Autonomous Orbit Control of LEO Satellites

Luís Miguel da Luz Bessa Ferreira
luis.luz.ferreira@tecnico.ulisboa.pt

Instituto Superior Técnico, Universidade de Lisboa, Portugal
October, 2021

Abstract

Countless benefits and applications of space technology led to a significant rise in the number of satellites being launched into space over the years to perform a wide range of missions. To satisfy the mission objectives, the satellites need to be operated at a certain nominal orbit. However, orbital injection errors and environmental perturbations cause the orbit to deviate from the nominal one. Recent improvements in on-board computational performance allow transferring the orbit correction process to on-board autonomous orbit control (AOC) algorithms. On that matter, this work is focused on the development of an autonomous algorithm to perform orbit correction and maintenance of a LEO satellite using novel electric propulsion solutions. Two different control methodologies are explored to achieve the proposed objective: linear quadratic regulator (LQR) and model predictive control (MPC). A control solution based on a combination of these methods is proposed. The MPC is used to correct large initial errors that result from the orbital injection, whereas the LQR assures the final convergence and orbit maintenance by counteracting the effect of orbital perturbations. The TerraSAR-X mission was selected as a test-bed to evaluate the algorithm performance. Simulation results show that the proposed AOC strategy effectively corrects orbit injection errors in different Keplerian elements. Moreover, the AOC algorithm demonstrates good performance for orbit maintenance activities, correcting the effect of orbital perturbations and maintaining the orbital error within the TerraSAR-X mission requirements, even when considering realistic sensors and actuators.

Keywords: Autonomous orbit control, LEO satellite, electric propulsion, model predictive control, linear quadratic regulator.

1 Introduction

Low Earth orbit (LEO) satellites are used to fulfil a variety of missions, from Earth and Space observation to communications and military applications. Most of the man-made objects orbiting the Earth are in LEO as they require less energy for orbit placement and, therefore, less associated cost. To satisfy the mission objectives, the orbit of these satellites is designed to fulfil certain requirements. However, launch injection errors and the effect of orbit perturbations cause the orbit to deviate from the one originally projected. This is normally not acceptable and, therefore, orbital maneuvers are performed to correct the orbital deviation. Usually, this process is completely managed by the ground-segment in a time-consuming process. The increase of on-board computational resources created a new opportunity for the implementation of on-board algorithms to control the spacecraft orbit automatically without ground intervention. In this context, on-board autonomous orbit control (AOC) arises as the answer to meet mission requirements and reduce the workload of operational teams. Simple control strategies, such as the linear quadratic regulator (LQR), have been used to develop autonomous orbit control algorithms (see [1]). Recent improvements in computational performance enable the use of more powerful control strategies, namely the model predictive control (MPC), to explore new AOC algorithms and enhance control performance. Moreover, the emergence of electric propulsion solutions allows for more precise and efficient orbit control, which enable novel AOC strategies.

One of the first groups to work on autonomous orbit control was Microcosm that tested their orbit control kit (OCK) on-board the UoSAT-12 satellite, launched in April 1999 [2]. The success of that mission validated the use of autonomous orbit control to correct deviations from the reference orbit and maintain a satellite's long-term orbit parameters. Following that, in 2011, the PRISMA mission validated the autonomous orbit keeping algorithm to maintain the longitude of the ascending node of the MANGO spacecraft, within a certain tolerance [3]. Regarding the PRISMA mission, Sergio De Florio and Simone D'Amico developed an orbit control strategy based on the relative motion of two satellites. They approached the orbit control problem as a formation fly between a real satellite, affected by the orbital perturbations, and a virtual satellite that fulfils the orbit requirements imposed by the mission. A standard LQR controller was implemented to compute the control actions and perform orbit corrections to maintain the position error between the real and the virtual satellite below 250 m (root-mean-square). PRISMA mission was used as a test-bed to validate the control algorithms [1].

A more powerful, but more computationally demanding, control strategy was proposed by Tavakoli and Assadian. They investigated the use of MPC for autonomous orbit control of LEO satellites by applying a similar strategy as Florio and D'Amico. The orbit control problem was converted into a relative orbit control problem in which an MPC controller computes the finite horizon optimal firing times of the satel-

lite thrusters, in order to drive the satellite to the reference orbit, not affected by undesirable perturbations [2].

The majority of autonomous orbit control solutions use chemical propulsion to perform corrective maneuvers. However, the emergence of electric propulsion systems enabled new control strategies. Mirko Leomanni, Andrea Garulli, and Antonio Giannitrapani developed a solution to maintain a repeat ground track orbit of a spacecraft driven by electric propulsion [4]. The control approach proposed uses a hysteresis controller that provides a pulse-width modulated command to the thruster. Simulation results showed that the resulting control scheme was able to acquire and maintain consistently in-orbit the desired repeat ground track pattern, in a fully autonomous manner.

From the literature review, it is noticeable a clear lack of AOC strategies that account for large initial errors due to the orbital injection process. Navigation errors of the launcher and thrust inaccuracies originate an error between the targeted orbit and the actual injection orbit. In this work, a solution to overcome these injection errors and drive the initial orbit to the desired one is proposed. This solution is based on an advanced control strategy, namely the MPC. Despite the increased computational effort, this more powerful control technique offers enhanced capabilities to the control system, which are not present in the solutions usually proposed in the literature. The control strategy was developed based on the use of electric thrusters as the actuators to correct the satellite orbit, which is an emergent solution in the field of autonomous orbit correction and maintenance.

To summarise, the objective of this work is the development of an algorithm to perform autonomous orbit control of a LEO satellite using electric propulsion that can cope with a wide range of initial error conditions. This algorithm computes the required control actions to correct initial orbit injection errors and maintain the orbital parameters of the satellite close to the reference, counteracting the effect of orbital perturbations. Two different control solutions are used to achieve this objective: LQR and MPC. Due to its demanding orbit control requirements, the TerraSAR-X mission was selected as a test-bed to evaluate the algorithm performance (see [5]).

2 TerraSAR-X mission

The TerraSAR-X mission, which was operated in close formation with the Tandem-X mission at distances of a few hundred meters, is used as a test-bed for the proposed controllers, given its demanding orbit control requirements. TerraSAR-X is a German synthetic-aperture radar (SAR) satellite based on a public-private partnership agreement between the German Aerospace Center (DLR) and EADS Astrium GmbH, which was launched on 15^{th} June, 2007 [5].

The spacecraft characteristics and orbit properties of the TerraSAR-X mission are presented in Table 1. The orbit of TerraSAR-X is LEO, sun-synchronous and has to fulfill the requirements presented in Table 2 to accomplish its mission.

Table 1: Spacecraft and orbit properties of the TerraSAR-X mission [5].

