POLITECNICO DI TORINO

MASTER OF SCIENCE IN MECHATRONIC ENGINEERING

Master Degree Thesis

SPACECRAFT AUTONOMOUS GUIDANCE AND CONTROL

Supervisors:
Prof. Carlo Novara
Ing. Luca Massotti

Candidate:
Mattia Boggio

July 2019

Abstract

This thesis aims to develop an autonomous guidance and control for Earth monitoring missions, based on the Nonlinear Model Predictive Control (NMPC) approach, alternative to the standard planning designed on ground. As case study, the scenario in which a spacecraft (S/C) has to overfly a given point located on the Earth surface, when the on-board software is triggered by an alarm, is considered. The satellite must guarantee a fast revisit and persistent monitoring of the event and, when the alert is over, the return to the initial orbit. As Earth observation mission, the ESA Sentinel-2 one is studied. Firstly, the strategies for the autonomous mission planning are developed. Their goal is to produce feasible trajectories for guiding the S/C exactly on the coordinate in which the alert is occurred. Then, the NMPC is designed in order to generate a control input which forces the S/C to follow the reference trajectories, finding a suitable trade-off between the time to perform the maneuvers and the propellant consumption. Constraints on the maximum thrust supplied by the engines and on the S/C orbital dynamics are taken into account explicitly in the problem. The obtained results are compared with the ideal strategy, which consists in instantaneous and impulsive ΔV applied at the orbital node, and with the standard mission planning, in which the ideal situation is approximated with a consequent efficiency degradation. The results show that the NMPC approach can achieve better performances than the manual case and guarantee nearly the same optimality level as the ideal impulsive strategy, presenting however several advantages with respect to this situation. In particular, NMPC can allow a significantly higher level of autonomy and offer more flexibility and adaptation capability with respect to traditional approaches. Further improvements, especially on the reduction of fuel consumption, can be done through the development of NMPC with low-thrust policies.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In the past years, autonomous vehicles have become increasingly important in a wide variety of engineering fields, especially in the automotive sector. Recently, the concept of autonomy is developing as an open problem also in the context of advanced space missions. The increase of interest is due to the possibility of using the autonomous guidance in a wide range of space applications, such as low-thrust orbital transfer, orbit phasing, station keeping, deep space correction maneuvers and rendezvous. Traditionally, the whole mission planning is carried out on ground by means of the classical orbital mechanics methods (e.g. Lambert's problem solution, direct and indirect orbit optimization, etc.), designing a-priori a strict schedule of maneuvers to be followed by the S/C during the various phases of the mission. A significant advance in this context is represented by the autonomous mission planning, guidance and control approach. The aim is to obtain unmanned systems with the ability to plan their motion trajectories without communications with ground control stations, i.e. systems with the capability of accomplishing missions on its own. This can be done by the development of a real-time autonomous trajectory planner able to generate, through the optimization of some performance indexes, feasible trajectories which enable vehicle to reach a desired final destination within a specified time, respecting limitations of on-board control system capabilities and vehicle performances.

However the implementation of practical methods for space mission autonomous planning raises a number of challenging ranging from extended agency and long-term planning to dynamic recoveries and robust concurrent execution, all in the presence of tight real-time deadlines, sharp maneuvers, changing goals, scarce resource constraints, and a wide variety of possible failures. From this point of view, NMPC appears to be a very promising approach. Indeed, it can provide in an unified way optimal guidance strategies and the control laws also for complex MIMO nonlinear systems. Moreover, it is able to efficiently manage nonlinear and/or nonconvex constraints and handle the most common types of propulsion in a systematic way. Examples of real applications in which Model Predictive Control (MPC) - either linear or nonlinear - has already been used are: controlling the S/C during rendezvous/docking maneuvers [1], station keeping [2], orbit phasing [3] and deep space TCM [4].

In the last few years, many space programs (i.e. GOCE, GRACE, Sentinels, etc...) have been focused on monitoring the Earth in order to observe (from far above) events and phenomena which cannot be studied in-situ on the surface of the planet [5]. These kinds of missions, in the near future, will require the design of S/C which, from a parking orbit, are able to quickly overfly, with a narrow tolerance, any point on the Earth surface, after suitable alarms triggered by on-board computer (OBC) units or from some ground station [6], [7]. In order to accomplish such a mission profile, many complex and expensive plane-change orbital maneuvers could be required. Hence, the necessity arises of planning a mission profile optimized in terms of time required to perform a given maneuver and propellant consumption, in order to extend the mission duration as long as possible.

For these reasons, the thesis project has been focused on the development of an autonomous guidance and control for an Earth observation satellite. In particular, as case study the Copernicus Sentinel-2 mission has been considered.

1.1 Copernicus Sentinel-2 mission

1.1.1 Copernicus programme

Copernicus is a program for Earth observation, coordinated and managed by the European Commission (EC) in partnership with the European Space Agency (ESA). Its aim is to provide an accurate, timely and easily information in the domain of environment and civil security [8].

The priority applications of this mission, shown in Figure 1.1, are:

- ATMOSPHERE MONITORING, which is to provide information on air quality and chemical composition of the atmosphere.
- MARINE ENVIRONMENT MONITORING, which is to provide information on the state and dynamics of physical ocean and marine ecosystem.
- LAND MONITORING, which is to provide information on land use and land cover, cryosphere, climate change and bio-geophysical variables.
- CLIMATE CHANGE MONITORING, which is to provide information on climate variables, climate analyses and projections and indicator relevant to adaptation and mitigation strategies.
- EMERGENCY MANAGEMENT, which is to provide information for emergency response in relation to different types of disasters.
- SECURITY SERVICE, which is to provide information in support of the civil security in Europe.

Figure 1.1. Copernicus mission

For the achievement of all the operations defined in the Copernicus programme, ESA has developed a family of satellites, called **Sentinels** [9]. Each Sentinel is used for a very specific aim among those defined above; but what characterizes each of them is to provide high-resolution radar and optical images of our planet.

1.1.2 Sentinel-2 mission

Sentinel-2 is a wide-swath, high-resolution, multi-spectral imaging mission, supporting Copernicus Land Monitoring studies, including the monitoring of vegetation, soil and water cover, as well as observation of inland waterways and coastal areas. As presented in Figure 1.2, this mission is a constellation of two twin satellites (Sentinel-2A and Sentinel-2B) placed in the same orbit, phased at 180° to each other.

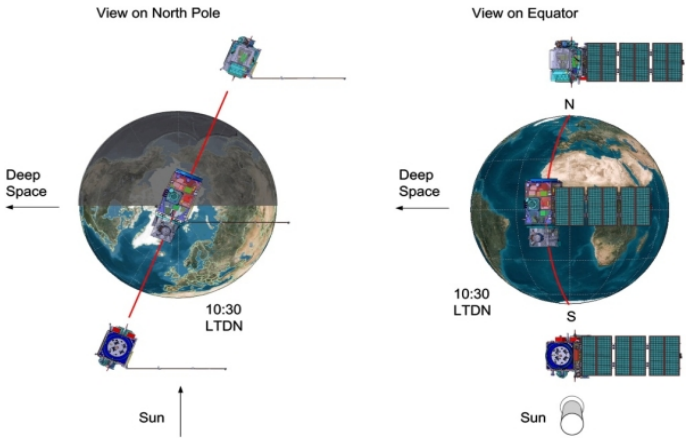

Figure 1.2. The Twin-Satellite SENTINEL-2

Mission objectives

Sentinel-2 provides a wide range of applications related to Earth's land and coastal water. The mission offers information for agricultural and forestry practices and for helping manage food security. Besides, satellite images are used to determine various plant indices such as leaf area chlorophyll and water content indexes. This is particularly important for effective yield prediction and applications related to Earth's vegetation.
As well as monitoring plant growth, Sentinel-2 can be used to map changes in land cover and to monitor the world's forests. It also provide information on pollution in lakes and coastal waters. Images of floods, volcanic eruptions and landslides contribute to disaster mapping and help humanitarian relief efforts. The aim is to provide, with Sentinel-2 data, continuity to multi-spectral imagery provided by the SPOT series of satellites and the USGS LANDSAT Thematic Mapper instrument.

So, the main applications of this mission are the monitoring of:

- land cover change;
- global crop;
- forest and vegetation;
- marine environmental;
- inland water;
- glaciers, ice extent and snow cover.

These applications ensure that Sentinel-2 makes a significant contribution to Copernicus themes.

MultiSpectral Instrument

The MultiSpectral Instrument (MSI) uses a push-broom concept. A push-broom sensor works by collecting rows of image data across the orbital swath and utilises the forward motion of the spacecraft along the path of the orbit to provide new rows for acquisition. The average period of observation over land and coastal areas is approximately 17 minutes and the maximum period of observation is 32 minutes. The Sentinel MSI sample 13 spectral bands: four bands at 10 metres, six bands at 20 metres and three bands at 60 metres spatial resolution. Only the GRI (GLOBAL REFERENCE IMAGE, map with landmarks) is completed at HR between 30-40 cm.

A detailed model of Sentinel-2, with the representation of the satellite platform and the communication links, is shown in Figure 1.3.

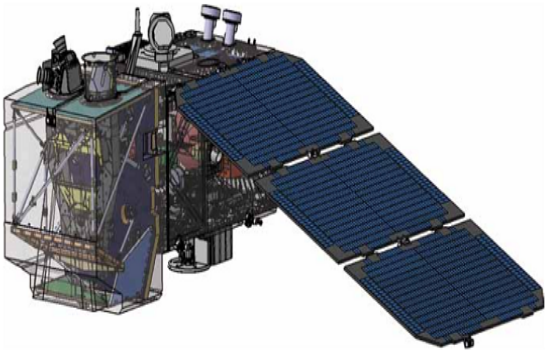

Figure 1.3. SENTINEL-2 model

Orbit characteristics

The Sentinel-2 mission orbit is **sun-synchronous**. A sun-synchronous orbit (SSO) is a nearly polar orbit around a planet, in which the satellite passes over any given point of the planet's surface at the same local mean solar time. In this way the solar illumination on the planet's surface is the same for every revolution; this condition helps the observation, since the light conditions remain approximately unchanged for each orbit.

Sentinel-2A and Sentinel-2B occupy the same orbit, but separated by 180°. Table 1.1 contains the main orbital informations for the two twin Sentinel.

Altitude (km)	Inclination (deg)	Period (min)	Cycle (days)	Ground-track deviation (km)	Local Time at Descending Node (hours)
786	98.62	100.6	10	+-2	10:30

Table 1.1. Orbit characteristics

The value of Local Time was chosen as a compromise between a suitable level of solar illumination and the minimisation of potential cloud cover. The Local Time value is close to the local overpass time of LANDSAT and almost identical to that of SPOT-5, permitting the integration of Sentinel-2 data with existing and historical missions, and contributing to long-term time series data collection.

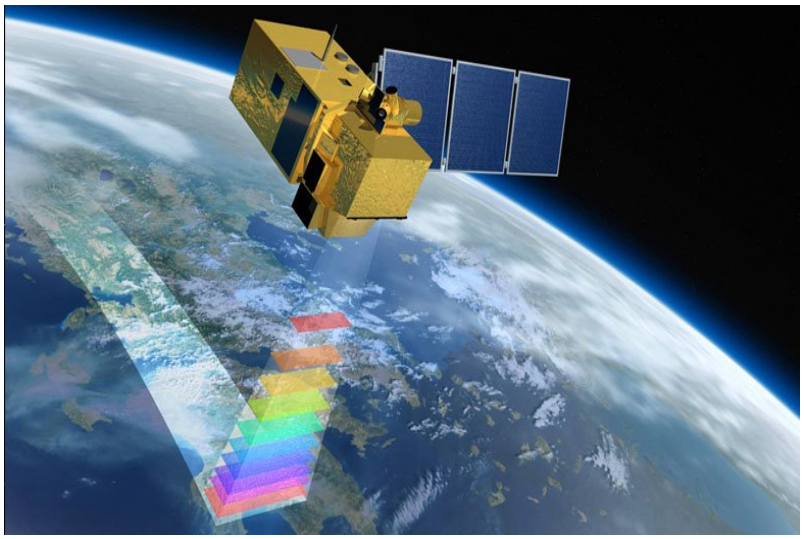

Figure 1.4. SENTINEL-2 field of view

Other features of Sentinel-2 orbit are:

- systematic global coverage of land surfaces from 56° S to 84° N, coastal waters, and all of the Mediterranean Sea;
- revisiting every 5 days under the same viewing angles. At high latitudes, Sentinel-2 swath overlap and some regions will be observed twice or more every 5 days, but with different viewing angle;
- 290 km field of view (reported in Figure 1.4).

1.2 Objectives and outline of the thesis

The aim of this thesis, developed in collaboration with **ESA-ESTEC**, is to implement an autonomous guidance alternative to the standard mission planning made on ground (GS) via the FOS. The problem has been addressed through the following steps:

- build a simulator in MATLAB reproducing the orbits of the Sentinel A;
- simulate alerts on the on-board software, which trigger the switch from remote/GS to autonomous mission planning;
- using Nonlinear Model Predictive Control (NMPC), simulate how the nominal mission changes for:
  - a fast revisit of the event;
  - a persistent monitor of the event;
  - the return to the nominal mission plan.

The thesis is organized in six chapter, including the current Chapter 1, that introduces the context in which the autonomous mission planning is developed.

Chapter 2 deals with the orbital dynamics required to define the motion of Sentinel-2A in its orbit.

Chapter 3 shows an overview of the Nonlinear Model Predictive Control (NMPC) approach used to realize the autonomous planning.

Chapter 4 describes the mission scenario considered and the strategies developed to implement the trajectory planner.

Chapter 5 presents the results obtained by applying to the NMPC the desired trajectory, computed through the planner, and compares the performances between autonomous and standard mission approaches.

Chapter 6 reports some concluding considerations about the thesis work and proposes possible future developments.

Chapter 2

Orbital dynamics

2.1 Basic physical principles

Orbital mechanics, as applied to artificial spacecraft, is based on celestial mechanics. The founders of this mechanics were Newton and Kepler. Indeed, Kepler provided three basic empirical laws that describe motion in unperturbed planetary orbits, while Newton formulated the more general physical laws governing the motion of a planet.

2.1.1 Kepler's laws

Kepler provided three empirical laws for planetary motion (Figure 2.1), based on Brahe's planetary observation:

1. The orbit of a planet is an ellipse with the sun located at one focus.
2. The radius vector drawn from the sun to a planet sweeps out equal areas in equal time intervals, that is the areal velocity is constant.
3. The squares of the periods of revolution of the planets are proportional to the cubes of the semimajor axis of their orbits.

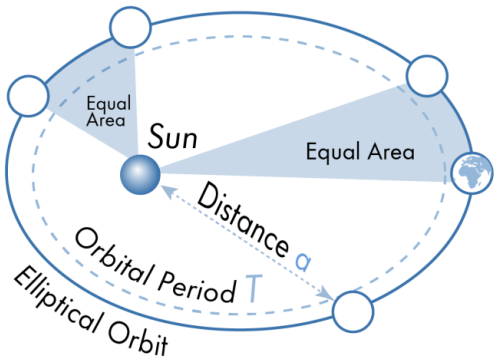

Figure 2.1. Kepler's laws of planetary motion

2.1.2 Newton's laws

Newton provided three laws of mechanics and one for gravitational attraction, formulated as follows:

1. A particle remains at rest or continues to move at a constant velocity, unless acted upon by an external force.

2. The rate of change of linear momentum of a body is equal to the force **F** applied on the body, where $\mathbf{p} = m\mathbf{v}$ is the linear momentum and

$$\mathbf{F} = \frac{d\mathbf{p}}{dt} = \frac{d(m\mathbf{v})}{dt}. \tag{2.1}$$

   In this equation, m is the mass of the body and **v** is the velocity vector.
   For a constant mass, this law takes the simplified form

$$\mathbf{F} = m\mathbf{a} \tag{2.2}$$

   where $\mathbf{a} = d\mathbf{v}/dt$ is the linear acceleration.

3. For any force $\mathbf{F}_{12}$ exerted by a particle 1 on a particle 2, there exists a force

$$\mathbf{F}_{21} = -\mathbf{F}_{12} \tag{2.3}$$

   exerted by particle 2 on particle 1.

4. Any two particles attract each other with a force

$$\mathbf{F} = \frac{Gm_1m_2\mathbf{r}}{r^3} \tag{2.4}$$

   where **r** is a vector of magnitude $r = |\mathbf{r}|$ along the line connecting the two particles with masses m_1 and m_2, and $G = 6.67 \times 10^{-11} Nm^2/kg^2$ is the universal constant of gravitation.

2.2 The two-body problem

As defined in [10], the two-body problem is an idealized situation in which, through the equation (2.4), the motion of two point particles that interact only with each other is determined. It is used to describe, for example, a satellite orbiting a planet, a planet orbiting a star and two stars orbiting each other (a binary star).

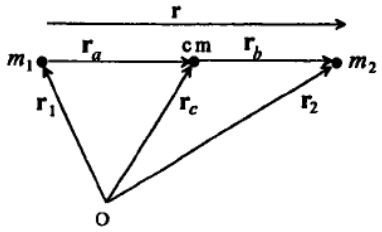

Figure 2.2. General two body problem

In Figure 2.2, m_2 exerts an attraction force $\mathbf{F}_1 = m_1\ddot{\mathbf{r}}_1$ on m_1, and m_1 exerts a force $\mathbf{F}_2 = m_2\ddot{\mathbf{r}}_2$ on m_2. The Newton's II law and gravity law yield the following equations :

$$\mathbf{F}_1 = m_1\ddot{\mathbf{r}}_1 = Gm_1m_2\frac{\mathbf{r}_2 - \mathbf{r}_1}{|\mathbf{r}_2 - \mathbf{r}_1|^3} \tag{2.5}$$

$$\mathbf{F}_2 = m_2\ddot{\mathbf{r}}_2 = Gm_1m_2\frac{\mathbf{r}_1 - \mathbf{r}_2}{|\mathbf{r}_1 - \mathbf{r}_2|^3} = -\mathbf{F}_1 \tag{2.6}$$

From (2.5) and (2.6), we obtain

$$\mathbf{r}_2 - \mathbf{r}_1 = -G(m_1 + m_2)\frac{\mathbf{r}_2 - \mathbf{r}_1}{r^3} \tag{2.7}$$

and, since $\mathbf{r} = \mathbf{r}_2 - \mathbf{r}_1$

$$\ddot{\mathbf{r}} + G(m_1 + m_2)\frac{\mathbf{r}}{r^3} = 0. \tag{2.8}$$

Equation (2.8) is the basic equation of motion for the two-body problem.
One of the main properties of this system can be derived from the definition of the center of mass (which from this moment will be indicated with *cm*).
The *cm* of the two-body system can be found from the equation $\sum m_j\mathbf{r}_j = 0$; it follows that $\mathbf{r}_a m_1 - \mathbf{r}_b m_2 = 0$. In Figure 2.2, $\mathbf{r}_c$ is the radius vector from the origin of the reference frame to the *cm* of the two-body system, and $\mathbf{r}_a$ and $\mathbf{r}_b$ are (respectively) the distance of m_1 and m_2 from the *cm*. It can be observed that $\mathbf{r}_a = \mathbf{r}_c - \mathbf{r}_1$ and $\mathbf{r}_b = \mathbf{r}_2 - \mathbf{r}_c$ or, equivalently, that $m_1(\mathbf{r}_c - \mathbf{r}_1) - m_2(\mathbf{r}_2 - \mathbf{r}_c) = 0$. Hence

$$\mathbf{r}_c(m_1 + m_2) = m_1\mathbf{r}_1 + m_2\mathbf{r}_2 \tag{2.9}$$

After two differentiations of (2.9) with time, and taking in consideration (2.5) and (2.6), we find that

$$\ddot{\mathbf{r}}_c = 0; \quad \dot{\mathbf{r}}_c = \text{const.} \tag{2.10}$$

The last equation means that although the *cm* in not accelerated, the system can be in rectilinear motion with constant velocity. Furthermore the center of mass can be chosen as the origin of an inertial frame.

2.2.1 Free restricted two-body equation

For $m_1 \gg m_2$ the equation (2.8):

$$\ddot{\mathbf{r}} + \mu \frac{\mathbf{r}}{r^3} = 0 \tag{FR2B}$$

where $\mu = Gm_1$ is the *gravitational parameter*. The (FR2B) is called the **free restricted two-body equation**.

The system defined by (FR2B) has the following properties:

1. the total mechanical energy is conserved;
2. the angular momentum is conserved;
3. the free response of the (FR2B) equation occurs on a plane.

The demonstrations of these properties, discussed in the next paragraphs, are taken from [11].

Energy conservation

Take the dot product of (FR2B) with $\dot{\mathbf{r}}$:

$$\begin{aligned} \ddot{\mathbf{r}} \cdot \dot{\mathbf{r}} + \frac{\mu}{r^3} \mathbf{r} \cdot \dot{\mathbf{r}} &= \frac{1}{2} \frac{d}{dt} (\dot{\mathbf{r}} \cdot \dot{\mathbf{r}}) + \frac{\mu}{2r^3} \frac{d}{dt} (\mathbf{r} \cdot \mathbf{r}) \\ = \frac{d}{dt} \frac{\dot{r}^2}{2} + \frac{\mu}{2r^3} \frac{d}{dt} r^2 &= \frac{d}{dt} \frac{\dot{r}^2}{2} + \frac{\mu \dot{r}}{r^2} = \frac{d}{dt} \Big(\frac{\dot{r}^2}{2} - \frac{\mu}{r} \Big) = 0. \end{aligned} \tag{2.11}$$

This proves the principle of energy conservation:

$$\dot{\mathcal{E}} = 0, \qquad \mathcal{E} = \text{const} \tag{2.12}$$

Indeed, $\mathcal{E} = \left(\frac{\dot{r}^2}{2} - \frac{\mu}{r} \right) = \left(\frac{v^2}{2} - \frac{\mu}{r} \right)$ is the total (mechanical) energy per unit mass, where:

- $\frac{v^2}{2}$ is the kinetic energy per unit mass;
- $-\frac{\mu}{r}$ is the potential energy per unit mass.

Then, for a given (constant) total energy $\mathcal{E}$, the corresponding orbital velocity is

$$v = \sqrt{2\mu/r + 2\mathcal{E}}. \tag{2.13}$$

Angular momentum conservation and planar motion

Take the cross product of $\mathbf{r}$ with (FR2B):

$$\mathbf{r} \times \ddot{\mathbf{r}} + \frac{\mu}{r^3} \mathbf{r} \times \mathbf{r} = \mathbf{r} \times \ddot{\mathbf{r}} = \dot{\mathbf{r}} \times \dot{\mathbf{r}} + \mathbf{r} \times \ddot{\mathbf{r}} = \frac{d}{dt} (\mathbf{r} \times \dot{\mathbf{r}}) = 0. \tag{2.14}$$

This proves the principle of angular momentum conservation:

$$\dot{\mathbf{h}} = 0, \qquad \mathbf{h} = \text{const} \tag{2.15}$$

Indeed, $\mathbf{h} = \mathbf{r} \times \dot{\mathbf{r}} = \mathbf{r} \times \mathbf{v}$ is the angular momentum per unit mass.
An important implication of $\mathbf{h}$ being constant is that $\mathbf{r}$ and $\mathbf{v}$ always remain in the same plane, called the **orbital plane**.

Orbit equation

With (FR2B) equation a geometric description of the Keplerian orbits can be derived. Take the cross product of (FR2B) with **h**:

$$\left(\ddot{\mathbf{r}} + \frac{\mu}{r^3}\mathbf{r}\right) \times \mathbf{h} = \frac{d}{dt}\left(\dot{\mathbf{r}} \times \mathbf{h} - \frac{\mu}{r}\mathbf{r}\right) = 0. \tag{2.16}$$

The above equation shows that

$$\dot{\mathbf{r}} \times \mathbf{h} - \frac{\mu}{r}\mathbf{r} = \text{const} = \mu\mathbf{e} \tag{2.17}$$

where **e** is the *eccentricity vector* and $e = |\mathbf{e}|$ is the *eccentricity*.

Take the dot product of **r** with (2.17):

$$\mathbf{r} \cdot (\dot{\mathbf{r}} \times \mathbf{h}) - \frac{\mu}{r}\mathbf{r} \cdot \mathbf{r} = \mu\mathbf{r} \cdot \mathbf{e}. \tag{2.18}$$

Knowing that $\mathbf{r} \cdot (\dot{\mathbf{r}} \times \mathbf{h})$ can be rewritten as $(\mathbf{r} \times \dot{\mathbf{r}}) \cdot \mathbf{h}$ and that $(\mathbf{r} \times \dot{\mathbf{r}}) \cdot \mathbf{h} = \mathbf{h} \cdot \mathbf{h} = h^2$, then (2.18) becomes

$$h^2 - \mu\, r = \mu\, r\, e\, \cos\theta \tag{2.19}$$

where θ (the angle between **r** and **e**) is called *true anomaly*.

Expliciting with respect to r and defining $p = h^2/\mu$ (p is called the *parameter* or *semi-latus rectum*), the **orbit equation** is obtained:

$$r = \frac{p}{1 + e\, \cos\theta}. \tag{ORE}$$

2.3 Orbit geometry

The (ORE) is the equation of a conical section. As shown in Figure 2.3, this is the general orbit equation from which different kinds of orbits evolve, namely: circular, elliptic, parabolic and hyperbolic. Motion under a central force results in orbits that are one of these conical sections.

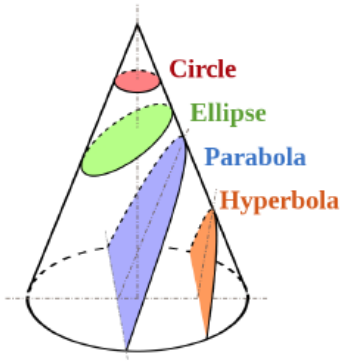

Figure 2.3. Conic sections

2.3.1 Analysis of Keplerian orbit

Starting from (ORE) equation, depending on the e value, the following conic sections are obtained:

1. circle, for $e = 0$;
2. ellipse, for $0 \leq e \leq 1$;
3. parabola, for $e = 1$;
4. hyperbola, for $e > 1$.

Each conic section, described in detail below on the basis on [12], has the following properties:

- the origin is located at one focus;
- θ is measured from the point on the conic nearest the focus;
- p determines the size;
- e determines the shape.

Circular Orbits

Setting $e = 0$ in the (ORE) equation, yields

$$r = p = \frac{h^2}{\mu} = \text{const} \tag{2.20}$$

Since $\dot{r} = 0$, it follows that $v = v_\perp$, that is the velocity of the body is perpendicular to the radius vector $\mathbf{r}$; so, the angular momentum formula $h = rv_\perp$ becomes simply $h = rv$. Substituting this expression for h into (2.20) and solving for v yields the velocity of a circular orbits

$$v = \sqrt{\frac{\mu}{r}} = \text{const} \tag{2.21}$$

The energy is then $\mathcal{E} = \mu^2/2h^2 = -\mu/2r < 0$. It means that as the radius goes up, the energy becomes less negative, i.e., it increases; in other words, the higher the orbit, the greater its energy.

