

# CONSTRAINED ATTITUDE PATH PLANNING VIA LEAST SQUARES MRP-BASED NURBS CURVES

R. Calaon\* and H. Schaub†

Maneuvering a spacecraft that is subject to orientation constraints is a nontrivial problem. Such orientation constraints can consist of, but are not limited to, keep-in zones that a certain body-fixed direction in the spacecraft must remain within, or vice versa, keep-out zones that a certain body-fixed direction must stay out of. This paper proposes a solution to the constrained attitude maneuvering problem applying path-planning algorithm to an undirected graph consisting of nodes sampled in the 3D Modified Rodrigues Parameters (MRPs) attitude space. Furthermore, this paper investigates the use of Non-Uniform Rational Basis Spline (NURBS) curves to obtain a smooth, time-dependent and twice-differentiable attitude reference trajectory from a sequence of constraint-compliant attitude waypoints. Different NURBS curves are presented and their performances compared in order to yield and effort-optimal solution.

## INTRODUCTION

Constraints in a spacecraft's orientation often present challenges in space missions, as they make maneuvering a spacecraft a nontrivial problem. These hard orientation constraints can be categorized in keep-out constraints and keep-in constraints. Keep-out constraints describe the problem where a body-fixed direction in the spacecraft must stay away from a certain inertial direction. This can be the case for a sensitive instrument, such as a camera or a star tracker, that must not point at a bright celestial object like the Sun. Often, such instruments have a certain field of view, which means that the keep-out zone consists of a cone in inertial space around the direction of the bright celestial object. Keep-in constraints represent the opposite problem, where the spacecraft is required to maneuver while maintaining a certain body fixed direction within a certain angular distance from an inertial direction. This can be the case for sun sensors, which must be able to see the Sun within their field of view at all times, or solar panels, for which the sunlight incidence angle must remain within certain bounds to ensure continuous power generation.

The solutions to the constrained attitude maneuvering problem that are found in literature can be broadly categorized into two groups: potential-function-based and path-planning-based. Solutions that rely on potential functions typically consist of an attractive potential, that causes the output control law to steer the spacecraft towards the desired attitude, and a repulsive potential, whose output control component steers the spacecraft away from the keep-out zone or the boundaries of the keep-in zone.<sup>1-4</sup> Such approaches are straightforward to implement and computationally cheap,

---

\*Ph.D. student, Ann and H.J. Smead Department of Aerospace Engineering Sciences, University of Colorado, Boulder, 3775 Discovery Drive, Boulder, CO, 80303

†Professor and Department Chair, Schaden Leadership Chair, Ann and H.J. Smead Department of Aerospace Engineering Sciences, University of Colorado, Boulder, 431 UCB, Colorado Center for Astrodynamics Research, Boulder, CO, 80309. AAS Fellow, AIAA Fellow.

but they can fail in the presence of complex geometries and/or overlapping constraints: whenever the resulting potential has local minima other than the desired target attitude, they can cause the spacecraft to remain stuck in the wrong configuration. Path-planning based approaches, on the other hand, usually rely on some sort of discretization of the attitude space to build a graph of constraint-compliant nodes, which can be navigated from the initial to the target attitude.<sup>5-7</sup> In these cases, graph-searching algorithms such as depth-first,<sup>8</sup> breadth-first<sup>9</sup> or A\*<sup>10</sup> are applied to navigate such graphs. In different approaches, the attitude space sampling is not deterministic, but rather stochastic: for example, Probabilistic RoadMaps<sup>11</sup> sacrifice the completeness of the attitude space mapping in exchange for faster execution times. These latter approaches do not suffer the problem of local minima. However, all they typically provide is a sequence of constraint-compliant attitude waypoints, which means that there is no further information about the attitude as a function of time or the angular rates required along the maneuver. Some recent contributions by Tan et al.<sup>12</sup> and Calaon et al.<sup>13</sup> provide solutions based on attitude sampling, combined with some form of interpolation to obtain a smooth reference trajectory from a sequence of attitude waypoints.

This paper is based on the Modified Rodrigues Parameters (MRPs) nonsingular attitude discretization presented in Reference 13. The focus is on the second part of the problem, which is how to obtain a smooth reference trajectory from a sequence of attitude waypoints. Reference 13 uses Non-Uniform Rational Basis Spline curves to obtain a twice differentiable reference trajectory that precisely interpolates all the baseline waypoints in the path provided by the graph-search algorithm. While effective, this approach displays some suboptimal behaviors that can be tied to the nature of the interpolating function: as the grid density  $N$  is increased to obtain a more accurate mapping of the obstacles, the function presents some parasitic oscillatory behavior to meet the requirement of interpolating every waypoint. These oscillations, ultimately, cause the total control effort required by the trajectory to be suboptimal, as mentioned in Reference 13 and highlighted by simulations in Reference 14. This paper proposes a different type of NURBS curve that performs a least-squares fit of the waypoints and the desired angular rates along the trajectory, in order to obtain a trajectory that still meets the requirements, but is desensitized with respect to the grid points and grid density. The goal is to achieve a smoother reference trajectory that requires a smaller control effort to be tracked by reducing the dependence on the chosen waypoints.

This paper is structured as follows: first, an overview of MRPs and their key properties and equations is given, together with a brief description of the attitude discretization used in this paper. Secondly, the requirements for the reference trajectory are defined, and the new Least Squares approximating NURBS is derived mathematically. The two NURBS curves, the interpolating curve and the LS approximating curve are compared to one-another in three different scenarios, to compare their performance with baseline A\* solutions. Finally, the effort-based A\* is applied to the same three scenarios and incorporating the two NURBS curves, to benchmark the performance of the two versus different grid density levels.

## **MRP WORKSPACE DISCRETIZATION**

### **Modified Rodrigues Parameters**

The Modified Rodrigues Parameters (MRPs) are a 3-dimensional coordinate set used to represent the attitude of a spacecraft that is able to rotate in the  $SO(3)$  space. MRPs can be defined from the

corresponding set of Principal Rotation Vector (PRV) and Principal Rotation Angle ( $\hat{e}, \Phi$ ) as:

$$\boldsymbol{\sigma} = \frac{1}{1 + \cos(\Phi/2)} \begin{Bmatrix} e_1 \sin(\Phi/2) \\ e_2 \sin(\Phi/2) \\ e_3 \sin(\Phi/2) \end{Bmatrix} = \hat{e} \tan(\Phi/4). \quad (1)$$

Being a minimal set, MRPs present a singularity. As Equation (1) shows, such singularity appears for  $\Phi = \pm 2\pi$ , which corresponds to a rotation of 360 degrees from the reference attitude. MRPs can be visualized in a 3D cartesian space, where the origin represents a null rotation with respect to the reference attitude, whereas every point located at an infinite distance from the origin describes a rotation of  $|\Phi| \rightarrow 2\pi$  from the origin. Moreover, for  $\Phi = \pm\pi$  it is  $\sigma = \|\boldsymbol{\sigma}\| = 1$ , therefore all the points in MRP space that lie on a unit sphere centered at the origin represent 180 degree rotations from the reference attitude. For every principal elements set ( $\hat{e}, \Phi$ ), it is possible to define a shadow set ( $\hat{e}, \Phi'$ ), with  $\Phi' = 2\pi - \Phi$ , which represents the same attitude. Assuming without loss of generality that  $0 \leq \Phi \leq \pi$ , the standard set ( $\hat{e}, \Phi$ ) represents the short rotation from the reference to the target attitude, whereas the shadow set ( $\hat{e}, \Phi'$ ) represents the long rotation ( $\pi \leq \Phi' \leq 2\pi$ ). This property can be exploited to obtain the shadow MRP set:

$$\sigma_i^S = \frac{e_i \sin(\Phi'/2)}{1 + \cos(\Phi'/2)} = \frac{e_i \sin(\Phi/2 - \pi)}{1 + \cos(\Phi/2 - \pi)} = \frac{-\sigma_i}{\sigma^2} \quad \text{for } i = 1, 2, 3. \quad (2)$$