TerraSAR-X characteristics	
Wet mass	1230 kg
Dimensions	5 m height $\times 2.4$ m diam.
Semi-major axis	6892.944 km
Eccentricity	0.0014252
Inclination	97.4401 deg
Altitude	$505 - 533$ km
Nominal revisit period	11 days

Table 2: TerraSAR-X mission requirements [5].

TerraSAR-X requirements	
LTAN	$18:00$ hrs ($\pm$ 0.25 hrs)
Tr	$\pm$ 500 m per revisit period

The local time of ascending node (LTAN) is used to describe sun-synchronous orbits. It is defined by the mean local time at which the spacecraft passes the Earth equator on the ascending branch of its orbit. The ground track repeatability defines with which accuracy the spacecraft should fly over a given path on the Earth's surface. The ground track repeatability is evaluated by the ground track repeatability error (Tr) which is the distance along the Earth surface between the satellite real position and its desired location.

The AOC problem can be formulated as a two satellite formation problem, in which one of them is virtual and not affected by non-gravitational orbit perturbations. This virtual satellite is referred to as the reference satellite, which complies with the mission requirements, having an error of zero in the ground track repeatability at all times. The reference satellite orbit to be used should be affected by the aspherical terms of the Earth gravity field, so that the orbital precession due to the oblateness of the Earth can be leveraged to keep the LTAN constant.

The ground track repeatability error (T_r), illustrated in Fig. 1 can be represented, at a given epoch, as the distance along the Earth surface between the real propagated satellite at point A and the reference satellite at point B.

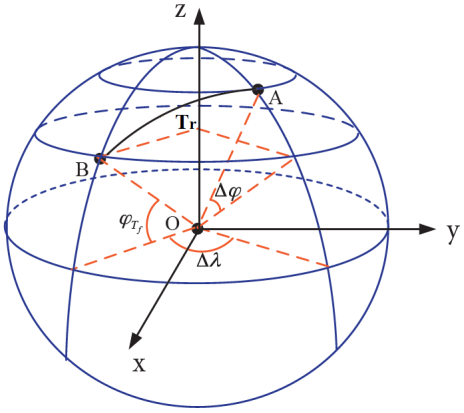

Figure 1: Ground track repeatability error schematic diagram [6].

The approximate formula for small values of T_r based on spherical geometry is given by (see [6])

$$T_r \approx R_\oplus \sqrt{\Delta\varphi^2 + (\cos\varphi_{T_f}\Delta\lambda)^2}, \tag{1}$$

where φ_{T_f} is the geocentric latitude of point B, and $\Delta\varphi$ and $\Delta\lambda$ are the geocentric latitude and longitude difference between points A and B, respectively.

3 Fundamentals of Orbital Motion

3.1 Reference Frames Definition

Two distinct reference frames are used in this work: the True of Date (ToD) coordinate frame and the Local Orbit (LOF) coordinate frame (see [7]).

The ToD coordinate frame has its origin coincident with the center of mass of the Earth and is approximately an inertial reference frame, even though its origin suffers an acceleration caused by other celestial bodies, mainly the Sun. This acceleration can be neglected for the accuracy required for the applications considered in this work given its small magnitude when compared with the magnitude of the gravitational acceleration caused by the Earth.

Throughout this work, the ToD frame is represented by the basis $\mathcal{I} = \{\mathbf{i_1}, \mathbf{i_2}, \mathbf{i_3}\}$. The $\mathbf{i_1}$ axis points from the center of mass of the Earth to the vernal equinox, at the current epoch. The $\mathbf{i_3}$ axis is along the axis of the Earth's rotation at the current epoch and pointing towards the North Pole. Finally, the $\mathbf{i_2}$ axis completes the right-handed orthogonal reference frame. Due to the precession of the Earth's rotation axis, the direction of the coordinate system axes varies with time. Therefore, the direction of these axes is defined relative to the current epoch of the analysis.

The Local Orbit frame is defined with respect to the orbit of the spacecraft and is represented by the basis $\mathcal{O} = \{\mathbf{o_1}, \mathbf{o_2}, \mathbf{o_3}\}$. The $\mathbf{o_3}$ axis points from the center of the Earth to the spacecraft, whereas the $\mathbf{o_2}$ axis points in the direction of the spacecraft's orbital angular velocity. The $\mathbf{o_1}$ axis completes the right-handed orthogonal reference frame, pointing in the direction of the motion for a circular orbit. This coordinate frame has its origin in the spacecraft's center of mass.

3.2 Orbital Motion Models

3.2.1 Cartesian Coordinates Representation

If the motion of a satellite is described relative to the ToD frame and perturbations are taken into account, then the equation of motion can be written in the form (see [8])

$$\ddot{\mathbf{r}} + \frac{\mu}{||\mathbf{r}||^3}\mathbf{r} = {}^I\mathbf{f}(\mathbf{r}, \mathbf{v}) + {}^I\mathbf{f}_c, \tag{2}$$

where $\mathbf{r}$ and $\mathbf{v}$ are, respectively, the position and the velocity of the spacecraft expressed in the ToD coordinate frame, ${}^I\mathbf{f}(\mathbf{r}, \mathbf{v})$ is a 3×1 vector that corresponds to an acceleration vector caused by orbital perturbations, μ is the standard gravitational parameter, and ${}^I\mathbf{f}_c$ is a 3×1 vector that corresponds to a control acceleration, expressed in the ToD reference frame, used to control the satellite motion

From (2), the set of first-order differential equations of the position and the velocity of a satellite, expressed in the ToD reference frame, is given by

$$\begin{cases} \dot{\mathbf{r}} &= \mathbf{v} \\ \dot{\mathbf{v}} &= -\frac{\mu}{||\mathbf{r}||^3}\mathbf{r} + {}^I\mathbf{f}(\mathbf{r}, \mathbf{v}) + {}^I\mathbf{f}_c \end{cases}. \tag{3}$$

3.2.2 Keplerian Elements Representation

Keplerian elements are parameters used to represent the spacecraft position in its orbit about the Earth, which provide immediate insight about the orientation and geometry of the orbit. In fact, the Cartesian position and velocity of the spacecraft may be useful for some computational applications but provide little insight into important aspects of the orbital motion [1], [9].