Elliptic Orbits

If $0 < e < 1$, then the denominator of (ORE) varies with true anomaly θ, but it remains positive, never becoming zero. The point on the ellipse at $\theta = 0°$ (point P in Figure 2.4) is called the *periapsis*, and the radius vector from the prime focus F of the ellipse to the periapsis is the minimum radius vector from the focus to any other point on the ellipse. Its value, found from the (ORE), is

$$r_p = \frac{p}{1+e}. \tag{2.22}$$

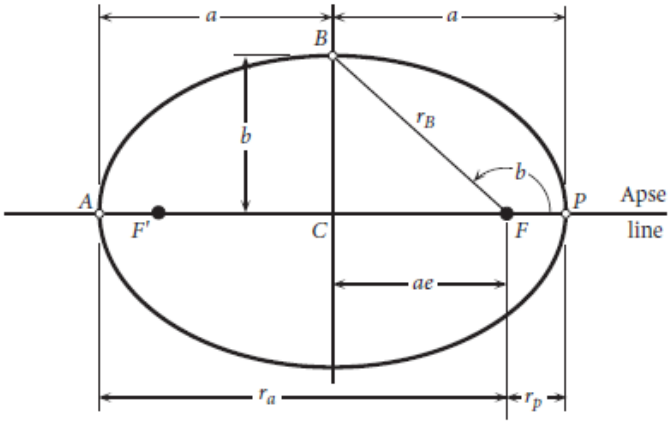

Figure 2.4. Elliptical orbit

For orbits around the Earth, which is considered to be located at the prime focus F, the periapsis is called *perigee*; r_p is the perigee distance from the focus. For orbits around the Sun, the periapsis is called *perihelion*.
The point on the ellipse (point A in Figure 2.4) with the maximum distance from the focus located at F is obtained by considering $\theta = 180°$; the value of this distance is:

$$r_a = \frac{p}{1-e} \tag{2.23}$$

Point A is called the *apoapsis* and r_a denotes the apoapsis radius vector. The apoapsis of an elliptic orbit in the solar system is called *aphelion*, while the apoapsis of an earth-orbiting spacecraft is called *apogee*.
Let $2a$ be the distance measured along the apse line from periapsis P to apoapsis A, as illustrated in 2.4. Then:

$$2a = r_p + r_a \tag{2.24}$$

Substituting (2.22) and (2.23) in (2.24), we get:

$$a = \frac{p}{1-e^2} \tag{2.25}$$

a is the semimajor axis of the ellipse. Solving (2.25) for p and putting the result into (ORE), yields an alternative form of the orbit equation:

$$r = a\frac{1-e^2}{1+e\cos\theta} \tag{2.26}$$

The center C of the ellipse is the point lying midway between the apoapsis and periapsis. The distance CF from C to F is

$$CF = a - FP = a - r_p = \frac{p}{1-e^2} - \frac{p}{1-e} = \underbrace{\frac{p}{1-e^2}}_{a} e = ae \tag{2.27}$$

Let B the point on the orbit which lies directly above C, on the perpendicular bisector of AP. The distance b from C to B is the semiminor axis. If the true anomaly of point B is β, then according to (2.25), the radial coordinate of B is

$$rb = a\frac{1-e^2}{1+e\cos\beta} \tag{2.28}$$

Starting from the above equation and knowing that the projection of r_B onto the apse line is ae, an interesting property of the ellipse can be found, that is $r_B = a$.
This property can be used to find the semiminor axis in terms of the semimajor axis and the eccentricity of the ellipse; indeed, according to the Pythagorean theorem,

$$b^2 = r_B^2 - (ae)^2 = a^2 - a^2e^2 \tag{2.29}$$

then:

$$b = a\sqrt{1-e^2} \tag{2.30}$$

Regarding the total energy, it is negative and equal to

$$\mathcal{E} = \frac{v^2}{2} - \frac{\mu}{r} = ... = -\frac{\mu}{2a} \tag{2.31}$$

while the velocity is

$$v = \sqrt{2\frac{\mu}{r} - \frac{\mu}{a}}. \tag{2.32}$$

The last equation is also known as **vis-visa equation**.

Parabolic Orbits

If the eccentricity equals 1, then the (ORE) equation becomes

$$r = \frac{p}{1+\cos\theta} \tag{2.33}$$

As the true anomaly θ approaches 180°, the denominator approaches zero, so that r tends towards infinity and consequently also the semimajor axis a.
According to (2.31), since $a \to \infty$, the energy of a trajectory for which $e = 1$ is zero, so that for a parabolic trajectory the conservation of energy is:

$$\frac{v^2}{2} - \frac{\mu}{r} = 0 \tag{2.34}$$

In other words, the speed anywhere on a parabolic path is:

$$v = \sqrt{\frac{2\mu}{r}} \tag{2.35}$$

The velocity in the above equation is the **escape velocity** necessary to leave the parabolic orbit around the central body located at the focus F. It is interesting also to notice that this escape velocity is larger by a factor of only $\sqrt{2}$ than the velocity of a circular orbit at the same distance r from F. This means that, to escape from a circular orbit requires a velocity boost of 41.4 percent.

Hyperbolic Orbits

If $e > 1$, the (ORE) formula describes the geometry of hyperbola shown in Figure 2.5.

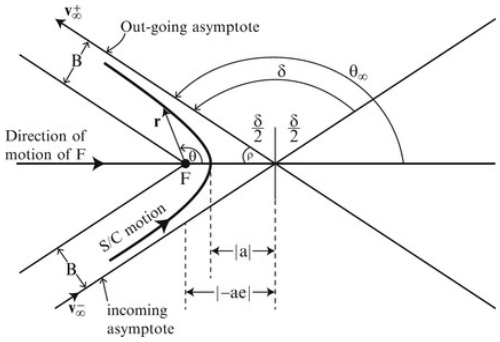

Figure 2.5. Hyperbolic trajectory

For this class of orbits, the total energy is positive and equal to

$$\mathcal{E} = \frac{\mu}{2a}. \tag{2.36}$$

This means that the kinetic energy of the satellite is larger than its potential energy, so that it is able to leave the gravitational attraction field of the central body. In other words a body passing close to a moving planet is subject to a velocity increase, without being captured by the planet gravity. This hyperbolic orbit feature is called **hyperbolic passage** and it is adopted in some interplanetary voyages, for example in US probe Mariner 10 mission (1973) to fly-by Venus and Mercury.
The conservation of energy for a hyperbolic trajectory is

$$\frac{v^2}{2} - \frac{\mu}{r} = \frac{\mu}{2a}. \tag{2.37}$$

According to (2.37), it is possible to find v_∞, that is the speed at which a body on a hyperbolic path arrives at infinity (i.e. $r \to \infty$)

$$v_\infty = \sqrt{\frac{\mu}{a}}. \tag{2.38}$$

This velocity is called *hyperbolic excess speed*.
Another relevant quantity is the *true anomaly of the asymptote* θ_∞; this is the true anomaly that goes to zero the (ORE), that is

$$\theta_\infty = \cos^{-1}(-1/e). \tag{2.39}$$

2.4 Reference frames

There are different types of reference systems used for the study of the motion of satellites and, more generally, of celestial bodies. The most common for the analysis of orbits around the Earth, are:

1. local vertical local horizontal frame (LVLH);
2. local orbit frame (LORF);
3. perifocal (perigee) frame (PF);
4. geocentric equatorial (GE) frame.

Local vertical local horizontal frame

It is a non-inertial reference system since, as shown in Figure 2.6, its origin is located directly on the satellite (P1) and then it is subject to accelerated motion. The unit vectors of this reference frame, are:

- $\mathbf{l}_3$ (local vertical), defined along the direction $P_0 \rightarrow P_1$ on the orbit plane;
- $\mathbf{l}_1$ (local horizontal), perpendicular to $\mathbf{l}_3$ on the orbit plane and sign concordant with the orbital velocity;
- $\mathbf{l}_2 = \mathbf{l}_3 \times \mathbf{l}_1$ (orbit pole), perpendicular to the orbit plane.

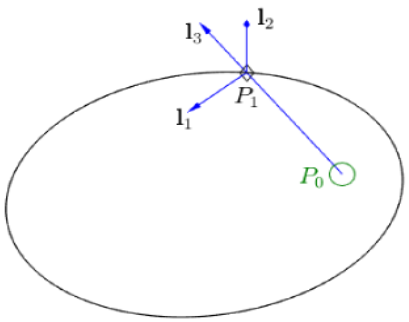

Figure 2.6. LVLH frame

Local orbit frame

Also for this reference frame the origin (P1) is placed on the satellite and then it is non-inertial. The unit vectors that define it, are:

- $\mathbf{o}_1$, instantaneous normalized velocity on the orbit plane and tangent to the orbit;
- $\mathbf{o}_2 = \mathbf{l}_2$, perpendicular to the orbit plane;
- $\mathbf{o}_3 = \mathbf{o}_1 \times \mathbf{o}_2$, on the orbit plane.

Perifocal frame

One of the most suitable reference system to describe the satellite motion is the Perifocal reference frame (Figure 2.7). It is an inertial reference system, since its origin is located in the center of the Earth (i.e. in one of the two fochi of the elliptic orbit) and then it moves with uniform rectilinear motion. The unit vectors that define this reference frame, are:

- $\hat{\mathbf{p}} = \mathbf{e}/e$, eccentricity unit vector passing through the periapis on the orbit plane;
- $\hat{\mathbf{w}} = \mathbf{o}_2 = \mathbf{l}_2$ (orbit pole), perpendicular to the orbit plane and concordant with momentum vector $\mathbf{h}$ of the satellite;
- $\hat{\mathbf{q}} = \hat{\mathbf{w}} \times \hat{\mathbf{p}}$, on the orbit plane.

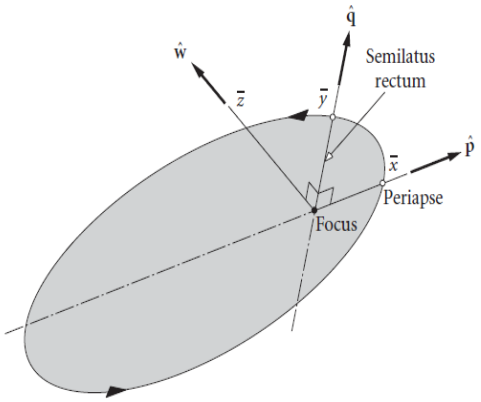

Figure 2.7. Perifocal frame

Geocentric equatorial frame

Also for this reference frame the origin is placed in the center of the Earth and then it is inertial. As reported in Figure 2.8, the fundamental plane is the equatorial frame and the unit vector that define it, are:

- $\hat{\mathbf{I}}$, defined along the vernal equinox direction, that is, the place where the Sun crosses the celestial equator in a northward direction in its annual apparent circuit around the ecliptic;
- $\hat{\mathbf{K}}$, defined along the north polar axis;
- $\hat{\mathbf{J}} = \hat{\mathbf{K}} \times \hat{\mathbf{I}}$, on the equatorial plane.

Note that this reference system does not rotate with the Earth, but remains relatively fixed against the background stars; furthermore it is independent on the spacecraft orbit.

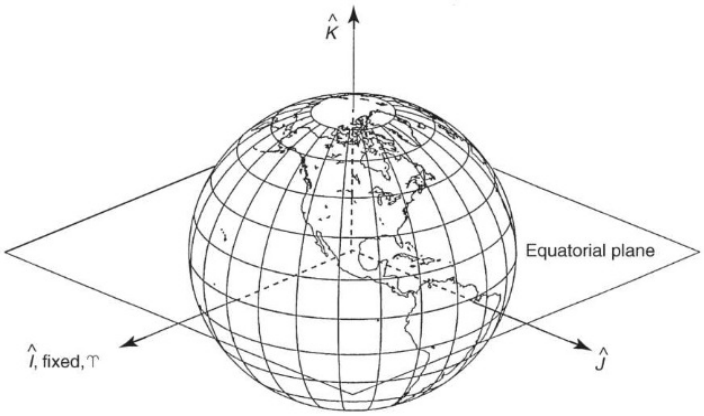

Figure 2.8. GE frame

2.5 Orbital elements

Once the main reference frames have been defined, it is possible to represent the satellite's motion with respect to them.
To completely define this motion, six scalar quantities are required, such as the three components of the velocity (**v**) and the position (**r**) vectors of the satellite at a certain instant of time in a three-dimensional Cartesian reference system. Alternatively, it is possible to use another set of six elements in order to uniquely identify both the shape and orientation of the orbit and the position of the satellite on it.
Respect to a Geocentric-Equatorial reference frame, these last quantities are called **orbital elements**. These elements, shown in Figure 2.9, are

- **semimajor axis** a: describes the size of orbit ellipse;
- **eccentricity** e: describes the shape of orbit ellipse;
- **inclination** i: the angle between the orbit plane and Earth's equatorial plane;
- **right ascension of ascending node** Ω: the angle from the vernal equinox to the ascending node (intersection between the orbital and the equatorial plane);
- **argument of perigee** ω: the angle from the ascending node to the eccentricity vector measured in the direction of satellite's motion;
- **true anomaly** $\nu \equiv \theta$: indicates the position of the satellite in its orbit.

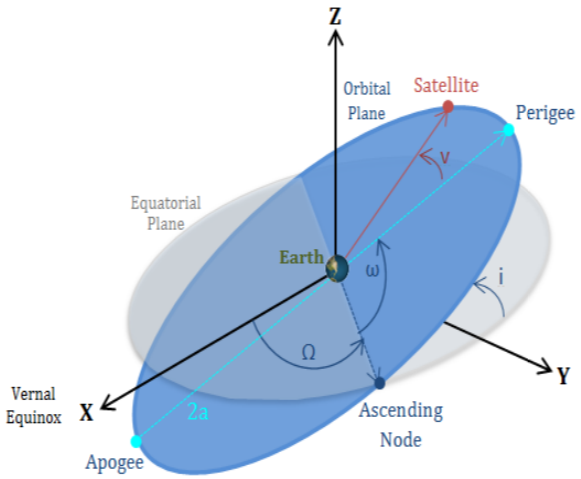

Figure 2.9. Orbital elements definition

This set of orbital elements can be divided into two groups: the dimensional elements and the orientation elements. The dimensional elements specify the size and the shape of the orbit and relate the position of spacecraft in the orbit to time; they are: semimajor axis, eccentricity and the true anomaly. The orientation elements specify the orientations of the orbit in space, which are: inclination, right ascension of ascending node and argument of perigee.
The quantities necessary to describe the satellite's motion, i.e. the velocity and the position vectors and the orbital elements, are related to each other; indeed, knowing one of these two sets, it is possible to derive the other on the basis of procedures described in [13] and reported in subsections 2.5.1 and 2.5.2.

2.5.1 Computation of the orbital elements based on r and v

Once position and velocity of the satellite, at a given instant of time, is assigned, the orbital elements can be derived. The step

$$(\mathbf{r}, \mathbf{v}) \rightarrow \{a, e, i, \Omega, \omega, \nu\}$$

can be performed using the following procedure:

1. Exploiting the conservation of mechanical energy $\mathcal{E}$, expressed by the equation (2.37):

$$\mathcal{E} = -\frac{\mu}{2a}$$

the value of the semimajor axis a is obtained:

$$a = -\frac{\mu}{2\mathcal{E}} \tag{2.40}$$

2. From (2.18), the eccentricity e is determined as a function of the module of the angular momentum:
$$e = \left| \frac{1}{\mu} \mathbf{v} \times \mathbf{h} - \frac{\mathbf{r}}{r} \right| \tag{2.41}$$

3. The value of the inclination i is obtained by determining the angle between the z axis of the Geocentric-Equatorial reference frame (whose direction is defined by the $\hat{\mathbf{K}}$ vector) and the direction of the $\mathbf{h}$ vector:
$$\cos i = \frac{\hat{\mathbf{K}} \cdot \mathbf{h}}{h} \quad \Rightarrow \quad i = \arccos\left(\frac{\hat{\mathbf{K}} \cdot \mathbf{h}}{h} \right) \tag{2.42}$$

4. The right ascension of ascending node Ω is derived through the following expression:
$$\cos \Omega = \hat{\mathbf{I}} \cdot \hat{\mathbf{n}} \quad \Rightarrow \quad \Omega = \begin{cases} \arccos(\hat{\mathbf{I}} \cdot \hat{\mathbf{n}}) & \text{if} \quad \hat{\mathbf{n}} \cdot \hat{\mathbf{J}} > 0 \\ 2\pi - \arccos(\hat{\mathbf{I}} \cdot \hat{\mathbf{n}}) & \text{if} \quad \hat{\mathbf{n}} \cdot \hat{\mathbf{J}} < 0 \end{cases} \tag{2.43}$$
where $\hat{\mathbf{n}}$ is a unit vector whose direction is the line of the nodes and whose orientation goes from the center of the ellipse to the ascending node; it is called *unit vector of the line of nodes* and is defined as:
$$\hat{\mathbf{n}} \triangleq \hat{\mathbf{K}} \times \frac{\mathbf{h}}{h}$$

5. The argument of perigee ω can be expressed as a function of the eccentricity unit vector $\hat{\mathbf{e}} \triangleq \mathbf{e}/e$ and of the unit vector of the line of nodes $\hat{\mathbf{n}}$ as:
$$\cos \omega = \hat{\mathbf{n}} \cdot \hat{\mathbf{e}} \quad \Rightarrow \quad \omega = \begin{cases} \arccos(\hat{\mathbf{n}} \cdot \hat{\mathbf{e}}) & \text{if} \quad \mathbf{e} \cdot \hat{\mathbf{K}} > 0 \\ 2\pi - \arccos(\hat{\mathbf{n}} \cdot \hat{\mathbf{e}}) & \text{if} \quad \mathbf{e} \cdot \hat{\mathbf{K}} < 0 \end{cases} \tag{2.44}$$

6. At last, observing that the true anomaly ν is the angle between the direction of the eccentricity unit vector $\hat{\mathbf{e}}$ (which identifies the perigee of the orbit) and the direction of the position unit vector $\hat{\mathbf{r}} \triangleq \mathbf{r}/r$, the following expression is obtained:
$$\cos \nu = \hat{\mathbf{e}} \cdot \hat{\mathbf{r}} \quad \Rightarrow \quad \nu = \begin{cases} \arccos(\hat{\mathbf{e}} \cdot \hat{\mathbf{r}}) & \text{if} \quad \mathbf{r} \cdot \mathbf{v} > 0 \\ 2\pi - \arccos(\hat{\mathbf{e}} \cdot \hat{\mathbf{r}}) & \text{if} \quad \mathbf{r} \cdot \mathbf{v} < 0 \end{cases} \tag{2.45}$$

2.5.2 Computation of r and v based on orbital elements

Known the set of orbital elements, the step
$$\{a, e, i, \Omega, \omega, \nu\} \rightarrow (\mathbf{r}, \mathbf{v})$$
represents the inverse problem with respect to the one previously considered.
For this purpose, it is convenient to use a Perifocal reference frame (Figure 2.7) in which the position vector of the satellite has the following expression:
$$\mathbf{r} = r \cos \nu \, \hat{\mathbf{p}} + r \sin \nu \, \hat{\mathbf{q}} \tag{2.46}$$

where the distance r from the center of attraction (coinciding with the origin of the reference system) is given, according to the orbital parameters, by the polar equation of the conic:

$$r = \frac{p}{1 + e\cos\nu} \qquad \text{with} \qquad p = a(1 - e^2)$$

The components of $\mathbf{r}$ in the Perifocal reference frame are then:

$$[\mathbf{r}]_{pwq} = r \begin{bmatrix} \cos\nu \\ \sin\nu \\ 0 \end{bmatrix} = \frac{a(1-e^2)}{1 + e\cos\nu} \begin{bmatrix} \cos\nu \\ \sin\nu \\ 0 \end{bmatrix} \tag{2.47}$$

Regarding the expression of $\mathbf{v}$, it must be kept in mind that, in the keplerian context, the $\hat{\mathbf{p}}$ and $\hat{\mathbf{q}}$ unit vectors are constant over time and therefore the Perifocal reference frame is assimilable to an inertial system, i.e.:

$$\frac{d\hat{\mathbf{p}}}{dt} \equiv \frac{d\hat{\mathbf{q}}}{dt} = 0$$

Deriving with respect to time the expression of $\mathbf{r}$ given by (2.46), we obtain:

$$\mathbf{v} = \frac{d\mathbf{r}}{dt} = (\dot{r}\cos\nu - r\,\dot{\nu}\sin\nu)\,\hat{\mathbf{p}} + (\dot{r}\sin\nu + r\,\dot{\nu}\cos\nu)\,\hat{\mathbf{q}} \tag{2.48}$$

At this point, it is necessary to express r and $r\,\dot{\nu}$, exclusively, as functions of the orbital elements. For this purpose, it is sufficient to derive with respect to time both members of the polar equation of the conic, obtaining:

$$\dot{r} = \frac{\partial r}{\partial \nu}\dot{\nu} = \frac{p(e\sin\nu)}{(1 + e\cos\nu)^2}\dot{\nu} = \frac{r\,\dot{\nu}(e\sin\nu)}{1 + e\cos\nu} = \cdots = \frac{\sqrt{\mu p}\,e\sin\nu}{r(1 + e\cos\nu)} = \sqrt{\frac{\nu}{p}}\,e\sin\nu \tag{2.49}$$

From the previous equation, the expression of $r\dot{\nu}$ can be found:

$$r\dot{\nu} = \sqrt{\frac{\mu}{p}}\,(1 + e\,\cos\nu) \tag{2.50}$$

Replacing (2.49) and (2.50) in (2.48), we get:

$$\mathbf{v} = \sqrt{\frac{\mu}{p}}\,[-\sin\nu\,\hat{\mathbf{p}} + (e + \cos\nu)\,\hat{\mathbf{q}}] \tag{2.51}$$

which represents the desired expression. In terms of components, the (2.51) can be written as:

$$[\mathbf{v}]_{pwq} = \sqrt{\frac{\mu}{p}} \begin{bmatrix} -\cos\nu \\ e + \cos\nu \\ 0 \end{bmatrix} \tag{2.52}$$

It can be noted that the components of $\mathbf{r}$ and $\mathbf{v}$, written in (2.47) and (2.52), depend only on the orbital elements a,e and ν. This is due to the fact that the expression of $\mathbf{r}$ and $\mathbf{v}$ previously computed refer to a Perifocal reference frame. For this reason the components do not contain neither the information relative to the orbital plane lay (which depend

on Ω and i angles), nor those related to the orientation of the orbit on its plane (through the ω angle).
In order to have $\mathbf{r}$ and $\mathbf{v}$ that depend on all the six orbital elements, it is necessary to define them in the Geocentric-Equatorial reference frame. To do this, it is sufficient to apply the rotation matrix $\mathbf{T}_{313}(\Omega, i, \omega)$ to the vectors defined in the Perifocal reference frame, in order to obtain:

$$[\mathbf{r}]_{IJK} = \mathbf{T}_{313}(\Omega, i, \omega)[\mathbf{r}]_{pwq} \tag{2.53}$$
$$[\mathbf{v}]_{IJK} = \mathbf{T}_{313}(\Omega, i, \omega)[\mathbf{v}]_{pwq} \tag{2.54}$$

where $\mathbf{T}_{313}(\Omega, i, \omega)$ is defined as a combination of the following three elementary rotation matrices:

$$\mathbf{T}_3(\Omega) = \begin{bmatrix} \cos\Omega & -\sin\Omega & 0 \\ \sin\Omega & \cos\Omega & 0 \\ 0 & 0 & 1 \end{bmatrix}$$
$$\mathbf{T}_1(i) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos i & -\sin i \\ 0 & \sin i & \cos i \end{bmatrix}$$
$$\mathbf{T}_3(\omega) = \begin{bmatrix} \cos\omega & -\sin\omega & 0 \\ \sin\omega & \cos\omega & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

then ($s \triangleq \sin$, $c \triangleq \cos$):

$$\mathbf{T}_{313}(\Omega, i, \omega) = \begin{bmatrix} c\Omega\, c\omega - s\Omega\, ci\, s\omega & -c\Omega\, s\omega - s\Omega\, ci\, c\omega & s\Omega\, si \\ s\Omega\, c\omega + c\Omega\, ci\, s\omega & -s\Omega\, s\omega + c\Omega\, ci\, c\omega & -c\Omega\, si \\ si\, s\omega & si\, c\omega & ci \end{bmatrix}$$

The computation of the position and velocity vectors, based on the orbital elements, is important for the analysis of the satellite's motion on its orbit. In fact, considering a Keplerian problem, note $\mathbf{r}$ and $\mathbf{v}$ at time t_0, it is possible to determine $\mathbf{r}$ and $\mathbf{v}$ at a time $t > t_0$. Starting from $\mathbf{r}(t_0)$ and $\mathbf{v}(t_0)$ and using the expressions described in the previous paragraph, the six orbital elements are determined. Furthermore, if the problem is Keplerian, the first five orbital elements remain constant while the only element that varies over the time is the true anomaly ν that identifies the position of the satellite along its orbit.
Since, given an instant of time $t > t_0$ it is possible to determine the value of $\nu(t)$, on the basis of the six orbital elements knowledge and applying the (2.53) and the (2.54), the $\mathbf{r}(t)$ and $\mathbf{v}(t)$ values are obtained. Moreover, in order to derive the position and velocity of the satellite at each point of the orbit, it is sufficient to vary ν from 0 to 2π.

2.6 Sentinel2 orbit definition through orbital elements

In this section the proceedings seen in subsections 2.5.1 and 2.5.2 are used to define the Sentinel2 orbit, that is the one necessary to perform the nominal mission (described in Chapter 1).

The spacecraft orbit is characterized by the following orbital elements[1]:

- $a \approx 7166\,\text{km}$;
- $e \approx 0.000127$;
- $i \approx 98.62°$;
- $\Omega \approx 327.4862°$;
- $\omega \approx 93.0135°$.

Starting from these orbital elements and applying (2.53) and (2.54) with ν that varies from 0 to 2π, it is possible to compute position and velocity of the satellite at each point of the orbit. Plotting the position vector, the resulting orbit is shown in Figure 2.10.

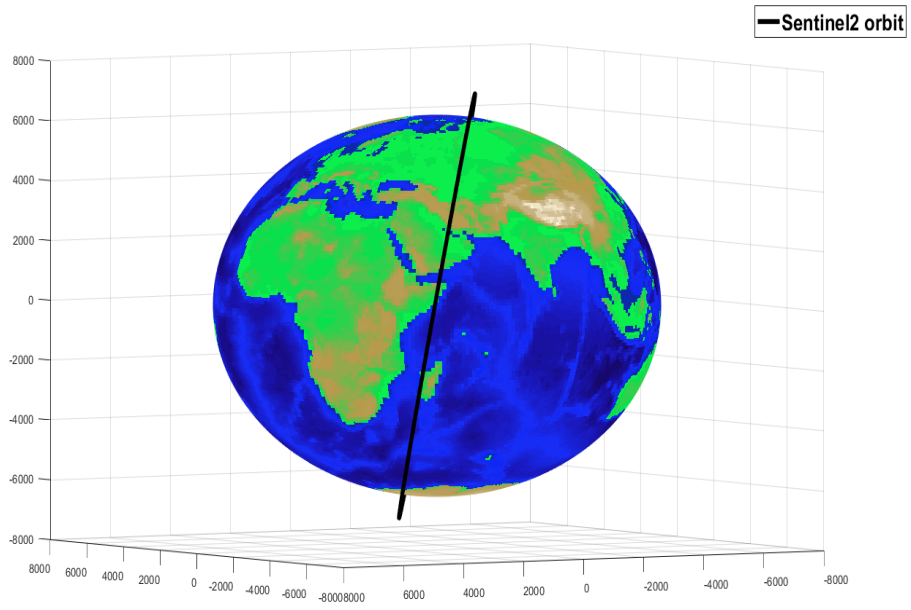

Figure 2.10. Obtained Sentinel2 orbit through orbital elements

[1]The information about the characteristics of Sentinel2 orbit are described on the ESA website [9].