From Equations (1) and (2) it can be shown that every MRP set obtained from the short rotation lies within the unit sphere in MRP space, whereas when the long rotation is chosen, the corresponding shadow MRP set lies outside of said unit sphere. This property makes MRPs a redundant set of coordinates, where every possible rotation in  $SO(3)$  (except the 0 and 360 degree rotation) is mapped to two distinct MRP sets in  $\mathbf{R}^3$ . This is a key property that is exploited in this paper, as it allows to represent the attitude space with a unit sphere centered in the origin of the MRP space. This can be done without loss of completeness, since the unit sphere completely describes the attitude space, and at the same time it avoids the singularity in the MRP formulation, which does not appear for points that lie inside the sphere.

The time derivatives of MRP sets  $\dot{\boldsymbol{\sigma}}$  and  $\ddot{\boldsymbol{\sigma}}$  can be correlated to the spacecraft's angular rate and acceleration vectors  ${}^{\mathcal{B}}\boldsymbol{\omega}_{\mathcal{B}\mathcal{N}}$  and  ${}^{\mathcal{B}}\dot{\boldsymbol{\omega}}_{\mathcal{B}\mathcal{N}}$  with respect to the inertial frame  $\mathcal{N}$ , expressed in the spacecraft's body frame  $\mathcal{B}$ . For ease of notation, such vectors will be referred to as  $\boldsymbol{\omega}$  and  $\dot{\boldsymbol{\omega}}$  in the following equations. The differential kinematic equations for the angular rates are:

$$\dot{\boldsymbol{\sigma}} = \frac{1}{4} [B(\boldsymbol{\sigma})] \boldsymbol{\omega} \quad (3)$$

$$\boldsymbol{\omega} = \frac{4}{(1 + \sigma^2)^2} [B(\boldsymbol{\sigma})]^T \dot{\boldsymbol{\sigma}} \quad (4)$$

where the  $[B(\boldsymbol{\sigma})]$  matrix is:

$$[B(\boldsymbol{\sigma})] = (1 - \sigma^2) [\mathbf{I}_{3 \times 3}] + 2[\tilde{\boldsymbol{\sigma}}] + 2\boldsymbol{\sigma}\boldsymbol{\sigma}^T \quad (5)$$

and  $[\tilde{\boldsymbol{\sigma}}]$  is the skew-symmetric matrix obtained from the 3D vector  $\boldsymbol{\sigma}$ . The equations for the angular accelerations are obtained differentiating Equations (3) and (4):

$$\ddot{\boldsymbol{\sigma}} = \frac{1}{4} \left( [B(\boldsymbol{\sigma})] \dot{\boldsymbol{\omega}} + [\dot{B}(\boldsymbol{\sigma}, \dot{\boldsymbol{\sigma}})] \boldsymbol{\omega} \right) \quad (6)$$

$$\dot{\boldsymbol{\omega}} = \frac{4}{(1 + \sigma^2)^2} [B(\boldsymbol{\sigma})]^T \left( \ddot{\boldsymbol{\sigma}} - \frac{1}{(1 + \sigma^2)^2} [\dot{B}(\boldsymbol{\sigma}, \dot{\boldsymbol{\sigma}})] [B(\boldsymbol{\sigma})]^T \dot{\boldsymbol{\sigma}} \right). \quad (7)$$

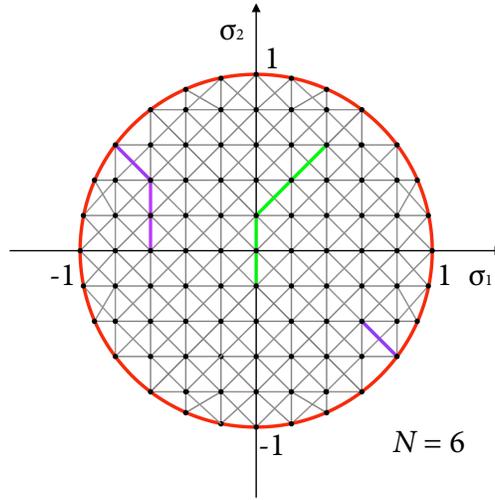
where:

$$[\dot{B}(\boldsymbol{\sigma}, \dot{\boldsymbol{\sigma}})] = (-2\boldsymbol{\sigma}^T \dot{\boldsymbol{\sigma}})[\mathbf{I}_{3 \times 3}] + 2[\dot{\boldsymbol{\sigma}}] + 2(\boldsymbol{\sigma} \dot{\boldsymbol{\sigma}}^T + \dot{\boldsymbol{\sigma}} \boldsymbol{\sigma}^T). \quad (8)$$

More details on MRPs and their properties are found in Reference 15.

### Workspace discretization and constraint representation

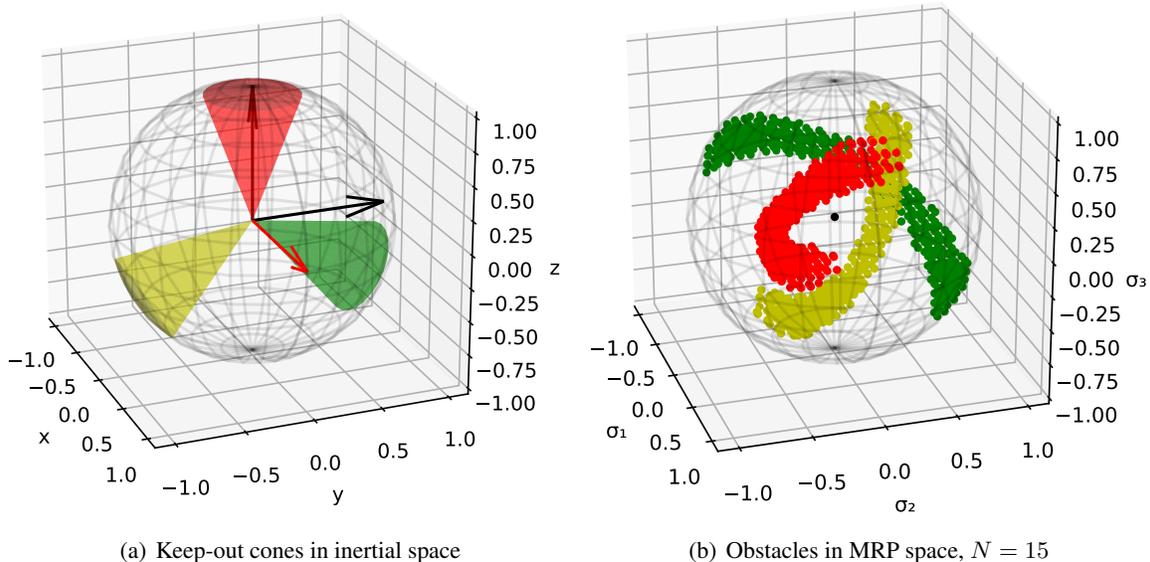
As previously mentioned, the attitude workspace can be reduced to a unit sphere in MRP space, centered at the origin, without loss of generality. For the present path-planning-based application, the attitude workspace is sampled in order to provide waypoints for the algorithm to navigate from the initial attitude to the target attitude. The sampling technique consists in discretizing the unit sphere with a cartesian grid of equally spaced nodes in MRP space. The grid density  $N$  is defined as the number of nodes contained along each principal cartesian semiaxis. Figure 1 shows a 2D rendering of the cartesian grid. Such cartesian grid constitutes the skeleton of an undirected graph, with the waypoints (nodes) in black, and the edges between two neighboring nodes drawn in gray. In reality, the MRP grid/graph exists in a 3D space, and each ‘internal’ node is connected to up to 26 neighboring nodes. This does not apply to the nodes located on the  $\sigma = 1$  boundary, which are instead connected to the proper neighboring nodes, and also to the neighboring nodes of their respective shadow set nodes. This enables the path-planning algorithm to switch to the opposite side of the unit sphere when the  $\sigma = 1$  boundary is encountered, thus performing MRP switching without entering the  $\sigma > 1$  domain. The paths in green and purple in Figure 1 represent, respectively, a shortest-distance path that does not require MRP switching, and one that does.