Gauss' Variational Equations (GVEs) adapted for near-circular orbits provide a convenient model to fully characterise the orbit and location of the satellite, using the set of Keplerian elements ${}^I\mathbf{x}_k = [a\ e_x\ e_y\ i\ \Omega\ u]^T$, where a is the semi-major axis, e_x and e_y are the components of the eccentricity vector, i is the inclination, Ω is the longitude of the ascending node, and u is the argument of latitude. Although TerraSAR-X initial orbit can be modeled as near-circular, during the orbit correction maneuvers the orbit eccentricity presents values that are not characteristic of a near-circular orbit. Therefore, an adaptation of the near-circular orbit system dynamics proposed in [8] is considered, as given by

$$\begin{cases} \dot{a} &= 2\frac{a^2}{\sqrt{\mu p}}\left[{}^O f_3 e \sin\theta + {}^O f_3 \frac{p}{||\mathbf{r}||}\right] \\ \dot{e}_x &= \sqrt{\frac{p}{\mu}}\left[{}^O f_3 \sin\theta + {}^O f_3(\cos E + \cos\theta)\right] \\ \dot{e}_y &= \sqrt{\frac{p}{\mu}}\left[-{}^O f_3 \cos\theta + {}^O f_3(\sin E + \sin\theta)\right] \\ \dot{i} &= {}^O f_3 \frac{||\mathbf{r}||}{\sqrt{\mu p}} \cos u \\ \dot{\Omega} &= {}^O f_3 \frac{||\mathbf{r}||}{\sqrt{\mu p}\sin i} \sin u \\ \dot{u} &= n + {}^O f_1\left[\frac{1}{e}\sqrt{\frac{p}{\mu}} - \frac{1-e^2}{e}\sqrt{\frac{a}{\mu}}\right]\sin\theta\left[1 + \frac{||\mathbf{r}||}{p}\right] - \\ & \quad {}^O f_2 \frac{||\mathbf{r}||}{\sqrt{p\mu}}\cot i \sin u - {}^O f_3\left[\frac{2||\mathbf{r}||}{\sqrt{\mu a}} + \left(\frac{1}{e}\sqrt{\frac{p}{\mu}} - \frac{1-e^2}{e}\sqrt{\frac{a}{\mu}}\right)\cos\theta\right] \end{cases}, \tag{4}$$

where ${}^O f_1$, ${}^O f_2$, and ${}^O f_3$ are the components of the acceleration vector felt by the spacecraft expressed in the Local Orbit frame and include all the perturbative and the control accelerations, p is the semi-latus rectum ($p = a(1-e^2)$), E is the eccentric anomaly, θ is the true anomaly, and n is the mean motion.

3.3 Perturbation forces

Perturbation forces cause a time-variation of the Keplerian elements, causing the orbit to deviate from the unperturbed Keplerian model. The most significant perturbation forces in low Earth orbits are the atmospheric drag (${}^I\mathbf{f}_{aero}$),

the solar radiation pressure (${}^I\mathbf{f}_{SRP}$), the third-body gravitational attraction (${}^I\mathbf{f}_{TBG}$) from the Moon and the Sun, and the non-spherical Earth gravity field (${}^I\mathbf{f}_{\oplus}$) [10]. These accelerations are expressed in the ToD reference frame. Therefore, a rotation to the Local Orbit frame is necessary in order to use them in the GVEs. The resulting perturbative acceleration (${}^O\mathbf{f} = [{}^O f_1 \ {}^O f_2 \ {}^O f_3]^T$), in the O-frame, is given by

$$ {}^O\mathbf{f} = {}^O\mathbf{f}_{aero} + {}^O\mathbf{f}_{SRP} + {}^O\mathbf{f}_{TBG} + {}^O\mathbf{f}_{\oplus}. $$

4 Control Design

TerraSAR-X mission requires that the ground track repeatability error is maintained within a specific range of values to fulfill the mission objectives. To achieve that, the projection on the Earth surface of the position of the real satellite should be driven and maintained close to the projection on the Earth of the reference satellite, so the ground track repeatability error is small enough. This is possible if the orbital position of the real satellite is driven to the reference satellite orbital position, considering that the projection of the satellite is directly dependent on its orbital position. Furthermore, by correcting the orbital velocity of the real satellite, its orbit will converge to the reference satellite orbit. This strategy will also maintain the LTAN within the range of values specified by the requirements since this is a property of the reference orbit. To achieve this objective, two different control strategies are considered: LQR and MPC.

4.1 LQR Design

4.1.1 Dynamic Model

In the ToD reference frame, the Cartesian representation of the error dynamics between the real satellite and the reference is a straightforward model, which simplifies the controller implementation.

Let ${}^I\mathbf{x} = [\mathbf{r}^T \ \mathbf{v}^T]^T$ be the vector that contains the position and the velocity of the real satellite expressed in the ToD reference frame. Recalling (3), the time derivative of vector ${}^I\mathbf{x}$ can be written as

$$ {}^I\dot{\mathbf{x}} = \begin{bmatrix} \mathbf{v} \\ -\frac{\mu}{||\mathbf{r}||^3}\mathbf{r} + {}^I\mathbf{f}(\mathbf{r},\mathbf{v}) \end{bmatrix} + \mathbf{B}\ {}^I\mathbf{f}_c, \tag{5} $$

where

$$ \mathbf{B} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. $$

As mentioned in Section 2, it is intended that the orbit precession still occurs in order to maintain the LTAN constant. Therefore, for the reference orbit, the only perturbative acceleration that is considered is the one caused by the Earth's gravity field and no input accelerations are considered. This orbit is modelled by

$$ {}^I\dot{\mathbf{x}}_r = \begin{bmatrix} \mathbf{v}_r \\ -\frac{\mu}{||\mathbf{r}_r||^3}\mathbf{r}_r + {}^I\mathbf{f}_r(\mathbf{r}_r,\mathbf{v}_r) \end{bmatrix}, \tag{6} $$

where ${}^I\mathbf{x}_r = [\mathbf{r}_r^T \ \mathbf{v}_r^T]^T$ is a vector that contains the position and the velocity of the reference satellite in the ToD reference frame and ${}^I\mathbf{f}_r(\mathbf{r}_r,\mathbf{v}_r)$ is a perturbative acceleration caused by the Earth's gravity field. To obtain a ground track error close to zero, the error $\boldsymbol{\epsilon} = {}^I\mathbf{x} - {}^I\mathbf{x}_r$ must converge to zero. Subtracting (6) from (5), the dynamic model in matrix form for the error between the real and the reference satellite is given by

$$ \dot{\boldsymbol{\epsilon}} = \begin{bmatrix} \mathbf{v} - \mathbf{v}_r \\ -\frac{\mu}{||\mathbf{r}||^3}\mathbf{r} + {}^I\mathbf{f}(\mathbf{r},\mathbf{v}) + \frac{\mu}{||\mathbf{r}_r||^3}\mathbf{r}_r - {}^I\mathbf{f}_r(\mathbf{r}_r,\mathbf{v}_r) \end{bmatrix} + \mathbf{B}\ {}^I\mathbf{f}_c. \tag{7} $$

Defining $\Delta\mathbf{v} = \mathbf{v} - \mathbf{v}_r$ and $\Delta\mathbf{r} = \mathbf{r} - \mathbf{r}_r$, it is possible to rewrite (7) as

$$ \dot{\boldsymbol{\epsilon}} = \begin{bmatrix} \Delta\mathbf{v} \\ g(\boldsymbol{\epsilon}) \end{bmatrix} + \mathbf{B}\ {}^I\mathbf{f}_c, \tag{8} $$

where

$$ g(\boldsymbol{\epsilon}) = -\frac{\mu}{||\Delta\mathbf{r} + \mathbf{r}_r||^3}(\Delta\mathbf{r} + \mathbf{r}_r) + \frac{\mu}{||\mathbf{r}_r||^3}\mathbf{r}_r + {}^I\mathbf{f}(\Delta\mathbf{r} + \mathbf{r}_r, \Delta\mathbf{v} + \mathbf{v}_r) - {}^I\mathbf{f}_r(\mathbf{r}_r,\mathbf{v}_r). $$

4.1.2 Linearization

The dynamic model for the error between the real and the reference satellite is given by (8). Notice that this model is nonlinear and, therefore, to implement an LQR controller based on this system dynamics, a linearization process must be carried out.