2.7 Sentinel2 orbit definition through integration of FR2B

The behaviour of the satellite, i.e. its position and velocity instant by instant, can be predicted by integration of differential equation (FR2B):

$$\ddot{\mathbf{r}} + \mu \frac{\mathbf{r}}{r^3} = 0 \quad \Rightarrow \quad \dot{\mathbf{v}} + \mu \frac{\mathbf{r}}{r^3} = 0$$

In order to perform a numerically integration, the satellite dynamics have to be described by a set of first order differential equations, called **state equations**, of the form

$$\dot{\mathbf{x}}(t) = f[\mathbf{x}(t), \mathbf{u}(t); t]$$
$$\mathbf{y}(t) = h[\mathbf{x}(t), \mathbf{u}(t); t]$$

where

- $\mathbf{x}(t) \in \mathbb{R}^{n_x}$: state;
- $\mathbf{u}(t) \in \mathbb{R}^{n_u}$: input;
- $\mathbf{y}(t) \in \mathbb{R}^{n_y}$: output;
- n_x: order of the system.

Since (FR2B) is a second order differential equation, it can be defined by two first order differential equations, in this way:

$$\dot{\mathbf{r}} = \mathbf{v} \tag{2.55}$$

$$\dot{\mathbf{v}} = -\mu \frac{\mathbf{r}}{r^3} \tag{2.56}$$

Defining the system states

$$\mathbf{x} = (\mathbf{r}, \mathbf{v}) = (x, y, z, \dot{x}, \dot{y}, \dot{z}) = (x_1, x_2, x_3, x_4, x_5, x_6)$$

the (5.1) and (5.2) can be rewritten as:

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \\ \dot{x_4} \\ \dot{x_5} \\ \dot{x_6} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -\frac{\mu}{r^3} & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{\mu}{r^3} & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{\mu}{r^3} & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} \tag{2.57}$$

The numerical integration is implemented using the TR-BDF2 method. It is an implicit Runge-Kutta[2] formula with a trapezoidal rule step as its first stage and a backward differentiation formula of order two as its second stage; by construction, the same iteration matrix is used in evaluating both stages.

[2]The Runge–Kutta methods are a family of implicit and explicit iterative methods, which include the well-known routine called the Euler Method, used in temporal discretization for the approximate solutions of ordinary differential equations (ODE).

To perform the integration, the following parameters must also be defined:

- integration interval
- initial state.

As integration interval, a period of the satellite's orbit was considered. Let A_p the total orbital area, the period P of an elliptical orbit (according to the third Kepler law) is given by

$$P = 2\pi\sqrt{\frac{a^3}{\mu}} \tag{2.58}$$

For Sentinel2 the period P is equal to $6015\,s$. Instead, as initial state, r_0 and v_0 are considered (i.e. position and velocity at time $t = 0$). They were computed, using the orbital elements that characterize the Sentinel2 orbit and applying (2.53) and (2.54) with $\nu = 0$.

Plotting the position vector, the resulting orbit reported in Figure 2.11.

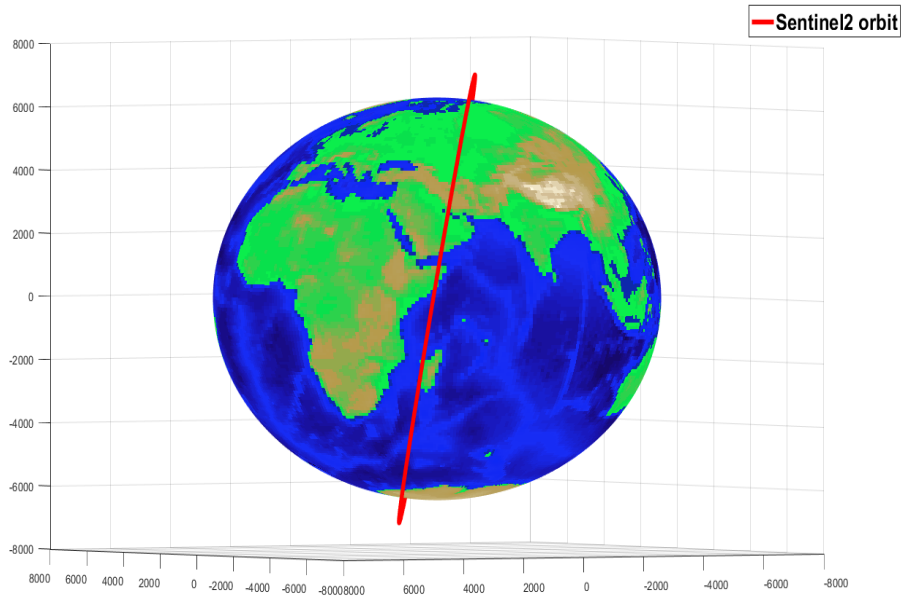

Figure 2.11. Obtained Sentinel2 orbit through integration

As can be seen, the two approaches provide equivalent results. What characterizes each of them is the way in which the orbit is described. Indeed, in the first one the satellite motion is computed by varying the true anomaly ν from 0 to 2π; this not allow a direct association between position and velocity of the satellite with a precise instant of time. Instead in the second approach, through integration, it is possible to make this association; instant by instant, an orbit described in terms of position and velocity is obtained. For this reason, the derivation of the orbit through integration is more suitable for simulation and for the use of predictive control algorithm.

Chapter 3

Nonlinear Model Predictive Control

The purpose of this chapter is to provide an overview about the principle and mathematical elements of the Nonlinear Model Predictive Control (NMPC).

3.1 Overview

In many control problems, it is desired to design a stabilizing feedback such that a performance criterion is minimized while satisfying constraints on the controls and the states. Ideally, one would look for a closed solution for the feedback satisfying the constraints while minimizing the performance. However, since often the closed loop solution can not be found analytically, a repeated solution of an open-loop optimal control problem for a given state is used [14]. Control approaches using this strategy are defined **Model Predictive Control (MPC)**.

3.1.1 Model predictive control development and principle

Linear model predictive control becomes popular since the 70s of the past century. Originally, it is designed and used for control of large-scale processes, typically in the chemical and petroleum industry, since the slow dynamics of those systems allowed large control intervals and then enough time to perform the optimization. Thanks to progress in the area of optimization algorithms and hardware computational speed, this approach has become suitable for control of fast dynamical systems with time constants in micro- and millisecond range [15]. For these reasons, MPC has obtained an increase in popularity as a control technique used, not only in the process industry, but also in a wide range of applications from automotive and aerospace to clinical anaesthesia.

As reported in [16], the key ideas of this control strategy are:

- use of a model to predict the process output along a future time horizon;
- definition of a control sequence minimizing an objective (or cost) function;
- receding strategy, so that at each instant the prediction horizon is shifted forward (towards the future).

Starting from these concepts, different types of predictive algorithms have been developed. The difference between them is only on the type of model used to represent the system and the cost function to be minimized.

The main advantages of this strategy are:

- it can be used to control a great variety of systems, also with very complex dynamics or with multi-inputs and multi-outputs (MIMO);
- it intrinsically compensates the delay in the response to a control action, called dead time;
- the resulting controller is defined by an easy-to-implement control law;
- it permits, during the design process, the treatment of the constraints on the states and output of the system obtaining a constrained optimization problem.

However it also has its drawbacks. One of these is the amount of computations required to derive the control law in cases where it has to be carried out at every sampling time (since the dynamic of the system changes), especially when the constraints are considered. Another disadvantage is the need for an appropriate model of the process to be available. Indeed the design algorithm is based on prior knowledge of that and therefore the more the model used is different from the real system, the more the resulting control law will be less accurate.

3.1.2 Nonlinear model predictive control development and principle

Many systems are, however, inherently nonlinear. The inherent nonlinearity together with an increase in the constraints (linear and nonlinear) on the system dynamics, imply operating conditions often near the boundary of the admissible region. Then, linear models are, in those cases, not sufficient to describe the process dynamics adequately and nonlinear models must be used. The variant of Model Predictive Control, characterized by the use of nonlinear system models in the prediction, is called **Nonlinear Model Predictive Control (NMPC)** [17].

As in linear model predictive control, NMPC involves the repetitive solution of an optimal control problem at each sampling time in a receding horizon fashion. While these problems are convex in linear MPC, in the nonlinear case they are not convex anymore; this determines the presence of more local minima and therefore a much greater number of computations for each sample, even without providing any hard guarantees on the solution. So, ensuring a global (or at least sufficiently good) solution to the resulting nonlinear optimization problem can be a process not feasible for real-time requirements. Furthermore, other limiting factors are the development of nonlinear dynamic models and state estimators. Then, the NMPC can not be used to solve any control problem; for these reasons, its industrial development is still limited. However, it is a powerful approach of great promise that has proven itself in several applications. With further research in the direction of numerical implementation technology and modeling and

state estimation methods, it may strengthen its position as the most powerful method available for certain classes of systems [18].

3.2 NMPC strategy

In general, the model predictive control problem is formulated as solving on-line a finite horizon open-loop optimal control problem subject to system dynamics and constraints involving states and controls. The methodology of all controllers belonging to the MPC family is shown in Figure 3.1.

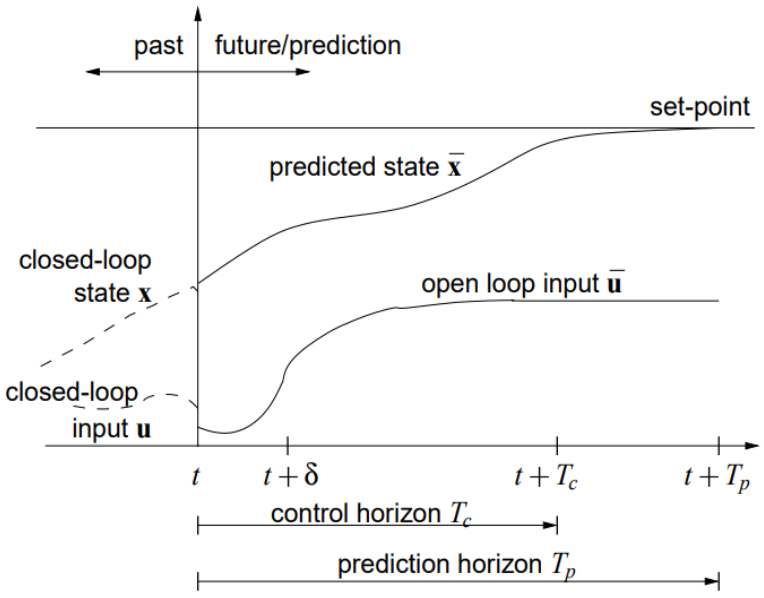

Figure 3.1. NMPC strategy

Based on measurements obtained at time t, at each time step the controller:

1. predicts the future dynamic behaviour of the system over a **prediction horizon** T_p;
2. determines (over a **control horizon** $T_c \leq T_p$) the command input such that a predetermined open-loop performance objective function is optimized, in order to obtain the best prediction (i.e., the prediction closest to the desired behaviour).

In an ideal situation, i.e. no disturbances, no model-plant mismatch and infinite prediction horizons, the input function found at time $t = 0$ could be applied to the system for all times $t \geq 0$. However, in general, this is not possible, since the true system behaviour is different from the predicted one. Then, it is necessary to incorporate some feedback mechanism, in order to implement the obtained open-loop manipulated iput function only until the next measurement becomes available. The time interval between each measurement is assumed to be fixed and equal to δ; this value, also indicated as Ts, is

called **sampling time**. Using the new measurement at time $t + \delta$, the whole procedure (prediction and optimization) is repeated in order to find a new input function with the control and prediction horizons moving forward. Furthermore the calculation of the command input at each time step based on the system behaviour, allows the inclusion of constraints on state and input.

3.3 Basic structure

In order to implement the strategy described in section 3.2, the structure shown in Figure 3.2 is used [16].

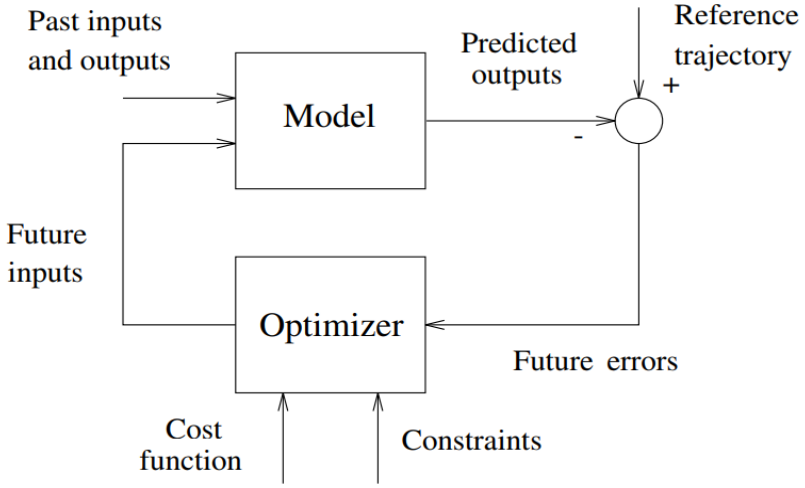

Figure 3.2. NMPC basic structure

The two basic elements of this structure are:

- the model
- the optimizer.

Model

As can be seen in Figure 3.2, the model is used to predict the future plant outputs, based on past and current values and on the proposed optimal future control actions. Then, the chosen model must be able to describe the dynamics of the process in order to provide a proper prediction in the behaviour and at the same time be simple to implement. It can be divided in process model and disturbance model, which takes into account the effect of unmeasurable inputs, noises and model errors. The process model can be described in different ways, but the most used description, especially in the academic research community, is the **state space representation**. Indeed, it allows a very simple derivation

of the controller, even for the multivariable case, and an easier expression of stability and robustness criteria.

Optimizer

The aim of the optimizer is to provide the control actions by minimizing a cost function that also takes into account the presence of constraints on the system. If the cost function is quadratic, its minimum can be obtained as an explicit function (linear) of past inputs and outputs and the future reference trajectory. Instead, in the presence of inequality constraints the cost function becomes more complex and therefore its minimization requires much higher computational costs. The control action provided by the optimizer is such that drives the process to accomplish the specified requirements, fulfilling at the same time the specified constraints.

3.4 Control problem formulation

Let consider a generic nonlinear system described by state space representation:

$$\begin{aligned} \dot{x} &= f(x,u) \\ y &= h(x,u) \end{aligned} \tag{3.1}$$

where $x \in \mathbb{R}^n$ is the state, $u \in \mathbb{R}^{n_u}$ is the command input and $y \in \mathbb{R}^{n_y}$ is the output of the system.
It is assumed that the state is measured in real-time, with a sampling time T_s, so that the measurements are

$$x(t_k) \quad \text{with} \quad t_k = T_s k \ \text{ for } \ k = 0,1,\dots$$

As defined in section 3.2, the NMPC strategy is based on two key operations:

- prediction
- optimization.

In subsections 3.4.1 and 3.4.2, taking from [11], a detailed treatment of these two operations is carried out.

3.4.1 Prediction

At time t a prediction of the state and output, over an interval $[t, t+T_p]$, is obtained by integration of 3.1 or a model of it. In general the predicted output $\hat{y}$, at a certain instant of time $\tau \in [t, t+T_p]$, is a function of the initial state $x(t)$ and of the input signal $u(t:\tau)$ (that is the input defined in the interval $[t,\tau]$):

$$\hat{y}(\tau) \equiv \hat{y}(\tau, x(t), u(t:\tau))$$

In order to simplify calculations, it is reasonable to assume the input as a constant signal after a time $T_c \in [T_s, T_p]$, previously defined as control horizon:

$$u(\tau) = u(t + T_c), \qquad \tau \in [t + T_c, t + T_p]$$

Note that, as shown in Figure 3.1, u is an **open-loop input**, since its value in the interval $[t, t + T_p]$ does not depend on the value assumed instant by instant by the state x in that interval.

3.4.2 Optimization

The aim of this control technique is to generate, at each sampling time, an input signal $u^*(t : \tau)$ such that the prediction $\hat{y}(\tau, x(t), u^*(t : \tau)) \equiv \hat{y}(u^*(t : \tau))$ has the desired behaviour for $\tau \in [t, t + T_p]$. The concept of desired behaviour is expressed by defining the **objective function**:

$$J(u(t : t + T_p)) \doteq \int_t^{t+T_p} \left(\|\tilde{y}_p(\tau)\|_Q^2 + \|\tilde{u}(\tau)\|_R^2 \right) d\tau + \|\tilde{y}_p(t + T_p)\|_P^2 \tag{3.2}$$

where $\|\tilde{y}_p(\tau)\| \doteq r(\tau) - \hat{y}(\tau)$ is the predicted tracking error and $r(\tau) \in \mathbb{R}^{n_y}$ is the reference to track. The symbols $\|\cdot\|_X$, introduced in the expression above, are weighted vector norms and their integrals are square signal norms.

So to get the desired behaviour, the input signal $u^*(t : t + T_p)$ is chosen as one minimizing the objective function $J(u(t : t + T_p))$. In other words, this means minimizing the three terms that compose it, which are

1. $\|\tilde{y}_p(\tau)\|_Q^2$: this term is the tracking error norm, that is the difference, instant by instant, between the reference signal and the predicted output;

2. $\|\tilde{u}(\tau)\|_R^2$: this term allows to manage trade-off between performance and command activity;

3. $\|\tilde{y}_p(t + T_p)\|_P^2$: this term gives further importance to the final tracking error.

The square weighted norm of a generic vector $v \in \mathbb{R}^n$ is

$$\|v\|_Q^2 \doteq v^T Q v = \sum_{i=1}^{n} q_i v_i^2, \quad Q = \text{diag}(q_1, \ldots, q_n) \in \mathbb{R}^{n \times n}$$

where $q_i \geq 0$ are the weights of the matrix Q. The values assigned to the matrices Q, R and P are fundamental for the NMPC design, since they regulate the optimization process. Indeed, depending on this weights, more or less importance can be given to the minimization of each of the three terms.

One of the main features of the model predictive control is the possibility to consider constraints on the system during the optimization process. Then the minimization of $J(u(\cdot))$ can be subjected both to constraints given by the dynamics of the system

$$\begin{aligned}\dot{\hat{x}}(\tau) &= f(\hat{x}(\tau), u(\tau)) \\ \hat{x}(t) &= x(t) \\ \hat{y}(\tau) &= h(\hat{x}(\tau), u(\tau))\end{aligned}$$

and to constraints on predicted state/output (for example obstacles or collision avoidance) and on the input (for example input saturation).

3.5 Mathematical formulation

The general formulation of the optimization problem, described in subsection 3.4.2, is:

$$\begin{aligned}&u^*(t : t + T_p) = \arg\min_{u(\cdot)} J\left(u(t : t + Tp)\right) \\ &\text{subject to:} \\ &\quad \dot{\hat{x}}(\tau) = f(\hat{x}(\tau), u(\tau)), \quad \hat{x}(t) = x(t) \\ &\quad \hat{y}(\tau) = h(\hat{x}(\tau), u(\tau)) \\ &\quad \hat{x}(\tau) \in X_c,\ \hat{y}(\tau) \in Y_c,\ u(\tau) \in U_c \\ &\quad u(\tau) = u(t + T_c), \quad \tau \in [t + T_c, t + T_p]\end{aligned} \tag{3.3}$$

To solve this problem on-line, at each sampling time, an efficient numerical algorithm is needed, since its formulation is in general non-convex. Moreover, the objective function $J(u(\cdot))$ is a function of the signal $u(\cdot)$, which can be seen as a vector with an infinite number of elements; hence, the optimization involves an infinite number of decision variables.

In order to overcome this problem, the input signal can be parametrized in the following way:

$$u(\tau) = \sum_{i=1}^{m} c_i \phi_i(\tau) = c\phi(\tau) \tag{3.4}$$

where $c = [c_1, \ldots, c_m] \in \mathbb{R}^{n_u \times m}$ are the parameters, while $\phi(\tau) = [\phi_1(\tau), \ldots, \phi_m(\tau)]^T \in \mathbb{R}^{m \times 1}$ are the basic functions. In general these basic functions can be

- rectangular functions:

$$\phi_i(\tau) = \begin{cases} 1, & \tau \in [t + (i-1)T_s, t + iT_s] \\ 0, & \text{otherwise} \end{cases}$$

- polynomial functions:

$$\phi_i(\tau) = (t-\tau)^{i-1}.$$

In both cases, the input is kept constant over the prediction horizon:

$$u(\tau) = c_1 = \text{const}, \quad \tau \in [t, t+T_p]$$

Note that in case of rectangular functions, $u(\tau)$ is piece-wise constant.

Since, with the parametrization, the input signal is represented by the finite dimension matrix c, the optimization problem can be reformulated as follows:

$$\begin{aligned} & c^* = \arg \min_{c \in \mathbb{R}^{n_u \times m}} J(c) \\ & \text{subject to:} \\ & \quad \dot{\hat{x}}(\tau) = f(\hat{x}(\tau), u(\tau)), \quad \hat{x}(t) = x(t) \\ & \quad \hat{y}(\tau) = h(\hat{x}(\tau), u(\tau)) \\ & \quad u(\tau) = c\phi(\tau) \\ & \quad \hat{x}(\tau) \in X_c, \ \hat{y}(\tau) \in Y_c, \ u(\tau) \in U_c \\ & \quad u(\tau) = u(t+T_c), \quad \tau \in [t+T_c, t+T_p] \end{aligned} \tag{3.5}$$

The resulting optimal input is:

$$u^*(\tau) = c^*\phi(\tau)$$

The optimal solution to the minimization problem $u^*(t : t+T_p)$, computed at time t, is an open-loop input; indeed, it depends on $x(t)$ but not on $x(\tau)$, with $\tau > t$. Then, if this signal is applied for the entire time interval $[t, t+T_p]$, the control does not perform any feedback action which could increase the precision or reduce errors and disturbance effects. To overcome this open-loop behaviour and obtain a feedback control algorithm, the so-called **receding horizon strategy** has to be introduced. It is defined by the following steps:

1. At time $t = t_k$, the optimal input $u^*(t : t+T_p)$ is computed via optimization.
2. Only the first input value $u(\tau) = u^*(t = t_k)$ is applied and keep it constant for $\forall \tau \in [t_k, t_{k+1}]$.
3. Steps 1-2 are repeated for $t = t_{k+1}, t_{k+2}, \ldots$

In this way, instant by instant, the optimal input u^* is computed. This signal is applied to the process for a time interval equal to T_s; after that, through the optimization algorithm, a new optimal input is computed depending on the current state. As shown in Figure 3.3, using this strategy a **closed loop control** is obtained; indeed, the controller is capable of sensing the state evolution at each sampling time and then reacting to possible uncertainty errors and unpredicted disturbances.

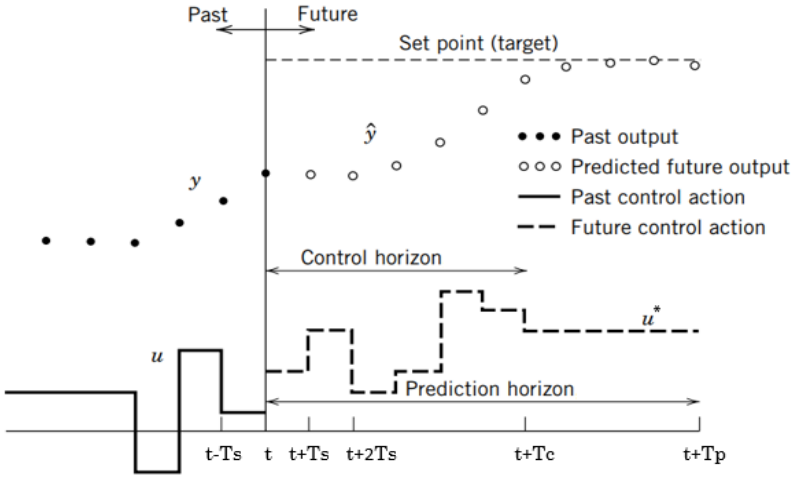

Figure 3.3. NMPC with reciding horizon strategy

3.6 NMPC design

3.6.1 Control scheme

The NMPC algorithm can be implemented in a MATLAB function and inserted in the Simulink block scheme as shown in Figure 3.4.

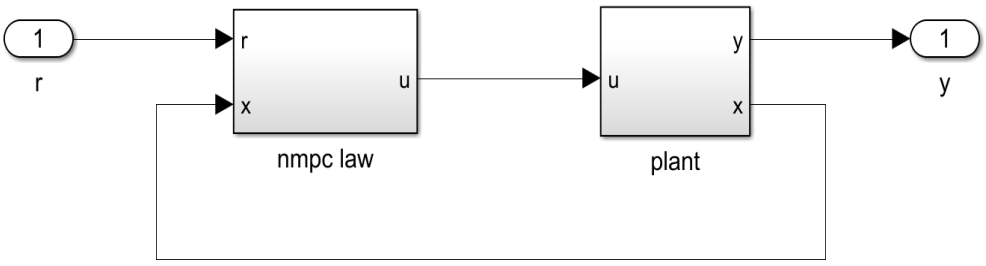

Figure 3.4. Control scheme

The plant block contains a detailed description of the system, by the use of the state space representation defined in the equations (3.1). Instead the model used in the nmpc law, necessary for prediction and optimization operations, is an approximate version

of (3.1):

$$\dot{\hat{x}} = \hat{f}(\hat{x}, u)$$
$$\hat{y} = \hat{h}(\hat{x}, u)$$

where $\hat{f}$ and $\hat{h}$ are approximations of f and h functions.
It is important to use an approximate model that

- simplifies the description of the system: this is useful for increasing the velocity of the prediction and optimization algorithm;
- does not neglect the main features of its dynamics.

3.6.2 Parameters choice

The design phase consists in acting on the NMPC parameters by a trial and error procedure in simulation, in order to obtain a configuration which can optimize performance and minimize complexity of the control algorithm. This parameters are: sampling time T_s, prediction horizon T_p, control horizon T_c, number of parametrization parameters m and weight matrices.

Sampling time

The sampling time T_s determines the rate at which the controller executes the control algorithm. In many situations, this parameter is given and cannot be chosen; indeed, it depends on the hardware that hosts the controller. In the particular cases where its design is available, the choice must be a trade-off between a sufficiently small value to deal with the plant dynamics (according to Nyquist-Shannon sampling theorem) and a not too small value to avoid numerical problems and slow computation.

Prediction and control horizon

The prediction horizon T_p determines how far the controller predicts into the future. Also in this case the choice, always available for the designer, must be a trade-off between a sufficiently large value to increase the closed-loop stability properties and a not too large value to avoid reducing the short-time tracking accuracy.

The control horizon T_c is the time after which the input signal is assumed constant. In other words, it indicates for how many time intervals, along the prediction horizon, the controller can change the input to obtain the desired behaviour. Then, a small control horizon involves few computations for the optimization, while a large value determines an improvement of the performance but at the same time an increase in the algorithm complexity. Usually, if a polynomial parametrization is used, it is convenient to set $T_c = T_p$; otherwise, $T_c = T_s$.

Number of parametrization parameters

The parametrization is an operation that transforms a space of infinite dimension into one of finite dimension equal to m. It was used to overcome the problem of minimizing a function with respect to a vector with an infinite number of elements, i.e. the input signal $u(\cdot)$. In general the smaller the parameter m, the smaller the number of decision variables involved in the optimization and then the lower the complexity of the algorithm. Moreover, a low number of parameters is enough to obtain a satisfactory control performance. For this reasons, usually $m = 1$ is chosen.

