. The controller then brings the spacecraft to the desired position, 20 m behind the leader, and its speed to the same as the leader's in the shortest possible time. The actuator's response sign corresponds to which thruster is on: 40 N for when the thrust is on the same direction of the leader's tangential velocity (considering its orbit around the Earth); -40 N when the thrust is against its orbital velocity.

The effects of input shaping are clearly observed for this scenario. In fact, the results of the same manoeuvre with shaped input, in Figure 2.9, show that the delay caused by input shaping makes it impractical for this kind of optimal solution. A more sophisticated approach, even for a time/fuel optimal control, would change the control profile of a *bang-off-bang* of a given number of switches to another profile with more switches. This is quickly addressed by Singh and Singhose in [7] and it is a nontrivial solution. In fact, the name optimal is somewhat misleading – the solution is optimal for the weights selected for (2.19) and for the selected profile with its fixed number of switches. Here is presented a simple modulation just to illustrate the effects of the delay caused by the shaper.

Velocity, for this kind of reference change, will always settle no matter what the delay from input shaping is, only the settling time will vary. Unfortunately, positioning does not share the same fate. As for the oscillations, a closer look in the system's response (Figure 2.10) indicates significant levels for both speed and position oscillations. A comparison between both shaped and unshaped responses for the solar panels deflection (Figure 2.11)

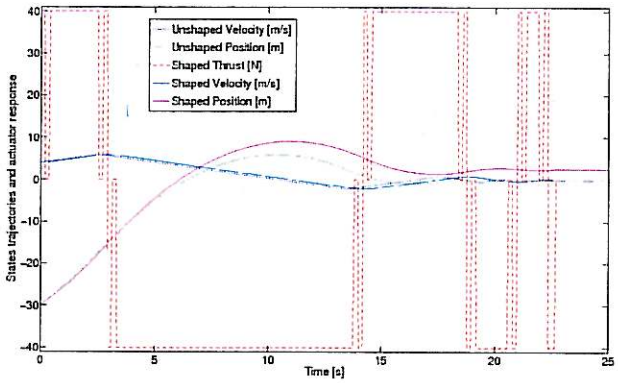

Figure 2.9: State trajectories in minimum time bang-bang shaped control.

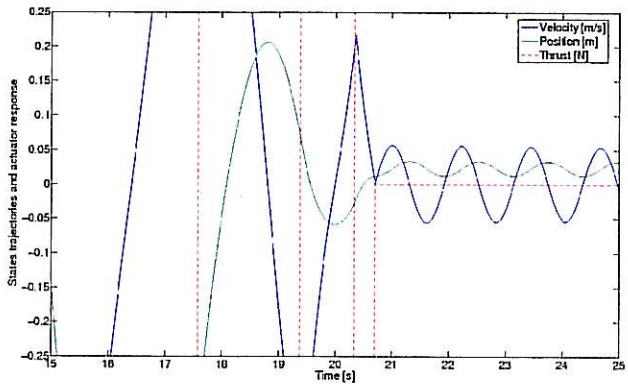

Figure 2.10: Closer inspection of the states trajectories on a bang-bang unshaped control.

validates the results obtained previously – the input shaper manages to keep the vibration in acceptable values throughout the whole manoeuvre. But another drawback from this input shaper becomes obvious: it cannot modify command pulses that are shorter than twice the shaper's pulse.

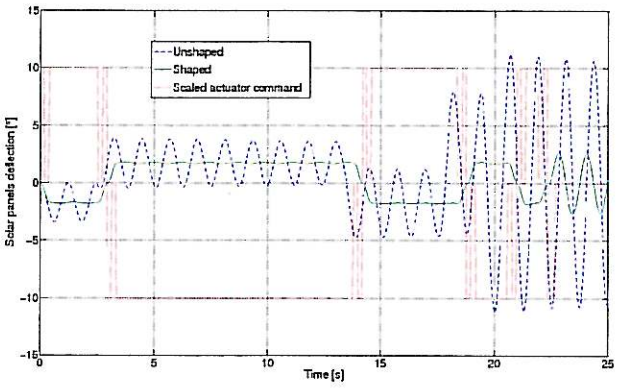

Figure 2.11: Solar panels deflection in both shaped and unshaped minimum time control.

Chapter 3

Input Shaping

Simply put, input shaping consists in converting a reference signal of a system (an actuator or a plant), like in the generalised structure presented in Figure 3.1. A pre-filter can be regarded as an input shaper, for instance.

Historically, input shaping dates from the late 1950s, with Smith's *posicast control* [8, 9]. Its limitations, high sensitivity to modelling errors, were soon discovered and the field didn't become very active until the article *Preshaping Command Inputs to Reduce System Vibration* by Singer and Seering was published in 1990[1] [11].

3.1 Posicast Control

Posicast control splits the reference signal into two parts. So, for a step reference, the single step is replaced by two steps that added together equal the original step, like the example in Figure 3.1. The size of the steps, and the delay before introducing the second step are derived from the system dynamics. The theory is quite simple and compact. Consider the second order system in the general polynomial form

$$H(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}. \tag{3.1}$$

[1]This article is referenced as the original work in robust shaping in all literature I came across. However, they had already published about this topic in the paper [10], from 1988.

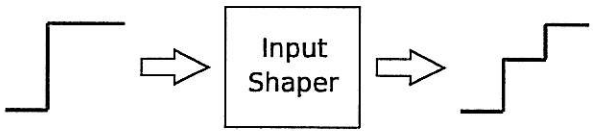

Figure 3.1: Input shaping of a step reference signal.

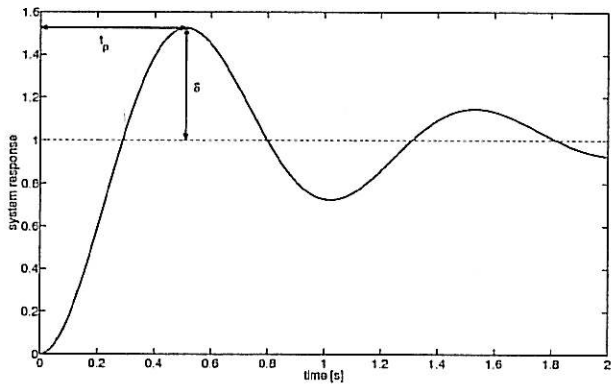

Figure 3.2: Step response for a lightly damped system.

An underdamped system with $\zeta = 0.2$ and a natural frequency $\omega_n = 2\pi$ rad/s would have the step response presented in Figure 3.2. The corresponding damped natural frequency is

$$w_d = w_n\sqrt{1-\zeta^2}, \tag{3.2}$$

thus, $w_d = 6.16$ rad/s or 0.9798 Hz. From control literature [12], the peak time, when the system reaches $1+\delta$, is

$$t_p = \frac{\pi}{w_d} \tag{3.3}$$

and the overshoot is

$$\delta = e^{-\pi\zeta/\sqrt{1-\zeta^2}}, \quad 0 \leq \zeta < 1. \tag{3.4}$$

In most applications, this oscillatory behaviour is undesirable and, quite often, unacceptable. Posicast control offers a simple and yet elegant solution to this by preshaping the system's reference. This shaper is given by the function $1+P(s)$, with

$$P(s) = \frac{\delta}{1+\delta}\left(e^{-t_p s} - 1\right). \tag{3.5}$$

The term $e^{-t_p s}$ represents a delay of t_p seconds. Hence, posicast operates by "holding" a part of the reference signal, represented by $\delta/(1+\delta)$, and then "releasing" this signal after t_p seconds, when the expression $e^{-t_p s} - 1$ becomes zero and $1 + P(s) = 1$. Figure 3.3 shows the system response for the shaped reference. This new reference is the original step convolved with $1 + P(s)$. As with virtually any technique, cancelling the vibration does not come without a cost. Posicast, like any other vibration cancelling input

České vysoké učení technické v Praze
FAKULTA ELEKTROTECHNICKÁ
Katedra řídicí techniky
121 35 Praha 2, Karlovo nám.13
Tel.: 224922372, Fax: 224918646

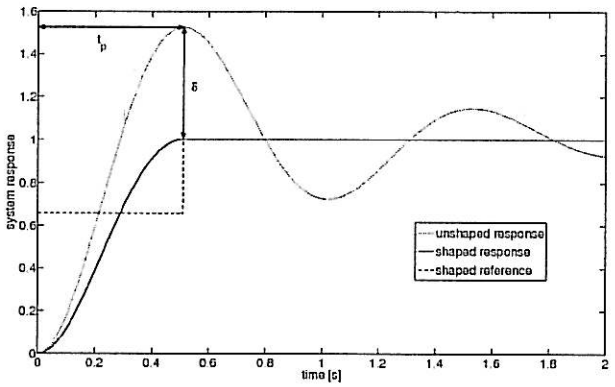

Figure 3.3: Step response for a preshaped system.

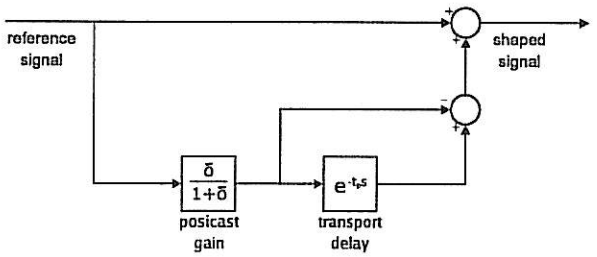

Figure 3.4: Block diagram of the posicast shaper.

shaper, induces a system delay. From Figure 3.3 is clear that the rise time[2] takes longer for the shaped system than for the unshaped.

The transcendental function $e^{-t_p s}$ may be approximated by a rational function of two polynomials of order p in the numerator and order q in the denominator, known as the (p, q) *Padé approximant* [12]. In block diagrams, $e^{-t_p s}$ is the transport delay with t_s s delay, and the posicast shaper can be constructed as in Figure 3.4.

[2]The definition of rise time varies but, typically, is the time it takes for the output to reach 90% of the step height.

Chapter 4

Formation Flying

Formation of satellites involves two or more satellites in active, real-time, cooperation. As opposed to constellation of satellites, where each satellite maintains itself in its own frame, formation flying requires the maintenance of a relative frame between the satellites. For instance, a constellation could be treated as individual orbital adjustments of several independent satellites, a case like the NAVSTAR constellation of 24 satellites for the Global Positioning System, which are separately controlled from Earth.

4.1 Formation Flying Missions

Formation flying usually involves several satellites that perform the task of a single larger satellite. This application is fairly recent[1] but several missions are planned for launch in the next following years. Table 4.1 [14]lists ESA's (alone or in cooperation with other agencies) planned missions for the next years. Apart from those, and the missions that are already deployed, there are several other missions planned by other space agencies such as NASA (e.g. Constellation-X Observatory) and SSC (e.g. PRISMA).

The mission requirements can be divided into:

science requirements: the specifications from the virtual instrument that the satellite cluster should perform (e.g. orbit parameters, pointing precision);

engineering requirements: which are derived from the requirements and constraints defined by the science requirements (e.g. relative distance precision, orbit considerations for positioning and maintenance).

The importance of formation flight for science is that it enables new technologies that would be unfeasible (or even impossible) by using a single

[1]The precursor of formation flight is Earth Observing-1, launched on November 2000, that flew one minute behind Landsat 7 covering the same ground track in a leader-follower formation [13].

Mission name	Mass [kg]	Launch year (Projected)
LISA Pathfinder	470	2009
SWARM	1000	2009
PROBA-3	150	2009
MAX	200	2010
XEUS	undefined	2015+
NIRI (DARWIN)	500	2015+

Table 4.1: ESA's future formation flying missions

satellite. From the engineering point the benefits are modularity (failure of one of the satellites doesn't, necessarily, compromise the whole mission); design of smaller, less complex, satellites; faster deployment after detailed definition, which possibilitates using off-the-shelf technologies; and other cost-related benefits. The major drawbacks are the increase in communication and spacecraft control requirements.

4.2 Relative Motion

Describing the spacecraft dynamics with a rotational frame, as depicted in Figure 4.1, offers a more elegant way to study several problems in space dynamics. For instance, the inspection of small deviations from an ideal orbit caused by perturbations such as drag force and third body perturbations; relative motion between two or more spacecraft in nearby orbits; the *rendezvous* problem and others. The development offered here follows [13], with some modifications. Alternative solutions are treated in several works, though variations from the one presented here seem to be dominant. To mention a couple, [15] proposes linearised equations for relative motion that take into account disturbing forces caused by Earth's oblateness, known as J_2 disturbance. (Other J_n coefficients are at least 400 times smaller than J_2 and are usually neglected [6].) The effects of the common assumptions made in this kind of analysis – namely neglecting gravitational perturbations and circular orbit assumption – is studied in [16]. Results presented there and in other similar works can be used to improve the generalisations made here. The following development leads to Hill's linearised equations for relative motion, which are used to describe the spacecraft relative dynamics through the remainder of this work. One can extend the analysis done with Hill's equations to other models without any major setbacks.

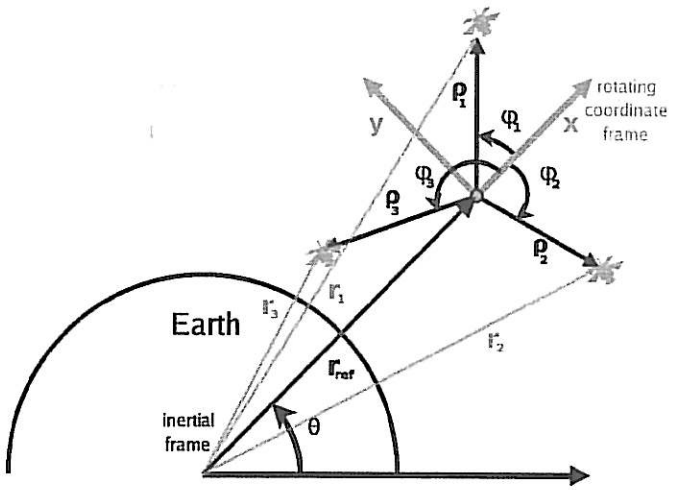

Figure 4.1: Rotating coordinate frame with origin at $\mathbf{r_{ref}}$ for a satellite cluster.

4.2.1 Relative Equations of Motion

The *two-body equation of motion*

$$\ddot{\mathbf{r}}_\mathbf{i} = -\mu_e \frac{\mathbf{r_i}}{{r_i}^3} + \mathbf{f_i}, \tag{4.1}$$

defines the motion of an i-th satellite orbiting Earth. $\mathbf{f_i}$ accounts for any external forces being it disturbances or control inputs. Assume $\mathbf{r_{ref}}$ for the reference spacecraft (or an ideal orbit), with unperturbed motion ($\mathbf{f_{ref}} = 0$); and $\mathbf{r_i}$ for a chaser spacecraft with a driving force $\mathbf{f_i}$ described by

$$\ddot{\mathbf{r}}_\mathbf{ref} = -\mu_e \frac{\mathbf{r_{ref}}}{{r_{ref}}^3}, \tag{4.2a}$$

$$\ddot{\mathbf{r}}_\mathbf{i} = -\mu_e \frac{\mathbf{r_i}}{{r_i}^3} + \mathbf{f_i}. \tag{4.2b}$$

The local coordinate frame (Figure 4.1) defines $\mathbf{x}$ as the axis along $\mathbf{r_{ref}}$; $\mathbf{y}$ is along the direction of the spacecraft's velocity; and $\mathbf{z}$, normal to the xy plane, completes the triad (x, y, z), with its origin at the reference spacecraft's centre of mass, or in an ideal point mass orbit. The absolute position for each spacecraft can be written in terms of the referential position vector $\mathbf{r_{ref}} = [r_{ref}\ \ 0\ \ 0]^T$ and the relative position vector $\boldsymbol{\rho_i} = [x\ \ y\ \ z]^T$ as

$$\mathbf{r_i} = \mathbf{r_{ref}} + \boldsymbol{\rho_i}, \tag{4.3}$$

which, from (4.2b), gives

$$\ddot{\mathbf{r}}_\mathbf{i} = -\mu_e \frac{\mathbf{r_{ref}} + \boldsymbol{\rho_i}}{\|\mathbf{r_{ref}} + \boldsymbol{\rho_i}\|^3} + \mathbf{f_i}. \tag{4.4}$$

Writing the relative position, in (4.3), as a difference between the orbit in study and the reference orbit $\boldsymbol{\rho_i} = \mathbf{r_i} - \mathbf{r_{ref}}$, the second derivative with respect to the inertial frame yields

$$ {}_I\ddot{\boldsymbol{\rho}}_i = \mu_e \left[\frac{\mathbf{r_{ref}}}{{r_{ref}}^3} - \frac{\mathbf{r_i}}{{r_i}^3} \right] + \mathbf{f_i}, $$

or

$$ {}_I\ddot{\boldsymbol{\rho}}_i = \mu_e \left[\frac{\mathbf{r_{ref}}}{{r_{ref}}^3} - \frac{\mathbf{r_{ref}} + \boldsymbol{\rho_i}}{\|\mathbf{r_{ref}} + \boldsymbol{\rho_i}\|^3} \right] + \mathbf{f_i}. \tag{4.5} $$