Starting by defining an equilibrium point given by $\Delta\mathbf{r}_0 = \Delta\mathbf{v}_0 = [0\ 0\ 0]^T$ and ${}^I\mathbf{f}_{c_0} = [0\ 0\ 0]^T$, which, by the definition of equilibrium point, provides $\dot{\boldsymbol{\epsilon}}_0 = [0\ 0\ 0\ 0\ 0\ 0]^T$, the small perturbation model for operations near this equilibrium point is given by

$$ \begin{cases} \Delta r_1 = \Delta r_{1_0} + \delta\Delta r_1 \\ \Delta r_2 = \Delta r_{2_0} + \delta\Delta r_2 \\ \Delta r_3 = \Delta r_{3_0} + \delta\Delta r_3 \\ \Delta v_1 = \Delta v_{1_0} + \delta\Delta v_1 \\ \Delta v_2 = \Delta v_{2_0} + \delta\Delta v_2 \\ \Delta v_3 = \Delta v_{3_0} + \delta\Delta v_3 \\ {}^I\mathbf{f}_c = {}^I\mathbf{f}_{c_0} + \delta{}^I\mathbf{f}_c \end{cases}. \tag{9} $$

Note that $\Delta\mathbf{r} = [\Delta r_1\ \Delta r_2\ \Delta r_3]^T$, $\Delta\mathbf{v} = [\Delta v_1\ \Delta v_2\ \Delta v_3]^T$, and $\mathbf{r}_r = [r_{r_1}\ r_{r_2}\ r_{r_3}]^T$, which are expressed in the ToD reference frame. The referred equilibrium point is chosen based on the control objective of driving the error between the real and the

reference satellite to zero. Therefore, it was decided to linearize (8) about the equilibrium point where the error is zero, given by $\boldsymbol{\epsilon} = [0\ 0\ 0\ 0\ 0\ 0]^T \Rightarrow \Delta\mathbf{r} = \Delta\mathbf{v} = [0\ 0\ 0]^T$. Remembering that $\boldsymbol{\epsilon} = [\Delta\mathbf{r}^T\ \Delta\mathbf{v}^T]^T$, it is possible to write $g(\boldsymbol{\epsilon})$ (see Section 4.1.1) as a Taylor series expansion, which provides a linear form of $g(\boldsymbol{\epsilon})$. Since the magnitude of the perturbative acceleration caused by the J_2 Earth gravity component is 10^3 times higher than the magnitude of other perturbative accelerations that affect the real orbit, for the sake of simplicity, it was decided to only consider the J_2 term influence in the system. With that in mind, and using the small perturbation model given by (9), it is possible to linearize the error dynamics (8), as given by

$$\delta\dot{\boldsymbol{\epsilon}} = \mathbf{A}_\epsilon \delta\boldsymbol{\epsilon} + \mathbf{B}_\epsilon \delta^I\mathbf{f}_c, \tag{10}$$

where

$$\mathbf{A}_\epsilon \delta\boldsymbol{\epsilon} = \begin{bmatrix} \delta\Delta\mathbf{v} \\ \frac{\partial g(\boldsymbol{\epsilon})}{\partial \Delta r_1}\Big|_{\boldsymbol{\epsilon}=0} \delta\Delta r_1 + \frac{\partial g(\boldsymbol{\epsilon})}{\partial \Delta r_2}\Big|_{\boldsymbol{\epsilon}=0} \delta\Delta r_2 + \frac{\partial g(\boldsymbol{\epsilon})}{\partial \Delta r_3}\Big|_{\boldsymbol{\epsilon}=0} \delta\Delta r_3 \end{bmatrix}$$

and

$$\mathbf{B}_\epsilon = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

The linearized dynamics for the error between the real and the reference satellite given by (10) operates around the equilibrium point where the error is zero and uses the deviations from this equilibrium $\delta\boldsymbol{\epsilon} = [\delta\Delta\mathbf{r}^T\ \delta\Delta\mathbf{v}^T]^T$ as the system state. For operating points far from zero, the linearized model is no longer valid.

4.1.3 LQR Synthesis

A MATLAB script was developed to evaluate the effect that the control input computed by the LQR has on the system and assess its performance in driving the error between the real satellite and the reference one to zero. These two satellites have the same physical characteristics as the TerraSAR-X spacecraft. The real satellite orbit is affected by the orbital perturbations caused by the Earth gravity field, the solar radiation pressure, the atmospheric drag, the third-body gravitational attraction, and by the control action computed by the controller. Furthermore, it is possible to define an initial orbital error for this satellite in any of its Keplerian elements.

The control input computed by the LQR controller takes the form of an acceleration $\delta^I\mathbf{f}_c$ that will affect the orbit of the real satellite. This control acceleration is limited by the actuators saturation, which depends on the selected thruster. This acceleration is calculated every simulation step and is obtained by the optimal linear control law for the linearized version of the system

$$\delta^I\mathbf{f}_c = -\mathbf{K}\,\delta\boldsymbol{\epsilon},$$

where the Kalman gain $\mathbf{K}$ is calculated using the MATLAB command

$$\mathbf{K} = \mathrm{lqr}(\mathbf{A}_\epsilon, \mathbf{B}_\epsilon, \mathbf{Q}, \mathbf{R}).$$

Matrices $\mathbf{A}_\epsilon$ and $\mathbf{B}_\epsilon$ are defined using the linearized dynamics for the error between the real satellite and the reference (10). Notice that $\mathbf{A}_\epsilon$ is dependent on the reference satellite Cartesian position on the ToD reference frame, which is updated every simulation step. However, to avoid an excessive computational effort, the Kalman gain might have a different update rate from the simulation step, since the most significant terms of the matrix $\mathbf{K}$ present a small variation over time, as verified in the simulations. Therefore, it is not necessary to update this matrix every simulation step. The calibration of matrices $\mathbf{Q}$ and $\mathbf{R}$ is done using a trial-and-error method and has, as a starting point, Bryson's method to reduce the level of arbitrariness (see [11]).