**Figure 1:** 2D MRP grid with two shortest-distance paths: i) green: no MRP switching: ii) purple: MRP switching<sup>13</sup>

Figure 1 shows the discretization of an attitude space where there are no rotational constraints. However, this paper deals with attitude path planning in the presence of such constraints, therefore, they must be accounted for in the graph. To this purpose, every node in the graph is checked for constraint compliance. Defining  ${}^B \hat{\mathbf{b}}_i$  the body-frame direction of the instrument and with  ${}^N \hat{\mathbf{s}}_i$  the inertial direction of the celestial object, constraint compliance can be verified using the formula:

$${}^B \hat{\mathbf{b}}_i \cdot [BN] {}^N \hat{\mathbf{s}}_i \leq \cos \beta_i \quad (9)$$



**Figure 2:** Three general keep-out constraints,  $\beta = 20 \text{ deg}^{13}$

where the  $<$  sign is used for keep-out constraints, the  $\geq$  for keep-in constraints, and  $[\mathcal{BN}]$  is the direction cosine matrix that maps a vector from the frame  $\mathcal{N}$  to the frame  $\mathcal{B}$ , and it is obtained as a function of the MRP set  $\sigma$ . The angle  $\beta_i$  in Equation (9) is the field of view (FOV) of the instrument. Figure 2 shows how a set of three keep-out constraints, in the presence of a sensitive instrument mounted along the body axis  ${}^{\mathcal{B}}\hat{b}_x$  map to the MRP space. Figure 1 (b) shows the constraint-incompliant nodes, color-coded to match the respective constraints. Such nodes are therefore removed from the graph and disregarded by the path-planning algorithm.

The most relevant features of the MRP discretization and obstacle representation have been highlighted in this section of the paper. For a more thorough description, the reader is referred to Reference 13.

### **PATH SMOOTHING: THE LEAST SQUARES APPROXIMATING NURBS**

Reference 13 showed how a sequence of constraint-compliant waypoints is not enough to provide an accurate reference for the desired slew maneuver, because it gives no insight on the required attitude, angular rate and angular acceleration as a function of time. Moreover, Reference 14 shows that a full reference trajectory can be tracked accurately by a spacecraft equipped with a set of reaction wheels, where the commanded torque obeys a Lyapunov-based feedback control law. The same cannot be said when the reference is only provided in terms of a series of attitude waypoints, in which case the tracking error between the reference and the actual attitude becomes significant. On the other hand, Reference 14 highlighted how the reference trajectory obtained from NURBS interpolation of the waypoints can require a global control cost that is suboptimal. This happens partially due to the constraint imposed on the interpolating NURBS curve: being forced to meet all the waypoints precisely, the interpolating curve presents wiggles between the waypoints that cause parasitic torques to appear when not needed. This phenomenon becomes more significant as the density of the waypoints increases, as it was initially highlighted in Reference 13.

This section develops a new type of Least Squares (LS) approximating NURBS curve, for which the condition of precise passage through the waypoints is not strictly enforced. The result is a curve that has the same properties of the interpolating NURBS curve, but has a smoother look, as it uses the waypoints as a baseline without being constrained to pass through them precisely.

Moreover, Reference 13 presents the requirement of obtaining a reference trajectory where the angular rate norm  $\|\boldsymbol{\omega}\|$  remains constant during the slew maneuver, in order to ensure that none of the limits on the individual angular rate components are violated.<sup>16</sup> This section aims to explore how to use the LS approximating NURBS not only to match the waypoints, but also to match a desired angular rate profile during the maneuver. Achieving the required angular rate norm profile by means of the NURBS curve only, instead of performing additional numerical manipulation as in Reference 13, removes a step in the effort-based A\* algorithm, ultimately improving its computational speed.

### NURBS curves properties

A NURBS (Non-uniform Rational B-Spline) curve is a parametric, piecewise-polynomial function that is characterized by a high level of smoothness. The piecewise-polynomial nature makes NURBS curves a good choice to fit large sets of data, because they do not suffer from the Runge's phenomenon, which arises when trying to fit the same large sets of data with one, high-order polynomial.<sup>17,18</sup> The degree of a NURBS is chosen by the user, and it does not depend on the number of data that needs to be fitted. Moreover, a NURBS curve of degree  $p$  is  $C^{p-1}$  in its entire domain, that is, continuous and differentiable  $p - 1$  times. For the present application, the desire is to obtain a NURBS curve that is at least  $C^3$ , such that both the first and second order derivatives are continuous and differentiable. This leads to a choice of a polynomial order  $p = 4$ . The general expression of a NURBS curve is:

$$\boldsymbol{\sigma}(u) = \sum_{i=0}^n N_{i,p}(u) \mathbf{P}_i \quad (10)$$

where  $u \in [0, 1]$  is the dimensionless parameter of the curve,  $\mathbf{P}_i$  are the  $n + 1$  control points, and  $N_{i,p}(u)$  are the basis functions of polynomial order  $p$  that are linearly combined to obtain the final piecewise-polynomial expression of  $\boldsymbol{\sigma}(u)$ . To define the basis function, it is necessary, first of all, to define a knot vector  $U$  containing the  $m + 1$  scalar terms:

$$U = \{\underbrace{0, \dots, 0}_{p+1}, u_{p+1}, \dots, u_{m-p-1}, \underbrace{1, \dots, 1}_{p+1}\}. \quad (11)$$

The  $p+1$  knots equal to 0 and 1 at the beginning and end of the knot vector are required to ensure that the curve passes exactly through the control points  $\mathbf{P}_0$  and  $\mathbf{P}_n$ . The intermediate knots correspond to the value of the parameter  $u$ , from now on defined as the dimensionless time, at which two different polynomial segments of the curve are joined. The basis functions are defined using the De Boor recursive formula:<sup>19</sup>

$$N_{i,0}(u) = \begin{cases} 1 & \text{if } u_i \leq u < u_{i+1} \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u).$$

The derivative of a NURBS curve is obtained as:

$$\boldsymbol{\sigma}'(u) = \sum_{i=0}^n N'_{i,p}(u) \mathbf{P}_i \quad (13)$$

where  $\boldsymbol{\sigma}'(u) = \frac{d\boldsymbol{\sigma}}{du}$  denotes the derivative with respect to the dimensionless time  $u$ . The derivatives of the basis functions  $N_{i,p}(u)$  are computed using the formula:<sup>17</sup>

$$N'_{i,p}(u) = \frac{p}{u_{i+p} - u_i} N_{i,p-1}(u) - \frac{p}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u). \quad (14)$$