The subscript I to the left indicates the derivative is with respect to the inertial frame. By using the transport theorem ${}_I\ddot{\boldsymbol{\rho}}_i$ can be expressed with respect to the *Local Vertical/Local Horizontal* (rotating) coordinate frame, represented by the subscript R, as [13, 17]

$$ {}_I\ddot{\boldsymbol{\rho}}_i = {}_R\,\ddot{\boldsymbol{\rho}}_i + 2\dot{\boldsymbol{\theta}} \times_R \dot{\boldsymbol{\rho}}_i + \ddot{\boldsymbol{\theta}} \times \boldsymbol{\rho_i} + \dot{\boldsymbol{\theta}} \times (\dot{\boldsymbol{\theta}} \times \boldsymbol{\rho_i}), \tag{4.6} $$

where $\boldsymbol{\theta} = [0 \;\; 0 \;\; \theta]^T$; and its first and second derivatives provide the angular velocity and acceleration, respectively. Now, expanding the cross products in (4.6) gives

$$ \begin{aligned} {}_I\ddot{\boldsymbol{\rho}}_i &= \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} + 2 \begin{bmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ 0 & 0 & \dot{\theta} \\ \dot{x} & \dot{y} & \dot{z} \end{bmatrix} + \begin{bmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ 0 & 0 & \ddot{\theta} \\ x & y & z \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \dot{\theta} \end{bmatrix} \times \begin{bmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ 0 & 0 & \dot{\theta} \\ x & y & z \end{bmatrix} \\ &= \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} + 2 \begin{bmatrix} -\dot{\theta}\dot{y} \\ \dot{\theta}\dot{x} \\ 0 \end{bmatrix} + \begin{bmatrix} -\ddot{\theta}y \\ \ddot{\theta}x \\ 0 \end{bmatrix} + \begin{bmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ 0 & 0 & \dot{\theta} \\ -\dot{\theta}y & \theta x & 0 \end{bmatrix}, \end{aligned} $$

with $[\mathbf{i} \;\; \mathbf{j} \;\; \mathbf{k}] = [\mathbf{x}/x \;\; \mathbf{y}/y \;\; \mathbf{z}/z]$, and thus

$$ {}_I\ddot{\boldsymbol{\rho}}_i = \begin{bmatrix} \ddot{x} - 2\dot{\theta}\dot{y} - \ddot{\theta}y - \dot{\theta}^2 x \\ \ddot{y} + 2\dot{\theta}\dot{x} + \ddot{\theta}x - \dot{\theta}^2 y \\ \ddot{z} \end{bmatrix}. \tag{4.7} $$

This, together with (4.5) form the base for the nonlinear equations of relative motion

$$ \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \mu_e \begin{bmatrix} \frac{1}{r_{ref}^2} - \frac{r_{ref}+x}{\|\mathbf{r_{ref}}+\boldsymbol{\rho_i}\|^3} \\ -\frac{y}{\|\mathbf{r_{ref}}+\boldsymbol{\rho_i}\|^3} \\ -\frac{z}{\|\mathbf{r_{ref}}+\boldsymbol{\rho_i}\|^3} \end{bmatrix} + 2\dot{\theta} \begin{bmatrix} -\dot{y} \\ \dot{x} \\ 0 \end{bmatrix} + \ddot{\theta} \begin{bmatrix} -y \\ x \\ 0 \end{bmatrix} - \dot{\theta}^2 \begin{bmatrix} x \\ y \\ 0 \end{bmatrix}. \tag{4.8} $$

This equation is referred to as the *True Model* as it accounts for nonlinearities and orbit eccentricity. The solution depends on the *classical orbit parameters* $(a, e, i, \Omega, \omega, \theta)$, where

ω (**or** $\dot{\theta}$**:** is the angular velocity;

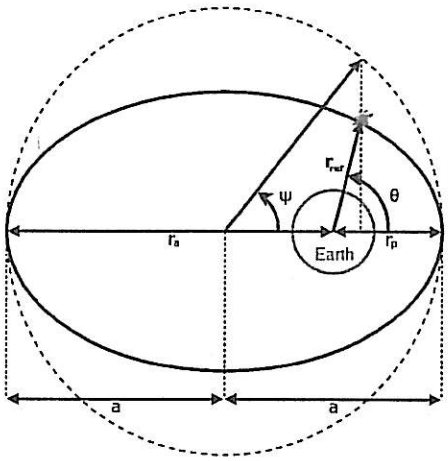

Figure 4.2: Elliptic orbit.

n: is the *mean motion* $n = \sqrt{\mu_e/a^3}$ ($n = \dot{\theta}$ for circular orbits);

a: is the *semimajor axis* $a = (r_a + r_p)/2$;

r_a: is the *apoapsis*, or *apogee* for Earth orbiting spacecrafts;

r_p: is the *periapsis*, or *perigee* for Earth orbiting spacecrafts;

e: is the *eccentricity* $e = (r_a - r_p)/(r_a + r_p)$;

i: is the *orbit inclination*, the angle between the orbital plane and the equatorial plane;

Ω: is the *right ascension of the ascending node*, the angle in the equatorial plane that separates the *node line* (where the orbital and the equatorial plane intersect) from the *vernal equinox* direction Υ;

θ: is the *true anomaly*, the angle between the major axis pointing the periapsis and $\mathbf{r_{ref}}$.

Figure 4.2 provides a quick reference for some of those parameters.

Then, in orbit parameters,

$$r_{ref} = \frac{p}{1 + e\cos\theta}, \tag{4.9a}$$

$$\dot{\theta} = \frac{h}{r_{ref}^2}, \tag{4.9b}$$

where $p = h^2/\mu_e$ and h is the magnitude of the *specific angular momentum* $\mathbf{h} = \mathbf{r} \times \mathbf{v}$. For elliptic orbits

$$p = a(1 - e^2), \tag{4.10a}$$

$$h = \sqrt{a(1 - e^2)\mu_e}. \tag{4.10b}$$

Thus, (4.9) becomes

$$r_{ref} = \frac{a(1-e^2)}{1+e\cos\theta}, \tag{4.11a}$$

$$\dot{\theta} = n\frac{(1+e\cos\theta)^2}{(1-e^2)^{3/2}}. \tag{4.11b}$$

The term

$$\begin{aligned}\|\mathbf{r_{ref}} + \boldsymbol{\rho_i}\|^3 &= [(r_{ref}+x)^2 + y^2 + z^2]^{3/2} \\ &= (r_{ref}^2 + 2r_{ref}x + \rho_i^2)^{3/2},\end{aligned}$$

can be linearised for relatively small values of ρ_i and, after a Taylor expansion, (4.4) can be written as

$$\ddot{\mathbf{r}}_\mathbf{i} = \mu_e \frac{\mathbf{r_{ref}} + \boldsymbol{\rho_i}}{r_{ref}^3}\left[1 - 3\boldsymbol{\rho_i}\left(\frac{\mathbf{r_{ref}}}{{r_{ref}}^2}\right)\right] + \mathbf{f_i} + \mathcal{O}(\rho_i^2). \tag{4.12}$$

Since, from (4.5),

$$\ddot{\boldsymbol{\rho}}_\boldsymbol{i} = \mu_e\frac{\mathbf{r_{ref}}}{r_{ref}^3} - \ddot{\mathbf{r}}_\mathbf{i},$$

substituting $\ddot{\mathbf{r}}_\mathbf{i}$ and arranging the equations yields

$$\ddot{\boldsymbol{\rho}}_\boldsymbol{i} = \frac{\mu_e}{{r_{ref}}^3}\left[-\boldsymbol{\rho_i} + 3\left(\frac{\mathbf{r_{ref}}}{r_{ref}}.\boldsymbol{\rho_i}\right)\frac{\mathbf{r_{ref}}}{r_{ref}}\right] + \mathbf{f_i} + \mathcal{O}(\rho_i^2). \tag{4.13}$$

From the arrangement of the rotating axis, as seen in Figure 4.1, and considering the property of the LHLV frame

$$\frac{\mathbf{r_{ref}}}{r_{ref}} = \begin{bmatrix}1\\0\\0\end{bmatrix},$$

(4.13) becomes

$$\ddot{\boldsymbol{\rho}}_\boldsymbol{i} = \frac{\mu_e}{r_{ref}^3}\begin{bmatrix}2x\\-y\\-z\end{bmatrix}. \tag{4.14}$$

Substituting in (4.7) gives

$$\begin{bmatrix}\ddot{x}\\\ddot{y}\\\ddot{z}\end{bmatrix} = \begin{bmatrix}2\frac{\mu_e}{r_{ref}^3} + \dot{\theta}^2 & \ddot{\theta} & 0\\ -\ddot{\theta} & \dot{\theta}^2 - \frac{\mu_e}{r_{ref}^3} & 0\\ 0 & 0 & -\frac{\mu_e}{r_{ref}^3}\end{bmatrix}\begin{bmatrix}x\\y\\z\end{bmatrix} + \begin{bmatrix}0 & 2\dot{\theta} & 0\\ -2\dot{\theta} & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}\dot{x}\\\dot{y}\\\dot{z}\end{bmatrix} + \begin{bmatrix}f_x\\f_y\\f_z\end{bmatrix},$$

which, with the results in (4.11), writing in terms of θ is defined as the state-space equations

$$\begin{bmatrix} \ddot{x} \\ \dot{x} \\ \ddot{y} \\ \dot{y} \\ \ddot{z} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} 0 & 2\frac{\mu_e(1-e\cos\theta)^3}{a^3(1-e^2)^3} + \dot{\theta}^2 & 2\dot{\theta} & \ddot{\theta} & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ -2\dot{\theta} & -\ddot{\theta} & 0 & \dot{\theta}^2 - \frac{\mu_e(1-e\cos\theta)^3}{a^3(1-e^2)^3} & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -\frac{\mu_e(1-e\cos\theta)^3}{a^3(1-e^2)^3} \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\begin{bmatrix} \dot{x} \\ x \\ \dot{y} \\ y \\ \dot{z} \\ z \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} w_x \\ w_y \\ w_z \end{bmatrix}, \quad (4.15)$$

where **u** are the control inputs and **w** are the disturbances.

4.2.2 Hill's Equations

Based on (4.15), the convenient Hill's (or Clohessy-Wiltshire) equations are derived under two assumptions:

1. the distance between the chaser and the rotational frame centre, ρ, is relatively small;

2. the reference orbit is circular (Figure 4.3).

The first assumption was already necessary to obtain (4.12). If a more complete, nonlinear, simulation is needed, (4.8) with its 10 states accounts for nonlinearities, but the challenge becomes modelling appropriate dynamics for all the states while by using Hill's equations the dynamics can be defined as a compact LTI system. The effects and consequences of those assumptions are well presented in [16].

Circular orbits are, in fact, a special case of elliptic orbit where:

$$\ddot{\theta} = 0,$$
$$\dot{\theta} = \dot{\psi} = n = \sqrt{\frac{\mu_e}{a^3}},$$
$$a = r_{ref},$$
$$e = 0.$$

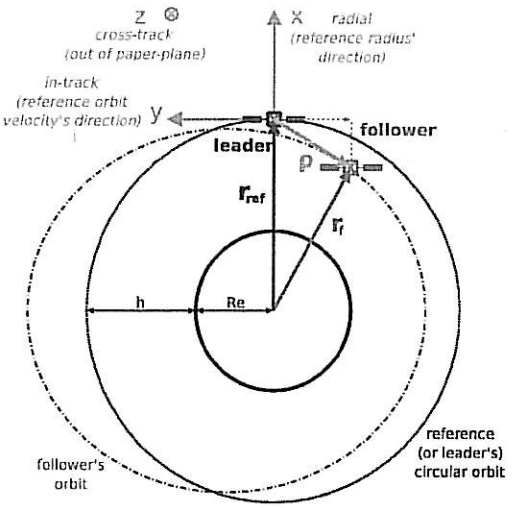

Figure 4.3: Relative coordinate frame for spacecraft formation. R_e is Earth's radius; h is the leader orbit's height; $\mathbf{r_{ref}}$ is the leader's orbit radius, assumed a circular orbit with constant radius $R_e + h$; $\mathbf{r_f}$ is the follower's orbit radius; and $\boldsymbol{\rho}$ is the relative distance between the satellites, $\boldsymbol{\rho} = \mathbf{r_f} - \mathbf{r_{ref}}$.

Hence, (4.15) becomes

$$\begin{bmatrix} \ddot{x} \\ \dot{x} \\ \ddot{y} \\ \dot{y} \\ \ddot{z} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} 0 & 3n^2 & 2n & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ -2n & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -n^2 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \dot{x} \\ x \\ \dot{y} \\ y \\ \dot{z} \\ z \end{bmatrix} + \begin{bmatrix} u_x \\ 0 \\ u_y \\ 0 \\ u_z \\ 0 \end{bmatrix} + \begin{bmatrix} w_x \\ 0 \\ w_y \\ 0 \\ w_z \\ 0 \end{bmatrix}. \tag{4.16}$$

Or, in terms of its components,

$$\ddot{x} - 2n\dot{y} - 3n^2 x = f_x, \tag{4.17a}$$

$$\ddot{y} + 2n\dot{x} = f_y, \tag{4.17b}$$

$$\ddot{z} + n^2 z = f_z, \tag{4.17c}$$

defined as

x, radial motion: same direction from the centre of the Earth to the relative coordinate's origin;

y, in-track motion: same direction from leader's velocity. If x and z are both zero, it can be interpreted as how far behind (or ahead) the follower is from the leader;

z, cross-track motion: perpendicular to the reference orbit plane. Can be interpreted as the deviations from the orbit plane.

From (4.17) it can be observed that the radial and in-track motion are coupled, but are both independent from the cross-track motion.

4.2.3 Unperturbed Motion

A possible configuration for a satellite cluster is to maintain a constant distance between the spacecrafts with the least possible effort. In theory, bodies with the same shape in the same orbit should experience equal perturbations and the motion is perfectly described by Hill's equations. Considering no input, one needs to find a configuration that would maintain the spacecrafts separated by

$$d = \sqrt{x^2 + y^2 + z^2}. \tag{4.18}$$

For spacecrafts in the same orbital plane $z = 0$.

To find a solution for this problem, (4.16) can be decomposed into radial/in-track dynamics

$$\begin{bmatrix} \ddot{x} \\ \dot{x} \\ \ddot{y} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} 0 & 3n^2 & 2n & 0 \\ 1 & 0 & 0 & 0 \\ -2n & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \dot{x} \\ x \\ \dot{y} \\ y \end{bmatrix} + \mathbf{f} \tag{4.19}$$

and cross-track dynamics

$$\begin{bmatrix} \ddot{z} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} 0 & -n^2 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \dot{z} \\ z \end{bmatrix} + \mathbf{f}. \tag{4.20}$$

For unperturbed motion, $\mathbf{f} = 0$ in both equations.

More generally, a desirable configuration could require that the formation sustains the same relative positions after a complete orbital period, a *drift-free* configuration. Another simpler example would be the chaser to maintain the exact same orbit from the leader with a small time delay (or advance), where y would be constant.

The *free force* solution for the system (4.19) [13]

$$\dot{x} = -c_1 n \sin(nt + \alpha), \tag{4.21a}$$
$$x = c_1 \cos(nt + \alpha) + c_2, \tag{4.21b}$$
$$\dot{y} = -2c_1 n \cos(nt + \alpha) - \frac{3}{2} n c_2, \tag{4.21c}$$
$$y = -2c_1 \sin(nt + \alpha) - \frac{3}{2} n c_2 t + c_3, \tag{4.21d}$$

and for the cross-track system (4.20)

$$\dot{z} = -cn \sin(nt + \beta), \tag{4.22a}$$
$$z = c \cos(nt + \beta), \tag{4.22b}$$

where c, c_i, α and β are integration constants.

Passive aperture

The free force solution in (4.19) is an ideal scenario for a satellite cluster since it requires no fuel to maintain the satellites in a relative periodic motion. The solutions to (4.19) are several; some well-known, same plane, are

$$\dot{x}_0 = -2ny_0, \tag{4.23a}$$

$$\dot{y}_0 = n\frac{x_0}{2}. \tag{4.23b}$$

Those allow a relative elliptic motion around the reference point. Figure 4.4 shows some examples of relative motion with the passive apertures from (4.23).

If $y_0 \neq 0$, all others nil, the satellite maintains a fixed distance behind or in front of the reference. The other configurations in Figure 4.4 create an elliptical motion relative to the reference orbit.

Passive apertures benefit from the natural orbit dynamics to maintain the satellites in a relative periodic position without external forces. Since the satellites are in very similar orbits, if they have similar aerodynamics their drag forces will be roughly the same and very little effort will be needed to maintain the cluster in formation. Constant distances, described by (4.18), can be achieved by with a circular aperture, though this would require the satellites to be in separate orbital planes since only with motion in the x-y plane it is not possible to achieve a circular passive aperture. A constant separation is also possible with small values for the in-track y distance.