4.2 MPC Design

4.2.1 Comparison between Keplerian Elements and Cartesian Coordinates

Similarly to the LQR, to design an MPC, it is necessary to define a state function that will be used during the optimisation process to predict the system behaviour over the prediction horizon. One possibility is to use Cartesian coordinates representation as the state function given the simplicity of this model when compared to the Keplerian elements representation. However, the implementation of this model leads to a difficult tuning of the MPC controller due to the large variation with time of the position and the velocity of the satellite expressed in Cartesian coordinates representation.

For this reason, the Keplerian elements representation for the orbital motion laws was selected as a state function. This set of equations allows for a simpler controller tuning given that the variation of the osculating orbital elements with time is smoother than the variation of the corresponding Cartesian components. Moreover, this allows for larger integration steps and therefore a faster computation process, despite the increased complexity of the model. The state dynamics based on Keplerian elements is not used as the dynamic model for the LQR since it leads to a more complex linearization process and result. For the MPC, it is not necessary to linearize the system since this control strategy allows to use nonlinear system dynamics as the state function.

4.2.2 State Function

The orbital motion model (4) provide a convenient model to fully characterise the orbit and location of the satellite, using the set of Keplerian elements ${}^{I}\mathbf{x}_k = [a\ e_x\ e_y\ i\ \Omega\ u]^T$ as the system state, which is then compared with the reference satellite set of Keplerian elements to correct the real satellite orbital position and velocity. For the design of the MPC, it is considered that ${}^{O}\mathbf{f} = {}^{O}\mathbf{f}_c + {}^{O}\mathbf{f}_{J_2}$, where ${}^{O}\mathbf{f}_c$ is the control acceleration vector expressed in the Local Orbit frame centered in the real spacecraft and ${}^{O}\mathbf{f}_{J_2}$ is the acceleration caused by the J_2 Earth gravity component, expressed in the same reference frame. Notice that, to reduce the computational effort during the optimisation process, only the J_2 Earth gravity component is considered.

4.2.3 MPC Synthesis

A MATLAB script was developed that implements the MPC and allows to assess its performance in driving the error between the real satellite and the reference to zero. The MPC computes the next control action as a result of an optimisation process that uses a prediction of the system behaviour during the prediction horizon. Notice that the state function defined in Section 4.2.2 is nonlinear. Thus, nonlinear MPC is used as the control strategy. This state function is used to obtain the system evolution during the prediction horizon. However, the MPC uses a discrete-time model for this prediction. Therefore, it is good practice to provide the controller with a discrete-time state function. Otherwise, MATLAB automatically discretizes the model using the implicit trapezoidal rule, increasing the computational effort. To hasten the optimisation process, the Forward Euler's method was used to discretize the state function (4).

An output function to compute the system output that will be compared with the reference need to be defined. Since the spacecraft will be equipped with sensors providing position and velocity and the state ${}^{I}\mathbf{x}_k$ can be directly obtained from these measurements, the system output is the state itself. Therefore, the output function is simply given by ${}^{I}\mathbf{y} = \mathbf{I}_6\,{}^{I}\mathbf{x}_k$, where $\mathbf{I}_6$ is the 6×6 identity matrix. The system output is used to compute the reference tracking term of the cost function. The cost associated with the control action is also considered by the manipulated variable cost term. To achieve good results, it is crucial to fine-tune the weight of these terms. The strategy implemented uses an empirical method that considers the magnitude of each of the terms evaluated by the cost function to find initial values for the tuning process. Contrasting with the LQR implementation, the MPC allows to define constraints on the control acceleration, so that the optimisation process only considers control actions that are within the saturation limits of the actuators.

The MPC controller performance can be assessed in order to evaluate the effect that the control input, computed by the controller, has on the system. A similar strategy to the one used to implement the LQR (Section 4.1.3) is proposed. The properties of the reference and the real satellites are as described in Section 4.1.3. Contrasting with the LQR implementation, in which case the system state is the error between the real satellite and the reference one, the MPC uses the set of Keplerian elements for near-circular orbits of the real satellite, expressed in the ToD reference frame, as the system state. The system output is compared with the reference satellite set of Keplerian elements, expressed in the same reference frame.

The control acceleration is computed using the MATLAB command

$$[\mathbf{mv}, \mathbf{opt}] = \text{nlmpcmove}(\mathbf{nlmpcobj}, {}^{I}\mathbf{x}_k, \mathbf{lastmv}, \mathbf{Ref}),$$

where **mv** is the next control acceleration to be used in the system, **nlmpcobj** is a MATLAB object containing the MPC design properties, ${}^{I}\mathbf{x}_k$ is the current system state, **lastmv** is a 3×1 vector containing the last control action used, and **Ref** is a $p_h \times 6$ matrix containing the Keplerian elements of the reference satellite during the prediction horizon. The MATLAB object **opt** contains initial guesses for the state and manipulated variable (control action) trajectories to be used in the next control intervals, until the end of the prediction horizon [12]. It is possible to use several of these guesses for the next control actions in order to avoid running the optimisation process at every simulation step and, in this way, reduce the computational effort. It is good practice to use this strategy when the orbital error is large, given that, in this situation, the variations of the control acceleration, from one simulation step to the other, are not very significant when compared to a situation characterised by an orbital error closer to zero.

4.3 Control Solution

To achieve good performance for different magnitudes of the position error, four different controllers are implemented – MPC 1, MPC 2, LQR 1, and LQR 2 – that drive and maintain at zero the orbital error between the real satellite and the reference. The two LQR controllers are used to correct small orbital errors given that the LQR is able to correct errors close to zero with less computational effort than the MPC. LQR 1 is designed to correct small errors between the real satellite and the reference. LQR 2, on the other hand, is optimised to reduce the error in steady-state. Its purpose is to counter perturbative accelerations that affect the real orbit and handle sensors and actuators noise, in order to maintain the orbital error within the required values. The difference in the implementation of these two controllers is in the definition of matrices **Q** and **R**. The two MPC controllers are used to correct orbital errors far from zero since, in this case, the linearized model is no longer valid. Moreover, for large orbital errors, the control acceleration computed is saturated by the actuators. Therefore, the MPC provides better results since it accounts for actuators saturation in the optimisation process. MPC 1 is designed to correct larger orbital errors than

the MPC 2 and these two controllers differ in the following design parameters:

- Sample time, T_s - a higher sample time is used for MPC 1 (high orbital errors), given that a smaller propagation accuracy is required in this case;
- Prediction horizon, p_h - a higher prediction horizon is used for MPC 1 (high orbital errors), given that the response time is higher in this case;
- Weights - tuning of the weights to use in the cost function is also needed for each of the MPC controllers.