### Improved time spacing

To build a NURBS curve, whether it is an interpolating curve or a LS approximating curve, it is necessary to define the time tags  $\bar{u}_k$  for  $k = 0, \dots, q$ , which correspond to the values of the dimensionless time  $u$  for which the curve passes through the  $k$ -th waypoint, with  $q + 1$  being the total number of waypoints. This is strictly true for an interpolating NURBS curve, where  $\boldsymbol{\sigma}(\bar{u}_k) = \boldsymbol{\sigma}_k$ , whereas for the LS approximating curve, this does not happen in general. In a generic case, Reference 17 suggests to compute the  $\bar{u}_k$ 's proportionally to a valid distance metric between two consecutive waypoints, in the space where such waypoints exist. In this paper, such metric is defined as:

$$\begin{aligned} \hat{\theta}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) &= \min \{ \theta(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2), \theta(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2^S) \} \\ \theta(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) &= 4 \arctan \left[ \frac{(1 - |\boldsymbol{\sigma}_2|^2) \boldsymbol{\sigma}_1 - (1 - |\boldsymbol{\sigma}_1|^2) \boldsymbol{\sigma}_2 + 2 \boldsymbol{\sigma}_1 \times \boldsymbol{\sigma}_2}{1 + |\boldsymbol{\sigma}_1|^2 |\boldsymbol{\sigma}_2|^2 + 2 \boldsymbol{\sigma}_1 \cdot \boldsymbol{\sigma}_2} \right] \end{aligned} \quad (15)$$

which, effectively, consists in the principal rotation angle of a rotation from  $\boldsymbol{\sigma}_1$  to  $\boldsymbol{\sigma}_2$ , assuming that the shortest rotation is always performed.

A better sampling of the  $\bar{u}_k$ 's time tags can be obtained when the desired properties of the NURBS curve are incorporated into the appropriate formulation. In this implementation, one of the goals is to obtain a reference trajectory with a constant angular rate norm. With respect to the final trajectory, the total angular distance swept by the spacecraft in its rotational motion is given by:

$$S(t) = \int_0^t \|\boldsymbol{\omega}\| d\tau. \quad (16)$$

A second order Taylor series expansion of  $S(t)$  can be performed around the generic time instant  $t_k$ , giving:

$$\begin{cases} S_{k+1} = S_k + \dot{S}_k(t_{k+1} - t_k) + \frac{1}{2} \ddot{S}_k(t_{k+1} - t_k)^2 \\ \dot{S}_{k+1} = \dot{S}_k + \ddot{S}_k(t_{k+1} - t_k) \end{cases}. \quad (17)$$

where  $S_k = S(t_k)$ . Eliminating  $\ddot{S}_k$  from Equation (17), yields the expression:

$$S_{k+1} - S_k = \frac{\dot{S}_k + \dot{S}_{k+1}}{2} (t_{k+1} - t_k). \quad (18)$$

Now, according to the Fundamental Theorem of Calculus, it is  $\dot{S}_k = \|\boldsymbol{\omega}(t_k)\| = \omega_k$ . The difference  $S_{k+1} - S_k$  can be approximated, assuming that the waypoints  $\boldsymbol{\sigma}_k$  and  $\boldsymbol{\sigma}_{k+1}$  are close enough

together, with  $S_{k+1} - S_k \approx \hat{\theta}(\boldsymbol{\sigma}_k, \boldsymbol{\sigma}_{k+1})$ . This gives the improved algorithm to calculate the time spacing between the waypoints:

$$\begin{aligned} t_0 &= 0 \\ t_{k+1} &= t_k + 2 \frac{\hat{\theta}(\boldsymbol{\sigma}_k, \boldsymbol{\sigma}_{k+1})}{\omega_k + \omega_{k+1}} \end{aligned} \quad (19)$$

from which the  $\bar{u}_k$ 's can be obtained normalizing the  $t_k$ 's by  $t_q$ , i.e., the time tag corresponding to the final waypoint.

### Least Squares waypoint approximation

This subsection outlines the procedure to obtain a LS approximating NURBS curve from a set of  $q + 1$  attitude waypoints. As previously mentioned, the idea that motivated the use of a LS fit of the waypoints, instead of a precise interpolation, is to reduce the dependence from the waypoints. The waypoints provide guidance in the attitude space to avoid the keep-out zones, but other than that, they are artificial constructs and, therefore, there is no real need to track them precisely.

The approach followed in this subsection only interpolates the first and last attitude waypoints  $\boldsymbol{\sigma}_0$  and  $\boldsymbol{\sigma}_q$ , since the initial and final attitudes are the only two pieces of information that the reference trajectory must for sure comply with. Moreover, it is also desired to obtain a reference trajectory that matches the required angular rates at the beginning and at the end of the maneuver. This translates into endpoint constraints also on the derivatives  $\dot{\boldsymbol{\sigma}}_0$  and  $\dot{\boldsymbol{\sigma}}_q$ , which are computed from the angular rates  $\omega_0$  and  $\omega_q$  using Equation (3). As opposed to the interpolating NURBS, in the LS approximation the number of control points  $\mathbf{P}_i$  for  $i = 0, \dots, n$  can arbitrarily be chosen by the user.

The first two control points,  $\mathbf{P}_0$  and  $\mathbf{P}_1$ , and the last two,  $\mathbf{P}_{n-1}$  and  $\mathbf{P}_n$ , are determined by imposing the endpoint coordinates and derivatives mentioned above. This gives the linear,  $4 \times 4$  system:

$$\begin{pmatrix} N_{0,p}(0) & 0 & 0 & 0 \\ N'_{0,p}(0) & N'_{1,p}(0) & 0 & 0 \\ 0 & 0 & N'_{n-1,p}(1) & N'_{n,p}(1) \\ 0 & 0 & 0 & N_{n,p}(1) \end{pmatrix} \begin{bmatrix} \mathbf{P}_0 \\ \mathbf{P}_1 \\ \mathbf{P}_{n-1} \\ \mathbf{P}_n \end{bmatrix} = \begin{bmatrix} \boldsymbol{\sigma}_0 \\ \boldsymbol{\sigma}'_0 \\ \boldsymbol{\sigma}'_q \\ \boldsymbol{\sigma}_q \end{bmatrix}. \quad (20)$$

The remaining  $\mathbf{P}_i$  for  $i = 2, \dots, q - 2$  are determined using a least squares approach. Before getting into the actual least squares fit, the following quantities are defined:

$$\boldsymbol{\rho}_k = \boldsymbol{\sigma}_k - N_{0,p}(\bar{u}_k)\mathbf{P}_0 - N_{1,p}(\bar{u}_k)\mathbf{P}_1 - N_{n-1,p}(\bar{u}_k)\mathbf{P}_{n-1} - N_{n,p}(\bar{u}_k)\mathbf{P}_n \quad \text{for } k = 1, \dots, q - 1. \quad (21)$$

The  $\boldsymbol{\rho}_k$  terms are used to remove the dependence of the waypoints from the four control points that have already been pre-computed. Defining with  $\boldsymbol{\sigma}_k$  the waypoints to be fitted, and with  $\boldsymbol{\sigma}(\bar{u}_k)$  the output of the NURBS curve according to Equation (10), it is possible to set up an error function that is the sum of the quadratic errors, and only depends on the remaining  $\mathbf{P}_i$  for  $i = 2, \dots, n - 2$ :

$$f = \sum_{k=1}^{q-1} \|\boldsymbol{\sigma}_k - \boldsymbol{\sigma}(\bar{u}_k)\|^2 = \sum_{k=1}^{q-1} \|\boldsymbol{\rho}_k - \sum_{i=2}^{n-2} N_{i,p}(\bar{u}_k)\mathbf{P}_i\|^2. \quad (22)$$