Weight matrices

The optimization process, that is the minimization of the objective function J, can be regulated and controlled through the use of three diagonal square matrices Q, P and R, called **weight matrices**. Indeed, the tuning of each diagonal element of these matrices, allows to find a suitable trade-off between performances and command activity. In particular:

- Q is related to the tracking error minimization at each sampling time and then it regulates the optimization of the system state;
- P is related only to the final term of the tracking error minimization and then it regulates the optimization of the system output;
- R is related to the command effort minimization and then it regulates the optimization of the input.

As an example, we can consider the matrix Q defined as

$$Q = \begin{bmatrix} q_{11} & 0 & \dots & 0 \\ 0 & q_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & q_{nn} \end{bmatrix}$$

where n is the system order. Each diagonal element sets a certain weight (or penalty) to the associated state variable to minimize. The higher the weight value, the more importance assumes the variable during the optimization process.

The initial choice of matrix values, before adjusting the weights with a trial and error tuning due to several simulations, can be done according to the following strategy:

$$Q_{ii} = \begin{cases} 1 & \text{in the presence of requirements on } x_i \\ 0 & \text{otherwise} \end{cases}$$

$$P_{ii} = \begin{cases} 1 & \text{in the presence of requirements on } y_i \\ 0 & \text{otherwise} \end{cases}$$

$$R_{ii} = \begin{cases} 1 & \text{in the presence of requirements on } u_i \\ 0 & \text{otherwise} \end{cases}$$

After that, depending on the simulation results and on the system requirements, the values of Q_{ii}, P_{ii} and R_{ii} can be regulated, knowing that:

- an increase of Q_{ii} and P_{ii} determines respectively a decrease of x_i and y_i energy and then a reduction of oscillations and converging time;
- an increase of R_{ii} determines a decrease of u_i energy and then a reduction of command effort and fuel consumption.

3.7 NMPC properties

3.7.1 Stability

One of the main problems of this control technique is that with a finite prediction and control horizon, the predicted open and the resulting closed-loop behaviour is in general different. Consequently, there is no guarantee that the closed-loop system will be stable. As described above in order to improve the stability properties, a sufficiently large T_p must be chosen. Then, the most intuitive way to achieve stability is the use of an infinite prediction horizon. Indeed, in the nominal case feasibility at one sampling instances also implies feasibility and optimality at the next sampling instances. This follows from **Bellman's Principle of Optimality** [19]: the input and state trajectories computed as the solution of the NMPC optimization problem at a specific instance in time, are in fact equal to the closed-loop trajectories of the nonlinear system, i.e. the remaining parts of the trajectories after one sampling instance are the optimal solution at the next sampling instance. This fact also implies closed-loop stability.

However, the use of an infinite prediction horizon is impossible from a computational point of view. For this reason it is necessary to enforce the closed-loop stability using a finite T_p. The simplest possibility is to add a so called **zero terminal equality constraint** at the end of prediction horizon, i.e. to add the equality constraint

$$\hat{\mathbf{x}}(t + T_p, \mathbf{x}(t), \hat{\mathbf{u}}) = \mathbf{0}$$

to optimization problem. This leads to stability of the closed-loop, if the optimal control problem possesses a solution at $t = 0$, since the feasibility at one time instance does also lead to feasibility at the following time instances and a decrease in the value function. One of the main problem of the zero terminal equality constraint is that the system must be brought to the origin in finite time. Additionally, from a computational point of view, an exact satisfaction of a zero terminal equality constraint does require an infinite number of iterations in the nonlinear programming problem.
In order to overcome the problems due to the use of a zero terminal constraint, the so called **terminal region constraint** can be used:

$$\hat{\mathbf{x}}(t + T_p) \in \mathbf{\Omega}$$

where $\mathbf{\Omega} \in \mathbb{R}^n$ is a bounded, closed and connected set. If the terminal region $\mathbf{\Omega}$ is suitably chosen, then stability of the closed-loop system can be guaranteed [17].

3.7.2 Robustness

In real-world applications, the exact plant model is seldom known. This means that an approximated model $\hat{f}$, $\hat{h}$ is used for control design, instead of the "true" model f, h (this holds for any method). In general, this is not a problem since, thanks to the receding horizon strategy, standard NMPC is inherently robust; this means that it is characterized by a good robustness properties. If this property needs to be improved, different techniques can be implemented:

- min-max NMPC
- H_∞ NMPC
- parametrized controller NMPC.

However these techniques are not widely used since they require a high computational effort and thus cannot be applied to problems where a small T_s is required.

Chapter 4

Mission scenario for autonomous guidance and control

This chapter provides a description of the specific mission scenario and strategies used for the implementation of the autonomous guidance and control. Moreover for each strategy, a definition of the feasible trajectories, planned to carry out the mission, will be reported. They will be treated as the reference signals of the control system and will determine the optimal path to be followed by the satellite. At the end of the chapter, the main propulsion technologies, used for orbital maneuvers, will be introduced.

4.1 Mission scenario description

In this thesis, since an Earth Observation satellite is considered, the mission scenario concerns the revisit and monitoring of an event in precise geographical coordinates on the Earth. The aim is to develop a system which is able to switch the satellite from remote/GS to autonomous planning when an alert triggers the on-board software; alerts could be a volcanic explosion or a flooding or human disasters. In this alarm scenario the satellite, autonomously, should guarantee:

- a fast revisit of the event;
- a persistent monitor of the event;
- the return to the nominal mission.

To obtain the autonomous guidance, two strategy are developed. Each of them is based on a trajectory planner and a predictive control algorithm. The planner has the purpose of producing trajectories which can guide the satellite from the current/nominal orbit to the target orbit. Instead, the predictive control algorithm, implemented through the use of NMPC approach (introduced in Chapter 3), allows to optimize the different maneuvers to be performed for obtaining the autonomous mission. In particular, it provides a command input for forcing the satellite to follow, in an optimum way, the same trajectories defined by the planner.

4.1.1 NMPC for autonomous guidance and control

As we will see in more detail later, the autonomous guidance consists of a series of inclination changes with respect to the current/initial orbit, i.e. the one traveled by the satellite to carry out the Earth observation mission, to obtain the target/final orbit, i.e. the one along which the satellite overflies the event. The problem when dealing with orbital maneuvers, especially in the case of inclination change maneuvers, is that the concept of impulsive and instantaneous thrust action, called ΔV, is only an abstract approximation useful for computing the mission budgets and evaluating satellite performances. However, in the real standard space approach, the ΔV obtained in the ideal case must be splitted in finite time intervals based on real thruster features. This is a process that determines an intrinsic decrease of satellite performance and an increase of propellant budget due to the gravity and misalignments losses. Furthermore, if the process that splits the thrust is not optimal, the losses can increase. Then, it is necessary to use a technique which can ensure an optimization of the maneuver profile. From this point of view, NMPC appears to be a very promising approach for autonomous guidance and, more in general, for space mission applications. The reasons of its great development are that this control system allows to:

- determine an optimal solution of the problem in which it is involved;
- consider high prediction horizon (T_p) and sampling time (T_s);
- manage, efficiently, the trade-off between performance and command activity;
- deal with complex linear and non-linear constraints.

One of its applications, in a real space mission, is for controlling the spacecraft during a rendezvous, using the Clohessy-Wiltshire equations for relative motion between the chaser and the target.

In this work it will be shown that NMPC provides excellent results also for inclination changes, that is the maneuvers required for the autonomous guidance.

4.2 Alerts simulation on the spacecraft

As mentioned above, in 'normal' condition the spacecraft travels the nominal orbit in order to carry out the mission assigned by the Copernicus Programme. When an alert occurred, the satellite software is triggered by an alarm that switches the system from remote/GS to autonomous planning. This scenario can be simulated on MATLAB by an user interface control, called **uicontrol**, which allows to create an alarm button. When it is pressed, the satellite becomes ready to receive the target coordinates and then to change its mission for obtaining the autonomous guidance. In this thesis, the events can be both positions just overflown by the satellite and generic positions on the Earth. Clearly, the further the target is from current satellite position, the more complex are the maneuvers to apply for accomplishing the different missions.

To simulate the satellite behaviour before and after alerts on the on-board software and consequently the switching from remote to autonomous guidance, the following scenarios can be defined on MATLAB

1. nominal scenario: the spacecraft makes its orbit to perform the Earth observation mission (Figure 4.1)

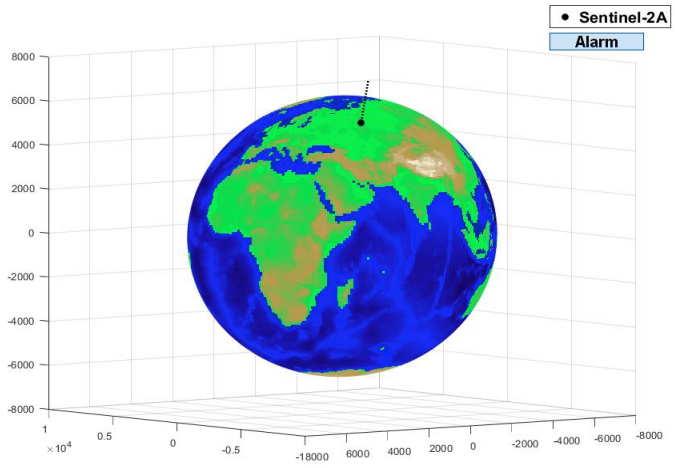

Figure 4.1. Satellite during nominal scenario before alert

2. alarm scenario: the spacecraft is warned on the presence of an alert from the user through the user interface button and becomes ready to receive the target coordinates and then to carry out the revisit and persistent monitor missions (Figure 4.2)

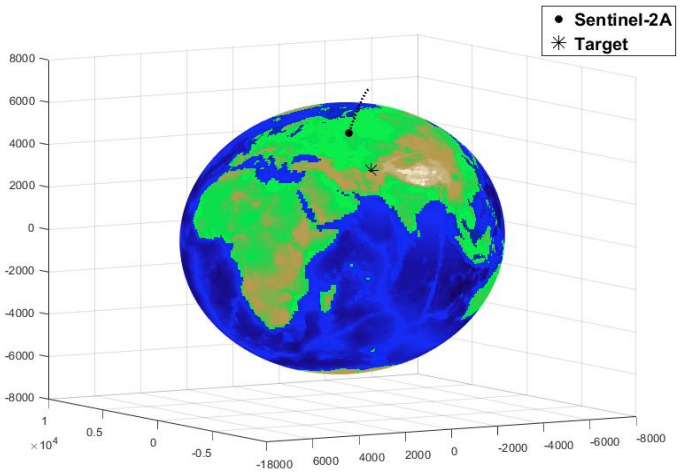

Figure 4.2. Satellite during alarm scenario

4.3 Target orbit definition

When an event occurred and the system switches from remote to autonomous, the satellite must be able to plan trajectories in order to reach the target orbit for a fast revisit, a persistent monitoring and the return to the nominal orbit. To accomplish these missions, two different strategies are developed. Both are based on the nominal orbit characteristics of the Sentinel-2. Indeed, in order to obtain Earth observation, the orbit is defined for a global coverage of the emerged lands between 56° S and 84° N, coastal waters and the entire Mediterranean Sea. Therefore, the satellite overflies, during the nominal mission, almost all the coordinates on the Earth, except those nearby the poles (Figure 4.3).

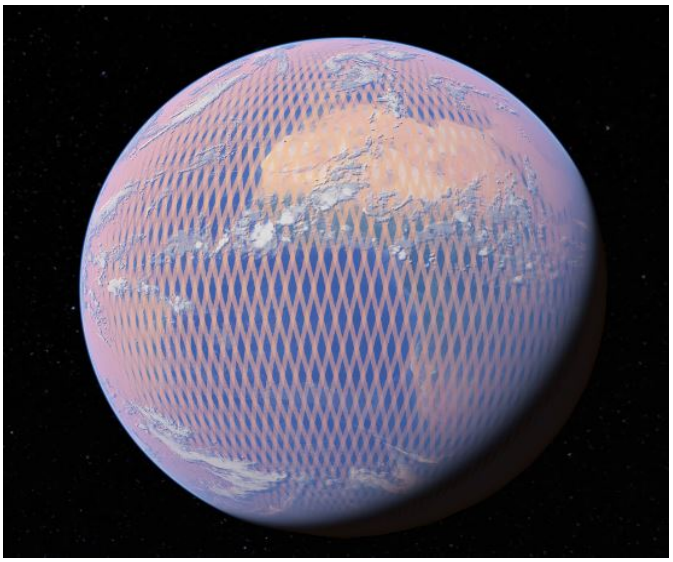

Figure 4.3. Sentinel-2 global coverage

Then, knowing the alert coordinate, by a search algorithm implemented on MATLAB, the target orbit can be found. This algorithm works by applying, for a properly number of time, a rotation matrix to the current orbit of the satellite. Indeed, its iterative application allows to simulate the Earth rotation effect on the orbit. For sake of simplicity, the elementary matrix about the z axis can be considered

$$T_1 = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.1}$$

where $\alpha = 0.4374$ rad, is the rotation angle occurred after each period of the orbit equals to 6015 s.

So, after each T_1 application, a new orbit is computed, rotated by an α angle with respect to the previous one, which covers a different Earth area. In this way, all the orbits for a full Sentinel-2 cycle are obtained; among these, there will also be the target orbit.

Considering the position, defined in section 4.2, the corresponding target orbit is shown in Figure 4.4.

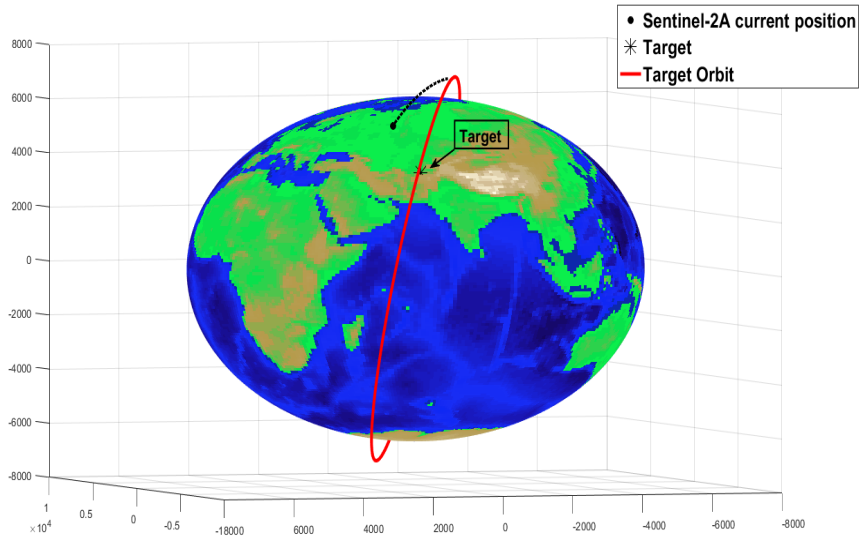

Figure 4.4. Target orbit associated to the event in which an alert occurred

4.4 Trajectory planner description

In a generic alarm scenario, the satellite must perform autonomously the required maneuvers to accomplish the three missions described above. The way in which it moves from the current to the target orbit is determined by the planner, which has the aim to generate feasible trajectories for guidance optimization.
Its implementation is based, firstly, on understanding in which area of the Earth the event occurred and its distance from the current satellite position. In order to achieve this, the Earth's surface is divided into four portions according to x and z values with respect to the Geocentric-Equatorial reference frame:

1. $x < 0$ and $z > 0$
2. $x < 0$ and $z < 0$
3. $x > 0$ and $z < 0$
4. $x > 0$ and $z > 0$

Figure 4.5 shown the four Earth's portions.

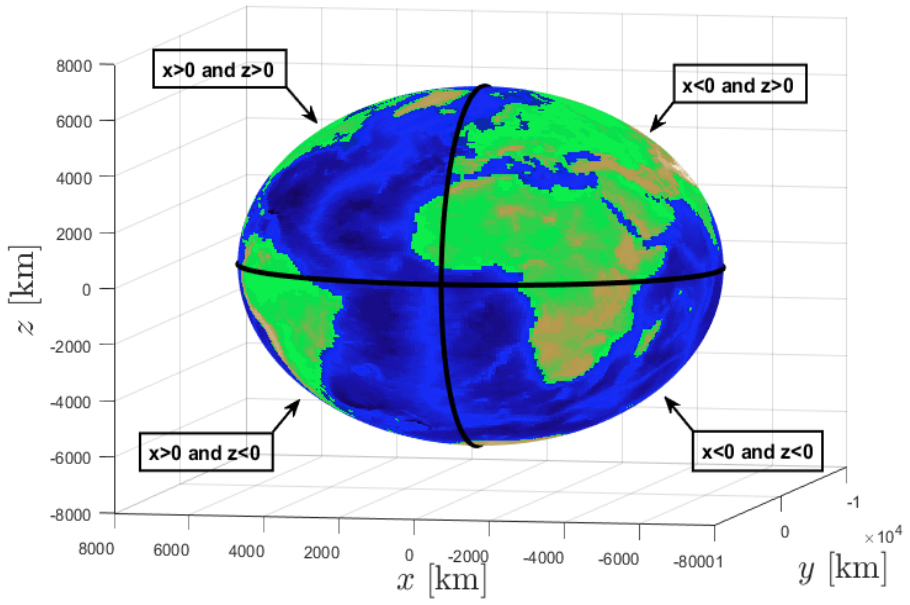

Figure 4.5. Earth's portions

So, knowing the event coordinates and projecting the current satellite position on the Earth, it is possible to place their geographic locations in one of the four portions defined above; in this way, the planner can derive the distance between them.
Moreover, in addition to this essential parameter, for the trajectories generation the planner must consider a main constraint, that is the required time to reach the exact coordinate during the revisit and the monitoring. Generally it is proportional to that required for the satellite to fulfill a complete orbit during the nominal mission for the Earth observation, equal to 6015 s. This time, besides being a constraint, can be also seen as an important parameter for finding the orbit along which the satellite overflies the event. Indeed, since a certain amount of time elapses between when the alarm triggers the software and when the revisit or monitoring mission is completed, due to Earth's rotation, there is a coordinate displacement. From this point of view, the orbit found in the previous section can not be considered, by the planner, as the target orbit to be reached in order to overfly the event during the revisit or the monitoring mission. The 'true' target orbit must be computed starting from that orbit and applying a rotation matrix which can take into account the angular displacement during the maneuvers. Also in this case, an elementary matrix about the z axis is used

$$T_2 = \begin{bmatrix} \cos\beta & -\sin\beta & 0 \\ \sin\beta & \cos\beta & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.2}$$

where $\beta = 7.2722 \cdot 10^{-5}$ rad, is the rotation angle occurred after each second.
Considering the above coordinates, for the first revisit mission the 'true' target orbit is

reported in Figure 4.6.

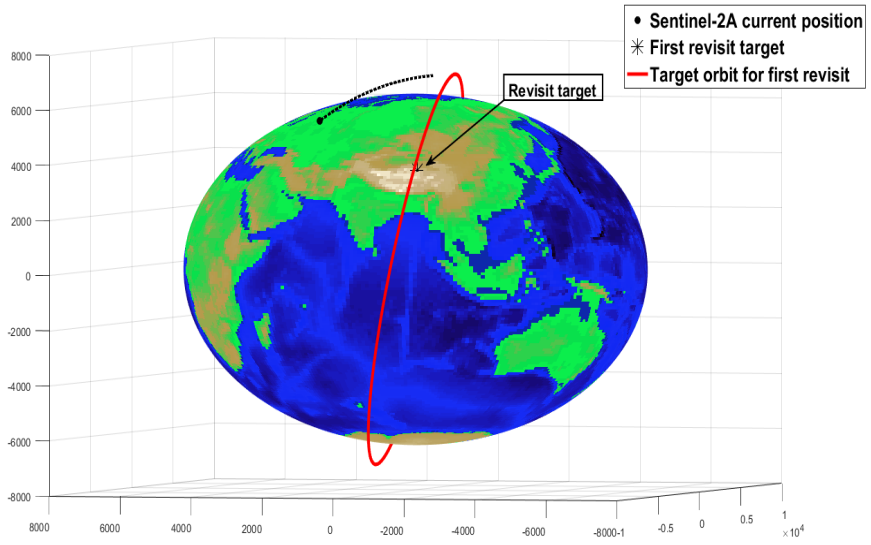

Figure 4.6. Target orbit associated to the event for the first revisit

Depending on the type of mission to be performed, T_2 must be applied for a different number of times; usually, in the revisit this number is greater than in the monitoring. Indeed, for the revisit mission the alert can be in a generic position of the Earth's surface. Then the greater the distance between the current position of the satellite and the coordinate to be overflown, the longer the time to reach the target and the higher the number of times the matrix T_2 must be applied. While regarding the monitor mission, since for any position the performed maneuvers are always the same, the number of applications of T_2 for each monitoring is fixed.

So summarizing, in the revisit and the monitoring case, the planner must compute the target orbit, along which the satellite will exactly overfly the event, through the application of T_2 matrix. For the revisit, since the alert coordinate can be in a generic position, the number of times for which T_2 must be applied, can not be defined a priori but changes from time to time. Then also the maneuvers to guide the satellite from current to target orbit change according to the alert position. Usually, as we will see later, the revisit mission determines much more sharp maneuvers than the monitoring mission, since the target could be very far from the current satellite position. The monitoring, instead, does not depend on the alert coordinate and therefore the number of times for which T_2 must be applied and consequently the maneuvers to be performed are fixed and always the same. The only variable parameter in this mission, since a persistent monitoring must be guaranteed, is the number of times for which the event must be overflown, i.e how long it must be kept under control. Clearly, this is related to the alert severity.

Once the monitoring of the event has been completed, the last mission of the autonomous guidance is the return to the nominal orbit, that is the one defined in the Copernicus Programme for the Earth observation. The planner must compute a trajectory which can bring the satellite exactly back to the starting sun-synchronous orbit in order to ensure that the Mean Local Solar Time (MLST) at the descending node is 10:30 (am).

4.5 Strategies definition

Once the target orbits, for carrying out the revisit and monitor missions, have been computed, the next step is to define the strategies used to derive the optimal satellite guidance. As previously reported, according to the type of strategy used, the way in which the planner computes the desired trajectories changes. In this project, two strategies are considered. Both of them have as main purpose to guide the satellite to the target in the shortest possible time and with high accuracy. Clearly, as we will in the Chapter 5, this involves a high command activity and consequently a high fuel consumption.
The two strategies considered are:

1. **Quasi-impulsive maneuvers strategy**, based on a search of the nodes between initial and target orbit and on the definition of a trajectory which determines the orbital changes exactly in one of them.

2. **Continuous manuevers strategy**, based on the definition of a trajectory that instant by instant, with a continuous change of inclination, guides the satellite from the current to the target orbit.

These two kinds of strategies are used to carry out the necessary maneuvers for the realization of the autonomous mission.

4.5.1 Quasi-impulsive maneuvers strategy

The first strategy is based on the concept of nodes, i.e. the intersections between two orbits. It consists in the application of the same maneuvers performed in the ideal situation, in which an impulsive and instantaneous ΔV is used for changing the inclination of the orbit exactly in one of the two nodes. Indeed, since in a maneuver for plan changes the inclination will be the only orbital parameter that will change, the maneuver must be performed in a node of the orbit (in the ascending node or in the descending node). The plane change manuevers carried out not in the nodes will result in the change of the right ascension of the ascending node (Ω) and then in worse performances. From this point of view, the idea of the first strategy is to accomplish a search of the nodes between current and target orbit and then to develop a planner capable of computing a trajectory which determines the orbital changes exactly in one of them. In order to obtain an optimization of the computed trajectories, an algorithm is developed for searching in which node is more suitable to implement the inclination change.

However in real case the application of impulsive maneuver is unfeasible; then, as we will see in Chapter 5, the NMPC will have the task of obtaining a quasi-impulsive maneuvers with performances quite analogue to the ideal case.
An example of how this strategy determines the autonomous guidance for the first revisit mission of the target, considered in the previous sections, is reported in Figure 4.7.

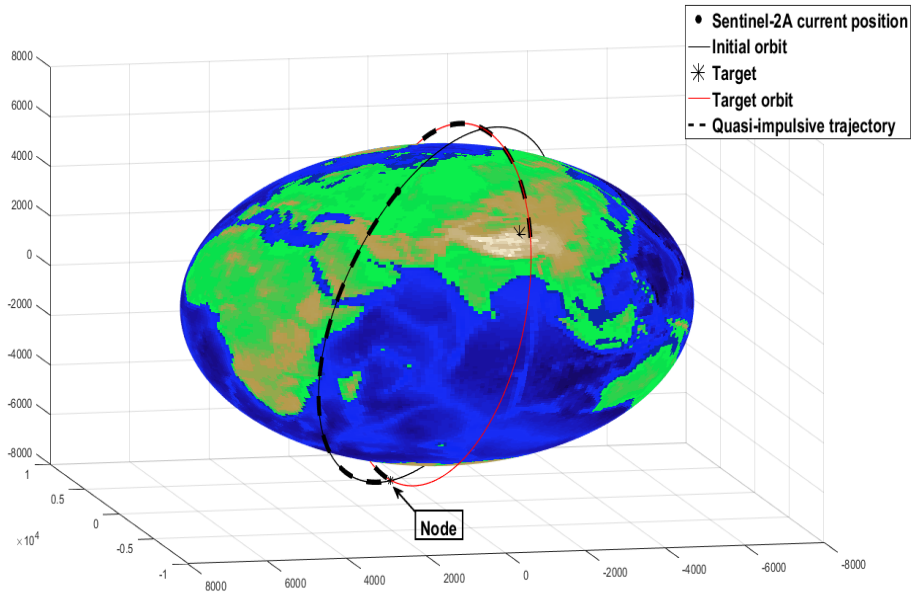

Figure 4.7. Quasi-impulsive maneuvers strategy for first revisit

4.5.2 Continuous maneuvers strategy

In the second strategy the continuous maneuvers concept is developed. It is based on the definition of a trajectory that, instant by instant, guides the spacecraft with a continuous change of inclination. This can be obtained by an interpolation between the initial and target orbit, allowing smooth maneuvers for the orbital change. In this case, a completely different strategy is considered with respect to the ideal case; indeed the inclination change is not applied at the nodes, but it is defined through a small orbital change at each instant of time. As we will see in Chapter 5, this approach can be advantageous in terms of command activity but at the same time the required ΔV and the propellant consumption considerably increase; this is due to the gravity and misalignments losses, which determine a performances decrease.
The main reason behind the development of this type of strategy has been to verify if the NMPC could handle situations where the command input should be applied for the whole duration of the mission. Indeed the future step will be to develop a strategy with continuous low-thrust maneuvers, in which the NMPC will be able to carry out the autonomous guidance in a reasonable time, clearly greater than the time used as a constraint in the two described strategy, and with a minimum fuel consumption.

Also for this strategy, the example of the first revisit of the same target, considered previously, is shown in Figure 4.8.