The orbital position error between the real satellite and the reference, at the current simulation step, is used to select the controller that will compute the next control acceleration. Therefore, at every simulation step, the orbital position error is evaluated to decide which controller is the best to use. For orbital position errors below a given threshold, a different controller is selected. However, if the orbital position error returns to a value above that threshold, the controller in use remains the same, except in the transition from LQR 2 to LQR 1. In this case, the controller in use returns to LQR 1 since LQR 2 is designed for steady-state.

5 Simulation Setup

The TerraSAR-X spacecraft physical characteristics and orbit described in Table 1 are used in the simulations.

The NavSBR GPS receiver is used for accurate navigation data measurements. The NavSBR has a 3σ position and velocity accuracy of 1 m and 0.02 m/s, respectively. This accuracy was modeled by adding Gaussian noise, with zero mean and 3σ equal to the receiver position and velocity accuracy, to the system output.

Electric propulsion systems present several advantages when compared to chemical propulsion, namely a higher specific impulse, the capability of operating high variations of thrust, and also provide better actuation accuracy [13]. On the other hand, the amount of thrust produced by electric engines is very small, which may lengthen the correction of large orbital errors, such as the ones resulting from the launcher orbital injection. Therefore, this type of propulsion is more suitable for low-thrust and long-duration applications, such as orbit maintenance maneuvers. However, most missions cannot afford embarking two different types of thrusters. Therefore, in this work, it was decided to equip the spacecraft solely with electrical propulsion, which is the system that offers better prospects of reducing the orbit maintenance errors.

TerraSAR-X has a total wet mass of 1230 kg, meaning that significant thrust values are required to correct large orbital errors. BUSEK BHT-8000 Hall thruster provides a high nominal thrust, which means that this actuator fits to achieve the control objectives. Nevertheless, one single thruster does not provide sufficient thrust to correct considerable orbital errors of a large satellite such as TerraSAR-X. Therefore, the strategy adopted uses $9 \times$ BUSEK BHT-8000 thrusters, three in each of the corresponding actuation axis. Table 3 summarises BUSEK BHT-8000 properties.

Table 3: BUSEK BHT-8000 thruster properties [14].

BUSEK BHT-8000	
Nominal thrust (max.)	450 mN
I_{sp}	2210 s
Power	8 kW
Thrust error (2σ)	1%
Qty.	9

Since chemical propulsion systems for this class of spacecraft can weigh more than 100 kg and electrical propulsion is nowadays the subject of intensive research and its thrust-to-mass ratio is expected to improve in the coming years, it was decided to maintain the original TerraSAR-X wet mass. Since no information is provided on the datasheet of BUSEK BHT-8000 about thrust accuracy, a general example of a mission that uses electric thrusters is used to estimate this parameter. According to [15], the difference between thrust demand and actual thrust for the QinetiQ T5 ion thrusters used in the ESA GOCE mission occurs within ± 1% of the thrust demand. Therefore, a similar thrust error is assumed for the BUSEK BHT-8000. This error is modeled by adding Gaussian noise, with zero mean and 2σ equal to 1% of the thrust demand, to the actuator's output.

The Forward Euler method, with a time step of 1 s, is used for the propagation of the real and the reference satellite since good propagation accuracy was achieved using this method without the need for excessive computational effort, as verified in the simulations.

Three threshold values, that are compared with the orbital position error between the real satellite and the reference, are defined to decide which controller will compute the next control acceleration. A variable threshold determines the transition from MPC 2 to LQR 1. The value of this threshold is empirically adapted to the simulation to perform, in order to improve the performance. The threshold values to determine the transition between each controller are presented in Table 4.

Table 4: Threshold values to select the controller.

Controller threshold	
MPC 1	≥ 100 km
MPC 2	100 km $\leftrightarrow$ variable
LQR 1	variable $\leftrightarrow 0.5$ km
LQR 2	≤ 0.5 km

To design the two MPC controllers, several control parameters must be selected. The values used are summarised in Table 5. A compromise is achieved between computational effort and control performance since it is not feasible to define a prediction time ($p_h\, T_s$) equal to the closed-loop response. For the MPC 2, since the orbital error is closer to zero, the propagation accuracy becomes more important

to achieve convergence. Therefore, a smaller sample time is chosen. The control horizon has the same dimension as the prediction horizon for both MPC controllers, since a decrease in the value of this parameter results in a loss of control performance, as observed in the simulations. For the MPC 1, all the control actions of the control vector (with dimension equal to the control horizon) computed by the MPC are used to control the real satellite, before a new control vector is obtained. On the other hand, for the MPC 2, only the first 40 elements of the control vector are used.

Table 5: MPC controllers parameters.

Parameters	MPC 1	MPC 2
T_s	10 s	5 s
p_h	50 steps	50 steps
m	50 steps	50 steps
$\mathbf{w^y}$	$[0.009\ 11\ 11\ 2\ 2\ 4]^T$	$[0.009\ 11\ 11\ 2\ 2\ 4]^T$
$\mathbf{w^u}$	$[0.1\ 0.1\ 0.1]^T$	$[0.1\ 0.1\ 0.1]^T$

To implement the two LQR controllers, the matrices **Q** and **R** need to be defined using the methodology presented in Section 4.1.3. For LQR 1, these matrices are given by

$$\mathbf{Q}_{LQR1} = \text{diag}\left(\left[\frac{1}{10^{-2}}\ \frac{1}{10^{-2}}\ \frac{1}{10^{-2}}\ \frac{1}{10^{-8}}\ \frac{1}{10^{-8}}\ \frac{1}{10^{-8}}\right]^T\right)$$

and

$$\mathbf{R}_{LQR1} = \text{diag}\left(\left[\frac{1}{1.7183^{-15}}\ \frac{1}{1.7183^{-15}}\ \frac{1}{1.7183^{-15}}\right]^T\right).$$

For LQR 2, the matrix **R** is adapted to work when the error in the system state is small. Therefore, the cost of the control action is reduced in order to deal with actuator and sensors noise and maintain the orbital error small enough to fulfil the mission requirements, despite the effect of orbital perturbations. Defining $\mathbf{Q}_{LQR2} = \mathbf{Q}_{LQR1}$, the matrix **R**, for LQR 2, is given by

$$\mathbf{R}_{LQR2} = \text{diag}\left(\left[\frac{1}{3.4285^{-14}}\ \frac{1}{3.4285^{-14}}\ \frac{1}{3.4285^{-14}}\right]^T\right).$$

An update time of 50 s is chosen to update the Kalman gain, for both LQR controllers, since the elements of **K** present small variations over time. This value offers good results while reducing the computational effort and is obtained using a trial-and-error methodology.

6 Simulation Results

In order to have a representative example of the type of orbital errors that can occur, the injection accuracy of the Vega-C launch system is used to set the initial errors. The injection accuracy (3σ of a Gaussian distribution) of the Vega-C launch system is presented in Table 6, based on [16]. In addition, an error in u is simulated in this section to represent the correction of an advance in the orbital position of the real satellite.