To minimize for  $f$ , its derivative with respect to the control points is equated to zero:  $\frac{df}{d\mathbf{P}_l} = 0$  for  $l = 2, \dots, n - 2$ . This gives:

$$\sum_{i=2}^{n-2} \left( \sum_{k=1}^{q-1} N_{l,p}(\bar{u}_k) N_{i,p}(\bar{u}_k) \right) \mathbf{P}_i = \sum_{k=1}^{q-1} N_{l,p}(\bar{u}_k) \boldsymbol{\rho}_k \quad \text{for } l = 2, \dots, n - 2. \quad (23)$$

For more details on this derivation the reader is referred to Reference 17. Defining  $N$  the  $(q - 1) \times (n - 3)$  matrix of scalars:

$$N = \begin{pmatrix} N_{2,p}(\bar{u}_1) & \dots & N_{n-2,p}(\bar{u}_1) \\ \vdots & \ddots & \vdots \\ N_{2,p}(\bar{u}_{q-1}) & \dots & N_{n-2,p}(\bar{u}_{q-1}) \end{pmatrix} \quad (24)$$

it is possible to rewrite Equation (23) in the matrix form:

$$(N^T N) \mathbf{P} = N^T \boldsymbol{\rho} \quad \text{with} \quad \mathbf{P} = \begin{bmatrix} \mathbf{P}_2 \\ \vdots \\ \mathbf{P}_{n-2} \end{bmatrix} \quad \text{and} \quad \boldsymbol{\rho} = \begin{bmatrix} \boldsymbol{\rho}_1 \\ \vdots \\ \boldsymbol{\rho}_{q-1} \end{bmatrix}. \quad (25)$$

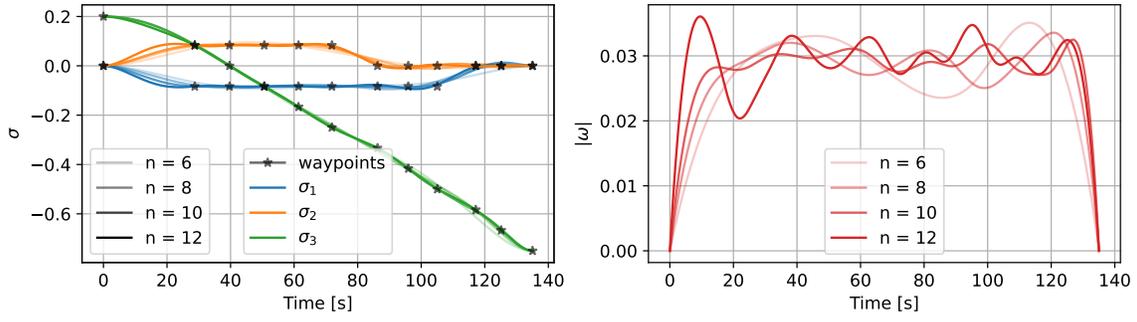
Equation (25) represents three underdetermined systems, because the control points  $\mathbf{P}_i$  and the terms  $\boldsymbol{\rho}_k$  are three-dimensional. The system can be solved choosing the minimum norm solution, which yields the vector  $\mathbf{P}$  for which the NURBS curve gives the Least Squares approximation of the  $q - 1$  intermediate waypoints:

$$\mathbf{P} = (N^T W N)^{-1} W N^T \boldsymbol{\rho} \quad (26)$$

where  $W$  is a square, diagonal, positive semidefinite matrix of size  $(q - 1)$  containing the weights associated with each waypoint. If  $W$  is the identity, all the waypoints are weighed equally. Equation (26) only works if the matrix  $N^T W N$  is full rank. It is important to note that the elements that compose the matrix  $N$  are strongly dependent from the knot vector  $U$  defined in Equation (11). DeBoor showed that when every knot span in  $U$  contains at least one  $\bar{u}_k$ , the matrix  $N^T W N$  is positive definite and well-defined.<sup>20</sup> The following algorithm for choosing the internal knots ensures that this is true:<sup>17</sup>

$$\begin{aligned} d &= \frac{q + 1}{n - p + 1} & i &= \text{int}(jd) & \alpha &= jd - i \\ u_{p+j} &= (1 - \alpha)\bar{u}_{i-1} + \alpha\bar{u}_i & \text{for} & & j &= 1, \dots, n - p. \end{aligned} \quad (27)$$

Figure 3 shows the performance of the LS approximating NURBS for a set of constraint-compliant MRP waypoints, where an angular rate norm of  $\omega^* = 0.03$  rad/s is required. The left plot shows different LS solutions obtained for varying numbers of control points, for each MRP component  $\sigma_j$ . It can be observed that for a smaller number of control points the approximation fits the waypoints less precisely. Conversely, as the number of control points is increased, the approximation fits the waypoints almost perfectly. The right-hand-side plot shows the angular rate norm associated with the LS approximating solutions, also displayed for different numbers of control points. It is observed that, while the angular rate norm is on average close to the desired value, it still oscillates noticeably. This is not surprising, since the required angular rate norm is not accounted for in this LS approximation.



**Figure 3:** LS approximating NURBS, attitude-based, for varying numbers of control points  $n$

### Least Squares waypoint and angular rates approximation

This subsection elaborates on the results of the previous one, enhancing the LS approximating NURBS curve to match the requirement on the angular rate norm. Because Equation (25) is an underdetermined system, it is possible to compute a minimum norm, LS solution that incorporates in its error function not only the error with respect to the waypoints, but also to the desired angular velocities at such waypoints. The requirement for the reference trajectory is to have a constant angular rate norm  $\|\omega\| = \omega^*$ . While this requirement cannot be enforced at the trajectory endpoints, it can be enforced in the intermediate waypoints.

Ideally, one should set up an error function for the MRP derivatives like:

$$g = \sum_{k=1}^{q-1} (\|\sigma'_k\|^2 - \|\sigma'(\bar{u}_k)\|^2)^2 = \sum_{k=1}^{q-1} \left[ \|\sigma'_k\|^2 - \sum_{j=1}^3 \left( \sum_{i=2}^{n-2} N'_{i,p}(\bar{u}_k) P_{i,j} \right)^2 \right]^2. \quad (28)$$

However, solving for the control points  $P_i$  that minimize the error function in Equation (28) is nontrivial, because due to the quadratic dependence of the terms in square brackets from the  $P_i$ , computing the solution would require nonlinear programming. A nonlinear program could be set up and solved. However, the LS fitting NURBS is supposed to run fast, because it is performed at every step of the effort-based A\* algorithm,<sup>13</sup> so, ideally, the solution sought should not feature iterative methods that could significantly affect the computational time. Instead, an estimate of the desired MRP derivatives  $\sigma'_k$  is provided as follows. As a first step, such derivatives are computed using central finite differences:

$$\hat{\sigma}'_k = \frac{\bar{u}_{k+1} - \bar{u}_k}{\bar{u}_{k+1} - \bar{u}_{k-1}} \cdot \frac{\sigma_k - \sigma_{k-1}}{\bar{u}_k - \bar{u}_{k-1}} + \frac{\bar{u}_k - \bar{u}_{k-1}}{\bar{u}_{k+1} - \bar{u}_{k-1}} \cdot \frac{\sigma_{k+1} - \sigma_k}{\bar{u}_{k+1} - \bar{u}_k} \quad \text{for } k = 1, \dots, q-1. \quad (29)$$

From Equation (3) it is possible to derive the following relation between the angular rate norm and the MRP derivative norm:

$$\|\dot{\sigma}\| = \frac{1 + \sigma^2}{4} \|\omega\|. \quad (30)$$

Knowing the desired angular rate norm  $\omega^*$ , it is possible to scale the estimated MRP derivatives obtained via finite differences, to make them correspond to an angular rate vector with the desired magnitude:

$$\sigma'_k = \frac{\hat{\sigma}'_k}{\|\hat{\sigma}'_k\|} \cdot \frac{1 + \|\sigma_k\|^2}{4} \omega^* \cdot t_q \quad \text{for } k = 1, \dots, q-1 \quad (31)$$

where the multiplicative term  $t_q$  appears since the derivatives are expressed with respect to the dimensionless time:  $\sigma' = \frac{d\sigma}{du} = \frac{d\sigma}{dt} \cdot \frac{dt}{du} = \dot{\sigma} \cdot t_q$ .