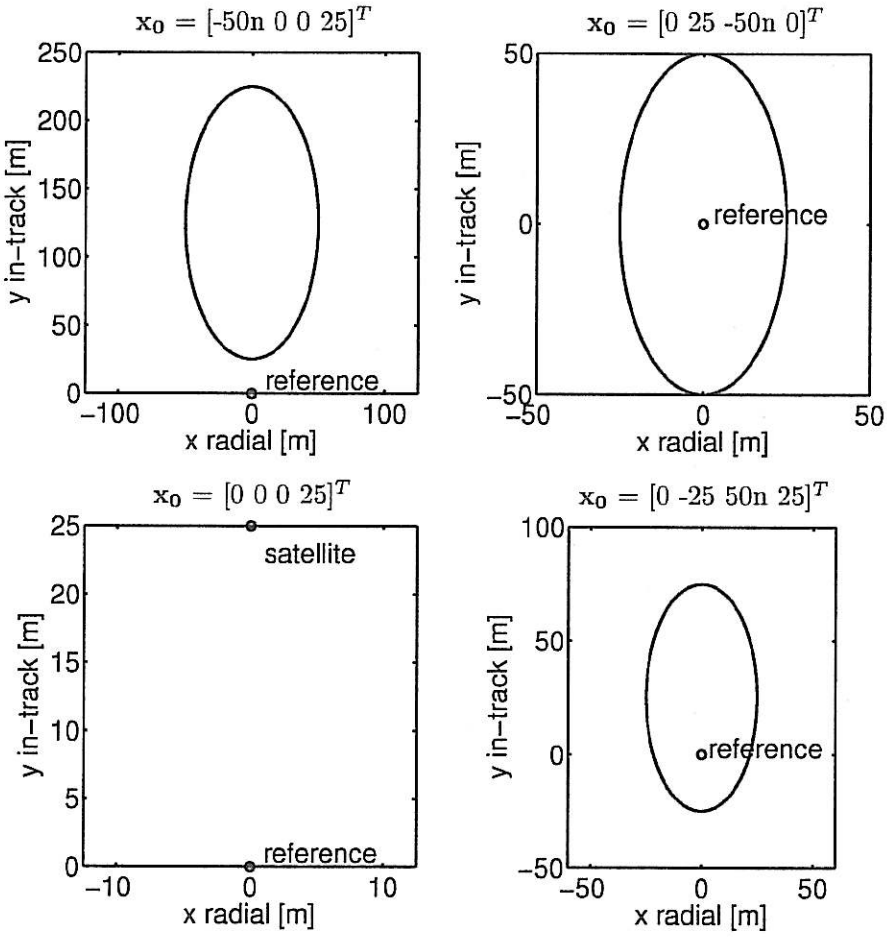

Figure 4.4: Relative motion with a passive aperture, with $\mathbf{x_0} = [\dot{x}_0\ x_0\ \dot{y}_0\ y_0]^T$.

Chapter 5

Orbital Control

When it comes to orbital control, all scenarios fall into two categories:

1. orbit insertion; and
2. orbit maintenance;

or variants of those. Orbit insertion encompasses all major orbital manoeuvres – deployment in some initial parking orbit, orbit transfer, orbit rendezvous, etc. The task is to position the spacecraft in the desired orbit, being it its original deployment or even when deorbiting the spacecraft at the end of its mission. Orbit maintenance takes into account minor translation changes, usually to compensate for disturbances that deviate the satellite from its planned orbit.

Fuel, among all factors that could end a satellite's mission[1], is the main element when determining a spacecraft's life span. No doubt, any control strategy used should take into account measures to minimise fuel consumption, consequently extending the spacecraft life cycle. Therefore, the categories above can be redefined as:

1. finding an optimal minimum fuel consumption trajectory to bring the satellite to a desired position, with a desired velocity, at a given time $t < \infty$; and
2. keeping the satellite in a desired orbit spending as less fuel as possible.

For the first, a possible approach is to use *Linear Programming*, explained in Section 5.1.1. For the second one, it is a matter of designing an appropriate controller that minimises the control effort, such as a Linear Quadratic Regulator (LQR), explained in Section 5.3.

[1] A description of the space environment and its hazards (e.g. solar panel degradation, atomic oxygen attack, atmospheric drag, etc.) can be found in [18].

5.1 Trajectory Planning

Defining an optimal trajectory consists in finding the velocity changes to be performed and at which time intervals to execute those changes. The more straightforward solution, of shifting the spacecraft to the desired relative position and velocity is unlikely to be a fuel-efficient solution in a formation configuration. In most times, it is interesting to take advantage of the free force orbit dynamics and shift the velocity in well defined intervals of time. For a single spacecraft, when only its absolute position is regarded, there may be not too many possibilities and the manoeuvre could be covered with a *Hohmann Transfer* or a *One-Tangent Burn*. In formation flying, or even for the rendezvous problem, the possibilities are innumerable.

5.1.1 Linear Programming

Linear Programming (LP) has well established algorithms to solve problems like the one fore-mentioned. Several software packages are available, both licensed and open-source, that can handle LP and its variants.

Of course we cannot simply defined the velocity shift in the trajectory expecting any spacecraft to be able to perform such changes. The magnitude of those shifts needs to be limited according to the limitations of the spacecraft's propulsion system. LP also encompasses those restrictions by defining linear constraints.

Formally, the definition of an LP problem is to minimise a vector $\mathbf{x}$, weighted by another vector c^T, obeying a set of constraints. It has the general form [19]

$$\min_x \; c^T x \qquad \text{subject to} \qquad \begin{aligned} Ax &= b \\ Gx &\leq h, \end{aligned} \tag{5.1}$$

where $G \in \Re^{m \times n}$ and $A \in \Re^{p \times n}$.

Several problems can be formulated as an LP problem. The question is how to define the variable involved. $\mathbf{x}$ needs to represent all inputs from t_0 to the final time T. Naturally, the system described in (4.16) needs to be discretised. A dynamic system can be written as

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \tag{5.2a}$$

$$\dot{\mathbf{y}} = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t). \tag{5.2b}$$

If the inputs changes only at the discrete intervals kT and the measurements are also sampled with a sampling rate $1/T$, the continuous system in (5.2) corresponds to the discrete dynamic system

$$\mathbf{x}(k+1) = \mathbf{A_d}\mathbf{x}(k) + \mathbf{B_d}\mathbf{u}(k), \tag{5.3a}$$

$$\mathbf{y}(k) = \mathbf{C_d}\mathbf{x}(k) + \mathbf{D_d}\mathbf{u}(k), \tag{5.3b}$$

where

$$\mathbf{A_d} = e^{\mathbf{A}T}, \quad \mathbf{B_d} = \left(\int_0^T e^{\mathbf{A}\tau} d\tau\right)\mathbf{B}, \quad \mathbf{C_d} = \mathbf{C} \quad \text{and} \quad \mathbf{D_d} = \mathbf{D}.$$

To adequate (5.3) as an LP problem, we need to find the unforced final state x_N, at the time instant NT, given the system starts with some initial state x_0 at k_0. This is done using the *state transition matrix* [20]

$$\mathbf{\Phi}(k, k_0) = \mathbf{A_d}(k-1)\mathbf{A_d}(k-2)\ldots\mathbf{A_d}(k_0). \tag{5.4}$$

with $\mathbf{\Phi}(k_0, k_0) = \mathbf{I}$. The complete system evolution, at any k-step is defined as

$$\mathbf{x}(k) = \mathbf{\Phi}(k, k_0)\mathbf{x_0} + \sum_{m=k_0}^{k-1} \mathbf{\Phi}(k, m+1)\mathbf{B_d}(m)\mathbf{u}(m), \tag{5.5a}$$

$$\mathbf{y}(k) = \mathbf{C_d}(k)\mathbf{\Phi}(k, k_0)\mathbf{x_0} + \mathbf{C_d}(k)\sum_{m=k_0}^{k-1} \mathbf{\Phi}(k, m+1)\mathbf{B_d}(m)\mathbf{u}(m) + \mathbf{D_d}(k)\mathbf{u}(k). \tag{5.5b}$$

By making $\mathbf{C_d} = \mathbf{I}$ and $\mathbf{D_d} = \mathbf{0}$, the system output will correspond to the system states $\mathbf{y}(k) = \mathbf{x}(k)$. The first element on the right-hand side in (5.5a), $\mathbf{\Phi}(k, k_0)\mathbf{x_0}$, can be derived from (5.4), prior to the solution of $\mathbf{u}(k)$. This latter needs to be described in a single vector that accounts for all the inputs in the interval $[k_0, N]$, such that (5.1) can be redefined as

$$\min_{u_{tot}} c^T u_{tot} \qquad \text{subject to} \qquad \begin{aligned} A_{tot}u_{tot} &= y_N - C_d(N)\Phi(N, k_0)x_0 \\ b_l &\leq u_{tot} \leq b_u, \end{aligned} \tag{5.6}$$

where

- A_{tot} defined by

$$\begin{aligned} A_{tot} = [&C_d(N)\Phi(N, k_0+1)B_d(k_0) \quad C_d(N)\Phi(N, k_0+2)B_d(k_0+1) \quad \ldots \\ &C_d(N)\Phi(N, N-1)B_d(N-2) \quad C_d(N)\Phi(N, N)B_d(N-1) \\ &C_d(N)\Phi(N, N+1)B_d(N)], \end{aligned} \tag{5.7}$$

  is the full state transition matrix that contains all the state transitions for each k-step $k \in [k_0, N]$;

- u_{tot} defined by

$$u_{tot} = \begin{bmatrix} u(k_0) \\ u(k_0+1) \\ \ldots \\ u(N-2) \\ u(N-1) \\ u(N) \end{bmatrix}. \tag{5.8}$$

is the complete input vector that contains all k-inputs;

- c is a weighting vector;
- b_l are the lowest values (lower boundary) the input u_{tot} can take;
- b_u are the highest values (upper boundary) the input u_{tot} can take;
- y_N is the desired final output (usually the state) where the system should be at the time instant $(N - k_0)T$;
- $\Phi(N, k_0)x_0$ is the unperturbed initial-state system response, would be the final state if $u(k+1) = u(k) = 0$ for $\forall k \in [k_0, N-1]$;
- $\Phi(k+1, k)$ is defined in (5.4);
- x_0 is the initial state.

Since those variable will always be matrices or vectors, I will drop the standard notation using bold characters to describe matrices (in bold capital letters) or vectors (in bold lower-case letters) in LP formulations like the one in (5.6).

The constraints in (5.6) can be summarised as the final output

$$\mathbf{y}(N) = \mathbf{C_d}(N)\mathbf{\Phi}(N, k_0)\mathbf{x_0} + \sum_{m=k_0}^{N} [\mathbf{C_d}(N)\mathbf{\Phi}(N, m+1)\mathbf{B_d}(m)]\mathbf{u}(m). \quad (5.9)$$

This equation is valid for any causal system, so $\mathbf{\Phi}(k, k+1) = 0 \ \forall k$. In fact, this can be regarded as an artifice to keep the indexes in simulations somewhat organised. Therefore, the last input $\mathbf{u}(N)$ can be ignored without any consequences, though if the LP is properly formulated they should be always zero anyway.

The dimensions for this LP's matrix and vectors are dependent on four factors:

1. the sampling frequency $1/T$ considered in the discretisation;
2. the number n_s of states in the system;
3. the number n_u of inputs to the system;
4. the time t_N allowed for the trajectory to be concluded.

They relate to each other as:

- the number of steps $N \approx \frac{t_N}{T}$;
- $A_{tot} \in \Re^{m \times n_s}$;
- $c,\ u_{tot},\ b_l,\ b_u \in \Re^{m \times 1}$;

České vysoké učení technické v Praze
FAKULTA ELEKTROTECHNICKÁ
Katedra řídicí techniky
121 35 Praha 2, Karlovo nám.13
Tel.: 224922372, Fax: 224918646

- $x_0,\ x_N,\ y_N \in \Re^{n_s \times 1}$;
- $u(k) \in \Re^{n_u \times 1}$;
- $\Phi(k+1,k) \in \Re^{n_s \times n_u}$;
- $\Phi(k_0,k_0) \in \Re^{n_s \times n_s}$;
- $m = (N+1)n_u$.

In a linear time-invariant (LTI) system, $\mathbf{A_d}(k+1) = \mathbf{A_d}(k)\ \forall k$ and can be simply written as $\mathbf{A_d}$. The same is valid for the other system matrices $\mathbf{B_d}$, $\mathbf{C_d}$ and $\mathbf{D_d}$. The latter was already disregarded in the formulation above, but one can follow an analogous procedure to derive an LP formulation including the $\mathbf{D_d}$ matrix. For instance, [4] provides an interesting survey on trajectory planning and gives a formulation with this consideration. This preliminary formulation presented here is based from [4].

Now, considering that the output are the measured states, $\mathbf{x}(k) = \mathbf{y}(k)$ and $\mathbf{C_d} = \mathbf{I}$, for an LTI system, (5.7) can be simplified as

$$A_{tot} = [\mathbf{A_d^{N-1}B_d} \quad \mathbf{A_d^{N-2}B_d} \quad \dots \quad \mathbf{A_d^1B_d} \quad \mathbf{A_d^0B_d} \quad \mathbf{0}], \tag{5.10}$$

and (5.9) becomes

$$\mathbf{y}(N) = y_N = \mathbf{A_d}^N\mathbf{x_0} + A_{tot}u_{tot}. \tag{5.11}$$

And last, the LP formulation in (5.6) becomes

$$\min_{u_{tot}} c^T u_{tot} \qquad \text{subject to} \qquad \begin{array}{l} A_{tot}u_{tot} = x_N - A_d^N x_0 \\ b_l \leq u_{tot} \leq b_u. \end{array} \tag{5.12}$$

Further considerations on convex optimality

Often, in optimisation, the minimum is regarded as $\min_x f(x) \to -\infty$ instead of $\min_x f(x) \to 0$. The latter is what we are trying to find here, for no matter if the commands are reverse or forward, any command will translate in energy consumption. Unfortunately LP provides solutions of the first type. To overcome this obstacle, which is not really a limitation or drawback for many other applications, we need to reformulate the problem in way that it will provide a solution of the type $\min_{u_{tot}} c^T u_{tot} \to 0$.

Tillerson [4] proposes, as explained in [19], to extend the equality matrix and the input vector as

$$A_{ext} = [A_{tot} \quad -A_{tot}], \tag{5.13}$$

and

$$u_{ext} = \begin{bmatrix} u_{tot}^+ \\ \\ u_{tot}^- \end{bmatrix}, \tag{5.14}$$

with A_{tot} defined in (5.7) and

$$u_{tot} = u_{tot}^{+} - u_{tot}^{-}. \tag{5.15}$$

The input constraints are defined as the maximum velocity shift that are feasible at any given step.

I preferred a more straightforward approach, if the LP solver allows, in redefining the B matrix in (4.16) and the input $\mathbf{u}(t)$ as

$$\mathbf{B_{ext}} = \begin{bmatrix} 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u_x^+ & u_y^+ & u_z^+ & u_x^- & u_y^- & u_z^- \end{bmatrix}. \tag{5.16}$$

with u_x^+, u_y^+, u_z^+ the inputs that induce a velocity change in the positive direction of the rotational coordinate system and u_x^-, u_y^-, u_z^- induce velocity changes in the reverse direction of the coordinate system. For satellites that have pairs of thrusters in the axis but with opposite direction this approach allows a direct command for each thruster (or a set of thrusters with the same direction) instead of interpreting positive commands as one direction and negative commands as its reverse. The inputs are then bounded as $0 \leq \mathbf{u}(t) \leq \dot{\mathbf{x}}_{max}(t)$. $\mathbf{u}(t)$ is used here to distinguish from $\mathbf{u}(k)$ used in the discretised system since the extended $\mathbf{B_d}$ will be different from (5.16). $\dot{\mathbf{x}}_{max}(t)$ is the maximum velocity shift (impulse) possible at a given time.

Both approaches, with the extended equality constraint matrix A_{ext} or the extended system matrix $\mathbf{B_{ext}}$, produce similar results solution and computationally-wise – the number of variables for the LP computing to find are the same.

5.1.2 Optimal Trajectory Planning

Now we can use the formulations from Section 5.1.1 with the system derived in Section 4.2.2. As an example, we could consider using Hill's equation, defined in (4.16), to solve a rendezvous problem. The spacecraft starts at some initial position x_0 and has to reach the reference orbit, when $x_N = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T$, at a given time t_N. Figure 5.1 presents the trajectory of a spacecraft that was given two orbital periods to achieve the desired position. Table 5.1 lists the parameters for this simulation. The cross-track z initial and final positions and speed are nil, so the spacecraft remains in the desired orbital plane throughout the whole manoeuvre.