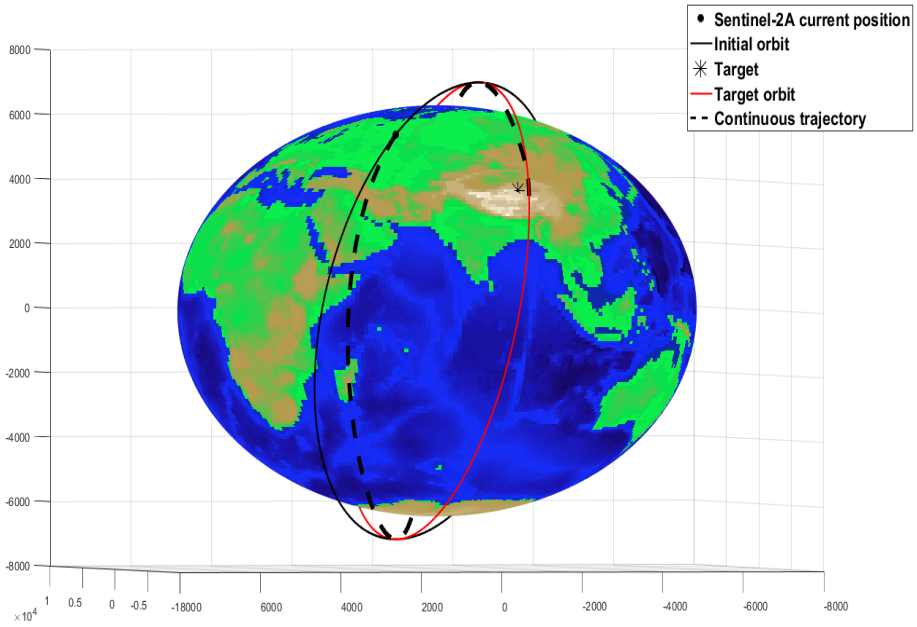

Figure 4.8. Continuous maneuvers strategy for first revisit

4.6 Space propulsion technologies

Propulsion system is any method used to produce thrust, which is the force that moves spacecraft and artificial satellite through air and space. Different propulsion systems generate thrust in different ways, but always through some application of the Newton's third law of motion. Space propulsion technologies can be framed in three different categories: escape propulsion (from Earth surface to orbit), in-space propulsion (in orbit) and deep space propulsion (from orbit to outer space). The launch vehicles currently used for escape propulsion rely on very mature technologies, but for in-space and deep space vehicles, there are prospects of significant technological advances. In this thesis, since we are dealing with a satellite in Earth Orbit, we will only consider the in-space propulsion technology.

4.6.1 The rocket equation

The space thruster behaviour follows the basic principle of a rocket: a device that can apply acceleration to itself using thrust by expelling part of its mass with high velocity can thereby move due to the conservation of momentum. The mathematical equation that describes the motion of vehicles based on the above principle is called **Tsiolkovsky rocket equation**.

This equation can be derived, considering the situation described in Figure 4.9.

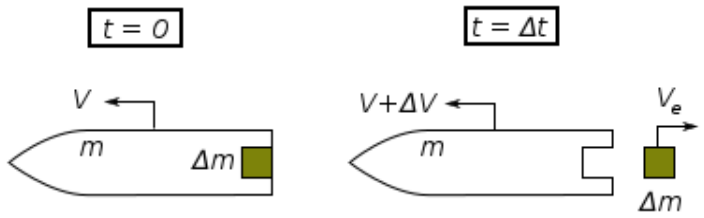

Figure 4.9. (Left) System of rocket and propellant at time $t = 0$. (Right) The system an instant later, after ejection of a small element Δm of combustion products.

The propellant ejection determines a momentum variation during Δt that can be expressed by the Newton's second law of motion as:

$$\sum F_i = \frac{P_2 - P_1}{\Delta t} \tag{4.3}$$

where P_1 is the momentum of the rocket at time t:

$$P_1 = (m + \Delta m)V$$

P_2 is the momentum of the rocket and exhausted mass at time $t + \Delta t$:

$$P_2 = m(V + \Delta V) + \Delta m V_e$$

and F_i is an external force, as for example atmospheric drag or solar pressure.
The velocity of the exhaust V_e in the observer frame is related to the velocity of the exhaust in the rocket frame v_e by (since exhaust velocity is in the negative direction):

$$V_e = V - v_e$$

Considering $dm = -\Delta m$ (since ejecting a positive Δm results in a decrease in mass) and $\sum F_i = 0$ (i.e. no external forces), the (4.3) yields:

$$\Delta V = v_e \ln \frac{m_0}{m_1} \tag{4.4}$$

where:

- ΔV (delta-v) is the maximum change of velocity of the vehicle;
- m_0 is the initial mass, including propellant;
- m_f is the final total mass without propellant;
- v_e is the effective exhaust velocity.

The parameter v_e is related to the thruster specific impulse I_{sp} and to the standard gravity g_0, by the following equation:

$$v_e = I_{sp} g_0$$

The Tsiolkovsky rocket equation has been derived by assuming that the body, whose motion is analyzed, is subject only to the action of the thrust exerted by the motor; it does not foresee the action of gravitational or aerodynamic forces. So, it would be exact only for the description of the rocket motion in the vacuum.
However, it can be effectively applied to orbital maneuvers in order to determine how much propellant is needed to change to a particular new orbit, or to find the new orbit as a result of a particular propellant burn.

4.6.2 In-space propulsion description

The region beyond Earth gravitational influence, until the Geostationary Earth Orbit (GEO) at 35.786 km above Earth surface, is defined as in-space. This region harbors all Earth monitoring systems, such as strategic communications assets, early warning, Earth observation, navigation, reconnaissance, surveillance and weather. Then, since we are dealing with a LEO spacecraft, in-space thrusters are considered in the study of the propulsion systems used for the autonomous mission. In general, their main functions are: primary propulsion, reaction control, station keeping, precision pointing, and orbital maneuvering. They divide into three main categories

1. chemichal: solid, liquid and hybrid;
2. cold gas;
3. electric.

Chemical propulsion

Thermodynamic/chemical thrusters are endothermic actuators since they do not need external power, but they exploit the propellant internal energy through combustion and chemical reactions. They can provide very high thrust as in the launch vehicles at the expense of a large ejected mass, since their ejection velocity is limited by the combustion/reaction temperature. Moreover, chemical thrusters are used in ON-OFF mode: force is delivered along much shorter intervals with respect to the interval between successive firing [21]. Then, since the main features are a high thrust capability, good levels of specific impulse and short ON mode time, they are used for abrupt and rapid maneuvers.

As reported in [20], depending on the type of fuel used, the chemical propulsion is divided into

- SOLID FUEL PROPULSION: in a solid rocket fuel grain, all the components required for vigorous combustion are mixed together and packed into a solid cylinder, as shown in Figure 4.10, into one substance. Once the combustion starts, it proceeds

until all the propellant is exhausted. There will be an oxidizer a fuel or some other solid hydrocarbon and an accelerant. When lit, the fuel grain will burn energetically, releasing a large volume of hot gases that are used to provide thrust.

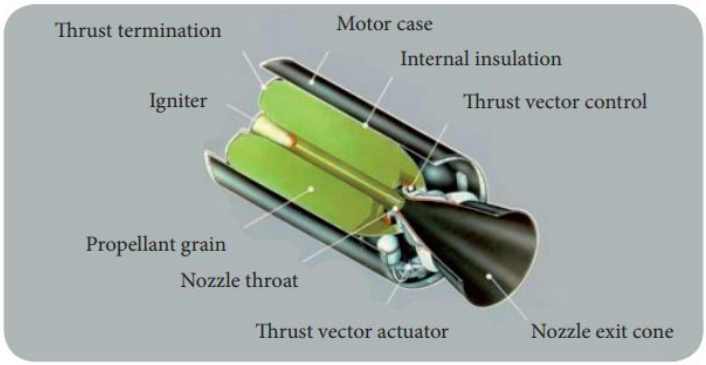

Figure 4.10. Solid chemical thrust

- LIQUID FUEL PROPULSION: propellant is comprised of two composites, i.e. fuel and oxidizer. They are stored separately in tanks in liquid phase and are pumped into the nozzle combustion chamber where burning occurs. Engine can stop the combustion and the thrust by turning off propellant flow. Liquid rockets tend to be heavier and more complex because of the pumps and storage tanks.

- HYBRID PROPULSION: as the name implies, it is a rocket with an engine which uses rocket propellants in two different phases - one solid and the other either gas or liquid. It consists of a pressure vessel (tank) containing the liquid oxidiser, the combustion chamber containing the solid propellant, and a mechanical device separating the two. When thrust is desired, a suitable ignition source is introduced in the combustion chamber and the valve is opened. The liquid propellant (or gas) flows into the combustion chamber where it is vaporized and then reacts with the solid propellant. Combustion occurs in a boundary layer diffusion flame adjacent to the surface of the solid propellant.

A kind of chemical thruster, which could be used to achieve the inclination changes for the autonomous guidance, is the **monopropellant hydrazine thruster**. The power for the propulsive reaction and resultant thrust is produced by the decomposition of hydrazine (N_2H_4) as it passes through a catalyst bed. These thrusters can operate in both steady state and pulse mode over a wide pressure range and are thus ideal for propulsion systems operating in blow-down mode. In Table 4.1,some units of monopropellant hydrazine thruster are reported.

	Thrust range (N)	Nominal specific impulse range (s)	Nominal mass flow range (g/s)
1N	0.320 ÷ 1.1	200 ÷ 223	0.142 ÷ 0.447
20N	7.9 ÷ 24.6	222 ÷ 230	3.2 ÷ 10.4
400N	120 ÷ 420	2080 ÷ 2155	58 ÷ 190

Table 4.1. Chemical monopropellant units produced by ArianeGroup [22]

Cold gas propulsion

A cold gas thruster is a propulsive device that uses pressurized inert gas as the reaction mass. The compressed gas is released through a propelling nozzle to generate a cold jet thrust. As shown in Figure 4.11, this thruster usually consists of simply a pressurized tank containing gas, a valve to control its release and a propelling nozzle, and plumbing connecting them. Because the gas is usually unheated, speed at the throat is low and very low performance is achieved. For this reason, cold gas propulsion is used for the following maneuvers/tasks:

- ultra-fine attitude control;
- make-up of solar radiation pressure and other environmental disturbances;
- real time drag-free control;
- formation flying maintenance/keeping/ mutual position control.

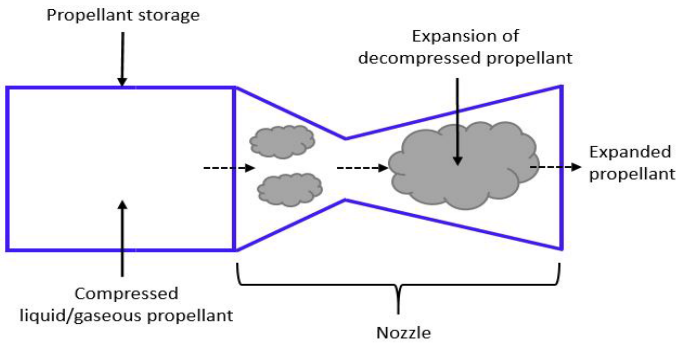

Figure 4.11. Cold gas propulsion

Electric propulsion

Electric propulsion is a technology aimed at achieving thrust with high exhaust velocities (usually greater than 10^3 m/s), which results in a reduction in the amount of propellant required for a given space mission or application compared to other conventional propulsion methods. This characteristic allows electrics rockets to run for a long time in a continuous mode. However, they generate a thrust of several orders of magnitude lower than the chemical engines (usually thrust-to-weight ratios not exceed 0.01) due to the current technological limits in the electrical power available on board the space probes. Then, since their main features are high fuel efficiency, very high specific impulse but low thrust capability, electric propulsion can be used for softer and longer maneuvers.

Electric propulsion systems for space use are typically grouped into three families, based on the adopted acceleration mechanism:

1. ELECTROTHERMAL PROPULSION: the electrothermal category groups the devices where electromagnetic fields are used to generate a plasma to increase the temperature of the bulk propellant. The acceleration mechanism (thermogasdynamic) is the same as in chemical propulsion systems: the propellant is led to a high temperature in a suitable space obtained in the thruster (similar to the combustion chamber) and then is allowed to expand through a nozzle. In this way the thermal energy supplied to the propellant is converted into kinetic energy and is therefore transformed into a useful form to generate a thrust. The main difference between electro-thermal and the chemical propulsion systems consists in the way in which the thermal energy is supplied to the propellant: in the first, in fact, the gas is heated or by resistances placed in direct contact with it (Resistojet) or through an arc electricity produced in the gas by the application of an appropriate potential difference (Arcjet). In general, performance of electrothermal systems in terms of specific impulse (I_{sp}) is somewhat modest (500 to 1000 seconds), but exceeds that of cold gas thrusters, monopropellant rockets, and even most bipropellant rockets.

2. ELECTROSTATIC PROPULSION: in electrostatic thrusters the propellant, after being ionized, is accelerated mainly by the *Coulomb force*, i.e. the application of a static electric field in the direction of the acceleration. The main electric propulsion systems based on electrostatic acceleration are

    - Gridded ion thruster, which uses high-voltage grid electrodes to accelerate ions with electrostatic forces as shown in Figure 4.12.
    - Hall-effect thruster, which is a type of ion thruster in which the propellant is accelerated by an electric field. It applies a magnetic field to limit the electrons' axial motion and then uses them to ionize propellant, efficiently accelerates the ions to produce thrust, and neutralizes the ions in the plume.

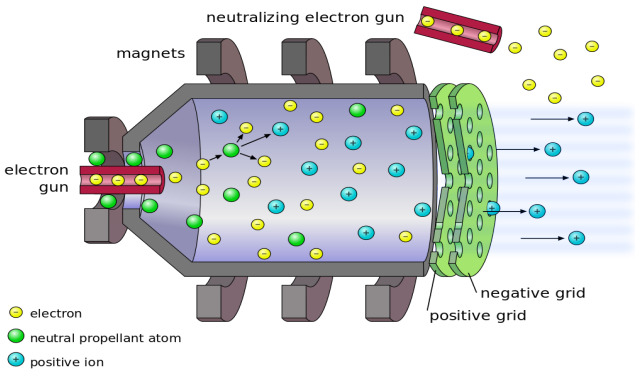

Figure 4.12. Gridded ion propulsion

3. ELECTROMAGNETIC PROPULSION: electromagnetic devices exploit the *Lorentz force* or the effect of electromagnetic fields (where the electric field is not in the direction of the acceleration) to accelerate the ions.

In recent years, propulsion systems are increasingly used in space mission. For example gridded ion thrusters are employed in the Artemis, GOCE and BepiColombo missions; in GOCE, they have the aim to provide real-time drag compensation to allow high accuracy measurement of the Earth gravitational field, while in BepiColombo, gridded ion thrusters will be used for the cruise to Mercury. Regarding the autonomous guidance, electric propulsion could be implemented in order to obtain greater efficiency of the consumed fuel; the only disadvantage would be an increase on the time required to complete the revisit and monitoring mission.

A kind of gridded ion thruster which would be used for the autonomous guidance is the **QinetiQ Electric Propulsion**. The main features of two classes of this system are reported in Table 4.2.

	Thrust range (mN)	**Specific impulse (s)**	**Power (W)**
T5	$0.6 \div 25$	> 3000	700
T6	$30 \div 230$	> 4000	5 k

Table 4.2. QinetiQ Kaufman Gridded Ion Engines [23]

Chapter 5

Simulation environment and results

In this chapter, on the basis of the mission scenario and corresponding strategies previously discussed, the simulation environment and the most significant results obtained on autonomous guidance and control will be presented.

5.1 Simulation environment design

To simulate the satellite behaviour during the autonomous mission, Simulink has been used. It is a software for modeling, simulation and analysis of dynamic systems, closely integrated with MATLAB.

Figure 5.1 shows the general scheme defined in Simulink for the simulations.

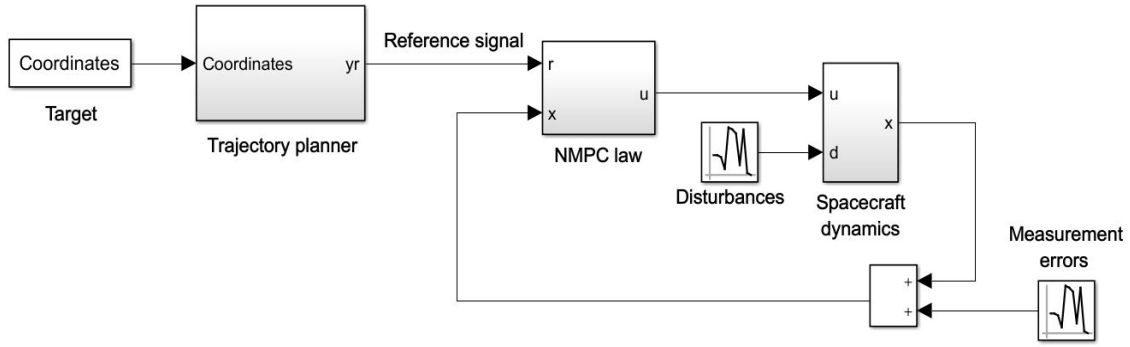

Figure 5.1. Simulation environment on Simulink

This scheme consists of four main blocks

1. TARGET: in this block the target coordinates are defined.
2. TRAJECTORY PLANNER: it includes the algorithm developed for the implementation of the corresponding strategy used for producing the desired trajectory.
3. SPACECRAFT DYNAMICS: this block contains the plant, i.e. a detailed description of the system through the use of state space representation.
4. NMPC LAW: the predictive control algorithm is implemented in this block.

In order to have a complete overview about how the simulation environment has been modeled, an accurate description of each block is reported below. Moreover, eventually, also disturbances and errors, considered in the control scheme, will be described.

5.1.1 Target block

As stated in Chapter 4, the strategies for autonomous guidance have been implemented considering alerts both in coordinates just flown over by the satellite and in generic positions on the Earth. In this block the target coordinates are reported. On the basis of them, the strategy implemented for carrying out the revisit and monitoring missions and the predictive control parameters tuning change.

5.1.2 Trajectory planner block

The planner has the aim to produce trajectories which can guide the satellite from the current/nominal orbit to the target orbit. The way in which these trajectories are defined depends on the kind of strategy used. In Chapter 4, the following ones have been presented:

1. Quasi-impulsive manuevers strategy.
2. Continuous maneuvers strategy.

So, in this block an algorithm is developed in order to implement the planner tasks; in particular, on the basis of the considered strategy, it contains all the steps required for the derivation of the trajectories. Among these, the main ones are:

- application of T_1 matrix for computing the orbit along which the satellite should observe the alert;
- application of T_2 matrix for considering the Earth rotation effect on the alert and then for finding the 'true' target orbits for revisit and monitoring;
- derivation of the reference trajectories for the fulfillment of the mission according to the chosen strategy.

Note that the reference trajectories are signals consist of position vectors $r \in \mathbb{R}^3$, defined instant by instant, which the satellite should meet during the autonomous mission. The reference signal y_r can be seen as the merger of the trajectories computed for the revisit, monitoring and return missions.

5.1.3 Spacecraft dynamics block

To simulate the Sentinel2 behaviour, a model can be defined as a mathematical description of physical properties of the system; usually this description is in terms of state space representation, i.e. a set of first order differential equations of the form:

$$\dot{\mathbf{x}}(t) = f[\mathbf{x}(t), \mathbf{u}(t); t]$$
$$\mathbf{y}(t) = h[\mathbf{x}(t), \mathbf{u}(t); t]$$

In Chapter 2, for the satellite orbit computation the free restricted two-body equation has been used:

$$\ddot{\mathbf{r}} + \mu \frac{\mathbf{r}}{r^3} = 0 \tag{FR2B}$$

Starting from this equation, two first order differential equations has been defined:

$$\dot{\mathbf{r}} = \mathbf{v} \tag{5.1}$$
$$\dot{\mathbf{v}} = -\mu \frac{\mathbf{r}}{r^3} \tag{5.2}$$

Finally, through a numerical integration, position and velocity values at each instant of time have been obtained.

If (FR2B) can be used to simulate, with a fair degree of accuracy, the satellite orbit, it is no longer suitable when, for the plant block, a detailed description of the spacecraft dynamics is required. Indeed NMPC performances are related to the model's ability to accurately describe the system to be controlled, since it adjusts the command input also taking into consideration the plant output. For this reason, it is necessary to introduce a new spacecraft model.

Orbit perturbations

Orbital dynamics is based on celestial mechanics, in particular on the Kepler's laws, which are empirical laws describing the motion of a body in unperturbed planetary orbits, and on the Newton's laws, which are general physical laws that imply the Kepler laws. Then, since the free restricted two-body equation has been derived on the basis of these laws, it can be used only to describe non-perturbed orbit, called *Keplerian orbits*.
A satellite in a real orbit, instead, is subject to perturbations:

- gravity potential harmonics perturbing the central force, due to an irregular mass distribution of planets (e.g. Earth polar flattening);
- third-body forces like those due to the Sun or Moon gravity;
- aerodynamic forces due to the residual atmosphere and wind at low-Earth orbits;
- solar/cosmic radiation;
- others, such as Earth radiation and tides, and spacecraft thermal radiation.

Since the Sentinel2 is in a low Earth orbit (LEO), among the perturbations reported above, drag is the most significant disturbing force. This perturbation can be modeled through the following equation:

$$\mathbf{F}_d = -\frac{1}{2}\rho C_D S |\mathbf{v}_{rel}| \mathbf{v}_{rel} \tag{5.3}$$

where

- ρ is the local atmospheric density;
- C_D, is the drag coefficient;
- S, is the spacecraft area projected along the direction of motion;
- $\mathbf{v}_{rel}$, is the relative velocity of the spacecraft with respect to the atmosphere; assuming a negligible atmospheric velocity, $\mathbf{v}_{rel} \cong \mathbf{v}$.

Moreover, the atmospheric density can be written by means of the barometric equation:

$$\rho(r) = \rho_0 \cdot \exp\left(-\frac{r - r_0}{H}\right)$$

where ρ_0 is the reference density, r_0 the height, H the scale height coefficient, while r is the distance from the planet center of mass.

Spacecraft model

To obtain an accurate and realistic model of the spacecraft, in addition to non-Keplerian perturbations, the mass variation due to firing of the engines must be take into account. It can be expressed through the classical Tsiolkovsky rocket equation:

$$m_f = m_0 \cdot \exp\left(-\frac{\Delta v}{v_e}\right) \tag{5.4}$$

where m_f is the mass at the end of the maneuver and m_0 is the initial mass.

After disturbances and mass variation have been defined, the next and final step is to introduce the model to be used in the simulation environment. It consists of a nonlinear MIMO system with seven state variables: six states (three position and three velocities) define the satellite motion and one state describes the change in mass due to propellant consumption during the manuevers. So the resulting spacecraft (S/C) model is:

$$\begin{aligned}
&\ddot{\mathbf{r}} = \mathbf{v} \\
&\ddot{\mathbf{v}} = -\mu\frac{\mathbf{r}}{r^3} + \frac{1}{m}(\mathbf{F}_d + \mathbf{d} + \mathbf{u}) \\
&\dot{m} = \begin{cases} 0 & \mathbf{u} = \mathbf{0} \\ -\mu/v_e & \mathbf{u} \neq \mathbf{0} \end{cases} \\
&\mathbf{F}_d = -\frac{1}{2}\rho\, C_D\, S\, v\, \mathbf{v}, \qquad \rho = \rho_0\, \exp\left(-\frac{r - r_0}{H}\right) \\
&\mathbf{u} = \mathbf{0} \qquad \text{if} \qquad m \leq m_b
\end{aligned} \tag{5.5}$$

where

- $\mathbf{x} = (\mathbf{r}, \mathbf{v}, m) \in \mathbb{R}^7$: state;
- $\mathbf{u} \in \mathbb{R}^3$: command input;
- $\mathbf{F}_d, \mathbf{d} \in \mathbb{R}^3$: disturbances;
- $\mathbf{y}$: output to control, that in the autonomous guidance it is the physical trajectory.

The quantities, which define the S/C model, can be divided in

- VARIABLES $\rightarrow$ $(\mathbf{r}, \mathbf{v})$: S/C position and velocity, $\mathbf{u}$: rocket thrust, $\mathbf{F}_d$: atmosphere drag, ρ: atmosphere density, $\mathbf{d}$: other disturbances, m: S/C mass (body + propellant)
- PARAMETERS $\rightarrow$ μ: gravitation parameters, C_D: drag coefficient, H: scale coefficient, S: S/C projected area, ρ_0 and r_0: reference density and height, v_e: effective exhaust velocity, m_b: body mass.

In Table 5.1 the parameters, which characterized the Sentinel-2 model, are reported.

Spacecraft Parameters	
Gravitation parameter μ	$4 \times 10^5\,\mathrm{km^3\,s^{-2}}$
Drag coefficient C_D	1
Scale coefficient H	8 km
S/C projected area S	$12\,\mathrm{m^2}$
Reference density ρ_0	$1.22\,\mathrm{kg\,m^{-3}}$
Body mass m_b	1016 kg

Table 5.1. Sentinel-2 parameters

Note that the effective exhaust velocity v_e is not indicated in Table 5.1; indeed, it changes based on the type of strategy used for the autonomous guidance.

5.1.4 NMPC law block

The NMPC law block includes the predictive control algorithm, that is the technique used to optimize the satellite maneuvers during the inclination changes. Its implementation requires the definition of:

- a prediction model;
- state and input constraints;
- the polynomial input parametrization m;
- sampling time (T_s), prediction (T_p) and control (T_c) horizon;
- weight matrices Q, P and R;
- tracking error tolerances;
- initial conditions of the system, i.e. $\mathbf{x}_0 = [\mathbf{r}_0, \mathbf{v}_0, m_0]$.

Among the items indicated above, prediction model, state constraints, and polynomial input parametrization remain unchanged and then they can be described a-priori without considering a specific situation. While since the other parameters change according to the mission scenario or strategy used, their design will be defined in section 5.2 when different cases will be taken into account.

Prediction model

For the prediction model a simplified and approximate version of (5.5) is used, not accounting for disturbances and mass variation during the prediction interval:

$$\begin{aligned} \ddot{\mathbf{r}} &= \mathbf{v} \\ \ddot{\mathbf{v}} &= -\mu \frac{\mathbf{r}}{r^3} + \frac{\mathbf{u}}{m} \\ \dot{m} &= 0 \end{aligned} \tag{5.6}$$

As seen in Chapter 3, it is important to use an approximate model which simplifies the description of the system, in order to increase the velocity of the prediction and optimization algorithm, but at the same time does not neglect the main features of its dynamics.

State constraints

On states, the following nonlinear constraint is considered:

$$X_c = \{x \in \mathbb{R}^7 : \|x(1:3)\| \geq 1.05 R_E\} \tag{5.7}$$

where R_E is the Earth mean radius and $x(1:3)$ is the state vector corresponding to the satellite position r. This constraint ensures that the S/C does not crash on Earth.

Polynomial input parametrization

The parametrization is an operation that transforms a space of infinite dimension into one of finite dimension equal to m. In general the smaller the parameter m, the lower the complexity of the algorithm. Since a low number of parameters is enough to obtain a satisfactory control performance, as polynomial input parametrization $m = 1$ is chosen.

5.1.5 Disturbances and errors

In the control scheme some perturbations are considered and simulated as random signals with zero mean and standard deviation (std) given as follows

- disturbances: $\text{std}(\mathbf{d}(t)) = 10^{-4}\, kg \cdot km/s^2$;
- measurement errors $\mathbf{w} = (\mathbf{w}_r, \mathbf{w}_v, \mathbf{w}_m)$:
  - position: $\text{std}(\mathbf{w}_r(t)) = 10^{-6}\, km$;
  - velocity: $\text{std}(\mathbf{w}_v(t)) = 10^{-8}\, km/s$;
  - mass: $\text{std}(\mathbf{w}_m(t)) = 10^{-6}\, kg$.