Having estimated the desired MRP rates, it is possible to proceed setting up a LS problem that is analogous to that introduced in the previous subsection. Let's define:

$$\rho'_k = \sigma'_k - N'_{0,p}(\bar{u}_k)\mathbf{P}_0 - N'_{1,p}(\bar{u}_k)\mathbf{P}_1 - N'_{n-1,p}(\bar{u}_k)\mathbf{P}_{n-1} - N'_{n,p}(\bar{u}_k)\mathbf{P}_n$$

for  $k = 1, \dots, q-1$  (32)

where the four control points are computed with the same procedure outlined in the previous subsection. It is possible to set up an error function that incorporates the squared errors with respect to the MRP waypoints and associated MRP rates:

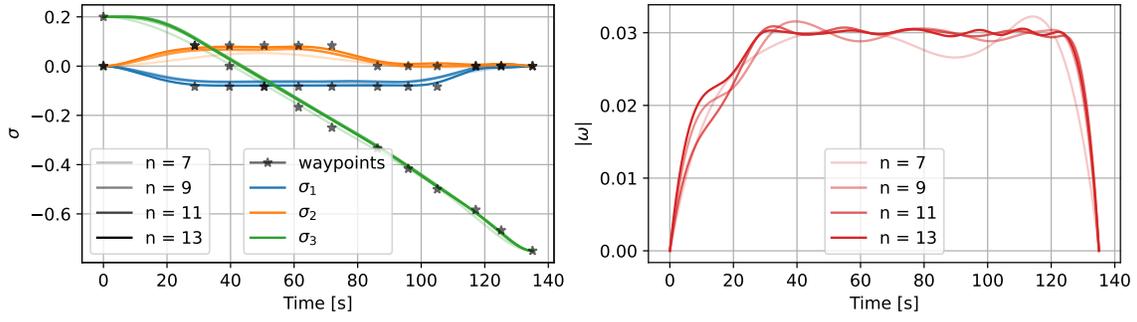
$$\begin{aligned} f &= \sum_{k=1}^{q-1} (\|\sigma_k - \sigma(\bar{u}_k)\|^2 + \|\sigma'_k - \sigma(\bar{u}_k)'\|^2) \\ &= \sum_{k=1}^{q-1} \left( \left\| \rho_k - \sum_{i=2}^{n-2} N_{i,p}(\bar{u}_k)\mathbf{P}_i \right\|^2 + \left\| \rho'_k - \sum_{i=2}^{n-2} N'_{i,p}(\bar{u}_k)\mathbf{P}_i \right\|^2 \right). \end{aligned} \quad (33)$$

Applying the same procedure to minimize  $f$  with respect to  $\mathbf{P}_i$  gives the same minimum norm solution as in Equation (26), only in this case the matrices that appear in the equation take the form:

$$N = \begin{pmatrix} N_{2,p}(\bar{u}_1) & \dots & N_{n-2,p}(\bar{u}_1) \\ \vdots & \ddots & \vdots \\ N_{2,p}(\bar{u}_{q-1}) & \dots & N_{n-2,p}(\bar{u}_{q-1}) \\ N'_{2,p}(\bar{u}_1) & \dots & N'_{n-2,p}(\bar{u}_1) \\ \vdots & \ddots & \vdots \\ N'_{2,p}(\bar{u}_{q-1}) & \dots & N'_{n-2,p}(\bar{u}_{q-1}) \end{pmatrix}, \quad \mathbf{P} = \begin{bmatrix} \mathbf{P}_2 \\ \vdots \\ \mathbf{P}_{n-2} \end{bmatrix}, \quad \boldsymbol{\rho} = \begin{bmatrix} \rho_1 \\ \vdots \\ \rho_{q-1} \\ \rho'_1 \\ \vdots \\ \rho'_{q-1} \end{bmatrix} \quad (34)$$

with  $N$  being a  $(2q-2) \times (n-3)$  matrix and  $\boldsymbol{\rho}$  being a  $(2q-2)$  vector, while  $\mathbf{P}$  remains a  $(n-3)$  vector. The weight matrix  $W$  has size  $(2q-2) \times (2q-2)$  and it can be used to give more weight, within the LS approximation, to the MRP waypoints or the MRP derivatives. It should be noted that, according to this formulation, the angular rate  $\omega$  is a function of both  $\sigma$  and  $\dot{\sigma}$ : this means that if the error on  $\sigma$  is large, the desired  $\omega^*$  will not be achieved even if the error on  $\dot{\sigma}$  is small. For this reason, it is conceptually wrong to attribute a higher weight to the MRP derivative elements than the MRP waypoints, since it is the latter that need to be approximated well enough for the rest of the approximation to hold.

Figure 4 shows the LS solutions, for varying numbers of control points, obtained using the formulation that accounts for waypoints and angular rates, with equal weight. In the left-hand-side plot, the approximation yields curves that approximate the waypoints less tightly, compared to Figure 3, but still recover the distribution of the waypoints faithfully. The lack of accuracy in tracking the waypoints is due to the fact that the minimum norm solution is trying, in this case, to optimize not only the waypoints, but also the angular rates. More interestingly, it can be observed from the right-hand-side plot that the angular rate norm, in this case, is much closer to the desired target  $\omega^* = 0.03$  rad/s, and the approximation is better for a higher number of control points.



**Figure 4:** LS approximating NURBS, attitude- and angular rate-based, for varying numbers of control points  $n$

## PERFORMANCE STUDY OF NURBS INTERPOLATION VS LS APPROXIMATION

Once the mathematical formulation for the LS approximating NURBS has been laid out, the next step is to compare its performance to the interpolating NURBS. The motivation for using the LS approximating NURBS is to relax the dependence of the trajectory from the waypoints, which serve as a guidance for the trajectory but are, ultimately, an ‘artificial’ constraint. Removing the constraint of precisely hitting the waypoints would allow the trajectory to have a smoother profile, which would ultimately lead to a smaller control effort. The interpolating NURBS, on the other hand, caused the trajectory to wiggle between the waypoints, which caused some ‘parasitic’ torques to appear and ultimately increase the required control effort. This section compares the performances of the two NURBS curves in three different scenarios with different sets of constraints. Moreover, the comparison is performed, for each scenario, for a varying level of grid density  $N$ . The trajectories are based, for each scenario, on the sequence of waypoints obtained applying a basic implementation of the A\* algorithm, using the metric presented in Equation (15). For the LS approximating NURBS,  $n + 1$  control points are used, where  $n = q + 2$ , that is, one control point per waypoint plus two control points to match the endpoint derivatives. This is the same number of control points required by the interpolating NURBS. The computed trajectories are, therefore, not effort-optimal, but this is not relevant for the comparisons that this section aims to make. Lastly, the computational cost of the two NURBS are also compared.