Table 6: Vega-C injection accuracy (3σ) [16].

Injection accuracy	
a	15 km
e	0.0012
i	0.15 deg
Ω	0.2 deg

6.1 Combined injection error

Using a combination of errors in the Keplerian elements between the real and the reference satellite, it is possible to define three different sets of injection errors with different magnitudes. The worst-case scenario error, the one with the highest magnitude, is defined considering the maximum injection error in all Keplerian elements shown in Table 6 and an additional error in u. The other two errors are also characterised by a combination of errors in all Keplerian elements, but with a smaller magnitude in each of them. These three orbital errors are defined in Table 7.

Table 7: Combined injection errors between the real and the reference satellite.

Error	a (km)	e_x	e_y	i (deg)	Ω (deg)	u (deg)
a)	-15	$+0.0012$	0	$+0.15$	$+0.2$	$+2$
b)	$+8$	$+0.0006$	0	-0.07	$+0.1$	$+1.2$
c)	$+3$	-0.0002	0	-0.04	-0.07	-0.5

The controller performance is assessed for a simulation with a duration equal to 7 complete orbits of the reference satellite (approximately 39830 s). In this simulation, ideal actuators and sensors are considered, in the sense that no noise interference is simulated. Later in this document, the controller performance is evaluated in a simulation with realistic actuators and sensors.

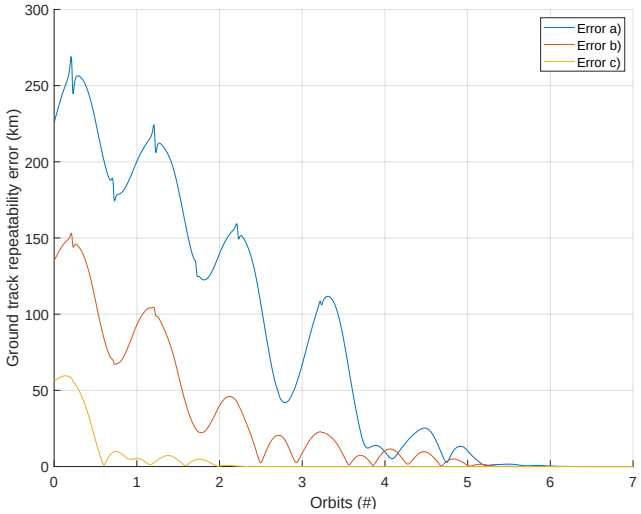

Figure 2: Ground track repeatability error between the real and the reference satellite for the errors defined in Table 7.

Figure 2 illustrates the ground track repeatability error, throughout the simulation, for each of the initial errors defined in Table 7. Notice that all of them converge to zero, ensuring that the ground track repeatability error requirement is fulfilled. In this case, the transition from MPC 2 to LQR 1 occurs at a position error magnitude of 8 km.

Further analysis is performed for the worst case scenario injection error (error a)). Figure 3 illustrates the variation of the Keplerian elements error between the real and the reference satellite, for this initial error. The correction of all Keplerian elements is successfully achieved after approximately 6 complete orbits (34081 s), at the transition from LQR 1 to LQR 2. However, due to the large errors to be corrected, this process is very demanding with significant oscillations in all Keplerian elements, mainly in the semi-major axis and the eccentricity vector, which justifies the need to adapt the near-circular orbit system dynamics proposed in [8] to the system dynamics given by (4).

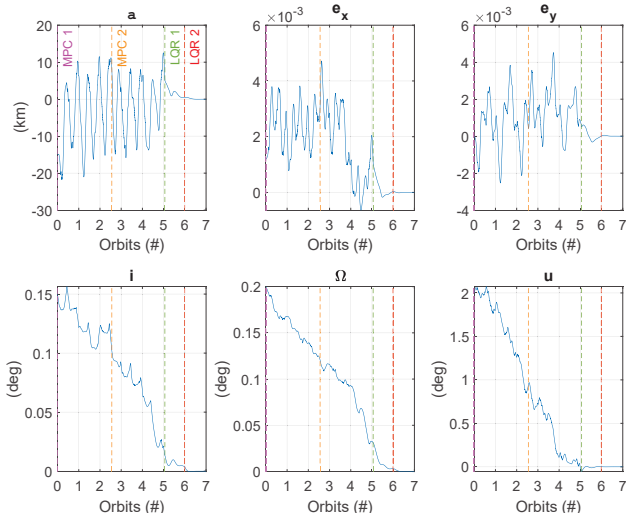

Figure 3: Keplerian elements error between the real and the reference satellite for error a). The transition between the controller in use is marked with a dashed line.

6.2 Combined injection error with realistic sensors and actuators

Error a), defined in Section 6.1, is used to assess the controller performance in a simulation with realistic actuators and sensors, as defined in the simulation setup. Figure 4 illustrates the evolution of the ground track repeatability error for a simulation with a duration equal to 7 complete orbits of the reference satellite (approximately 39830 s). In this case, the transition from MPC 2 to LQR 1 occurs at a position error magnitude of 8 km. As presented, the objective to reduce the value of this error to less than 500 m is still achieved after approximately 5.5 complete orbits (30985 s), despite using realistic sensors and actuators that induce noise in the system. Actually, the effect of the non-idealities of the sensors and the actuators is not visible when compared to the magnitude of the initial error, as illustrated in Fig. 4.

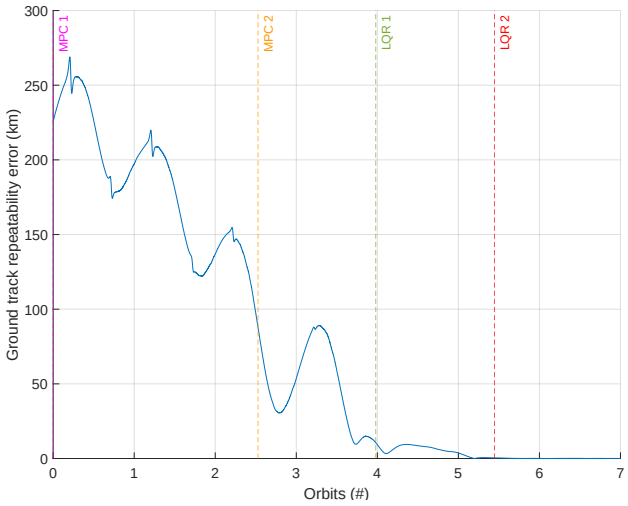

Figure 4: Ground track repeatability error between the real and the reference satellite for initial error a), using realistic sensors and actuators.