These types of plot provide an interesting insight to relative motion. Figure 5.1 shows the evolution of the spacecraft's position and speed during the manoeuvre, and Figures 5.2 and 5.3 show the satellite dynamics in time,

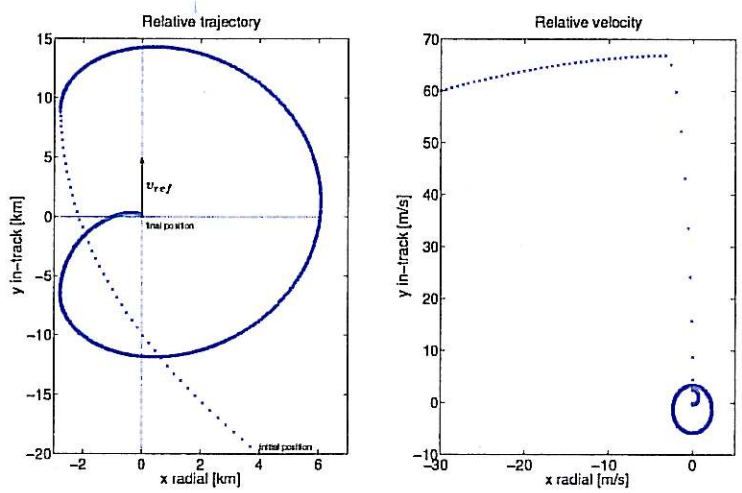

Figure 5.1: Spacecraft trajectory projected on the xy-plane during two orbital periods in a rendezvous problem. The cross marks represent one step each, the relative trajectory plot is the trajectory from the satellite as seen from the reference orbit, v_{ref} indicates the direction of the reference orbit velocity. The velocity plot shows the satellite's velocity progress for the duration of the manoeuvre. It can be compared with the trajectory plot to interpret the complete states evolution.

x_0	$[-30\ 4000\ 60\ -20000\ 0\ 0]$
T	10 s
a	$R_e + 5000$ km
t_f	201 min
N	2416

Table 5.1: Parameters for the rendezvous simulation in Figure 5.1

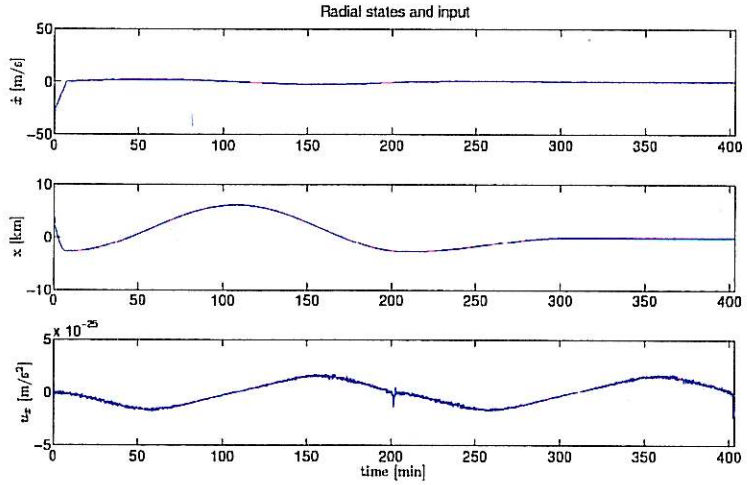

Figure 5.2: Radial satellite dynamics in a rendezvous problem.

with $u_j = u_j^+ - u_j^-$, $j = \{x, y\}$. The satellite is behind (in term of in-track position) the desired position and to the right (further from the orbit's centre). The resulting optimal trajectory increases the satellite's in-track velocity and also increases its relative radial velocity. The speed changes are more evident on the early stages of the manoeuvre, and then only minor shifts are noticed at the end. This is indicated by how spaced the cross marks are at the beginning of the orbit shift. It is worth pointing that a shift in speed does not necessarily has to be induced by an active input, the natural orbital dynamics with its state coupling induces unforced shifts in the orbit dynamics. Figure 5.4 shows the unperturbed states dynamics. Since those initial states are not a solution for the force free equations (4.21) listed in Section 4.2.3, the satellite will drift away from the desired orbit. After only two orbital periods, the satellite drifts almost 5000 km away from the reference orbit.

Actuator limitations

In real-world applications, rarely ever the actuator have the characteristics required to reproduce the signals plotted in Figures 5.2 and 5.3, i.e. unlimited slew rate, high precision and unlimited output. Moreover, rocket thrusters usually accept on-off commands, as explained in Section 2.2, specially the ones used in larger translational changes, like orbital (re)allocation. Those require higher thrust levels than usually required in orbit maintenance.

The first limitation can be accommodated by placing constraints in the inequality defined in the general LP formulation (5.1), $Gx \leq h$. Slew rate

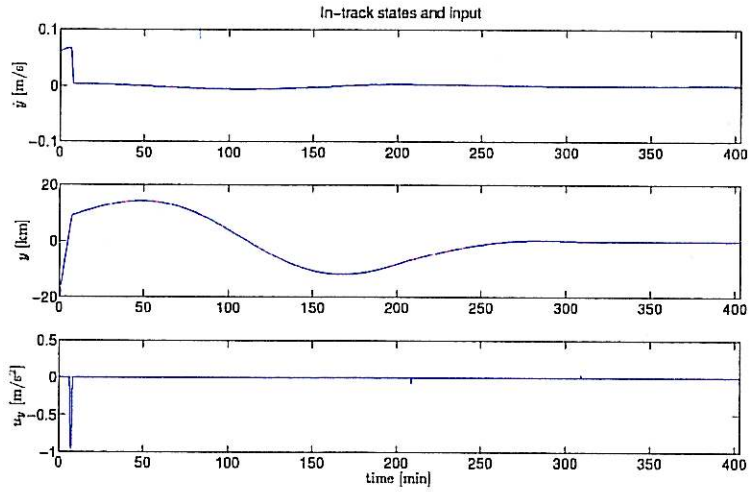

Figure 5.3: In-track satellite dynamics in a rendezvous problem.

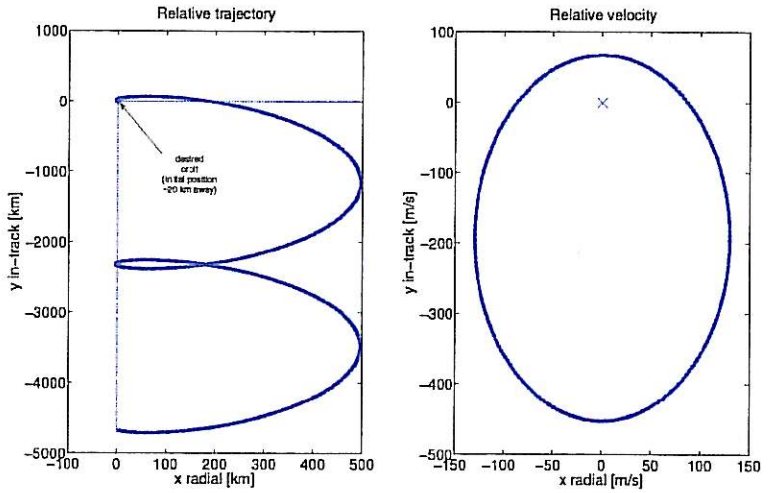

Figure 5.4: Unperturbed satellite's relative dynamics.

is how fast the actuator can shift or, in discrete systems, how much it can vary from one step to another. This is expressed as

$$-s_j^{rate} \leq u_j(k+1)u_j(k) \leq s_j^{rate}, \tag{5.17}$$

or, in the inequality matrix format, as

$$S = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & \dots & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & & & & \ddots & \ddots & \\ 0 & 0 & \dots & -1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -\mathbf{I} & \mathbf{I} & & \dots & \mathbf{0} \\ & -\mathbf{I} & \mathbf{I} & & \\ & & \ddots & \ddots & \\ \mathbf{0} & \dots & & -\mathbf{I} & \mathbf{I} \end{bmatrix}. \tag{5.18}$$

The dimension of the identity matrices $\mathbf{I}$ will be the same as the inputs', and one $\{-\mathbf{I}, \mathbf{I}\}$ pair shall be needed for each k^{th} step in the trajectory for each constraint. The inequality for this is defined in the LP formulation as

$$\begin{aligned} \min_{u_{tot}} \quad & \text{subject to} \quad A_{tot}u_{tot} = x_N - A_d^N x_0 \\ & \begin{bmatrix} S \\ -S \end{bmatrix} u_{tot} \leq \begin{bmatrix} s_x^{rate} \\ s_y^{rate} \\ s_z^{rate} \\ \vdots \\ s_x^{rate} \\ s_y^{rate} \\ s_z^{rate} \end{bmatrix} \\ & b_l \leq u_{tot} \leq b_u. \end{aligned} \tag{5.19}$$

Naturally, the boundaries can be also defined inside the inequality matrix G.

This is an interesting aspect of Linear Programming. By extending the inequality matrix, several input constraints can be defined, giving LP a certain level of modularity. There are limitations, of course, and one is how to define constraints for binary inputs, needed for on-off actuators. This could be formulated, instead, in an Integer Programming (IP) problem. Again, there are several software tools capable of performing this computation, but the convex optimisation offered by LP is no longer guaranteed. Another issue is that IP (or 0-1 Programming), is much more computationally demanding than LP, for it if defined without restrictions in terms of branching, it could expand up to 2^n nodes, n the number of variables. For the rendezvous example just presented, this would represent 2^{9668} nodes!

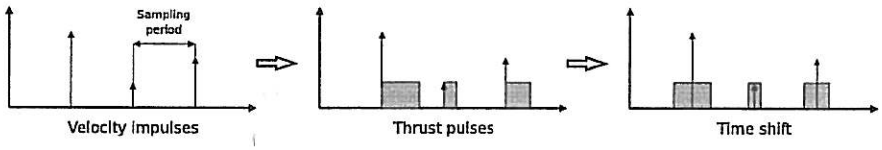

Figure 5.5: Mapping the velocity impulses into thrust pulses.

There are workarounds to avoid this exponential growth. A branching strategy could be defined, or the problem could be structured for *dynamic programming*. This first appeared in the 1940s, and is based on R. E. Bellman's *principle of optimality* [21]:

> An optimal policy has the property that no matter what the previous decision (i.e., controls) have been, the remaining decisions must constitute an optimal policy with regard to the state resulting from those previous decisions.

With this concept, instead of doing a search through all possible combinations, at every new node created, an optimality policy is tested. In that way, the number of nodes grows linearly instead of exponentially. Lewis and Syrmos [21] provide a good introduction on dynamic programming. A complete optimal strategy could make use of all those techniques, in a compromise between performance, computational requirements and similarities to real systems.

Velocity shifts to thrust conversion

An option to proceed with LP and yet fulfil the thruster's on-off constraint is to convert the velocity shifts in thrusts. The solution for the optimal minimum fuel trajectory given by the LP formulations in (5.6) or, more specifically, (5.19) provides the velocity shifts that need to occur at each step. Now what we need is to map the velocity impulses to thrust pulses, as outlined in Figure 5.5.

The procedure is:

1. calculate how long the thruster needs to be on to perform the change in velocity defined by the magnitude of its impulses;

2. find the centre of each impulse and place the centre of each pulse on the same position, this removes the delays caused by this conversion;

3. round the pulses to fit the new input vectors, so the smaller are the steps of the thrust pulses compared to the step of the calculated impulses (sampling period in Figure 5.5), the better is the approximation.

This approach is quite limited, though. Apart from the errors caused by the approximations, for very small values of Δv, the pulses will also be very short, possibly shorter than the minimum impulse bit (Section 2.2). To obtain the Δv impulses, the model also needs to be discretised considering the impulse response approximation instead of the more common Zero-Order-Holder.

5.2 Trajectory Tracking

The open-loop trajectory planned in Section 5.1 hardly seems a safe approach. Especially when considering the hours-long manoeuvres that involves an change. A small deviation from the original trajectory could represent thousands of kilometres of error at the final state. The velocity shifts Δv or even the accelerations defined for the optimal trajectory would tolerate no errors from the actuators. It is necessary to create a feedback control structure and it seems natural to assume that the optimal trajectory defined in Section 5.1 is a good place to start defining a complete feedback control strategy for orbital manoeuvres.

5.2.1 LQR Controller Design

Section 2.5 briefly described some concepts from optimal control. The Linear Quadratic Regulator (LQR) is a special case in optimal control theory. The reasoning behind choosing to use an LQR controller is that it is quite straightforward to implement and many software packages contain tools to design it fast and effectively. Before choosing it I had tried to tune a controller by hand. After a few hours without success on tuning the controller, changing to an LQR proved to be a reasonable and effortless option.

It consists in minimising a quadratic cost function

$$J(t_0) = \frac{1}{2}x^T(T)S(T)x(T) + \frac{1}{2}\int_{t_0}^{T}(x^TQx + u^TRu)dt \tag{5.20}$$

for the general linear system

$$\dot{x}(t) = \mathbf{A}x(t) + \mathbf{B}u(t). \tag{5.21}$$

(2.19), defined previously, is a more general format of the cost function in (5.20). The weighting matrices Q and R are selected by the control engineer, roughly dictating how much effort should be made to minimise the states in x or the inputs in u. All matrices in (5.20) are assumed to be positive semi-definite or positive definite as

$$S(T) \geq 0, \qquad Q \geq 0, \qquad R > 0, \qquad \text{all symmetric}$$

Linear system:

$$x(k+1) = \mathbf{A_d}x(k) + \mathbf{B_d}u(k) \tag{5.24}$$

Cost function:

$$\frac{1}{2}x^T(N)S(N)x(N) + \frac{1}{2}\sum_{k=k_0}^{N-1}(x^T(k)Qx(k) + u^T(k)Ru(k)) \tag{5.25}$$

Riccati equation:

$$S(k) = A_d^T[S(k+1) - S(k+1)B_d(B_d^T S(k+1)B_d + R)^{-1}B_d^T S(k+1)]A_d + Q \tag{5.26}$$

Kalman gain:

$$K(k) = (B_d^T S(k+1)B_d + R)^{-1}B_d^T S(k+1)A_d + Q \tag{5.27}$$

Optimal input:

$$u(k) = -K(k)x(k) \tag{5.28}$$

Table 5.2: Discrete linear quadratic optimal regulator

and, usually, diagonal matrices. $S(t)$ is the solution to the *Riccati equation*

$$-\dot{S}(t) = A^T S(t) + S(t)A - S(t)BR^{-1}B^T S(t) + Q, \qquad t \leq T. \tag{5.22}$$

The optimal control input is then defined as

$$u(t) = -R^{-1}B^T S(t)x(t)$$

or, in terms of the *Kalman gain*

$$K(t) = R^{-1}B^T S(t),$$
$$u(t) = -K(t)x(t). \tag{5.23}$$

The discrete counterparts of (5.20)-(5.23) are defined in Table 5.2 [21]. Both sets of equations are presented since this works uses interchangeably continuous and discrete systems, unless in situations where is mandatory to use a discretised system like in the LP formulation presented in Section 5.1.2.

In LTI systems, with slow varying reference signal, (5.22) can be approximated to a limiting solution $S(\infty)$ when $S(t)$ converges for $t \ll T$. Thus, $\dot{S}(t) = 0$, and the *Algebraic Riccati Equation* (ARE) is defined as

$$0 = A^T S + SA - SBR^{-1}B^T S + Q. \tag{5.29}$$

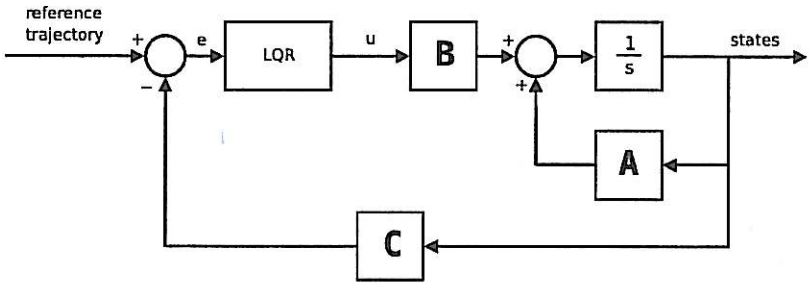

Figure 5.6: LQ regulator.

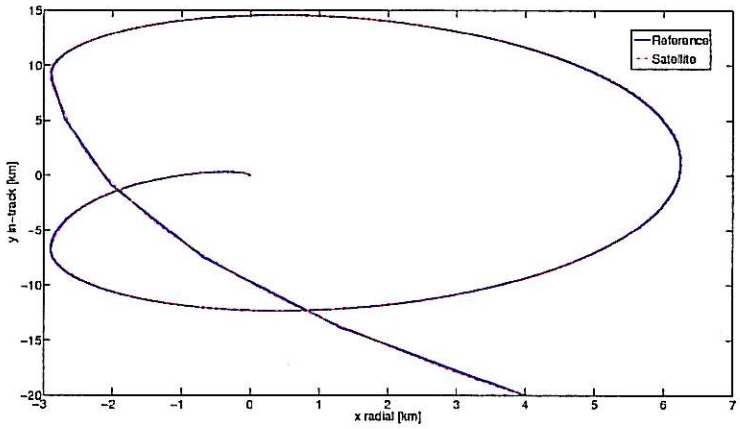

Figure 5.7: Simulated trajectory for a satellite tracking an optimal minimum fuel trajectory.