5.2 Simulation of the implemented strategies and obtained results

Considering a generic scenario, the S/C must perform autonomously the required maneuvers for visiting the point where an alert is triggered, in a time close to one orbit period. The further step is to persistent monitoring the target. Depending on the alert severity, the number of target inspection can change; in this work only one monitoring is considered. When the alert is over, the satellite must return to the nominal orbit. The main mission constraint is the required time to reach the coordinates. All the maneuvers are obtained through an algorithm able to generate trajectories for driving the satellite from the current orbit to the one whose ground track crosses the target. The definition of these trajectories is related to the type of strategy used: the first one is based on quasi-impulsive maneuvers, while the other on continuous maneuvers. In the first strategy, the S/C covers the initial orbit up to the node; then, after the node, the target orbit is followed. In the second strategy, the trajectory is obtained as an interpolation between the two reference orbits. In both the missions, the target orbit can be defined, thanks to the Sentinel-2 orbit characteristics, as a plane change of the nominal orbit.
Once that the different trajectories are computed, they are merged in a single signal which is used as the reference into the NMPC loop. The aim is to generate a control input which forces the satellite to follow the optimal trajectories defined in the reference.

To highlight the NMPC performances in comparison with the ideal and manual approaches, the main parameters characterizing the S/C orbit are analyzed considering three different cases according to the distance between the coordinates, in which the alert occurred, and the current position of the satellite:

1. coordinates just flown over by the satellite;
2. coordinates near the current area which is overflying by the satellite during the Earth observation mission;
3. coordinates far from the current area which is overflying by the satellite during the Earth observation mission.

Usually, as we will see, the revisit and return missions are more expensive than monitoring mission, because the alert can be in a generic coordinate on the Earth surface.

5.2.1 Quasi-impulsive maneuvers strategy

In the quasi-impulsive strategy, the autonomous guidance is implemented through the application of the same maneuvers performed in the ideal situation, in which an impulsive and instantaneous ΔV is applied in order to change the inclination of the orbit exactly in one of the two nodes with the best possible performances. However in the real case this type of maneuver is unfeasible since the thrust budget cannot be delivered in an infinitesimal time interval. Then, the idea is to use the NMPC for obtaining an optimal quasi-impulsive maneuver with performances analogue to the ideal case.

In order to obtain the best NMPC configuration, which can handle and optimize non-impulsive maneuvers, several trial and error simulations have been executed. The tuning procedure is related to the choice of the sampling time T_s, prediction horizon T_p, control horizon T_c, the weight matrices Q, P and R, the tracking error tolerance and the input constraints. Table 5.2 shows the design parameters chosen to guarantee a satisfactory trade-off between performances requirements.

NMPC Parameters	
Sampling time T_s	50 s
Prediction horizon T_p	500 s
Control horizon T_c	500 s
Q weight matrix	$0\mathbf{I}_3$
P weight matrix	$100\mathbf{I}_3$
R weight matrix	$1000\mathbf{I}_3$
Position tolerance	$[1,1,1]^T$ km
Input upper bound	$+T_{MAX}$
Input lower bound	$-T_{MAX}$

Table 5.2. NMPC parameters for quasi-impulsive strategy

The identity matrices of Q and P are constituted of three rows since the reference

signal is defined as the trajectories produced by the planner, i.e. the sets of position vector $r \in \mathbb{R}^3$ (as described in section 5.1.2). Also the weight matrix $R \in \mathbb{R}^{3\times3}$, since the command input is a vector of three elements. The values of matrix R are greater than those of matrix P since in this type of strategy the most important parameter to optimize is the fuel consumption; indeed, the goal is to obtain values quite analogue to the ideal ones. However the values of matrix P are very high, as the converging time to the target orbit is still an important parameter in the quasi-impulsive strategy. As far as concerned the prediction horizon T_p, a value of about a tenth of the orbital period was chosen. This allows the NMPC to increase the closed-loop stability properties and to know in advance the orbital changes. Moreover since T_p is not so high, the NMPC can react quickly enough to the inclination changes to implement at the nodes. The control horizon T_c is chosen equal to T_p, while the sampling time T_s is an order of magnitude lower than T_p. Indeed, since the satellite dynamics is not fast, values of tens of seconds can be assigned to T_s. In this way, numerical problems and slow computation are avoided. Other parameters setted for the NMPC design are the position tolerances; they define how much the predictive algorithm can 'fail' in producing the optimized trajectory with respect to the reference one. Regarding the input upper/lower bounds, they are related to the specific mission, i.e. to the distance between alert and current position of the S/C, and to the type of space propulsion used for the realization of the autonomous guidance; note that T_{MAX} represents the maximum thrust supplied by the engine. In Table 5.2, the initial conditions of the system are not indicated. Indeed they do not represent NMPC parameters, to be designed to optimize the control algorithm, but they define the conditions from which to start the simulation of the autonomous guidance. Then, their values will be indicated for each simulated situation. The last parameter to set is the *effective exhaust velocity*. For quasi-impulsive strategy $v_e = 21\,\mathrm{km\,s^{-1}}$ is chosen. Once all the parameters have been designed, the simulated results are shown, considering different alert positions.

Alert in coordinates just overflown by the satellite

This is the simplest situation to analyze as the orbit, required to overfly the corresponding target, is exactly the nominal one, i.e. the orbit on which the satellite is travelling. Then in this case, the application of T_1 matrix is not required. An example of this type of situation is reported in Figure 5.2.

For carrying out the revisit mission, the current satellite orbit must be rotated, through the application of T_2 matrix, for taking into account the angular displacement during the maneuvers. A similar procedure can be used to find the target orbit for the monitoring mission. Then an alarm in a coordinate just flown over by the satellite can be considered the simplest situation, indeed

- the procedures for obtaining the revisit and monitoring missions are quite similar; this means less fuel consumption
- the search algorithm for finding the target orbit is useless since it can be associated with the current orbit of the satellite; this mean less computational cost.

But simple does not means non-realistic; indeed this can be the most common scenario, with respect to an alert in a generic position, if the satellite is able to detect, through the acquired images, some anomalous event.

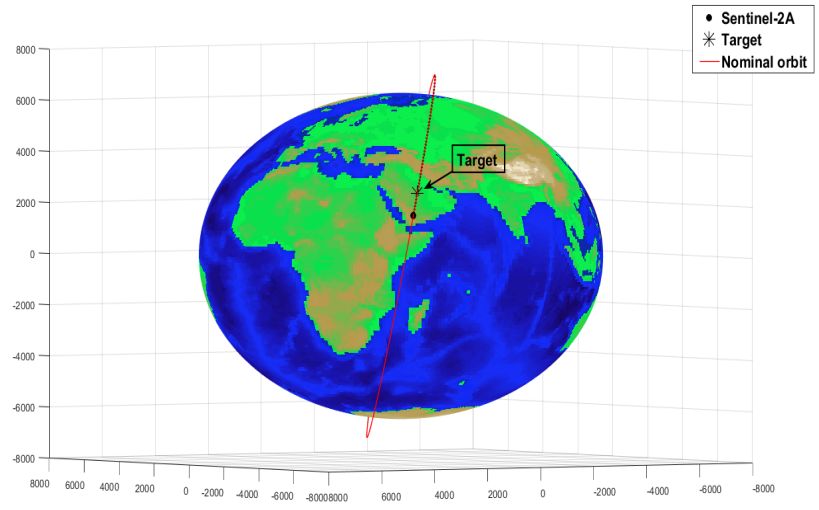

Figure 5.2. Alert in a position just overflown by the satellite

Figure 5.3 shows the trajectory produced by the planner for the realization of revisit and monitor missions.

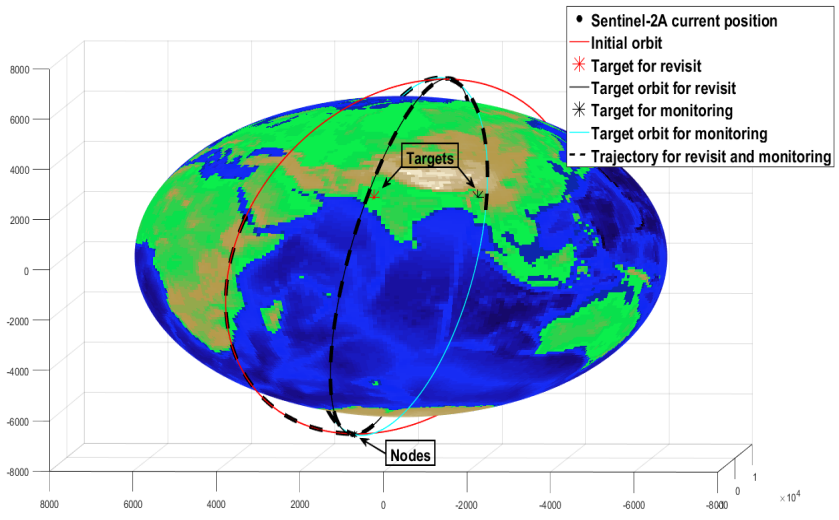

Figure 5.3. Reference trajectory for the revisit and monitoring missions

In order to highlight how this strategy operates, in Figure 5.4 the reference trajectory is divided in three sub-trajectories. As can be seen, the transfer from the initial orbit to

the target one always occurs near the node; in an ideal situation, i.e. with the possibility of generating instantaneous and impulsive ΔV, this guarantees the best possible performances, as gravity and misalignments losses are avoided.

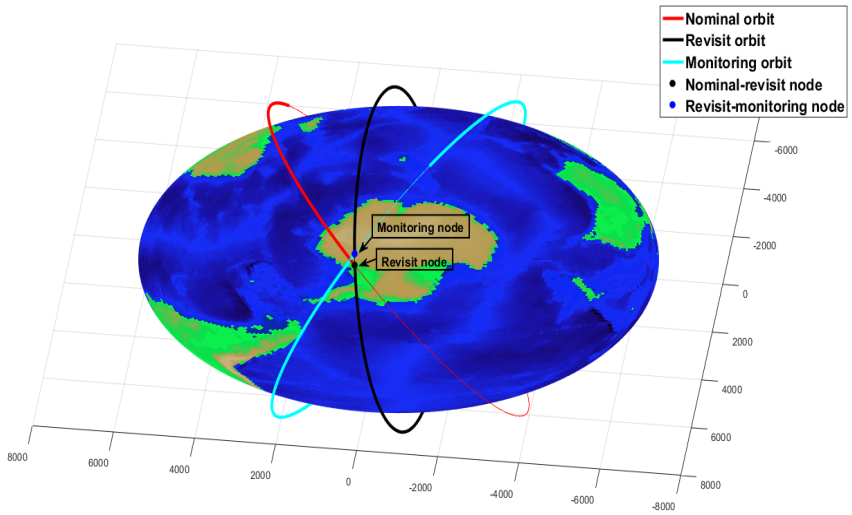

Figure 5.4. Nominal-revisit and revisit-monitoring orbital changes

When the monitor mission is over, the planner has to produce the required trajectory for returning to the the nominal orbit (Figure 5.5).

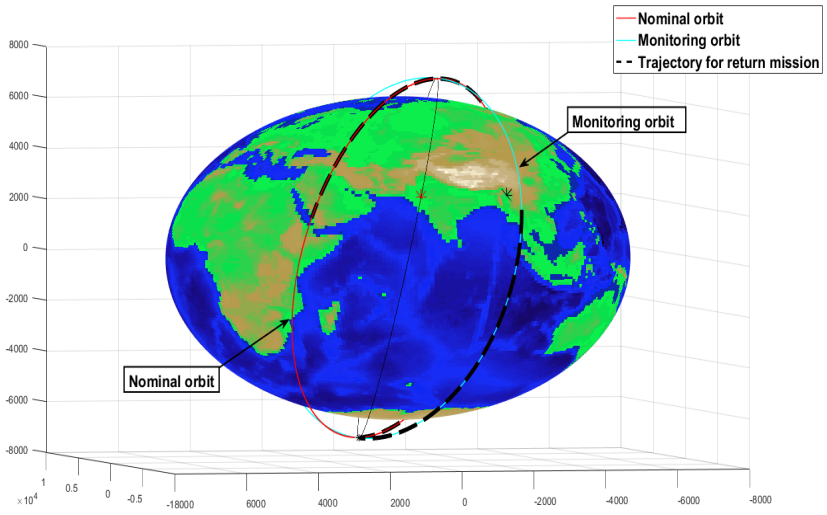

Figure 5.5. Monitoring-nominal orbital change for return mission

The procedure is the same as the other two missions; the nodes between the monitoring orbit and the nominal one are searched in order to carry out the orbital change in

one of them. It should be noted that the exact return to the initial orbit allows to preserve the sun-synchronous features of the Sentinel-2 orbit, in particular the Mean Local Solar Time (MLST) at the descending node equals to 10 : 30 am. Once the trajectories are produced, they can be used as the reference for the NMPC, which has the main aim to produce a control input for forcing the satellite to follow the planned trajectories. Before to execute the NMPC, the input upper/lower bounds and the initial conditions have to be indicated

- $T_{MAX} = 11\,\text{kN}$, then:
  - input upper bound=$+11\,\text{kN}$
  - input lower bound=$-11\,\text{kN}$;
- initial conditions:
  - $\mathbf{r}_0 = [-5.9756,\ 3.4753,\ 1.8555] \cdot 10^3$ km
  - $\mathbf{v}_0 = [-1.0736,\ 1.9665,\ -7.1324]$ km/s
  - $m_0 = 2200$ kg.

It should be noted that m_0 defines the initial mass of the S/C, that is body + propellant; since the dry mass of the satellite is 1016 kg, the fuel available for the completion of the mission is 1184 kg.

The results produced by the NMPC, designed with the parameters indicated in Table 5.2, are presented in the Figures 5.6, 5.7, 5.8 :

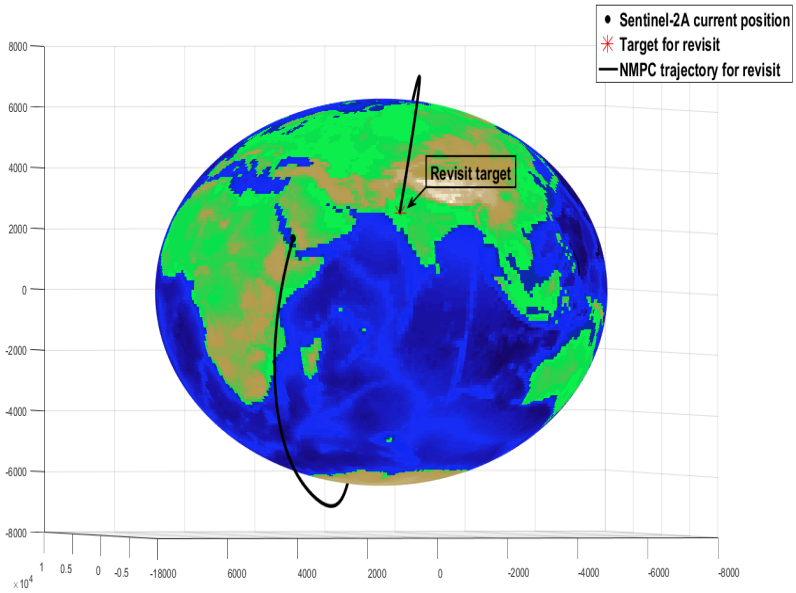

Figure 5.6. NMPC optimized trajectory for revisit mission

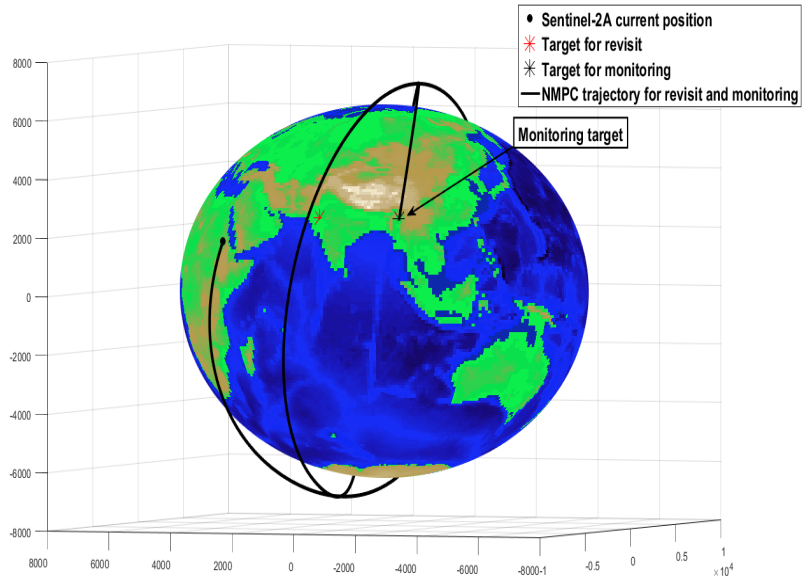

Figure 5.7. NMPC optimized trajectory for revisit and monitoring missions

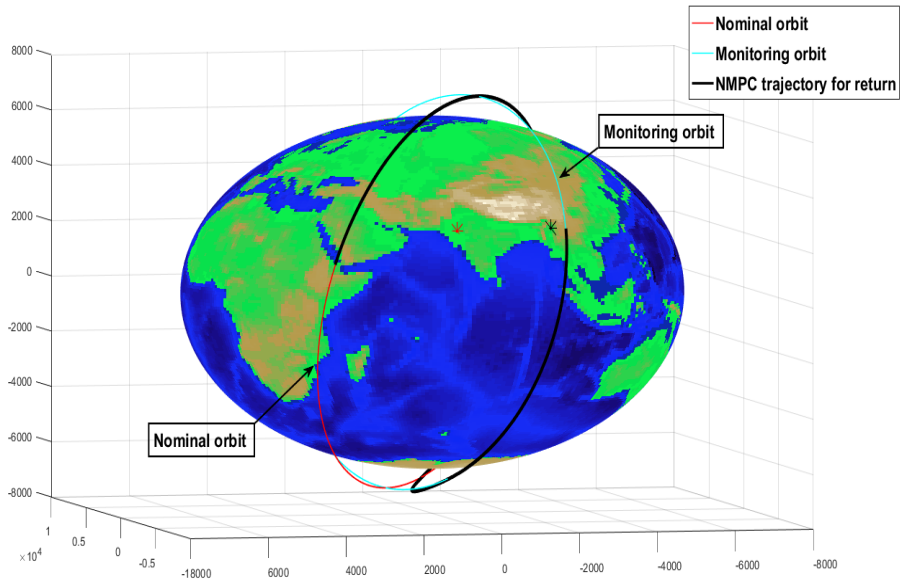

Figure 5.8. NMPC optimized trajectory for return missions

In Figure 5.9 the optimum quasi-impulsive maneuvers produced by the NMPC for the revisit and monitoring autonomous guidance are shown .

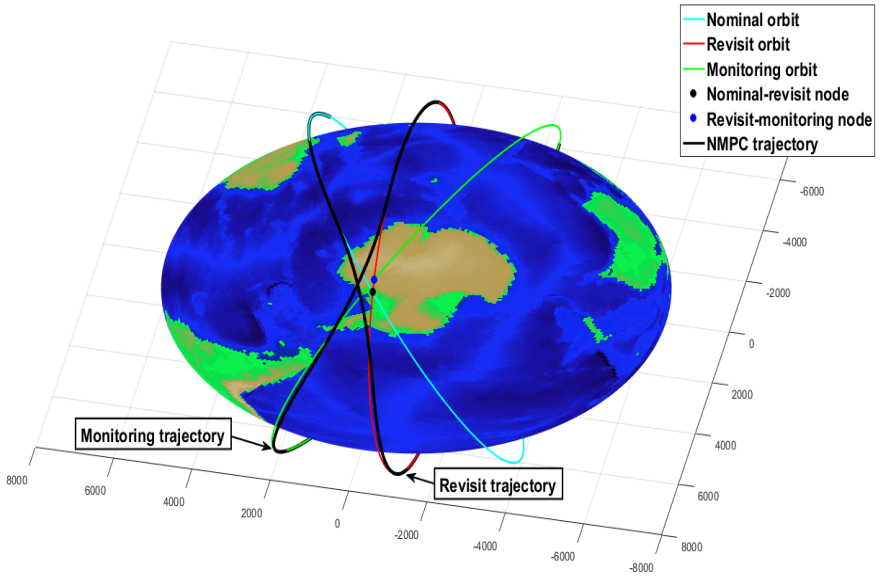

Figure 5.9. Non-impulsive NMPC maneuvers near the nodes

As can be seen, the NMPC is able to carry out the revisit and monitoring missions with continuous maneuvers applied to the satellite for times longer than 1000 s; although the orbital changes are not instantaneous, the required time constraints to overfly the targets are still respected. For this situation, since the maneuvers for carrying out the autonomous guidance are not so sharp, it is possible to simulate a manual guidance. In this way, a comparison between standard mission planning designed on ground and autonomous mission planning designed through NMPC technique can be presented. The manual mission is obtained through the application of the maximum thrust used for accomplishing the autonomous guidance, that is $T_{MAX} = 11$kN. As in the quasi-impulsive strategy, also for this manual approach the maneuvers for orbital changes are not instantaneous and they are applied when the S/C is near to the node. The idea is to consider the same situation of the autonomous case, in order to evaluate the differences between the two methods. Clearly, in the manual planning the obtained orbit is not exactly the desired orbit, that is the target one, as there are small errors especially for the eccentricity; this is due to not so high accuracy of this approach. While, using the NMPC, the S/C is guided perfectly and autonomously on the target orbit.

In Figures 5.10 and 5.11 the main performances characterizing autonomous guidance for the above alert, are plotted. In particular in Figure 5.10 the NMPC propellant consumption is compared both with the manual case and with ideal case, which expects the best performance possible; indeed, as mentioned before, in the ideal situation the orbital changes are performed through impulsive and instantaneous maneuvers. In the Figure 5.11, instead, NMPC and manual performances are compared in terms of ΔV required to accomplish the mission.

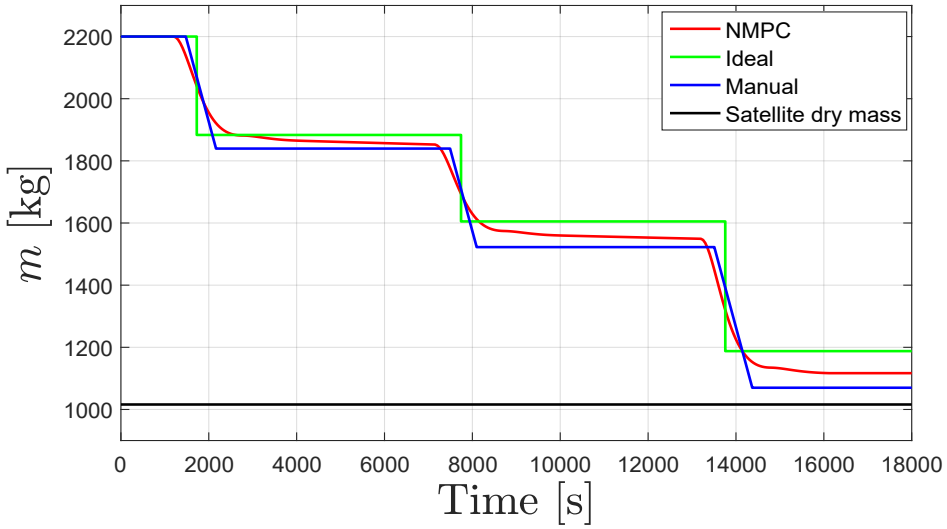

Figure 5.10. Satellite Mass Variation Comparison

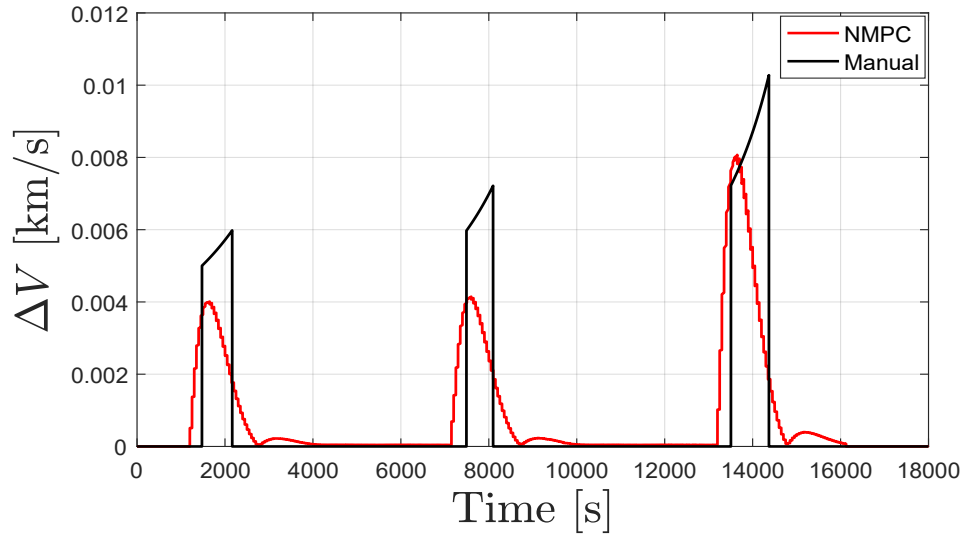

Figure 5.11. ΔV comparison

The simulated results in Figure 5.10 show that the NMPC strategy is almost able to reproduce the same performance of the ideal case, with a small worsening due to the engines action delivery in a finite but small amount of time (few minutes). However the application of impulsive and instantaneous ΔV are unfeasible in real application. Then, more significant and consistent results can be obtained if autonomous guidance is compared with the standard manual approach. In this case with NMPC approach

better performances are achieved both for the used ΔV to performed the maneuvers and for the propellant consumption. In Tables 5.3, 5.4 and 5.5 the performances of NMPC strategy, ideal approach and standard mission planning are summarized, reporting also the required time for the revisit and the monitoring missions.

	Fuel Consump. Revisit (kg)	**Fuel Consump. Monitoring (kg)**	**Fuel Consump. Return (kg)**
Ideal Maneuvers	316.7	278.3	417.3
Quasi-Impulsive Maneuvers	332.4	292.8	436.5
Manual Maneuvers	360.5	317.1	452.5

Table 5.3. Fuel consumption of the proposed approaches

	ΔV Revisit (km/s)	**ΔV Monitoring (km/s)**	**ΔV Return (km/s)**
Ideal Maneuvers	3.2635	3.3577	6.3234
Quasi-Impulsive Maneuvers	3.4403	3.5803	6.8159
Manual Maneuvers	3.7583	3.9727	7.4069

Table 5.4. Required ΔV for the completion of the mission

	Revisit Time (s)	**Monitoring Time (s)**
Ideal Maneuvers	5835	5955
Quasi-Impulsive Maneuvers	5836	5955
Manual Maneuvers	5838	5960

Table 5.5. Time for revisit and monitor the target

It can be noted that with the NMPC there is a fuel saving of about 70 kg with respect to the manual approach; also the required ΔV is less in the autonomous guidance and control. Moreover the performance of the manual mission planning get worse if the orbital changes become more sharp, as can be seen for the return mission.

In addition, in Figure 5.12 the control action produced by the NMPC is shown; this plot is useful to describe how the NMPC works for obtaining a command activity in the range defined by the user, that in this case is between -11 kN and $+11$ kN.