The effect of using realistic sensors and actuators becomes visible after reaching the steady-state. Figure 5 shows the evolution of the ground track repeatability error during three complete orbits of the reference satellite (approximately 17070 s), after the seventh complete orbit of this satellite. At this point, the system is in steady-state and LQR 2 is controlling the system. Notice that the ground track repeatability error presents values up to 0.21 km. These error values are well within the requirements for this parameter (0.5 km), which confirms the effectiveness of the control strategy.

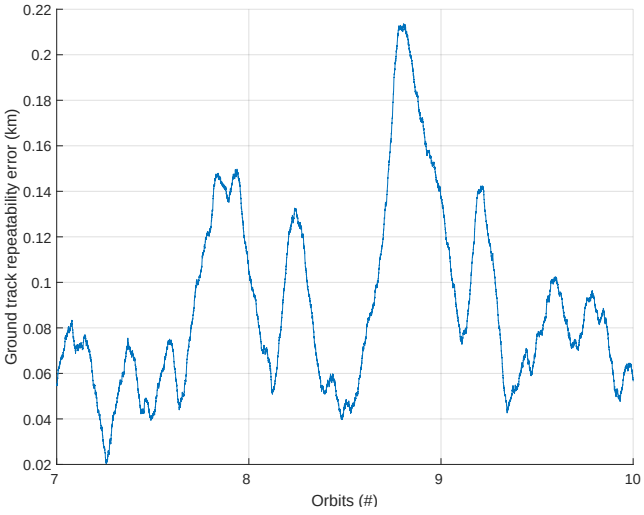

Figure 5: Ground track repeatability error between the real and the reference satellite after reaching the steady-state, using realistic sensors and actuators.

7 Conclusions and Future Work

Although good results were obtained, the controller tuning was proven difficult and its performance dependent on the initial orbital state, due to the complexity of the system

and the relatively small prediction window used in the design of the MPC. The computational effort required to compute the control actions limited the dimension of the prediction window, decreasing the control performance. A strategy to simplify the system model can be explored to reduce the computational effort and, in that way, enable the increase of the prediction horizon. The obtained MPC results in correcting large initial orbit errors were interesting to overcome the limitations of the LQR related to the linearization process. Nonetheless, LQR is a computationally efficient strategy and presented good results for orbit maintenance routines, when the orbit error is close to zero.

The AOC strategy developed in this work demonstrated effective results in correcting orbit injection errors in different Keplerian elements. The TerraSAR-X mission requirements were fulfilled even in the worst-case scenario when a combination of maximum injection error in all Keplerian elements was used as the initial state. However, the actuation induces an initial oscillatory response of some orbital parameters (Fig. 3) before reaching the steady-state. It is expected that a larger prediction horizon will significantly improve the results obtained, at the cost of increased computational effort. Moreover, the AOC algorithm demonstrated good performance for orbit maintenance activities. The controller was capable of counteracting the effect of orbital perturbations, maintaining the orbital error close to zero. The use of realistic sensors and actuators, which introduced noise in the system, did not impact the initial convergence of the proposed algorithm. In the steady-state, the impact of these non-idealities is more evident and caused the ground track repeatability error to present values up to 0.21 km, still well within the requirements defined for the TerraSAR-X mission. The performance of the developed AOC algorithm should also be evaluated for a smaller satellite. In this case, the required thrust to perform orbit correction activities is smaller and, therefore, the electric propulsion is best suited.

The use of electric propulsion was proven effective for orbit maintenance activities. However, the correction of large initial errors required the use of multiple thrusters to achieve sufficient actuation force to correct the orbit of TerraSAR-X, due to its relatively high mass. Future improvements on the electric propulsion topic might overcome this problem with the development of more powerful thrusters. Future work should also compare the use of chemical propulsion with the one proposed in this research, which is based on electrical propulsion, taking into consideration convergence time and propellant and energy costs. Moreover, a solution that combines the two types of propulsion mentioned should not be discarded.

References

[1] Sergio De Florio and Simone D'Amico. Optimal autonomous orbit control of remote sensing spacecraft. *19th AAS/AIAA Space Flight Mechanics Meeting*, 2014.

[2] M.M. Tavakoli and Nima Assadian. Model predictive orbit control of a low earth orbit satellite using Gauss's variational equations. *Proceedings of the Institution of Mechanical Engineers, Part G, Journal of Aerospace Engineering 228(13):2385-2398*, 2014.

[3] De Florio S, D'Amico S, and Radice G. Flight results of the precise autonomous orbit keeping experiment on the PRISMA mission. *Journal of Spacecraft and Rockets 50(3):662-674*, 2012.

[4] Mirko Leomanni, Andrea Garulli, and Antonio Giannitrapani. An adaptive groundtrack maintenance scheme for spacecraft with electric propulsion. *Acta Astronautica*, 2019.

[5] Simone D'Amico, C. Arbinger, and M. Eineder. Precise ground-in-the-loop orbit control for low earth observation satellites. *18th ISSFD Meeting, Munich*, 2004.

[6] Yuanbo Wu, Mingjun Pu, and Donghong Wang. Multi-objective optimization method for repeat ground-track orbit design considering the orbit injection error. *Journal of Aerospace Technology and Management 10*, 2018.

[7] F. Landis Markley and John L. Crassidis. *Fundamentals of Spacecraft Attitude Determination and Control*. Springer, 2014.

[8] Karel F. Wakker. *Fundamentals of Astrodynamics*. Faculty of Aerospace Engineering, TU Delft, 2015.

[9] Oliver Montenbruck and Simone D'Amico. Proximity operations of formation-flying spacecraft using an eccentricity/inclination vector separation. *Journal of Guidance Control and Dynamics, Vol. 29, No. 3*, 2006.

[10] Michel Capderou. *Handbook of Satellite Orbits - From Kepler to GPS*. Springer, 2014.

[11] E. Okyere, A. Bousbaine, G.T. Poyi, A.K. Joseph, and J.M. Andrade. LQR control design for quad-rotor helicopters. *The Journal of Engineering 2019(17)*, 2018.

[12] MathWorks. nlmpcmove. `https://www.mathworks.com/help/mpc/ref/nlmpc.nlmpcmove.html#mw_f32e081b-0763-4b1a-8df4-2b4b325ecbc3`, visited on September 4, 2021.

[13] S. Marcuccio, F. Ceccanti, and M. Andrenucci. Control strategies for orbit maintenance of LEO small satellites with feep. *Spacecraft Propulsion, Third International Conference*, 2000.

[14] BUSEK, Co. Inc. *BUSEK BHT-8000 Hall Effect Thruster*, 2019.

[15] Neil Wallace, Peter Jameson, Christopher Saunders, Michael Fehringer, Clive Edwards, and Rune Floberghagen. The GOCE ion propulsion assembly – lessons learnt from the first 22 months of flight operations. *32nd International Electric Propulsion Conference*, 2011.

[16] Arianespace Inc. *Vega C User's Manual, Issue 0 Revision 0*, 2018.