The following inertia tensor is used for the spacecraft, which is modeled after a 3-unit cubesat:

$${}^B[\mathbf{I}] = \begin{bmatrix} 6.67 & 0 & 0 \\ 0 & 41.87 & 0 \\ 0 & 0 & 41.87 \end{bmatrix} \cdot 10^{-3} \text{ kg} \cdot \text{m}^2. \quad (35)$$

The control effort is estimated, like in,<sup>13</sup> as the integral over the trajectory of the norm of the instantaneous control torque:

$$U = \int_0^T \|\mathbf{L}\| dt \quad \text{with} \quad \mathbf{L} = {}^B[\mathbf{I}]{}^B\dot{\boldsymbol{\omega}}_{\mathcal{B}\mathcal{N}} + {}^B\boldsymbol{\omega}_{\mathcal{B}\mathcal{N}} \times {}^B[\mathbf{I}]{}^B\boldsymbol{\omega}_{\mathcal{B}\mathcal{N}}. \quad (36)$$

All the scenarios presented in this section feature rest-to-rest maneuvers with an angular rate norm  $\omega^* = 0.03$  rad/s.

## Scenario 1: eigenaxis rotation

The first scenario is very simple, as it features an eigenaxis rotation about the  ${}^B\hat{\mathbf{b}}_3$  axis. Only one sensitive instrument with a field of view of 20 deg is oriented along the  ${}^B\hat{\mathbf{b}}_1$  axis, and one keep-out zone is present along the inertial direction  ${}^N\hat{\mathbf{s}} = [-1, 0, 0]$ . The initial and final attitude are, respectively,  $\sigma_0 = [0, 0, 0.1]$  and  $\sigma_q = [0, 0, -0.75]$ . Figure 5 shows the trajectory of the sensitive boresight in inertial space, projected onto a 2D plane. All the trajectories obtained for different grid densities  $N$  overlap along the same projection on the 2D plot. More interesting are the results shown in Figure 6, which also shows that the trajectories match. This happens because of the simple nature of the eigenaxis rotation, for which all the guidance waypoints lie along the  $\sigma_3$  axis. For all values of  $N$  the angular rate norm is approximated very well, as it stays very close to the target value. Figure 7 shows the comparison between the two types of NURBS: as far as control effort, the LS approximating curve performs better across all grid densities, yielding in some cases a significant reduction. The computational cost of the LS approximating curve is, on the other hand, higher. Despite that, it scales comparably to the interpolating curve as the grid density (and therefore the number of waypoints  $q$ ) increases.

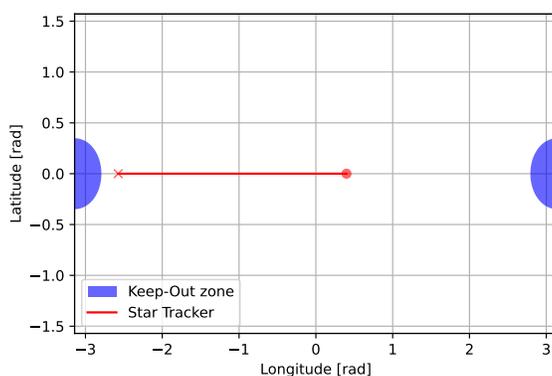


Figure 5: Boresight plot, scenario 1

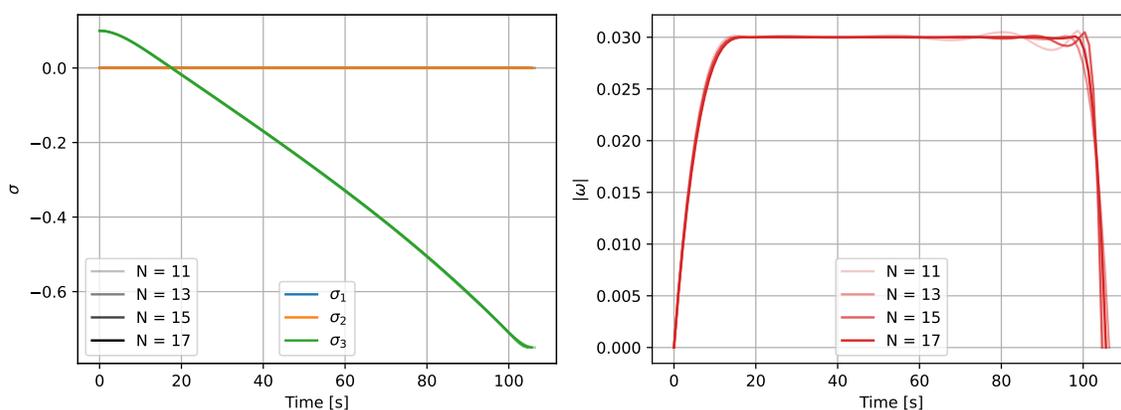
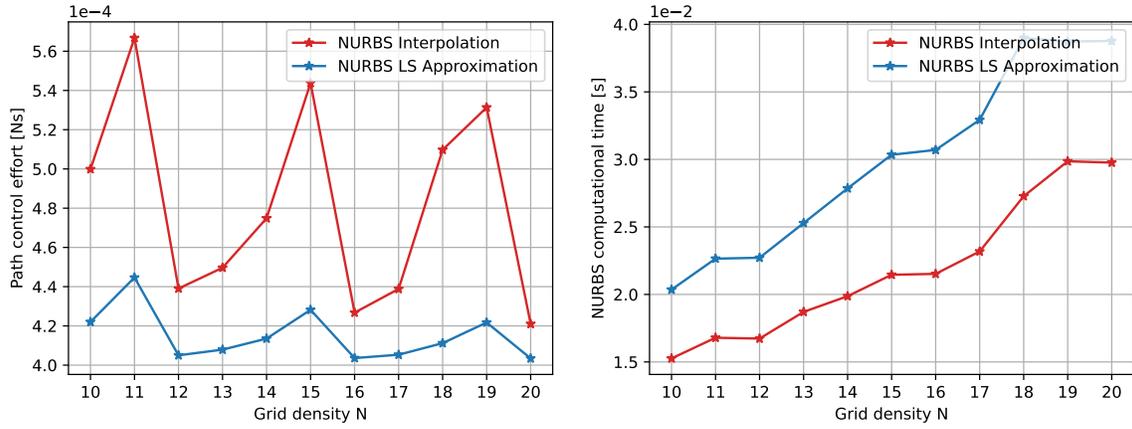


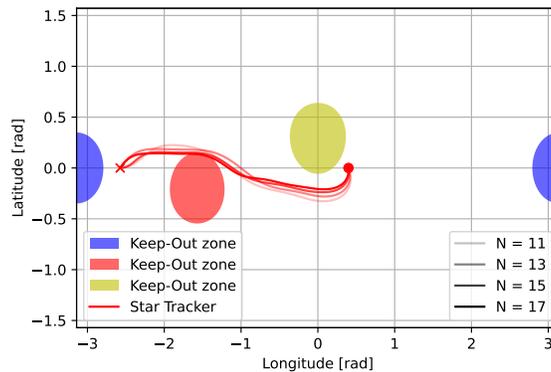
Figure 6: Attitude  $\sigma$  and angular rate  $\|\omega\|$  for varying grid densities  $N$ ; scenario 1