The suboptimal gain then becomes

$$K(\infty) = R^{-1}B^T S(\infty). \tag{5.30}$$

There are several configurations that can make use of those equations to set a LQR controller. The one adopted here is depicted in Figure 5.6. More complex structures would include a compensator and some other filters, but the design here is being kept as general as possible to be able to include extensions at any step defined through this work. For the LQR controller used here, the requirements are that it adequately follows the trajectory, without major specifications in performance. Figure 5.7 shows the minimum fuel trajectory (originally presented in Figure 5.1) and how the satellite follows it. For this simulation, it was considered an on-off thruster with the same parameters used originally in Chapter 2 and defined in Table 2.2.

The satellite follows the reference trajectory almost perfectly. The small deviations are due to the smoother trajectory that the satellite takes. This happens because of the values selected for the weighting matrix Q, a diagonal matrix with equal elements. By keeping it much smaller than the values in

R, the minimisation in (5.20) makes a much larger effort to keep the control inputs small than to minimise the states, in this case redefined to be the tracking error. For all the simulations, the elements in Q were tens or hundreds of time smaller than the elements in R. The system could be forced to have a tighter tracking performance, but that would represent a much greater fuel expenditure.

Also, the tracking performance will be dependent on the actuator constraints. The nonlinearities considered in this simulation was a dead-zone, so the actuator doesn't constantly shift on and off; and a signum function, of the form

$$\begin{aligned} f(x) &= \text{signum}(x) \\ f(x) &= \begin{cases} -1 & \text{for} \quad x < 0 \\ 0 & \text{for} \quad x = 0 \\ 1 & \text{for} \quad x > 0, \end{cases} \end{aligned} \tag{5.31}$$

multiplied by the actuator gain, i.e. the thrust. With a dead zone in the interval $[-0.1 \quad 0.1]$, the system behaved accordingly and the control effort was rather small, as seen in Figure 5.8. The most effort is at the beginning of the manoeuvre, then later only small adjustments are made to keep the satellite in the desired trajectory.

The complete control designed in this chapter followed the guidelines for linear systems and then stretched for the nonlinear case. Certainly, not the most appropriate design but its results were suitable for the purpose here. Nonlinearities are very particular to each system and a more in-depth analysis would drift from the higher level approach that I am trying to proportionate in this work. The exact, or statistical, characteristics of each actuator would have to be derived from experimentation. Then it would be a matter of modelling it and trying with different parameters – e.g. the dead zone interval.

5.3 Orbit Maintenance

Orbit maintenance should require only small active velocity shifts. Maintaining a satellite cluster in formation, in a passive aperture, requires even less interference from the satellite systems, only to avoid one of the satellites to drift away from the formation. The challenges in orbit maintenance can be divided in two parts:

- maintaining the cluster in formation, in the desired structure required to accomplish the scientific mission; and
- keeping the whole platform in the desired orbit, no matter what the reference point is.

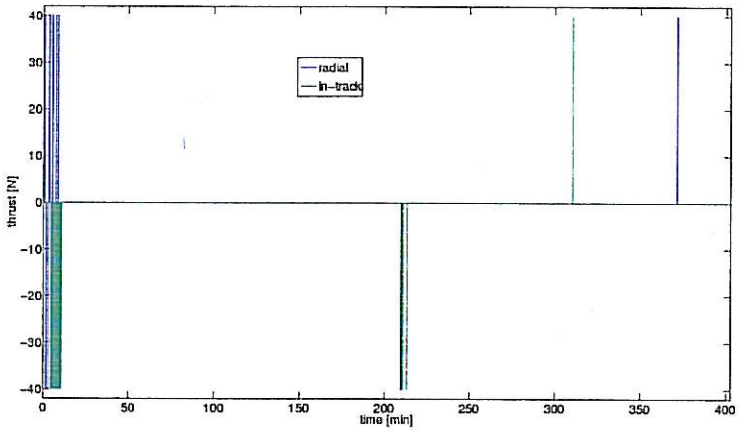

Figure 5.8: Control effort for the rendezvous manoeuvre.

The importance of each will dictate the amount of effort expended in keeping each frame – the relative or the absolute.

Input shaping of the reference would be of little use on formation keeping – there is no shift in the reference signal – and, henceforth, is left aside in this thesis. The amount of thrust would be so small and so would be the vibrations and its consequences. There are increasingly researches on formation keeping without using rocket thrusters. For example, [22] uses hybrid propulsion using coulomb and electrostatic forces in the order of mN to keep the relative formation.

Chapter 6

Vibration Analysis and Simulations

This chapter evolves from the analysis presented in Chapters 3 and 4. It has the purpose of presenting general guidelines on how to extend the system to include the dynamics of the flexible satellite. I allowed myself a certain level of abstraction for no matter how deep is the analysis, the points raised here will be the same:

- how to extend the rigid satellite dynamics to include flexible modes;
- how vibration affects relative motion; and
- present how input shaping affects the performance of linear and non-linear systems.

The first abstraction is to consider the vibration modes in only one direction. Say the satellite counts only with a pair of on-off thrusters, and has to perform the same rendezvous manoeuvre provided in the example in Chapter 4. Instead of rotating the satellite to align the thruster with the direction of the velocity shifts Δv, we could benefit from the coupling in Hill's equation for relative motion and bring the satellite to the reference frame's origin only by applying active inputs in just one of the axis. From Figure 5.8 makes more sense to select the axis where the control effort is smaller, in radial motion; or simulate both and compare the fuel costs.

For that, the LP problem needs to be redefined for a pair of inputs, (u_x^+, u_x^-) or (u_y^+, u_y^-). Matrix $\mathbf{B_{ext}}$ for the same orbital planes (no cross-track motion), in (5.16), becomes

$$\mathbf{B_{ext}} = \begin{bmatrix} 1 & -1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} u_x^+ \\ u_x^- \end{bmatrix}, \tag{6.1}$$

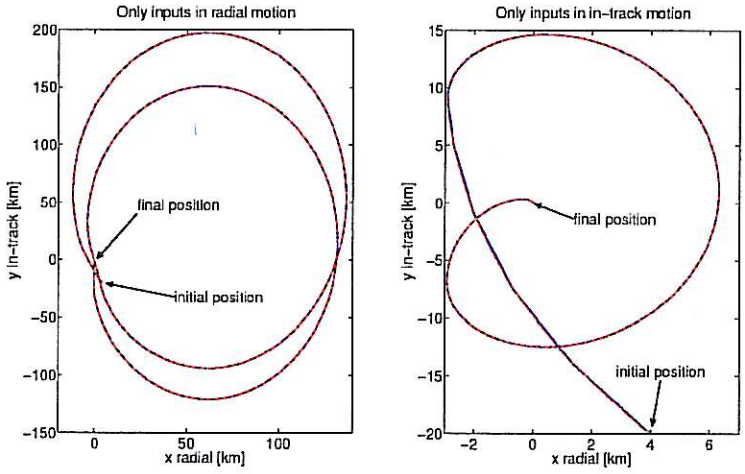

Figure 6.1: Trajectories with inputs constrained in a single axis of motion.

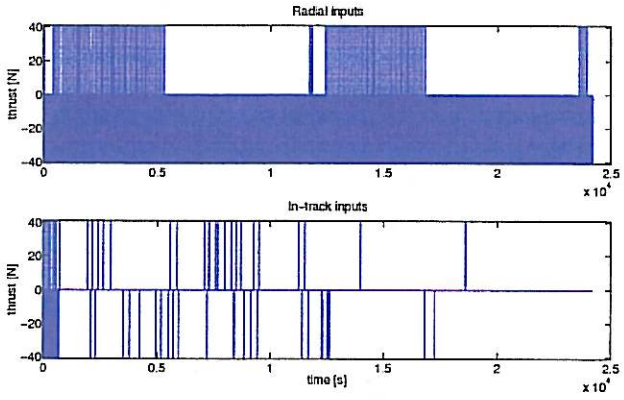

Figure 6.2: Rocket thrust needed to perform the manoeuvres in Figure 6.1.

and the analogous for in-track motion would have 1 and -1 in the third line.

Figure 6.1 compares the trajectories (planned and simulated for non-linear actuators) for only radial or in-track inputs. The control effort for both plots can be seen in Figure 6.2. The trajectory using solely thrust in the radial direction, for this example, needs much more intervention than if using only in-track thrusts. In fact, it is marginally stable for the first case – the values for the LQR controller had to be more conservative than the previous example in Chapter 4. The total radius for the manoeuvre is also much larger, and the amount of fuel necessary to perform this rendezvous is over 10 times greater if only radial thrusts were used instead of in-track thrusts.

6.1 Extended Flexible System

Now, I will extend the system matrices to accommodate also the vibration modes. The flexibility analysed here is the one derived in Section 2.1.1 though a complete system, with all three motions and with a full description of the flexibility in all spacecraft axis, is easily extendible from this analysis. The resulting system, with just one flexible mode on each axis, would need 12 state variables.

Disregarding the cross-track motion, which is decoupled from the radial/in-track plane, and considering flexible modes only in the in-track axis provides the system in state-space description

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \\ \mathbf{y} &= \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}, \end{aligned} \tag{6.2}$$

with the system matrices defined as

$$\mathbf{A} = \begin{bmatrix} 0 & 3n^2 & 2n & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ -2n & 0 & 0 & 0 & \frac{2k_d}{m_1 l} & \frac{2k}{m_1 l} \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -k_d\left(\frac{2}{m_1 l^2} + \frac{1}{m_2 l^2}\right) & -k\left(\frac{2}{m_1 l^2} + \frac{1}{m_2 l^2}\right) \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{6.3a}$$

$$\mathbf{B} = \begin{bmatrix} \frac{1}{m_1+2m_2} & 0 \\ 0 & 0 \\ 0 & \frac{1}{m_1} \\ 0 & 0 \\ 0 & -\frac{1}{m_1 l} \\ 0 & 0 \end{bmatrix} \tag{6.3b}$$

$$\mathbf{C} = \mathbf{I} \tag{6.3c}$$

$$\mathbf{D} = \mathbf{0}. \tag{6.3d}$$

$\mathbf{B}$, in (6.3b), could be defined with solely the second column, if there is no input in the radial axis.

6.1.1 Vibration Damping for Linear Actuators

The system in described by (6.3), is an abstraction. In any real system, there will be flexible modes in all axes, and depending on the sort of actuator and how the satellite's attitude evolves in time, a time varying system description would be required for a more accurate system. That is very particular to each system and will not be treated here. This development can be used for other systems, not just satellite motion.

I will make use of the posicast control presented in Section 3.1. This implementation is merely illustrative, so I make no considerations on robustness, the most noticeable deficiency of posicast shapers. More complex shapers are addressed in the several works from Singhose and/or Singer (e.g. [23, 24, 25]). Besides, in linear systems, one interesting aspect of input shaping is that its parameters are only dependent on the flexible modes, so modelling errors in parameters that do not affect the flexible modes also do not affect the input shaper effectiveness.

For the open-loop system, the posicast parameters, δ and t_p, are only defined by the flexible appendage characteristics, described by the spring's torsion coefficient k and damping from internal frictions k_d; with

$$\omega_n = \sqrt{\frac{k}{m_2}} \tag{6.4}$$

and

$$\zeta = \frac{k_d}{2}\sqrt{km_2}. \tag{6.5}$$

t_p and δ are obtained from (3.3) and (3.4), respectively. For the feedback structure depicted in Figure 6.3, ζ and ω_n (and, consequently, t_p and δ) need to be derived from the new system poles. It is well known that the open-loop poles are the eigenvalues from matrix $\mathbf{A}$, as

$$p = \text{eig}(\mathbf{A}).$$

Considering the system in Figure 6.3, (6.2) can be rewritten as

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{Ax} + \mathbf{Bu} \\ &= \mathbf{Ax} + \mathbf{BKe} \\ &= \mathbf{Ax} + \mathbf{BK}(\mathbf{r} - \mathbf{Cx}) \\ &= (\mathbf{A} - \mathbf{BKC})\mathbf{x} + \mathbf{BKr}, \end{aligned} \tag{6.6}$$

where $\mathbf{r}$ is the reference states trajectory and $\mathbf{K}$ is the [Kalman] gain from the LQR controller. Hence, the poles are

$$p = \text{eig}(\mathbf{A} - \mathbf{BKC}),$$

from where the damping ratio ζ and natural frequency ω_n can be obtained by expanding the characteristic equation

$$(s - p_i)(s - p_i^*) = s^2 + 2\zeta\omega_n + \omega_n^2,$$

with $(p_i,\ p_i^*)$ the complex conjugates of interest. The necessary equations are summarised on Table 6.1. Depending on the system, there will be more complex pairs than just one. Even Hill's equation has some slow oscillating

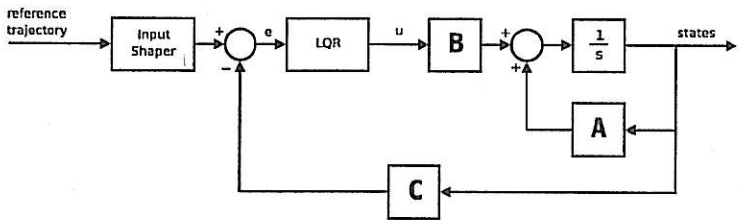

Figure 6.3: Block diagram for the feedback system with the input shaper.

Complex conjugate poles:

$$\begin{aligned}(s+p_i)(s+p_i^*) &= (s+\zeta\omega_n - j\omega_d)(s+\zeta\omega_n + j\omega_d) = 0 \\ &= (s+\zeta\omega_n)^2 + \omega_d^2 = 0\end{aligned} \tag{6.7}$$

with solutions

$$\begin{aligned}s &= -\zeta\omega_n \pm j\omega_d \\ \omega_d &= \mathrm{Im}(s) \\ \zeta &= \sqrt{\frac{\mathrm{Re}^2(s)}{\mathrm{Re}^2(s)+\omega_d^2}}\end{aligned}$$

Overshoot:

$$\delta = e^{-\pi\zeta/\sqrt{1-\zeta^2}} \tag{6.8}$$

Peak time:

$$t_p = \frac{\pi}{\omega_d} \tag{6.9}$$

Posicast shaper:

$$1 + P(s) = 1 + \frac{\delta}{1+\delta}(e^{-t_p s} - 1) \tag{6.10}$$

Table 6.1: Equations for posicast control

states	$\mathbf{x}$	$[\dot{x}\ x\ \dot{y}\ y\ \dot{\theta}\ \theta]$
initial states	x_0	[-30 4000 60 -20000 0 0]^T
final states (desired)	x_N	[0 0 0 0 0 0]^T
final time	t_f	24200 s
torsion coefficient	k	37.5 kg.rad^2.s^{-2}
damping constant (friction)	k_d	0.045 kg.rad^2.s

Table 6.2: Simulation parameters for the flexible satellite

overshoot	δ	0.9902
peak time	t_p	0.6103 s

Table 6.3: Posicast parameters for the closed-loop system

mode. It is up to the control engineer to select which poles cause the unwanted oscillations and design a shaper that suppresses them. One posicast shaper is needed for each pair of complex poles. Hence, one set of parameters (overshoot δ (3.4) and peak time t_p (3.3)) is obtained from each pair of conjugate poles.

The equations on Table 6.1 are just general guidelines. System zeros would have effect on the system damping factor ζ and, consequently, on the damped frequency ω_d. For the system in (6.6), there are no zeros that affect the last state, the flexible appendages deflection θ. Hence, the results from Table 6.1 hold for this system.

Simulations

For the simulations, the flexible appendages parameters are the same previously used (Table 2.2) and are repeated here in Table 6.2, which also lists the simulation parameters. Table 6.3 lists the corresponding posicast parameters, calculated from Table 6.1, for the system depicted in Figure 6.3.

One more consideration is necessary before the actual simulations can take place. Because of its structure (Figure 6.3), the delay operator $e^{-t_p s}$ needs to be initialised to some non-nil value in order to keep the error e small. In other words, the terms inside the parentheses in (6.10) should cancel, i.e. $e^{-t_p s} = 1$. If the posicast is constructed as the block diagram in Figure 3.4, the transport delay block should be initialised in order to cancel the posicast $P(s)$ node, approximating the delay as

$$e^{-t_p s} \approx x_0 \frac{\delta}{1+\delta} \qquad \text{for} \quad t < t_p, \tag{6.11}$$

where x_0 are the system initial states. When $t > t_p$, though, the system has already evolved to different values from the initial conditions, and when the delay block shifts from the initialised values in (6.11) to $r(t - t_p)\delta/(1 + \delta)$

there is a fast drop in e which induces more oscillation than if the system was unshaped.