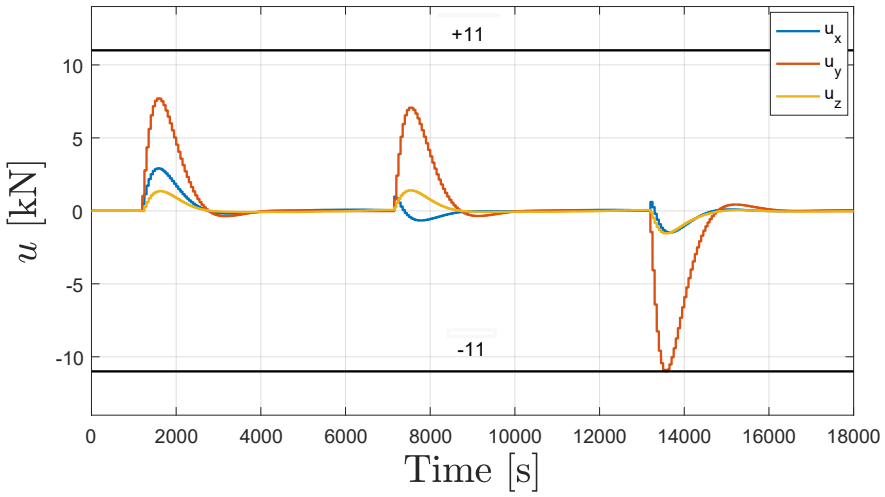

Figure 5.12. Quasi-Impulsive Maneuvers Control Action

Finally, the errors on orbital parameters are shown in Figure 5.13 in order to evaluate the effectiveness of the controller. Note that a greater simulation time is considered for highlighting the tracking error performances of the NMPC.

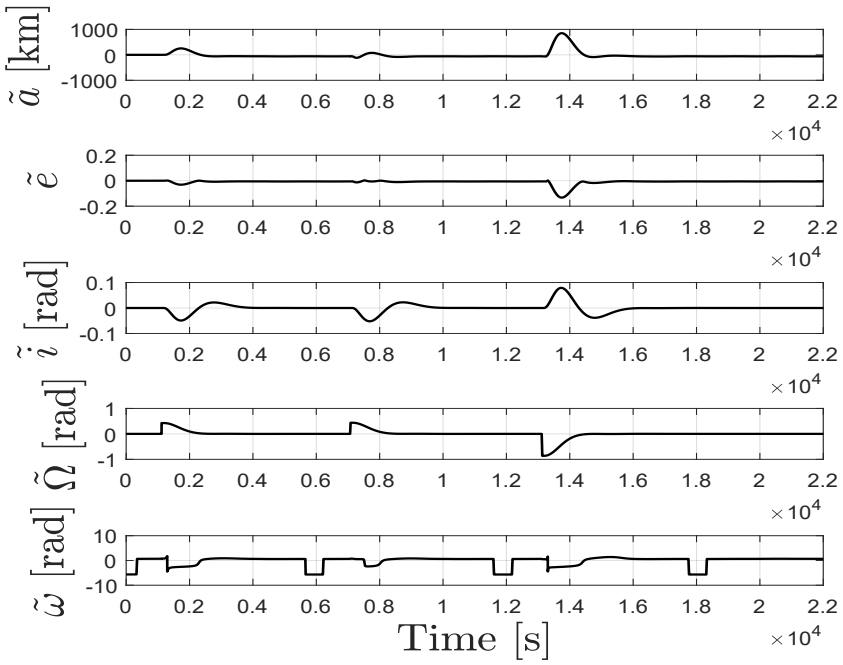

Figure 5.13. Tracking Error in Terms of Orbital Parameters.

Alert near the current area flown over by the satellite

In this case a generic alert, occurred in coordinates near the area which the satellite is flying over during the Earth observation mission, is considered. This means that the maneuvers, required to perform the revisit and return are not much more sharp with respect to those obtained in the previous situation. An example of this type of alert is reported in Figure 5.14.

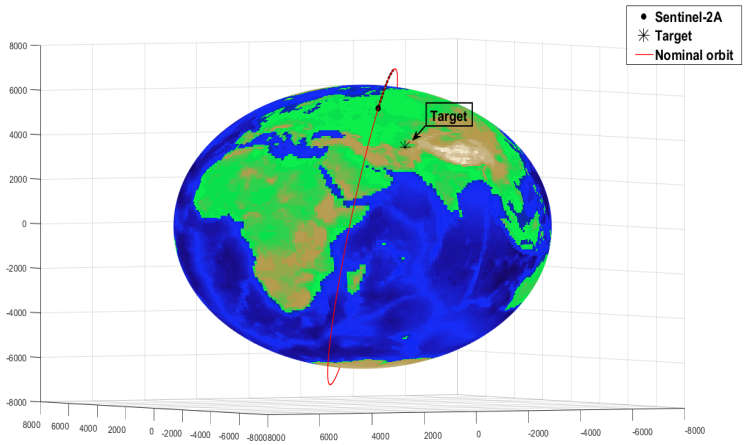

Figure 5.14. Alert near the current satellite position

Once the coordinates in which the alert occurred has been defined, since it is located in a generic position on the Earth, the corresponding orbit along which the satellite can observe the event must be computed. To get it, the search algorithm defined in Chapter 4 has to be used. It works applying the T_1 rotation matrix for the properly number of time. The resulting orbit is shown in Figure 5.15.

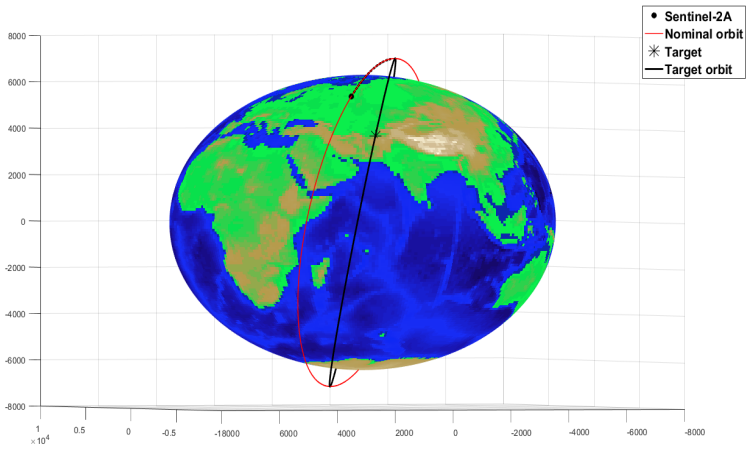

Figure 5.15. Resulting target orbit

However the computed orbit is not the revisit target orbit; indeed, since a certain amount of time elapses between when the alarm triggers the software and when the revisit or monitoring mission is completed, due to Earth's rotation, there is a coordinate displacement. It can be taken into account through the application of T_2 matrix. The target orbit required for the fast revisit of the event is reported in Figure 5.16.

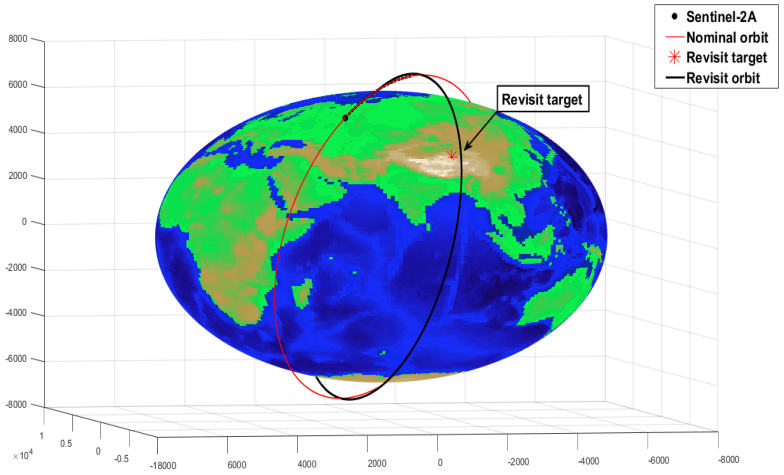

Figure 5.16. Orbit for the revisit mission

A similar procedure can be used to find the orbit for the monitoring mission. Once the target orbits are computed, the planner can produce the required trajectory for the realization of the autonomous guidance as shown in Figure 5.17.

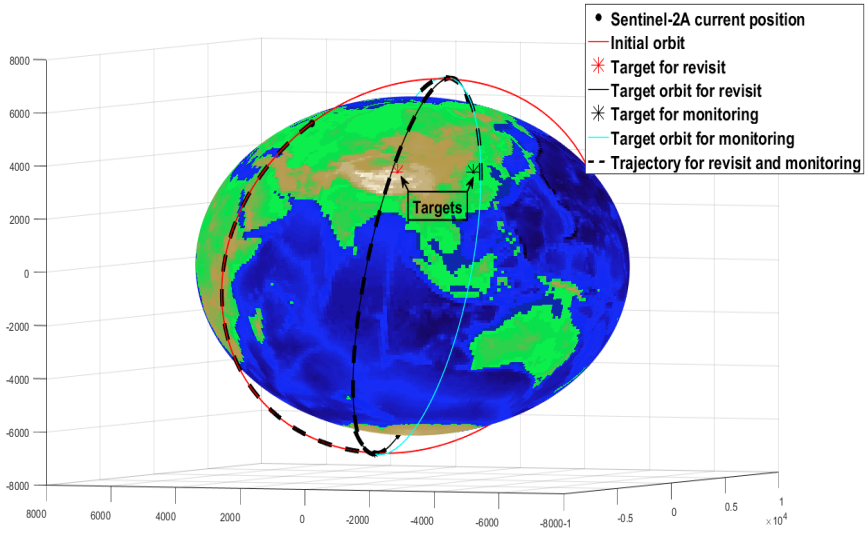

Figure 5.17. Reference trajectory for the revisit and monitoring missions

When the monitor mission is over, the planner has to produce the required trajectory for returning to the the nominal orbit. Then the results of the planner can be used as the reference for the NMPC. The input upper/lower bounds and the initial conditions, different than the before situation, are

- $T_{MAX} = 14\,\text{kN}$, then:
  - input upper bound=$+14\,\text{kN}$
  - input lower bound=$-14\,\text{kN}$;
- initial conditions:
  - $\mathbf{r}_0 = [-4.2398,\ 1.7129,\ 5.5049] \cdot 10^3$ km
  - $\mathbf{v}_0 = [-4.5279,\ 3.7209,\ -4.6435]$ km/s
  - $m_0 = 2600$ kg.

The initial mass and the maximum thrust are greater than the previous situation, since the maneuvers for revisit and return are more sharp. The results produced by the NMPC for the revisit and monitoring missions are shown in Figure 5.18.

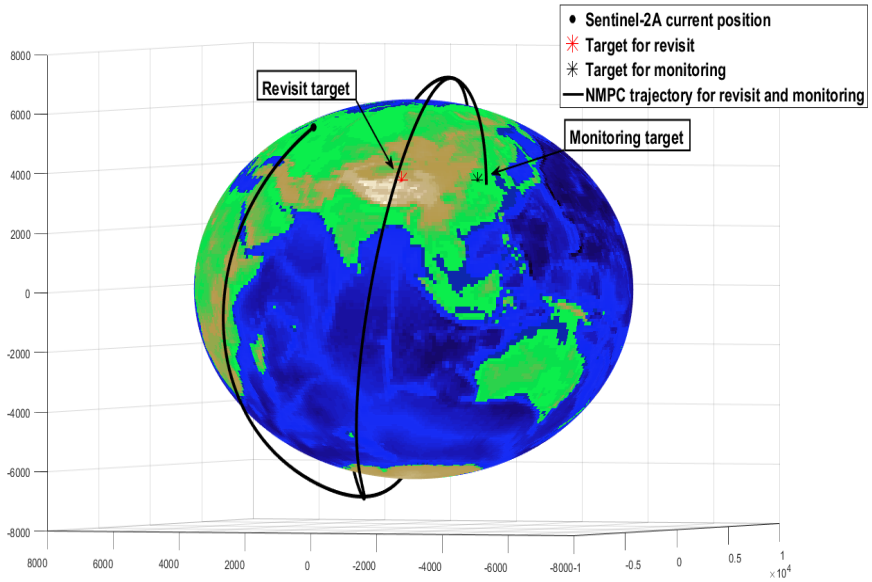

Figure 5.18. NMPC optimized trajectory for the revisit and monitoring missions

As can be seen, the NMPC is able to carry out the revisit and monitoring missions with continuous maneuvers applied to the satellite. Also for this situation the main performances characterizing autonomous guidance and control are compared with ideal and manual approaches in order to evaluate whether the results found in the previous case are confirmed or not. In Figure 5.19 the fuel consumption characterizing autonomous guidance, ideal and manual cases are plotted, while in Figure 5.20 NMPC and manual approaches are compared in terms of required ΔV.

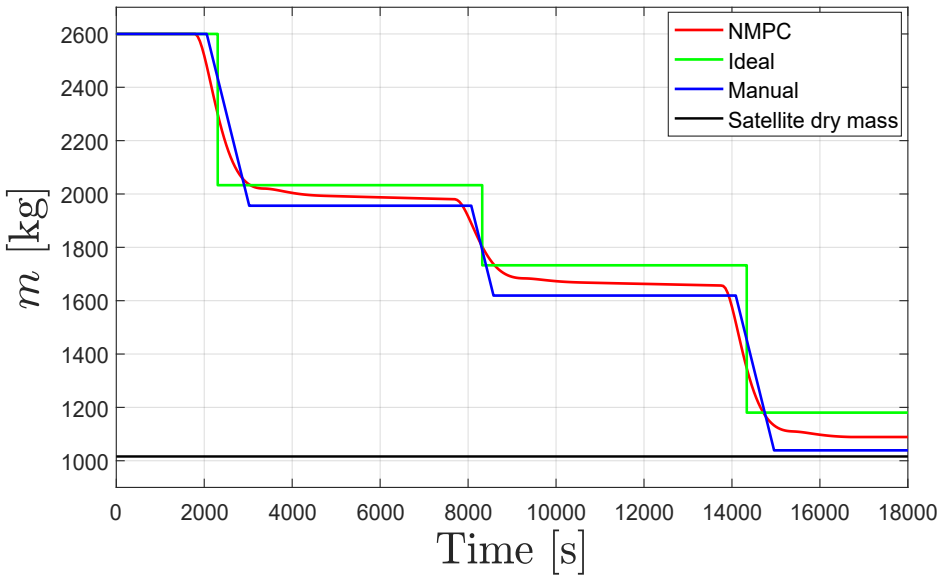

Figure 5.19. Satellite mass variation comparison

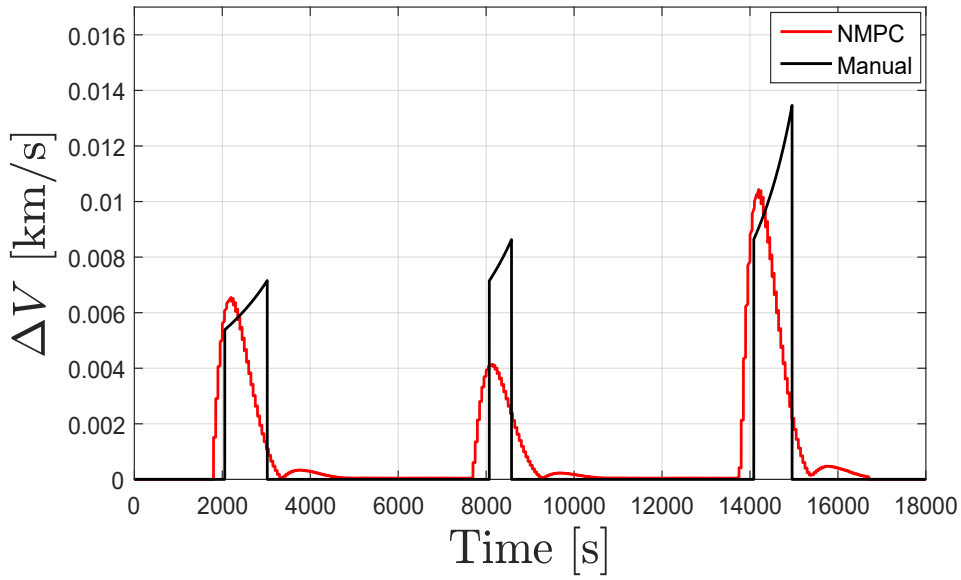

Figure 5.20. ΔV comparison

The results found in the previous situation are also supported in this case where a target in a generic position on the Earth is considered; indeed Figure 5.19 shown that NMPC produces better performances than the manual approach and nearly the same

ideal level as the impulsive strategy. This means that the optimum profile produced by NMPC, in order to accomplish the autonomous guidance, does not degrade with the increasing complexity of the maneuvers to be performed. The results in Tables 5.6, 5.7 and 5.8 confirm what is reported in the previous graphs.

	Fuel Consump. Revisit (kg)	**Fuel Consump. Monitoring (kg)**	**Fuel Consump. Return (kg)**
Ideal Maneuvers	566.9	300.4	552.3
Quasi-Impulsive Maneuvers	602.5	309.9	569.7
Manual Maneuvers	638.9	337.9	581.6

Table 5.6. Fuel consumption of the proposed approaches

	ΔV Revisit (km/s)	**ΔV Monitoring (km/s)**	**ΔV Return (km/s)**
Ideal Maneuvers	5.1655	3.3569	8.0603
Quasi-Impulsive Maneuvers	5.5363	3.5402	8.6493
Manual Maneuvers	5.9263	3.9684	9.3178

Table 5.7. Required ΔV for the completion of the mission

	Revisit Time (s)	**Monitoring Time (s)**
Ideal Maneuvers	6244	5955
Quasi-Impulsive Maneuvers	6245	5955
Manual Maneuvers	6248	5960

Table 5.8. Time for revisit and monitor the target

For this generic alert, with NMPC approach there is a fuel saving of about 80 kg and a reduction of the required ΔV of more than 8% with respect to the manual one. Moreover in Table 5.7, it should be noted that the manual simulated mission planning determines an increase in the amount of ΔV used when the required maneuvers for orbital changes become more sharp, as already seen in the previous case.

In Figure 5.21 the control action produced by the NMPC is shown.

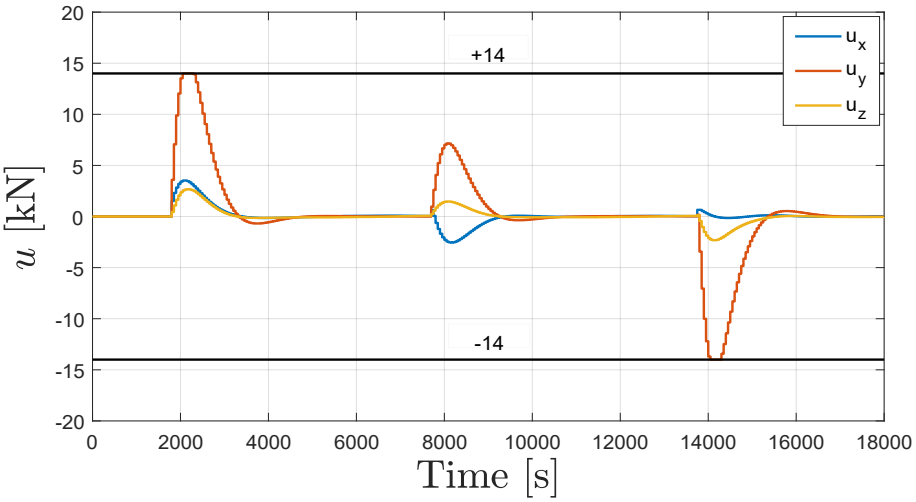

Figure 5.21. Quasi-Impulsive Maneuvers Control Action

Finally, the errors on orbital parameters are reported in Figure 5.22 in order to evaluate the effectiveness of the controller.

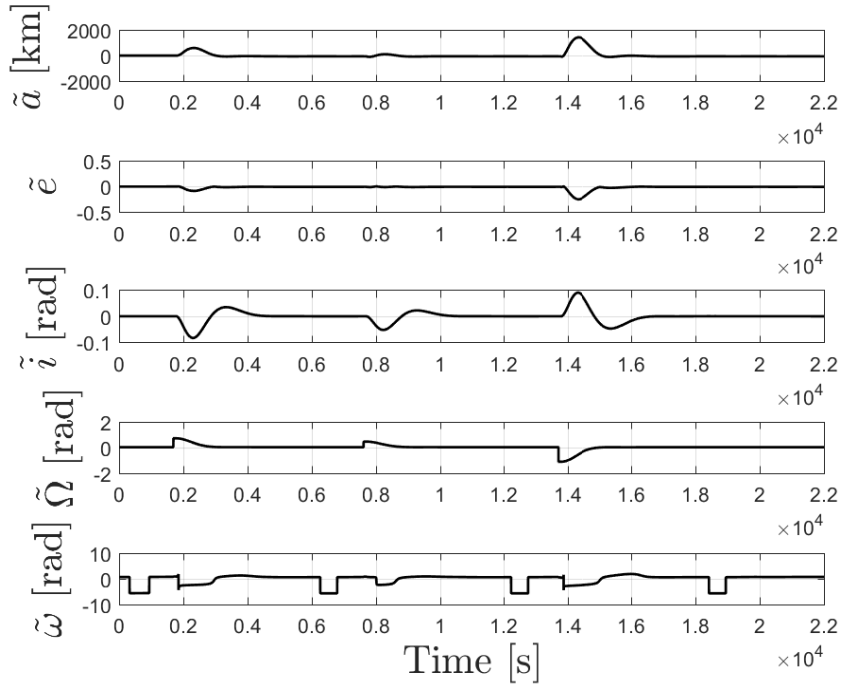

Figure 5.22. Tracking Error in Terms of Orbital Parameters.

Alert far from the current satellite position

In this last case an alert, occurred in coordinates far from the current area flown over by the satellite, is considered. An example of this type of alert is reported in Figure 5.23.

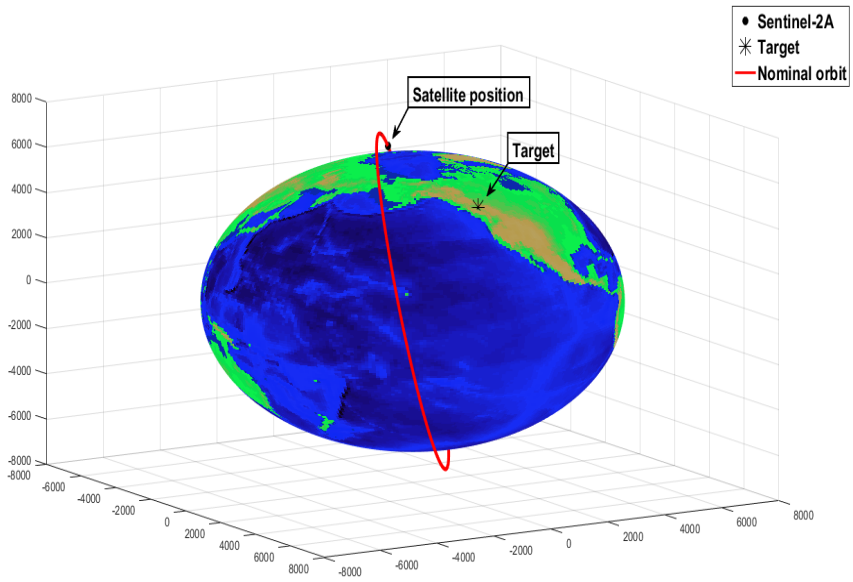

Figure 5.23. Alert in coordinates far from the current satellite position

As can be seen, the target is located in an Earth's portion opposite to the current position of the satellite. Then the maneuvers, required to perform the autonomous guidance and control, can be more sharp than the previous situations. For this reason, the maximum thrust and the satellite initial mass are greater than the previous alert. In particular:

- $T_{MAX} = 16\,\text{kN}$, then:
  - input upper bound=$+16\,\text{kN}$
  - input lower bound=$-16\,\text{kN}$;
- initial conditions:
  - $\mathbf{r}_0 = [-4.2398,\ 1.7129,\ 5.5049] \cdot 10^3$ km
  - $\mathbf{v}_0 = [-4.5279,\ 3.7209,\ -4.6435]$ km/s
  - $m_0 = 2800$ kg.

The results produced by the NMPC for the revisit and monitoring missions are shown in Figure 5.24.

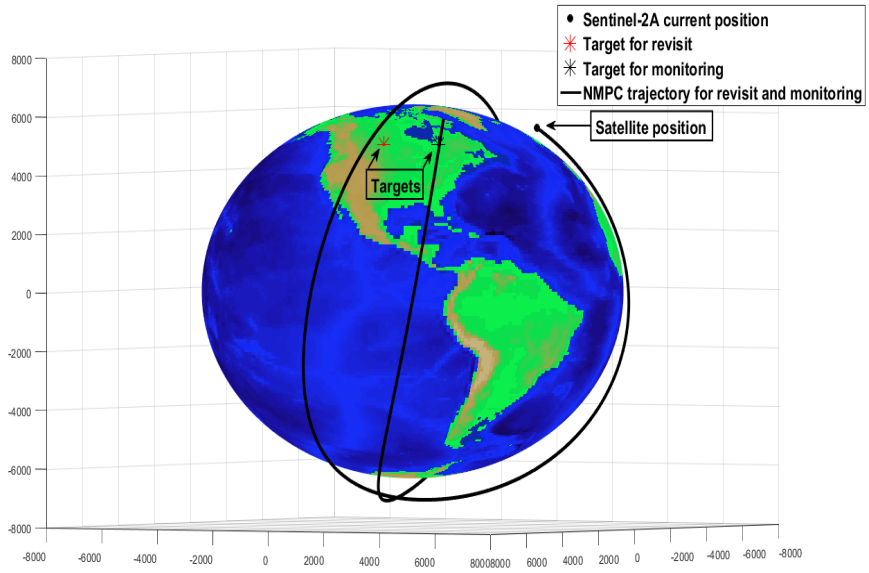

Figure 5.24. NMPC optimized trajectory for the revisit and monitoring missions

In Figure 5.25 the fuel consumption characterizing autonomous guidance for the above alert, are plotted and compared only with ideal case.

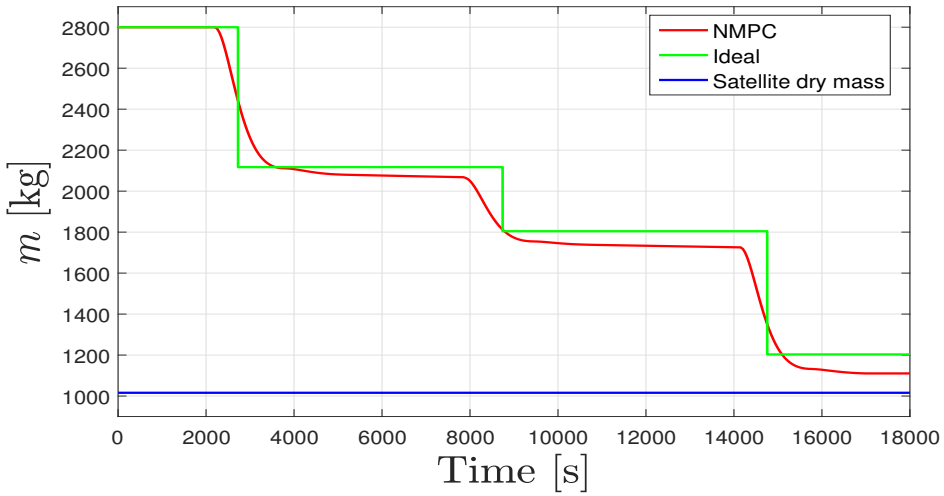

Figure 5.25. Satellite mass variation comparison

Also in this last situation, where the maneuvers for accomplishing the revisit, monitoring and return missions are very sharp, the NMPC can guarantee nearly the same optimality level as the ideal impulsive strategy. This means that the optimization process is not affected by mission complexity.

5.2.2 Continuous maneuvers strategy

This strategy is based on the definition of a trajectory that instant by instant guides the spacecraft with a continuous change of inclination, obtained by an interpolation between the initial and target orbit, allowing to perform smooth maneuvers. In this case, a completely different strategy is considered with respect to the ideal situation; indeed the inclination change is not applied at the nodes, but it is defined through a small orbital inclination at each instant of time. This approach can be advantageous in terms of command activity but at the same time the ΔV required and the propellant consumption considerably increase. As already mentioned in Chapter 4, with this approach the NMPC performances, in case of application of the control input continuously throughout the mission, can be analyzed and exploited for the development of a future low-thrust maneuver strategy.