This is reflected in the $\dot{\theta}$ and θ responses in Figure 6.1.1. During the first 100 s, before the abrupt change in the thruster's output rate, the vibrations for the unshaped response are smaller. Later, when the posicast reaches a regular operational regime, the vibrations in the shaped system decrease greatly and there are virtually no oscillations after 250 s. Posicast removes all the undesirable flexible modes from the system. In this case, the vibration levels are very small with and without input shaping due to the controller response, that tries to minimise the control effort throughout all the manoeuvre. By changing the LQR weighting matrices, Q and R, to a more aggressive stance forcing the LQR to minimise the states over the inputs produces the results seen in Figure 6.5. The posicast control still performs as desired, but the control effort is too great. This proves that the input shaper is effective for even high levels of oscillation.

The delay induced by the posicast is seen in Figures 6.6 and 6.7, that shows the respectives shaped and unshaped references and trajectories for the initial second of the rendezvous manoeuvre. In the shaped system, the controller only starts acting after the time delay t_p has passed. For this initial second, the states from both shaped and unshaped simulations overlap. They only slightly drift apart from the unshaped desired trajectory, but those plots are interesting to note how large is the difference between the references.

For the remaining part of the manoeuvre, the tracking performance is as expected from the previous results in Chapter 5. With the exception of the vibration, both shaped and unshaped systems behave similarly, and only a slight difference in the fuel consumption is noticed, the shaped system consuming less. The delay persists even when the system reaches its steady-state, but the states difference is, at most, on the order of 10^{-7} – i.e., the states from the unshaped system are no more than 10^{-7} (m or m/s) closer to the final reference than the states from the shaped system. So the final states are practically the same for both shaped and unshaped systems, their difference being smaller than the steady-state error (that can be easily removed with a set of integrators at the error signal). Figure 6.8 shows the final states for the in-track variables.

For this simulation, the shaped and unshaped references are equal at the end of the manoeuvre. This might not be the case in several different scenarios and the time delay could represent an inconvenient. One way to overcome it is to start the manoeuvre t_p s before the planned time. Another way is to define the delay in the posicast shaper dynamically. As the states progress to the end of the manoeuvre the control effort becomes smaller[1],

[1]This is generally true for passive apertures, but can differ a lot depending on the initial conditions and desired final states.

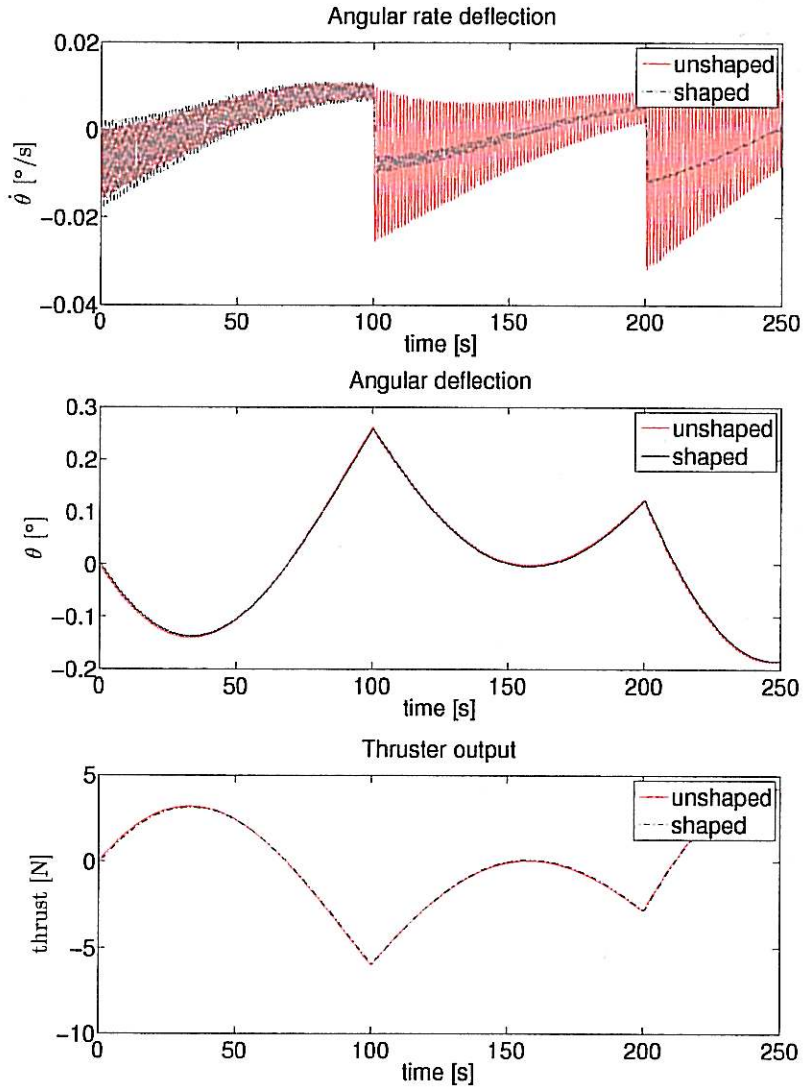

Figure 6.4: Angular rate deflection in the beginning of the manoeuvre.

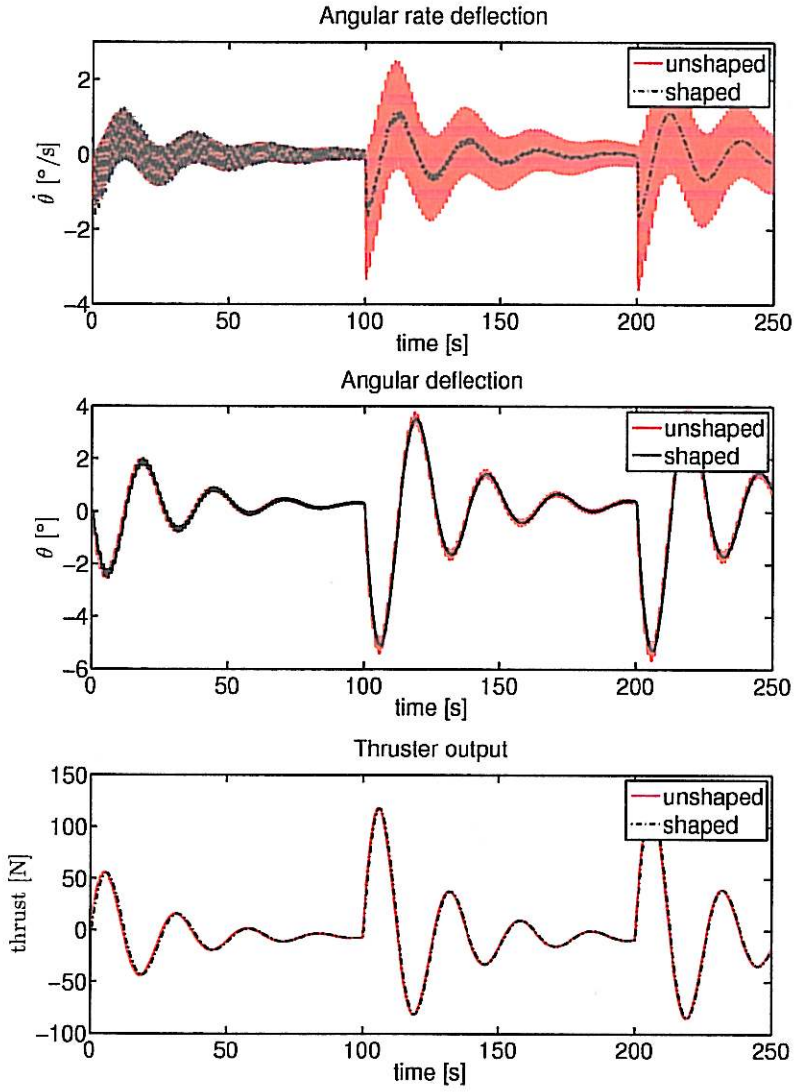

Figure 6.5: Angular rate deflection in the beginning of the manoeuvre with increased states weighting matrix Q and decreased control weighting matrix R in the LQR controller.

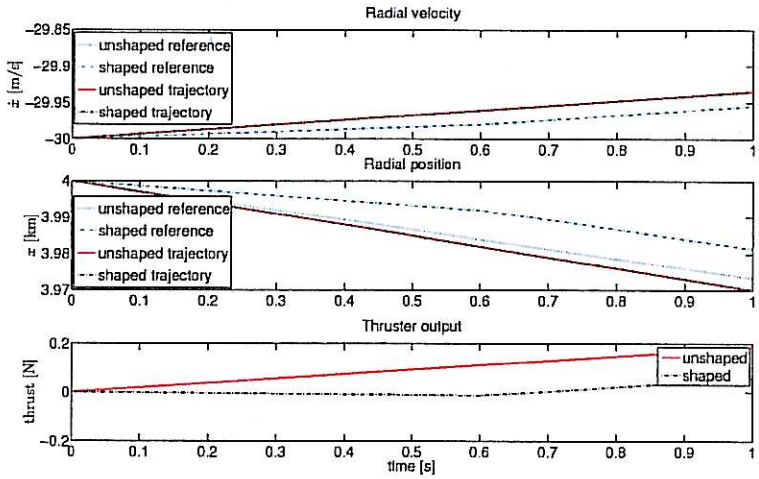

Figure 6.6: Shaped and unshaped references and radial states trajectories for the first second of the rendezvous manoeuvre.

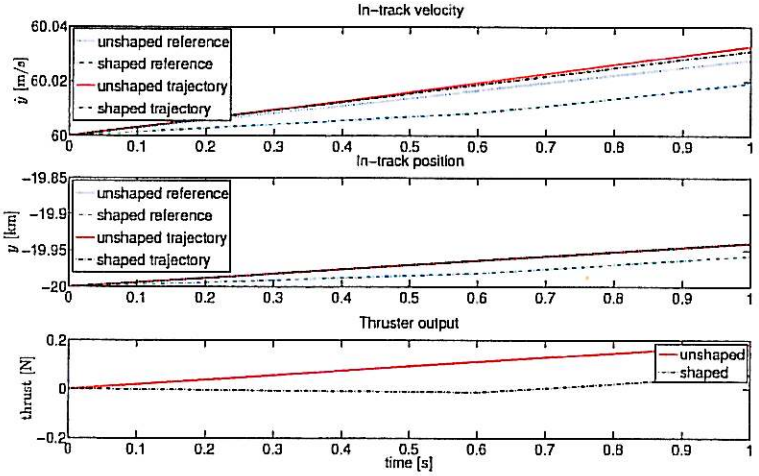

Figure 6.7: Shaped and unshaped references and in-track states trajectories for the first second of the rendezvous manoeuvre.

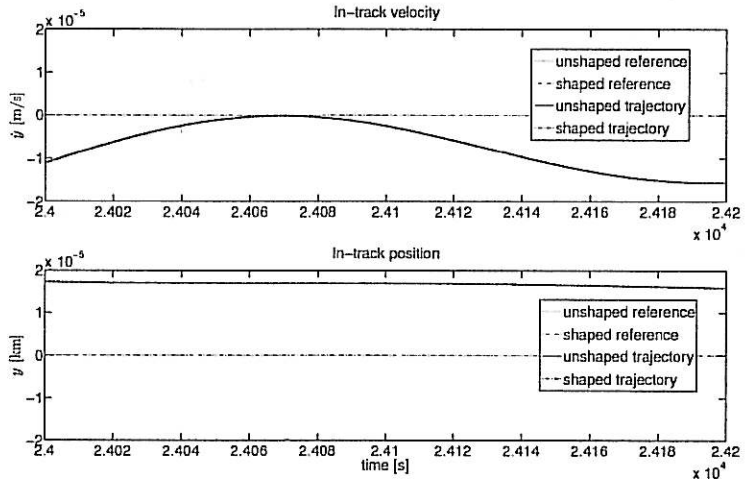

Figure 6.8: Shaped and unshaped references and in-track states at the end of the manoeuvre.

and so the source of the vibrations becomes smaller and the time delay can be safely decreased. The first option is more general, though.

The rendezvous manoeuvre from Chapter 5 was maintained here just to emphasise the differences between the different configurations for each simulation. Other simulations with different initial and final states were performed, and they produced similar results in the aspects illustrated throughout this work. This next section also makes use of the same rendezvous optimal trajectory planned with the LP formulated in Section 5.1.1. It extends the development made in this section to the nonlinear system explained at the end of Section 5.2.1.

6.1.2 Vibration Damping for Nonlinear Actuators

As Singer simply put [26, 10]: "no general statement can be made regarding the application of the new technique [input shaping] to nonlinear systems since each nonlinearity poses unique problems". In other words, the systematic development done here with posicast control or some other pole-zero cancelling technique is no longer valid for nonlinear systems. Mathematical rigour needs to be replaced by numerical techniques.

Some articles offer more rigorous treatments for nonlinear systems of the on-off type like the one treated here [7]. They make use of bang-bang or bang-off-bang control to find minimum time or minimum fuel, respectively, and extend it with the pole-zero cancelling method to mitigate the vibrations. This offers a very interesting solution if the system performs satisfactorily under a restricted number of actuator switches – in bang-bang/bang-off-bang control the designer has to define a specific number of switches that

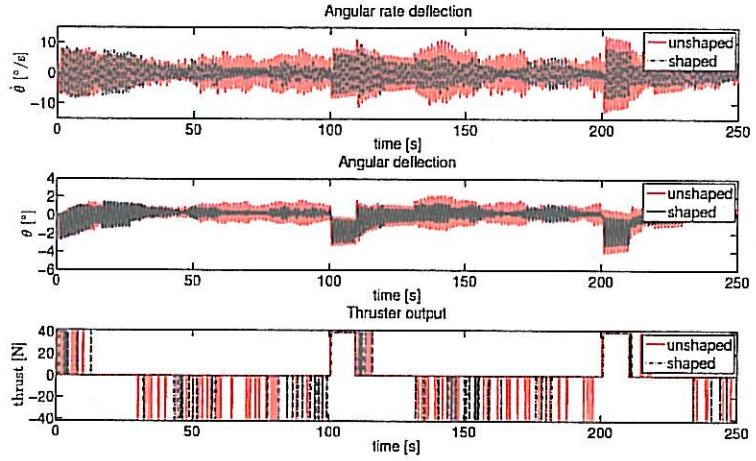

Figure 6.9: Angular deflection in the beginning of the manoeuvre with a nonlinear on-off actuator.

the actuator will perform for each change of reference.

What if it simply works? By using the same parameters from Table 6.3 for the posicast shaper the system performs substantially better. For the same initial 250 s, the plots for the linear actuator in Figures 6.1.1 and 6.5, the oscillations are reduced and the controller shifts significantly less, as the plots in Figure 6.9 show. In fact, there is almost a 10% decrease in the fuel consumption in the satellite following the shaped reference. Certainly, this result does not reflect an effective gain in this system since the LQR controller was projected for a linear actuator and fine tuning it would probably produce less expressive results. But this shows that input shaping can potentially have interesting side effects other than just smoothing the undesirable oscillations.

Another aspect from this type of nonlinearity [on-off], is that the oscillations are now much more prominent. This is because of the shape of the actuator output that enters the system. The hard edged, squared-like, thrust bursts that come from the actuator excites several modes from the system. The thrust pulses can be decomposed as a sum of sine waves of frequency ω, 3ω, 5ω, ..., where ω is the frequency of the pulse; or, more generally, the inverse of the pulse width. With the several pulses that are seen from Figure 6.9, it is very likely that one of the decomposed frequencies will match (or be very close to) those of the oscillating modes of the system. Hence, vibration can be much more inconvenient when dealing with step and square wave inputs.

For the tracking performance, the system with shaped reference had

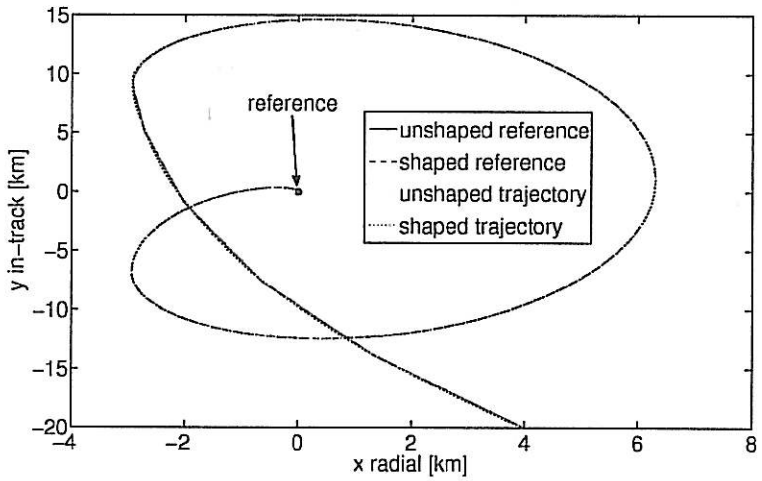

Figure 6.10: Relative motion with on-off actuators.

no significant degradation when compared to the unshaped system (Figure 6.10).

The plots in Figure 6.10 are in the order of kilometres so it is quite hard to notice the differences between the shaped and unshaped trajectories. A reconfiguration of a passive aperture, described by (4.23), is presented in Figure 6.11, and the detail of the final resulting orbit is detailed in Figure 6.12. The shaped and unshaped references practically overlap on this final configuration, the shaped reference having a slightly smaller diameter compared to the unshaped. This was already expected, from the results presented by Biediger [3], and can be seen in Figure 6.13.

6.1.3 Final Considerations

Input shaping (IS), though easily implemented for linear systems, is not as elegant for nonlinear systems. Unless the designer is willing to sacrifice the flexibility of an LQR (or operationally similar controller) by using a bang-off-bang control design, with a restricted number of actuator commands, an input shaper for nonlinear systems would have to be designed for each different manoeuvre.

The small delay caused by the shaper can be decreased dynamically. In that way, the shaped trajectory reaches the end of the manoeuvre with the same states of the unshaped trajectory. The effects of using a dynamic delay and the benefits of IS for different manoeuvres can be seen in Table 6.6. It compares:

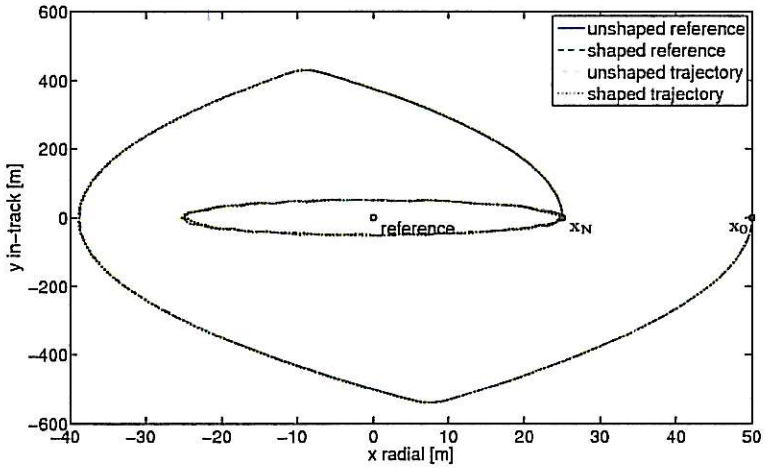

Figure 6.11: Reconfiguration of a passive aperture with $\mathbf{x_0} = [0\ 50\ -100n\ 0]^T$ and $\mathbf{x_N} = [0\ 25\ -50n\ 0]^T$.

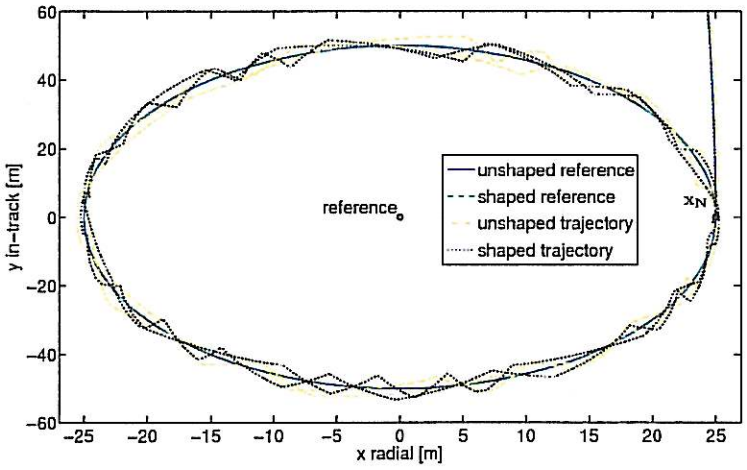

Figure 6.12: Detail of the final aperture after the reconfiguration.

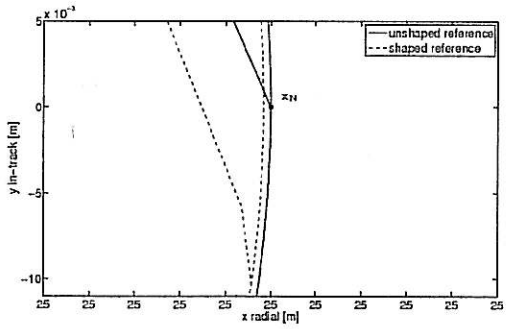

Figure 6.13: Radial error for the shaped reference induced by the shaper's delay. The two lines on the left are the final calculated trajectories approaching the final position before entering the new passive aperture configuration. The remaining lines on the right are the final relative motion passive trajectories. The shaped radius is slightly smaller than the unshaped, on the order of 10^{-5}.

- the *quadratic mean* vibration, given by

$$\dot{\theta}_{RMS} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\dot{\theta}_i^2}, \tag{6.12}$$

  with N the total number of steps for the simulation, and i the i^{th} step of the simulation;

- the *root mean squared error* (RMSE) of the satellite positioning (x, y), given by

$$e_{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\sqrt{(x_i^* - x_i)^2 + (y_i^* - y_i)^2}}, \tag{6.13}$$

  with (x_i^*, y_i^*) the optimally determined positions and (x_i, y_i) the simulated positions; and

- the fuel consumption, in terms of total velocity shifts $\sum \Delta v$, throughout the manoeuvre;

for several unshaped and shaped trajectories, with and without dynamically decreasing the delay of the shaper.

The manoeuvres parameters, initial state $\mathbf{x_0}$ and final state $\mathbf{x_N}$, and the final state error $\mathbf{e_N}$ is displayed in Table 6.4, for simulations with a linear actuator, and Table 6.5, for simulations with nonlinear actuators. The length for all manoeuvres is the same – 10 min – and the initial and

man. #	x_0	x_N	rigid sat. e_N	rigid sat. Δv
L01	[0 50 -0.052019 0]	[0 25 -0.02601 0]	[-0.060065 -13.442 7.9928 -57.839]	3.0666
L02	[-0.052019 0 0 50]	[-0.02601 0 0 25]	[0.057455 36.97 -21.849 55.262]	8.2589
L03	[0.052019 0 0 -50]	[0.02601 0 0 -25]	[-0.057455 -36.97 21.849 -55.262]	8.2589
L04	[0 100 -0.10404 0]	[0 50 -0.02601 0]	[-0.10189 -25.818 15.346 -98.121]	6.0596
L05	[0 15 -3 15]	[0 0 0 0]	[0.059284 12.132 -6.6225 55.713]	2.8048
L06	[0 0 0 300]	[0 0 0 30]	[-2.1411 70.749 -36.411 -2064.7]	37.401
L07	[0 0 0 -300]	[0 0 0 -30]	[2.1411 -70.749 36.411 2064.7]	37.401
L08	[-1 5 -1 5]	[1 -5 1 -5]	[-2.5362 1559.8 -820.32 -2848.6]	176.76
L09	[0 0 10 1000]	[1 -3 0 500]	[-3.6065 638.52 -580.75 -3692.2]	128.01
L10	[10 1000 0 0]	[0 500 1 -3]	[3.3789 -9187 4572.8 5655.1]	902.42

man. #	unshaped e_N	shaped (static) e_N	shaped (dynamic) e_N	rig. e_{RMSE}
L01	[-0.060066 -13.442 7.9927 -57.84]	[-0.062584 -13.419 7.9873 -60.258]	[-0.060844 -13.396 7.9746 -58.571]	182.34
L02	[0.057458 36.97 -21.849 55.265]	[0.064335 36.947 -21.855 61.879]	[0.059244 36.834 -21.799 56.941]	499.4
L03	[-0.057458 -36.97 21.849 -55.265]	[-0.064335 -36.947 21.855 -61.879]	[-0.059244 -36.834 21.799 -56.941]	499.4
L04	[-0.1019 -25.818 15.345 -98.124]	[-0.10672 -25.778 15.339 -102.76]	[-0.10328 -25.724 15.308 -99.425]	363
L05	[0.059285 12.132 -6.6225 55.714]	[0.061181 12.13 -6.6235 57.54]	[0.059821 12.103 -6.6137 56.222]	161.17
L06	[-2.1411 70.748 -36.41 -2064.7]	[-2.1295 71.395 -36.813 -2053.5]	[-2.149 70.338 -36.293 -2072.5]	2353.9
L07	[2.1411 -70.748 36.41 2064.7]	[2.1295 -71.395 36.813 2053.5]	[2.149 -70.338 36.293 2072.5]	2353.9
L08	[-2.5362 1559.8 -820.32 -2848.6]	[-2.2772 1560.9 -821.42 -2599.3]	[-2.5397 1558.7 -820.48 -2853.2]	10300
L09	[-3.6065 638.52 -580.75 -3692.2]	[-3.4223 639.81 -581.59 -3515]	[-3.609 637.62 -580.8 -3695.6]	7459.5
L10	[3.3789 -9187 4572.8 5655.1]	[1.9395 -9188.3 4577 4270.3]	[3.3669 -9181.9 4574 5650.3]	52742

Table 6.4: Sample manoeuvres and its final state error for linear actuators

man. #	x_0	x_N	rigid sat. e_N	rig. Δv
N01	[0 50 -0.052019 0]	[0 25 -0.02601 0]	[0.003085 0.18846 0.093305 2.9498]	1.308
N02	[-0.052019 0 0 50]	[-0.02601 0 0 25]	[-0.0081234 0.11145 -0.47984 -7.808]	3.8
N03	[0.052019 0 0 -50]	[0.02601 0 0 -25]	[0.0081234 -0.11145 0.47984 7.808]	3.8
N04	[0 100 -0.10404 0]	[0 50 -0.02601 0]	[0.0060502 -0.066204 0.56202 5.8395]	2.488
N05	[0 15 -3 15]	[0 0 0 0]	[-0.0024639 -0.45028 -0.26122 -2.2993]	1.058
N06	[0 0 0 300]	[0 0 0 30]	[26.426 767.29 56.882 25634]	23.952
N07	[0 0 0 -300]	[0 0 0 -30]	[-26.426 -767.29 -56.882 -25634]	23.952
N08	[-1 5 -1 5]	[1 -5 1 -5]	[-60.036 -19185 43.829 -57014]	23.982
N09	[0 0 10 1000]	[1 -3 0 500]	[-50.097 -18874 82.917 -46787]	23.978
N10	[10 1000 0 0]	[0 500 1 -3]	[82.793 17678 64.013 87965]	23.992
man. #	**unshaped e_N**	**shaped (static) e_N**	**shaped (dynamic) e_N**	**rig. e_{RMSE}**
N01	[0.003715 -0.0063201 0.38151 3.592]	[0.00308 0.0022967 0.80948 2.984]	[0.0035333 -0.096914 0.64674 3.4538]	2.5081
N02	[-0.00364 0.03329 -0.02891 -3.482]	[-0.004113 0.10979 -0.30855 -3.976]	[-0.0041165 0.05036 -0.075421 -4.015]	5.1743
N03	[0.00364 -0.03329 0.02891 3.482]	[0.004113 -0.10979 0.30855 3.976]	[0.0041165 -0.05036 0.075421 4.015]	5.1743
N04	[0.003926 0.018043 0.083665 3.787]	[0.0045127 -0.0014334 0.4595 4.370]	[0.0044571 0.03816 0.17716 4.3368]	3.1766
N05	[-0.003537 0.0201 -0.37937 -3.438]	[-0.002713 -0.18348 -0.36654 -2.596]	[-0.003218 -0.10105 -0.1065 -3.1137]	2.3148
N06	[26.426 767.3 56.884 25634]	[26.406 764.31 56.947 25615]	[26.409 755.89 57.204 25619]	9162.1
N07	[-26.426 -767.3 -56.884 -25634]	[-26.406 -764.31 -56.947 -25615]	[-26.409 -755.89 -57.204 -25619]	9162.1
N08	[-60.036 -19185 43.829 -57014]	[-60.068 -19179 43.604 -57048]	[-60.03 -19173 43.715 -57011]	57008
N09	[-50.097 -18874 82.917 -46787]	[-50.108 -18854 82.666 -46804]	[-50.087 -18851 82.72 -46784]	47842
N10	[82.793 17678 64.012 87965]	[82.867 17681 64.374 88036]	[82.811 17674 64.162 87983]	1.7498e+05

Table 6.5: Sample manoeuvres and its final state error for nonlinear actuators

man. #	unshap. Δv	$\dot{\theta}_{RMS}$	e_{RMSE}	shaped Δv	(sta. $\dot{\theta}_{RMS}$	del.) e_{RMSE}	shaped Δv	(dyn. $\dot{\theta}_{RMS}$	del.) e_{RMSE}
L01	3.0666e+00	1.1983e-04	1.8234e+02	3.0662e+00	1.1874e-04	1.8259e+02	3.0650e+00	1.1935e-04	1.8248e+02
L02	8.2589e+00	3.2032e-04	4.9940e+02	8.2582e+00	3.1966e-04	5.0006e+02	8.2537e+00	3.2002e-04	4.9980e+02
L03	8.2589e+00	3.2032e-04	4.9940e+02	8.2582e+00	3.1966e-04	5.0006e+02	8.2537e+00	3.2002e-04	4.9980e+02
L04	6.0596c+00	2.3290c-04	3.6300c+02	6.0591c+00	2.3229c-04	3.6346c+02	6.0563c+00	2.3259c-04	3.6328c+02
L05	2.8048e+00	1.0440e-04	1.6117e+02	2.8055e+00	1.0394e-04	1.6140e+02	2.8043e+00	1.0420e-04	1.6130e+02
L06	3.7402e+01	1.1104e-03	2.3539e+03	3.7380e+01	1.1098e-03	2.3543e+03	3.7467e+01	1.1120e-03	2.3599e+03
L07	3.7402e+01	1.1104e-03	2.3539e+03	3.7380e+01	1.1098e-03	2.3543e+03	3.7467e+01	1.1120e-03	2.3599e+03
L08	1.7676e+02	5.2500e-03	1.0300e+04	1.7669e+02	5.2036e-03	1.0323e+04	1.7690e+02	5.2531e-03	1.0322e+04
L09	1.2801e+02	3.7938e-03	7.4595e+03	1.2797e+02	3.7622e-03	7.4758e+03	1.2812e+02	3.7967e-03	7.4767e+03
L10	9.0242e+02	2.7150e-02	5.2743e+04	9.0216e+02	2.6864e-02	5.2883e+04	9.0314e+02	2.7163e-02	5.2851e+04
man. #	unshap. Δv	$\dot{\theta}_{RMS}$	e_{RMSE}	shaped Δv	(sta. $\dot{\theta}_{RMS}$	del.) e_{RMSE}	shaped Δv	(dyn. $\dot{\theta}_{RMS}$	del.) e_{RMSE}
N01	1.2640e+00	6.6519e-02	2.7867e+00	1.2700e+00	5.4924e-02	1.8264e+00	1.2520e+00	5.9105e-02	2.2592e+00
N02	3.8060e+00	7.1513e-02	5.5980e+00	3.6900e+00	6.3338e-02	5.0258e+00	3.6620e+00	6.5912e-02	5.3358e+00
N03	3.8060e+00	7.1513e-02	5.5980e+00	3.6900e+00	6.3338e-02	5.0258e+00	3.6620e+00	6.5912e-02	5.3358e+00
N04	2.5800e+00	6.3999e-02	3.3203e+00	2.4560e+00	5.8837e-02	2.3072e+00	2.4900e+00	5.3357e-02	3.1067e+00
N05	1.0500e+00	4.9250e-02	2.6942e+00	1.0320e+00	3.4973e-02	1.9673e+00	1.0560e+00	4.5985e-02	2.4086e+00
N06	2.3952e+01	7.2556e-02	9.1621e+03	2.3942e+01	7.2427e-02	9.1548e+03	2.3944e+01	7.3049e-02	9.1565e+03
N07	2.3952e+01	7.2556e-02	9.1621e+03	2.3942e+01	7.2427e-02	9.1548e+03	2.3944e+01	7.3049e-02	9.1565e+03
N08	2.3982e+01	5.6244e-02	5.7008e+04	2.3974e+01	5.6264e-02	5.7012e+04	2.3972e+01	5.6327e-02	5.6998e+04
N09	2.3978e+01	5.6196e-02	4.7842e+04	2.3964e+01	5.6186e-02	4.7830e+04	2.3964e+01	5.6355e-02	4.7822e+04
N10	2.3992e+01	5.6237e-02	1.7498e+05	2.3990e+01	5.6302e-02	1.7500e+05	2.3990e+01	5.6394e-02	1.7499e+05

Table 6.6: States and vibration comparison between shaped and unshaped trajectories

s/c #	symbol	$[\dot{x}_0\ x_0\ \dot{y}_0\ y_0]$	$[\dot{x}_N\ x_N\ \dot{y}_N\ y_N]$
I	$\triangle$	[0 50 -100n 0]	[-25n 0 0 -50]
II	$\triangleleft$	[-50n 0 0 -100]	[0 -25 50n 0]
III	$\triangledown$	[0 -50 100n 0]	[100n 0 0 50]
IV	$\triangleright$	[50n 0 0 100]	[0 25 -50n 0]

Table 6.7: Initial and final states in a cluster reconfiguration with a quarter of an orbital period for the manoeuvre

final states were selected at will, just to provide some variety and establish some conclusions.