As in Subsection 5.2.1, also in this case the best NMPC configuration is obtained through a trial and error procedure. Table 5.9 shows the design parameters chosen to guarantee a satisfactory trade-off between performances requirements.

NMPC Parameters	
Sampling time T_s	50 s
Prediction horizon T_p	800 s
Control horizon T_c	800 s
Q weight matrix	$0\mathbf{I}_3$
P weight matrix	$1000\mathbf{I}_3$
R weight matrix	$100\mathbf{I}_3$
Position tolerance	$[1,1,1]^T$ km
Input upper bound	$+T_{MAX}$
Input lower bound	$-T_{MAX}$

Table 5.9. NMPC parameters for continuous strategy

The values of matrix P are greater than those of matrix R since in this type of strategy, unlike the quasi-impulsive one, the most important parameter to optimize is the converging time, instant by instant, to the reference trajectories. However the values of matrix R are very high, as the fuel consumption is an important parameter especially

because with a continuous autonomous guidance the gravity and misalignments losses are greater than the ideal and quasi-impulsive approaches. As far as concerned the prediction horizon T_p, an higher value than the quasi-impulsive strategy is chosen in order to improve the closed loop stability and to have a broad view of the reference trajectory. Regarding the input upper/lower bounds, they are related to the specific mission, as well as the initial condition of the system. For this reason they will be indicated for each simulated situation.

The last parameter to set is the *effective exhaust velocity*. For this type of strategy $v_e = 25\,\mathrm{km\,s^{-1}}$ is chosen. In continuous maneuvers approach the exhaust velocity must be higher than the previous strategy, since orbital inclination changes applied continuously at each instant of time required an higher specific impulse. Once all the parameters have been designed, the simulated results are shown, considering the same alert positions of quasi-impulsive strategy.

Alert in coordinates just overflown by the satellite

Figure 5.26 shows the trajectory produced by the planner for the realization of revisit and monitor missions.

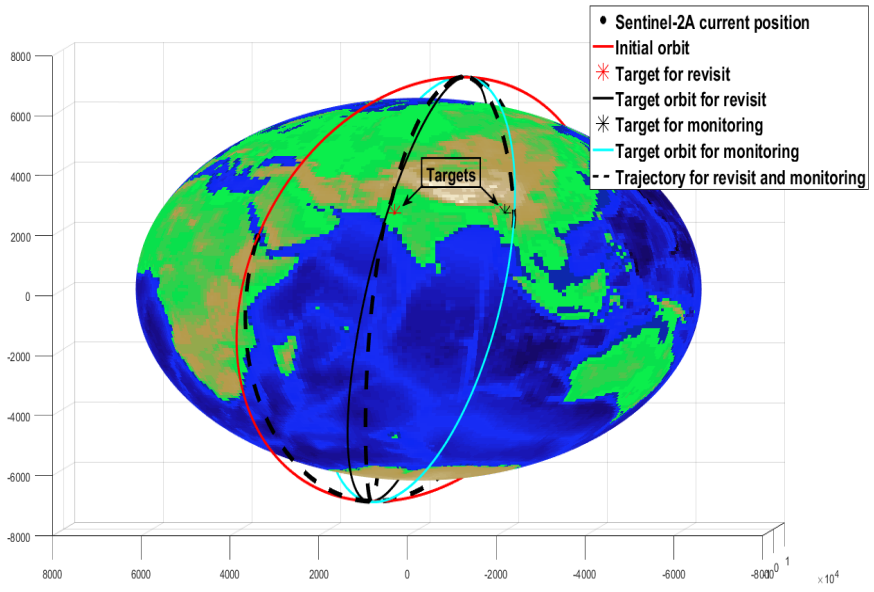

Figure 5.26. Reference trajectory for the revisit and monitoring missions

It should be noted that this strategy defines the reference trajectory as an interpolation between the initial and final orbits. In order to highlight this concept, in Figure 5.27 is reported also the part of the trajectory not shown in the previous figure.

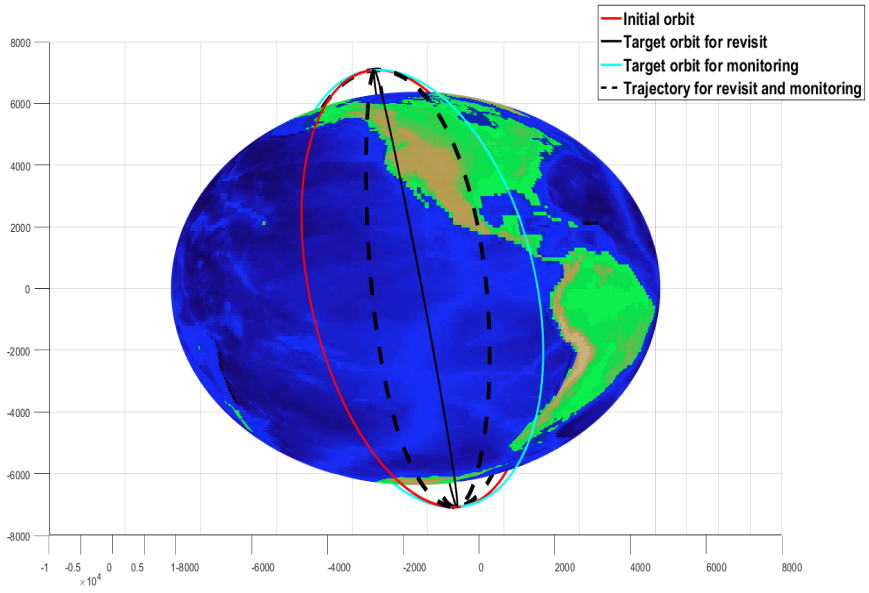

Figure 5.27. Other view of the reference trajectory for the revisit and monitoring missions

Once the trajectories are produced, they can be used as the reference for the NMPC, which has the main aim to produce a control input for forcing the satellite to follow the planned trajectories. Before to execute the NMPC, the input upper/lower bounds and the initial conditions have to be indicated

- $T_{MAX} = 4\,\text{kN}$, then:
  - input upper bound=$+4\,\text{kN}$
  - input lower bound=$-4\,\text{kN}$;
- initial conditions:
  - $\mathbf{r}_0 = [-5.9756,\ 3.4753,\ 1.8555] \cdot 10^3$ km
  - $\mathbf{v}_0 = [-1.0736,\ 1.9665,\ -7.1324]$ km/s
  - $m_0 = 2200$ kg.

With this strategy the required thrust is lower than quasi-impulsive one, since it is based on small and continuous inclination changes. However, this leads to a worsening of performance in terms of fuel saving. The results produced by the NMPC, designed with the parameters indicated in Table 5.9, are presented in Figure 5.28.

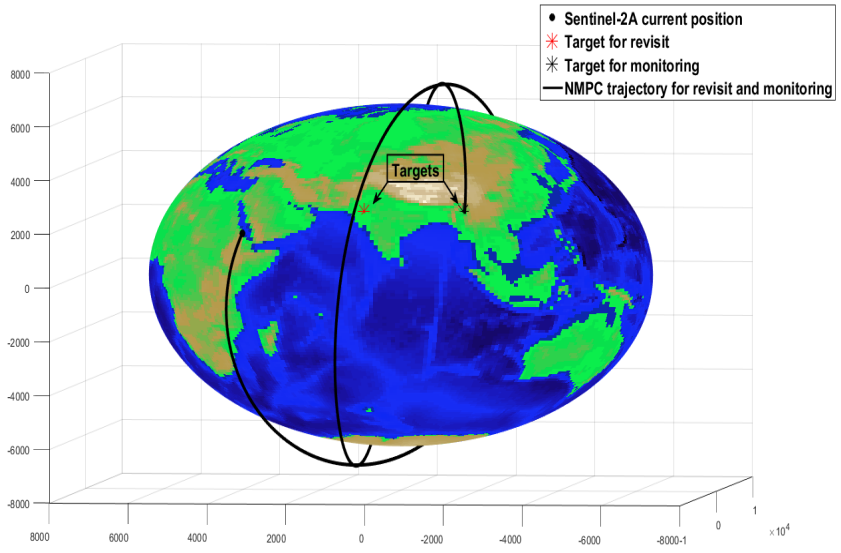

Figure 5.28. NMPC optimized trajectory for the revisit and monitoring missions

The NMPC is able to carry out the revisit and monitoring missions even if the maneuvers for the orbital changes are not applied in a quasi-impulsive way to the node, but through continuous and small inclination changes for the whole duration of the mission. In order to evaluate the performances produced by the NMPC technique, they are compared with the ideal and manual approaches as done in the previous strategies. In particular, in Figure 5.29 the fuel consumption characterizing autonomous guidance, ideal and manual cases are plotted.

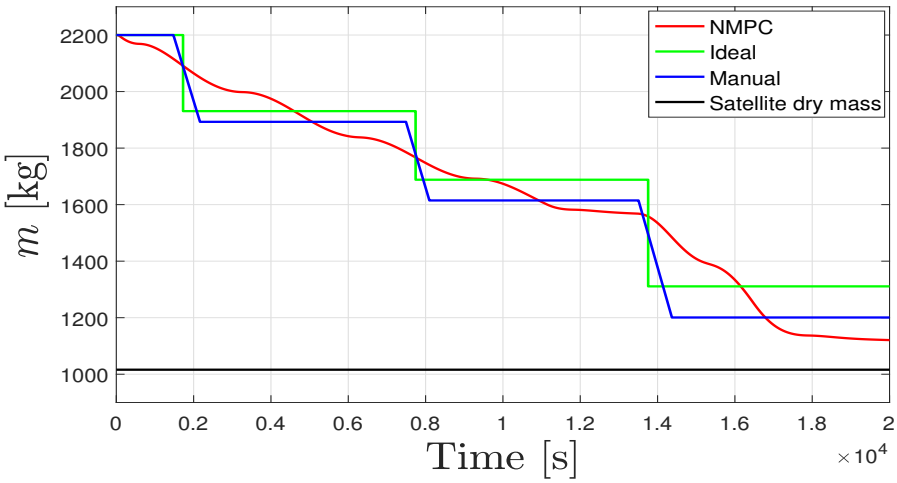

Figure 5.29. Satellite Mass Variation Comparison

The simulated results in Figure 5.29 show that, with a continuous strategy, the NMPC not only fails to guarantee the same optimality level of the ideal impulsive strategy, but also produces worse performances than the manual approach. This worsening, with respect to the quasi-impulsive strategy, is due to the fact that the maneuvers are completely performed outside the region of optimality and then many misalignments and gravity losses are introduced. The results in Tables 5.10, 5.11 and 5.12, in which also a comparison of the required ΔV is reported, confirm this worsening in the performances.

	Fuel Consump. Revisit (kg)	**Fuel Consump. Monitoring (kg)**	**Fuel Consump. Return (kg)**
Ideal Maneuvers	269.2	242.7	377.3
Continuous Maneuvers	352.9	301.2	419.5
Manual Maneuvers	307.1	278.1	414.1

Table 5.10. Fuel consumption of the proposed approaches

	ΔV Revisit (km/s)	**ΔV Monitoring (km/s)**	**ΔV Return (km/s)**
Ideal Maneuvers	3.2635	3.3577	6.3234
Continuous Maneuvers	4.3720	4.4499	7.9141
Manual Maneuvers	3.7583	3.9727	7.4069

Table 5.11. Required ΔV for the completion of the mission

	Revisit Time (s)	**Monitoring Time (s)**
Ideal Maneuvers	5835	5955
Continuous Maneuvers	5840	5980
Manual Maneuvers	5838	5960

Table 5.12. Time for revisit and monitor the target

It should be noted that NMPC technique determines also greater ΔV and times for revisit and monitoring than manual mission planning.

In Figures 5.30 and 5.31 the control action and the corresponding ΔV are reported.

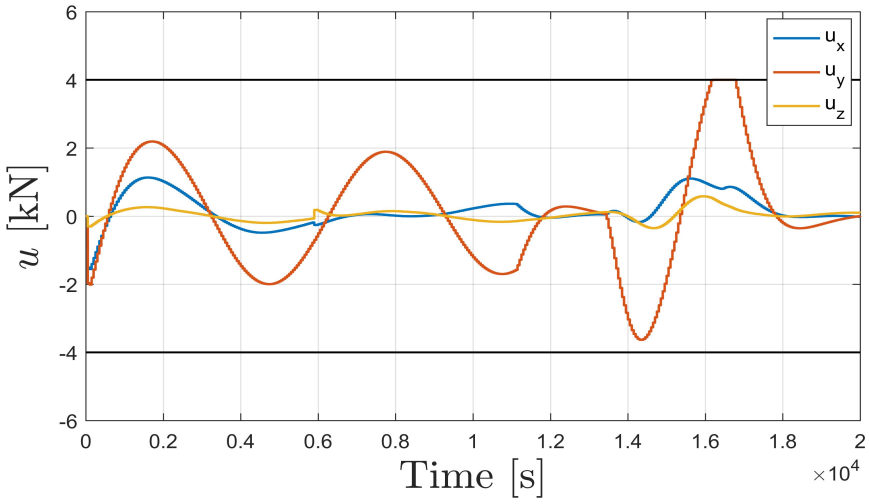

Figure 5.30. Continuous Maneuvers Control Action

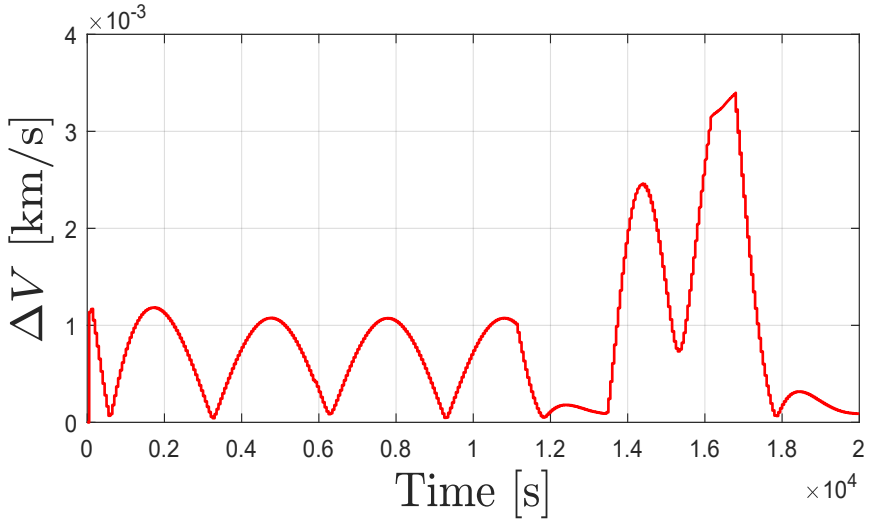

Figure 5.31. Required ΔV for continuous maneuvers

These figures highlight even more the continuous action produced by the NMPC; indeed, both the command and the ΔV are applied for the whole duration of the mission. This results in a reduction of the maximum required thrust, but produces worsening both in the fuel consumption and in the total required ΔV.

Finally, the errors on orbital parameters are shown in Figure 5.32 in order to evaluate the effectiveness of the controller.

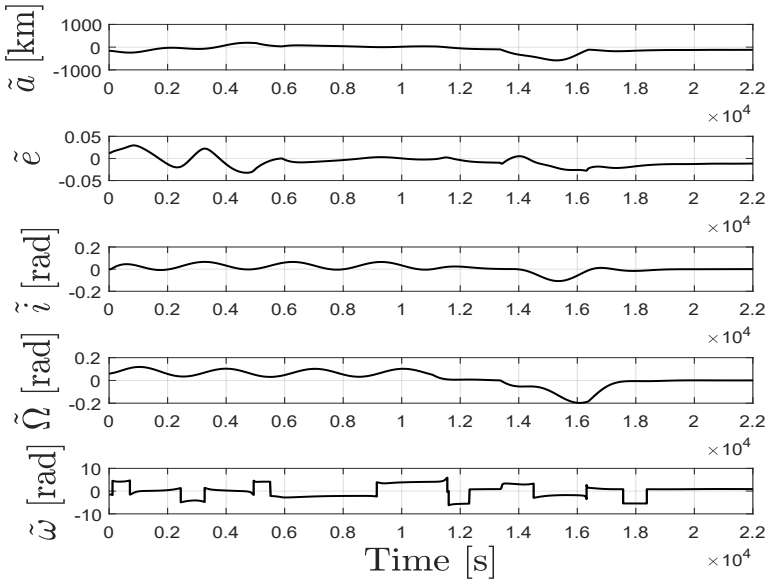

Figure 5.32. Tracking Error in Terms of Orbital Parameters.

Alert near the current area flown over by the satellite

Considering the same alert of the quasi-impulsive strategy, the optimized trajectory produced by the NMPC for the revisit and monitoring missions are shown in Figure 5.33.

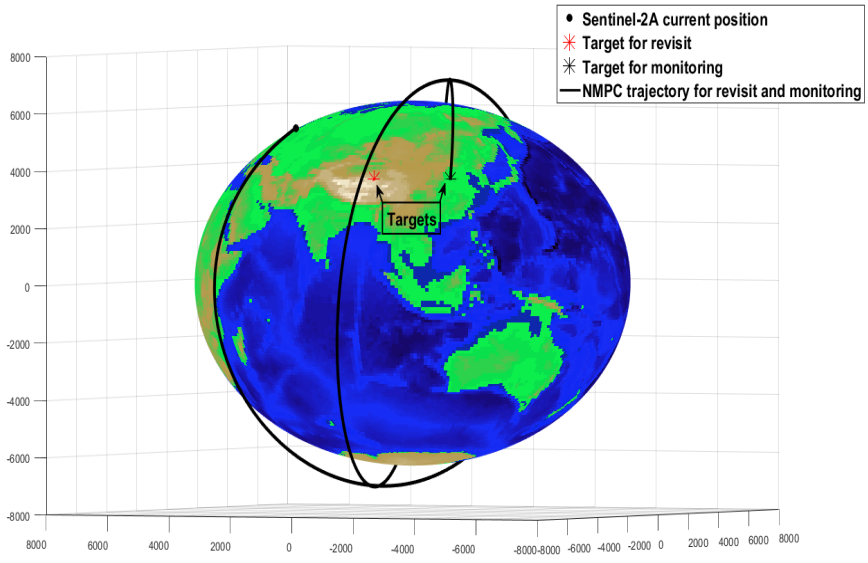

Figure 5.33. NMPC optimized trajectory for the revisit and monitoring missions

As can be seen, the NMPC can handle the optimization operation continuously for the whole duration of the mission, producing an optimized trajectory which is able to guide the satellite exactly on the alert coordinate.

Alert far from the current satellite position

Also for this alert, only the trajectory produced by the predictive control is considered and reported in Figure 5.34.

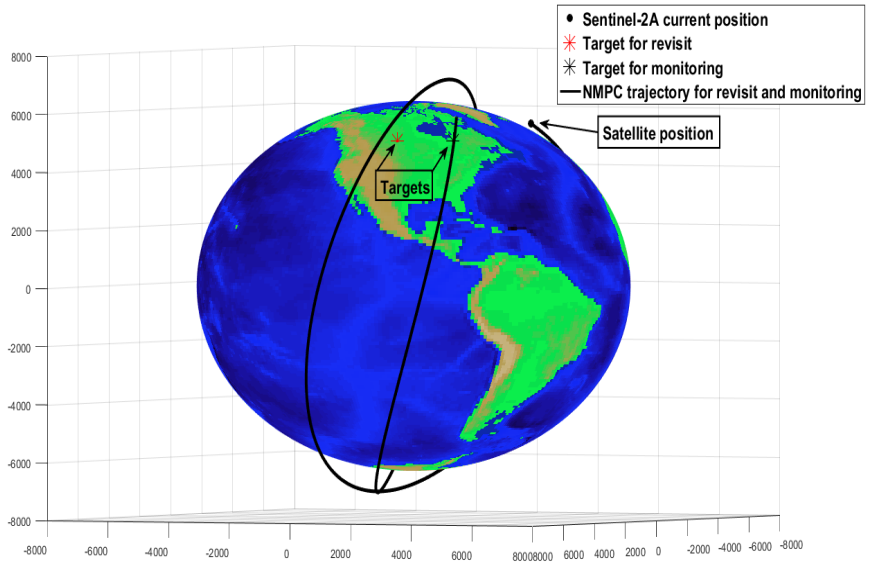

Figure 5.34. NMPC optimized trajectory for the revisit and monitoring missions

Chapter 6

Conclusions

In this thesis, an autonomous guidance and control approach for Earth observation, alternative to the standard space mission planning made on ground, has been presented. In particular, the aim of the project has been to develop an autonomous system, based on Nonlinear Model Predictive Control (NMPC) technique, with the ability to plan its motion trajectories for the monitoring of generic coordinates on the Earth, in which an alert occurred. As case study, the ESA Sentinel-2 mission has been considered.

First of all, the nominal orbit of the satellite has been simulated by the use of orbital elements and the integration of the classical two-body equation.

Therefore, the trajectory planner has been implemented. Its goal has been to produce feasible trajectories for guiding the Sentinel exactly on the target coordinates. From this point of view, two type of strategies for carrying out the revisit and monitoring missions have been taken into account: quasi-impulsive strategy and continuous strategy. The first approach consists in the application of the same maneuvers performed in the ideal situation, in which an impulsive and instantaneous ΔV is used in order to change the inclination of the orbit exactly in one of the two nodes. The second strategy, instead, is based on the definition of a trajectory (obtained by an interpolation between the initial and target orbit) that instant by instant guides the spacecraft with a continuous change of inclination, allowing to perform smooth maneuvers.

Then, after the development of the trajectory planner, a Nonlinear Model Predictive Control has been designed in order to generate a control input which forces the S/C to follow the reference trajectories, finding a suitable trade-off between the time to perform the maneuvers and the propellant consumption. Moreover, in the optimization operation constraints on the states, to ensures the S/C not to crash on Earth, and on the inputs, to account for thruster saturation, have been considered.

Finally, the autonomous guidance based on NMPC has been simulated, considering different alerts, and its performances have been compared with the ideal impulsive approach and manual mission planning. For the first strategy, the obtained results have

shown that the NMPC technique can achieve better performances than the manual case and guarantee nearly the same optimality level as the ideal impulsive strategy. However the application of impulsive and instantaneous ΔV are unfeasible in real application, while the NMPC method can be actually applied in a real mission allowing a significantly higher level of autonomy and offer more flexibility and adaptation capability with respect to traditional approaches. The second strategy, instead, has produced worse performances than ideal and manual situations due to high gravity and misalignments losses. Thus it does not provide improvements with respect to the previous strategy. However, the main reason behind the development of continuous maneuvers approach has been to verify if the NMPC could handle situations where the command input should be applied for the whole duration of the mission.

Future Works

Starting from the results presented in this thesis project, it is possible to carry out further developments.

In particular the concept of continuous and small orbital inclination changes can be exploited to develop a new strategy based on low-thrust maneuvers. The aim will be to produce continuous maneuvers with a different kind of propulsion (i.e. Ion thrusters, Hall thrusters), in order to reduce the propellant consumption with respect to the ideal case. Indeed electric propulsion is characterized by a high engine specific impulse, which ensures a higher propellant efficiency even though the ΔV could increases. However, a low-thrust maneuvers will not be able to ensure a fast overfly of target coordinate (which is the main constraints of Earth observation missions).

Bibliography

[1] L. Singh, S. Bortolami, L.A. Page. *Optimal Guidance and Thruster Control in Orbital Approach and Rendezvous for Docking Using Model Predictive Control*. Proceedings of AIAA Guidance, Navigation and Control Conference, Toronto, 2010.

[2] A. Weiss, U.V.Kalabić, S. Di Cairano. *Station keeping and momentum management of low-thrust satellites using MPC*. Aerospace Science and Technology, Volume 76, May 2018, Pages 229-241.

[3] M. Leomanni, G. Bianchini, A. Garulli, A. Giannitrapani. *State Feedback Control in Equinoctial Variables for Orbit Phasing Applications*. Journal of Guidance, Control and Dynamics, Volume 41, Issue 8, August 2018.

[4] S. Pelle, E. Gargioli, M. Berga, J. Pisacreta, N. Viola, A. Dalla Sega, M. Pagone. *Earth-Mars cyclers for a sustainable human exploration of Mars*. Acta Astronautica Volume 154, January 2019, Pages 286-294.

[5] L. Massotti, D. Di Cara, J.G. del Amo, R. Haagmans, M. Jost, C Siemes, P. Silvestrin. *The ESA Earth Observation Programmes Activities for the Preparation of the Next Generation Gravity Mission*. AIAA Guidance, Navigation, and Control (GNC) Conference, Boston, 2013.

[6] G. Rabideau, D. Tran, S. Chien, B. Cichy, R. Sherwood. *Mission Operations of Earth Observing-1 with Onboard Autonomy*. Proceedings of the 2nd IEEE International Conference on Space Mission Challenges for Information Techonolgy, 2006.

[7] S.A. Chien, R. Sherwood, D. Tran, B.Cichy, G. Rabideau, R. Castaño, A. Davies, et al. *Lessons Learned from Autonomous Spacecraft Experiment*. Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems, Ultrech, The Netherlands, 2005, Pages 11-18.

[8] Copernicus. [Online]. Available: `https://www.copernicus.eu/en`

[9] Sentinel, ESA. [Online]. Available: `https://sentinel.esa.int/web/sentinel/home`

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

[11] Carlo Novara. *Nonlinear control and aerospace applications*. Lecture notes.

[12] Howard Curtis. *Orbital Mechanics for Engineering Students*. Elsevier Aerospace Engineering Series,2005

[13] Giovanni Mengali, Alessandro A. Quarta. *Fondamenti di Meccanica del Volo Spaziale*. Pisa University Press, 2014.

[14] R. Findeisen, L. Imsland, F. Allgöwer, B.A. Foss. *State and Output Feedback Nonlinear Model Predictive Control: An Overview*. European Journal of Control Volume 9, Issues 2–3, 2003, Pages 190-206.

[15] M. Vukov, A. Domahidi, H. Joachim Ferreau, M. Morari, M. Diehl. *Auto-generated Algorithms for Nonlinear Model Predictive Control on Long and on Short Horizons*. 52nd IEEE Conference on Decision and Control, 2013, Pages 5113 - 5118.

[16] Eduardo F. Camacho, Carlos Bordons Alba. *Model Predictive Control*. Springer, 2 edition, 2007.

[17] Rolf Findeisen, Frank Allgöwer. *An Introduction to Nonlinear Model Predictive Control*. 21st Benelux Meeting on Systems and Control, 2002, Pages 119-141.

[18] Tor A. Johansen. *Introduction to Nonlinear Model Predictive Control and Moving Horizon Estimation*. Selected Topics on Constrained and Nonlinear Control, 2011, Pages 187-239.

[19] Richard E. Bellman. *Dynamic Programming*. Princeton University Press, Princeton 1957.

[20] M.C.V. Salgado, M.C.N. Belderrain, T.C. Devezas. *Space Propulsion: a Survey Study About Current and Future Technologies*. Journal of Aerospace Technology and Management, 2018.

[21] Luca Massotti. *Fundamentals of Space Technology*. Lecture notes.

[22] ArianeGroup. [Online]. Available:`https://www.ariane.group/en/`

[23] QinetiQ. [Online]. Available:`https://www.qinetiq.com/`