The next section investigates the effects of vibrations presented so far for a satellite cluster regarding their relative positions.

6.2 Multiple Satellites in Formation

To keep the trajectories of the independent satellites in a more comprehensive fashion, I will consider the satellite model first proposed in Chapter 5, with actuators in both x and y directions. The flexible mode in considered only in the y direction, as described in Section 6.1.

I will consider now a reconfiguration of a cluster with four satellites. They are required to decrease the diameter of their passive aperture, their original and final positions listed in Table 6.7. The free force motion of their original positions and their calculated trajectories are displayed in Figure 6.14. The spacecrafts move clockwise around the reference point, and their initial positions are indicate by the symbols in Table 6.7.

Another manoeuvre, more fuel efficient, would make use of the orbital dynamics and give the cluster a full orbital period for the reconfiguration. This manoeuvre can be seen in Figure 6.15, and the initial and final (desired) states are listed in Table 6.8. The fourth column, the desired final state, is half the initial state displayed on the third column, of the initial states. The necessary velocity shifts, Δv, are reduced by over 95% when compared to the same manoeuvre if given only a quarter of an orbital period.. Just for some insight on the importance of how much time the spacecrafts are given to reach the desired states, Table 6.9 lists the normalised fuel costs for the same initial and final states from Table 6.8 with different periods to perform the manoeuvre.

6.2.1 Effects of Flexibility in the Relative Distances

The relative distance between the satellites, similar to (4.18) is given by

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{6.14}$$

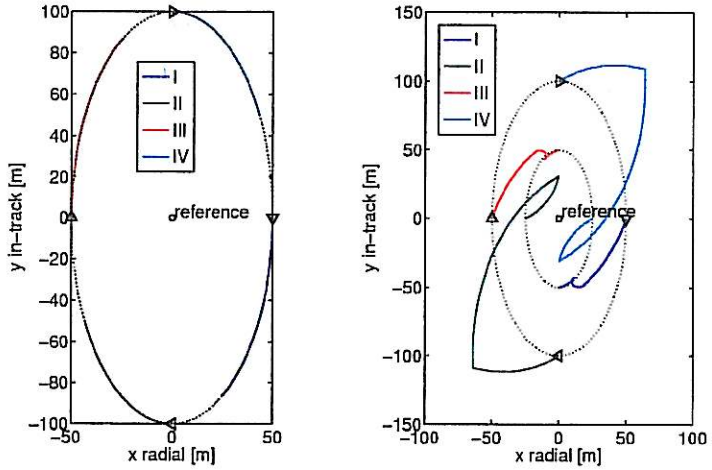

Figure 6.14: Free force motion and optimal trajectories for a four satellite cluster reconfiguration. For the free force, the satellites motion were simulated for 3000 s, or 50 min. For the minimum fuel trajectory planning the spacecrafts were given a quarter of an orbital period to reach their new orbits.

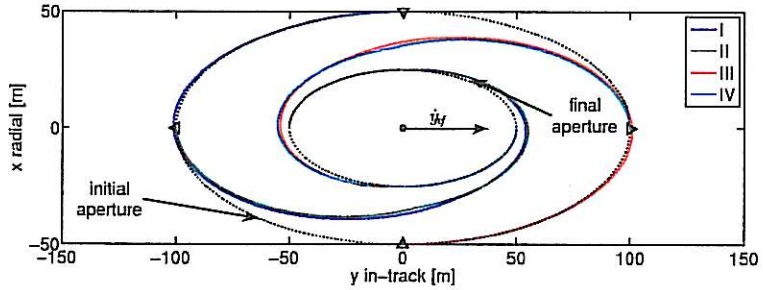

Figure 6.15: Optimal trajectory for the four satellite cluster reconfiguration with a full orbital period. The radial and in-track axes are shifted in this figure, so the centre of the reference orbit is downwards and the reference point is moving to the right on the inertial coordinate plane ($\dot{y}_{ref}$).

s/c #	symbol	$[\dot{x}_0\ x_0\ \dot{y}_0\ y_0]$	$[\dot{x}_N\ x_N\ \dot{y}_N\ y_N]$
I	$\triangle$	[0 50 -100n 0]	[0 25 -50n 0]
II	$\triangleleft$	[-50n 0 0 -100]	[-25n 0 0 -50]
III	$\triangledown$	[0 -50 100n 0]	[0 -25 50n 0]
IV	$\triangleright$	[50n 0 0 100]	[25n 0 0 50]

Table 6.8: Initial and final states in a cluster reconfiguration with a full orbital period

duration (orbital period)	fuel cost, s/c I & III (normalised Δv)	fuel cost, s/c II & IV (normalised Δv)
1/8	240	416
1/4	78.9	62.6
1/2	18.7	7.52
1	1.00	1.00
2	0.960	0.762
4	0.960	0.762
8	0.960	0.762

Table 6.9: Normalised fuel costs for different trajectory durations

with i and j the satellite index ($d_{ij} = 0$ for $i = j$, the same satellite). Since this simulation considers four satellites, the distances make a combination of six possibilities. Keeping track of those distances is paramount for a formation flying mission – the cooperation of each satellite in the cluster is more dependent on their relative positions than on their absolute (inertial) positions. This section investigates the effects of vibration on the four satellite cluster just presented.

The distances between the satellites are defined as in Figure 6.16. Figure 6.17 shows the distances between satellite I and the other three satellites – d_{12}, d_{13} and d_{14}. The unshaped (optimal) and the shaped desired (reference) trajectories overlap in this graph, making them indistinguishable.

A comparison of the performance for this manoeuvre for shaped and unshaped trajectories is presented in Tables 6.10, 6.12 and 6.13: Table 6.10 lists the Δv cost for the length of the orbital shift, one orbital period of 1207.9 s; Table 6.12 lists the quadratic mean vibration, given by; Table 6.13 lists the *root mean squared error* (RMSE), given by

$$e_{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(d_{ij}^{*} - d_{ij})^2}, \tag{6.15}$$

with d_{ij}^{*} the desired (from the estimated optimal trajectories) distance and d_{ij} the simulated distance.

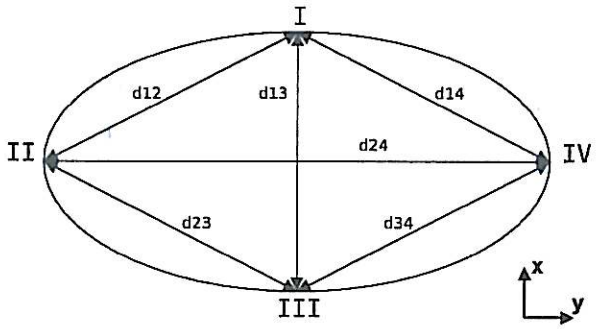

Figure 6.16: Definition of the relative distances between the four satellites.

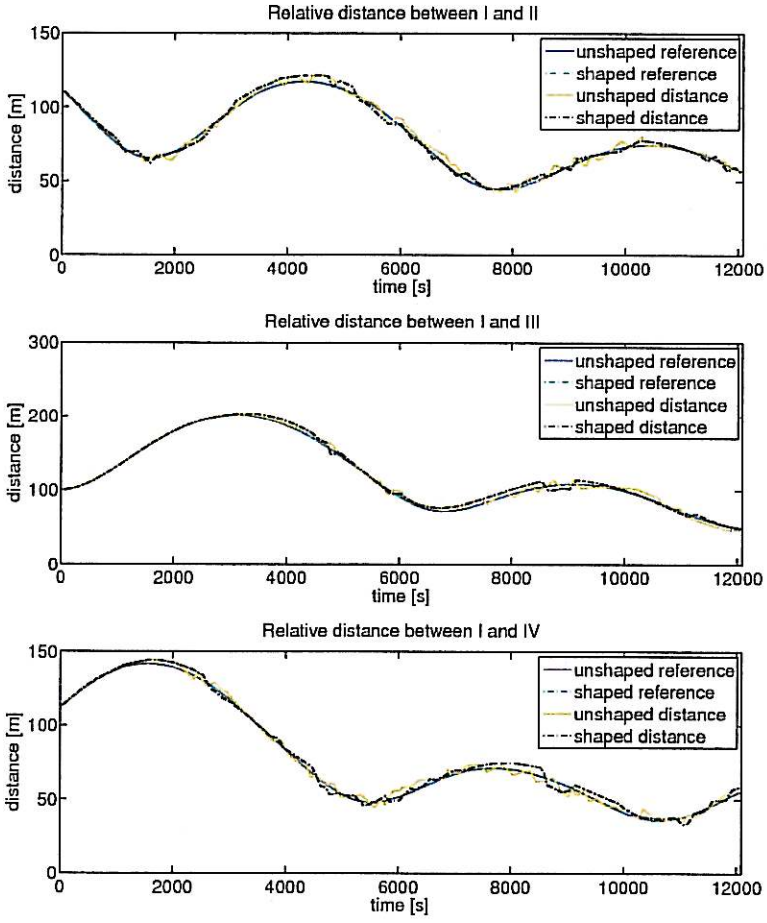

Figure 6.17: Simulated distances between satellite I and satellites II, III and IV.

satellite	I	II	III	IV
unshaped	1.8113	2.8679	1.8113	2.8679
shaped	0.4528	0.1509	0.4528	0.1509

Table 6.10: Total Δv cost, in m/s, for cluster reconfiguration with a nonlinear actuator

satellite	I	II	III	IV
unshaped	0.0495	0.0847	0.0495	0.0847
shaped	0.0519	0.0848	0.0519	0.0848

Table 6.11: Total Δv cost, in m/s, for cluster reconfiguration with a linear actuator

satellite	I	II	III	IV
unshaped	0.5520	0.6257	0.5520	0.6257
shaped	0.3934	0.4260	0.3934	0.4260

Table 6.12: Quadratic mean vibration (in terms of $\dot{\theta}_{rms}$, in ^{o}/s) for the four satellites during the manoeuvre

segment	d_{12}	d_{13}	d_{14}	d_{23}	d_{24}	d_{34}
unshaped	0.0098	0.0137	0.0093	0.0093	0.0148	0.0098
shaped	0.0094	0.0144	0.0107	0.0107	0.0146	0.0094

Table 6.13: Root mean squared error of the simulated trajectories

The results from Table 6.10 look quite promising. One must have in mind, though, that those are exaggerations of a not fully analysed nonlinear system. In fact, in the linear system, disregarding the actuator constraints, the results are slightly worse in terms of fuel savings when using IS, as seen from Table 6.11, and there are no significant improvements in reducing the vibration. That certainly changes for different manoeuvres, as already seen in Table 6.6, and also changes for different Kalman gains in the LQR controller.

Chapter 7

Conclusion and Future Prospects

I am yet to see an engineering work that is absolutely complete. There is always room for improvements. A compromise between time, costs and goals will decide when to conclude a work. I end this task by summarising the main points I wanted to show.

Vibration in spacecraft is often overlooked. The simple simulations in Chapter 2 already show that flexibility can be quite a problem in satellite control. Input shaping is a cheap, and yet powerful, technique to mitigate vibration. It could even represent savings in fuel, as the results in Chapter 6 show. However it would be precipitated to conclude this without further research. However, from my understanding, the energy that is dissipated in the flexible appendages needs a source: the fuel. A first proposal for a future work would be to investigate this more carefully and also experimentally.

This document addressed several topics that could be researched more in-depth, individually or in conjunction with another subject. Hopefully, this document provided some brief guidelines in:

- flexible spacecraft modelling;
- input shaping theory;
- formation flying theory;
- linear programming theory; and
- effects of flexibility on orbital manoeuvres and formation flying.

The latter, I believe, was inconclusive. To properly investigate the effects of vibration in formation flying a few more items should be considered:

- the positioning sensing system (with some knowledge of its noise characteristics);

- the complete propulsion system; and
- the attitude dynamics.

This complete model could lead to some important conclusions for formation flying strategies. My goal, considering the time span of this project, was to give some initial insights on this matter.

[11] N. C. Singer and W. P. Seering, "Preshaping command inputs to reduce system vibration," *Journal of Dynamic Systems, Measurement and Control*, vol. 112, pp. 76–82, March 1990.

[12] G. Franklin, J. Powell, and A. Emami-Naeini, *Feedback Control of Dynamic Systems (5th Edition)*. Prentice Hall, 2005.

[13] M. Guelman, "Spacecraft formation flying," lecture notes for the spacemaster course, Faculty of Aerospace Engineering, Asher Space Research Institute, Technion, Haifa, Israel, January 2006.

[14] E. S. Agency, "Science and technology webpage."

[15] S. A. Schweighart, "Development and analysis of a high fidelity linearized j_2 model for satellite formation flying," Master's thesis, Department of Aeronautical and Astronautical Engineering, Massachusetts Institute of Technology, June 2001.

[16] K. T. Alfriend, H. Schaub, and D.-W. Gim, "Gravitational perturbations, nonlinearity and circular orbit assumption effects on formation flying control strategies," Tech. Rep. AAS 00-012, American Institute of Aeronautics and Astronautics, February 2000.

[17] R. Alonso, J. L. Crassidis, and J. L. Junkins, "Vision-based relative navigation for formation flying of spacecraft," *American Institute of Aeronautics and Astronautics*, 2000.

[18] A. C. Tribble, *The Space Environment: Implications for Spacecraft Design*. Princeton University Press, 2003.

[19] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[20] C.-T. Chen, *Linear System Theory and Design (Oxford Series in Electrical & Computer Engineering)*. Oxford University Press Inc, USA, 1998.

[21] F. L. Lewis and V. Syrmos, *Optimal Control*. John Wiley & Sons Inc, 1995.

[22] V. Lappas, C. Saaj, D. Richie, M. Peck, B. Streetamn, and H. Schaub, "Spacecraft formation flying and reconfiguration with electrostatic forces," *American Institute of Aeronautics and Astronautics*, January-February 2007.

[23] W. Singhose, S. Derezinski, and N. Singer, "Extra-insensitive input shapers for controlling flexible spacecraft," *Journal of Guidance Control and Dynamics*, vol. 19, pp. 385–391, March-April 1996.

Bibliography

[1] A. E. Bryson, Jr, *Control of Spacecraft and Aircraft.* Princeton University Press, 1994.

[2] M. J. Doherty and R. H. Tolson, "Input shaping to reduce solar array structural vibrations," Tech. Rep. NASA/CR-1998-208698, Joint Institute for Advancement of Flight Sciences, Langley Research Center, Hampton, Virginia, August 1998.

[3] E. A. O. Biediger, *Vibration Reduction Using Command Generation in Formation Flying Satellites.* PhD thesis, School of Mechanical Engineering, Georgia Institute of Technology, March 2005.

[4] M. J. Tillerson, "Coordination and control of multiple spacecraft using convex optimization techinques," Master's thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, June 2002.

[5] S. S. Ge, T. H. Lee, and C. J. Harris, *Adaptive Neural Network Control of Robotic Manipulators (World Scientific Series in Robotics & Intelligent Systems).* World Scientific Publishing, 1999.

[6] M. J. Sidi, *Spacecraft Dynamics and Control: A Practical Engineering Approach.* Cambridge University Press, 1997.

[7] T. Singh and W. Singhose, "Tutorial on input shaping/time delay control of maneuvering flexible structures," in *Proceedings of the American Control Conference*, pp. 1717–1731, May 2002.

[8] O. J. M. Smith, "Posicast control of damped oscillatory systems," in *Proceedings of the IRE*, vol. 45, no. 9, pp. 1249–1255, September 1957.

[9] O. Smith, *Feedback Control Systems- McGraw-Hill Series in Control Systems Engineering.* McGraw Hill, 1958.

[10] N. C. Singer and W. P. Seering, "Preshaping command inputs to reduce system vibration," Tech. Rep. A.I.Memo No. 1027, Massachusetts Institute of Technology, January 1988.

[24] W. E. Singhose, L. J. Porter, and N. C. Singer, "Vibration reduction using multi-hump extra-insensitive input shapers," in *American Control Conference*, 1995.

[25] W. Singhose, N. Singer, and W. Seering, "Comparison of command shaping methods for reducing residual vibration," in *Proceedings of the 1995 European Control Conference*, 1995.

[26] N. C. Singer, *Residual Vibration Reduction in Computer Controlled Machines.* PhD thesis, Massachusetts Institute of Technology, January 1989.