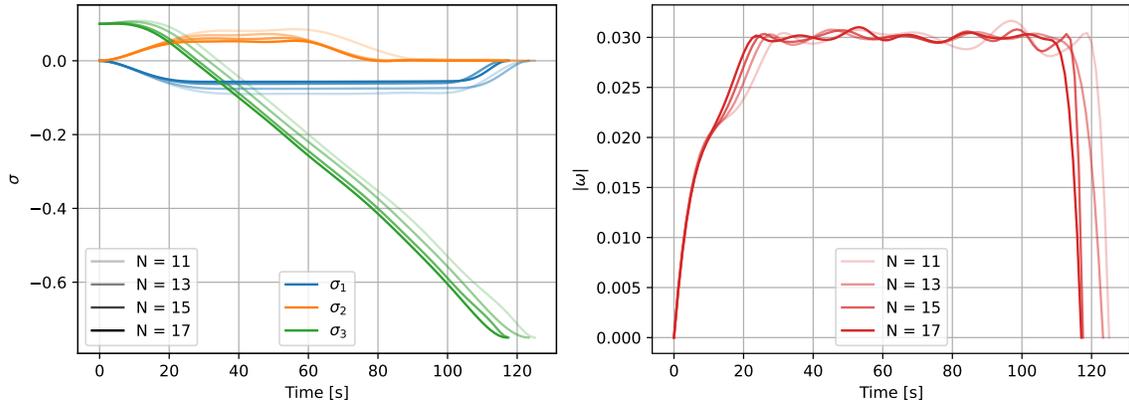
**Figure 7:** Control effort and computational time of interpolating NURBS vs LS approximating NURBS, for varying grid densities  $N$ ; scenario 1

### Scenario 2: multiple keep-out zones

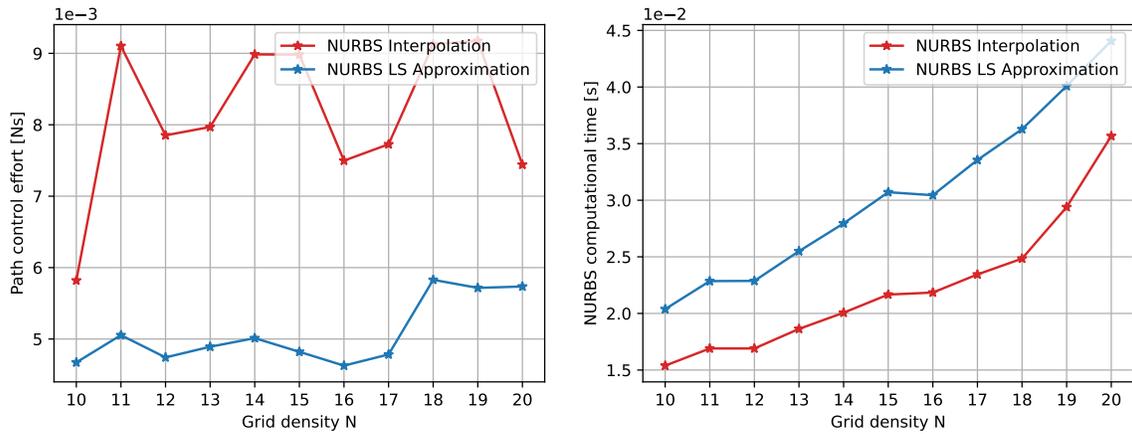
The second scenario is based off of the previous one, with the addition of two more keep-out zones along the inertial directions  ${}^{\mathcal{N}}\hat{s}_2 = [0, -0.981, -0.196]$  and  ${}^{\mathcal{N}}\hat{s}_3 = [0.958, 0, 0.287]$ . Figure 8 shows the trajectories of the sensitive instrument in inertial space, as it steers away from the keep-out zones. In this case, different trajectories are obtained for different grid densities  $N$ : this happens because, for different  $N$ , the attitude space is sampled with different nodes, and therefore the baseline path computed by A\* consists of different waypoints for each case. Regardless, the different trajectories remain fairly close to one another, although for high grid densities some brief constraint violations occur. Figure 9 shows that, even in this case, the angular rate is approximated quite well, although the plateau around  $\omega^* = 0.03$  rad/s is not as flat as for the previous scenario. This happens because, for scenario 1, all the baseline waypoints are aligned along a straight line, for which the approximations made in Equations (19) and (31) are much more accurate. It can also be observed that, for higher  $N$ , the angular rate norm is less oscillatory, thanks to a higher number of waypoints that provide more guidance for the output trajectory. Lastly, Figure 10 compares the performances of the two NURBS curves. In this scenario, the interpolating NURBS is associated



**Figure 8:** Boresights plot, scenario 2



**Figure 9:** Attitude  $\sigma$  and angular rate  $\|\omega\|$  for varying grid densities  $N$ ; scenario 2

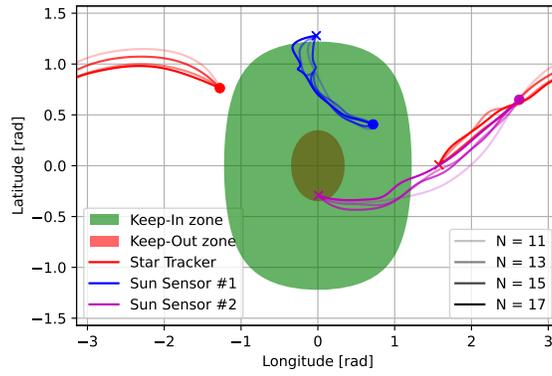


**Figure 10:** Control effort and computational time of interpolating NURBS vs LS approximating NURBS, for varying grid densities  $N$ ; scenario 2

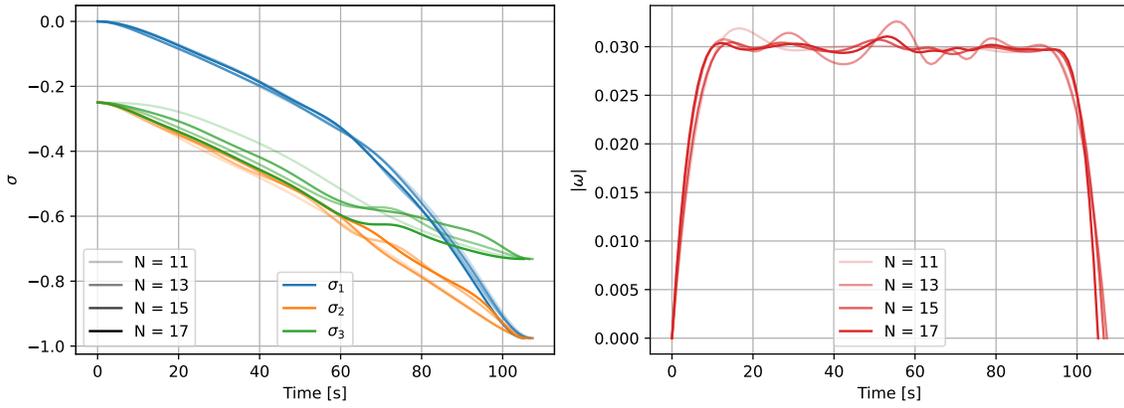
with a control effort that is almost twice as high as the LS approximating NURBS. Moreover, the control effort for the first one has increased significantly with respect to scenario 1, whereas for the second, the control effort remains closer to the values computed for the first scenario. For the computational cost, the same considerations can be made: the LS approximating NURBS is consistently more costly, but comparable to the interpolating NURBS, and they display a similar trend with increasing levels of grid density.

### Scenario 3: mixed keep-in and keep-out zones

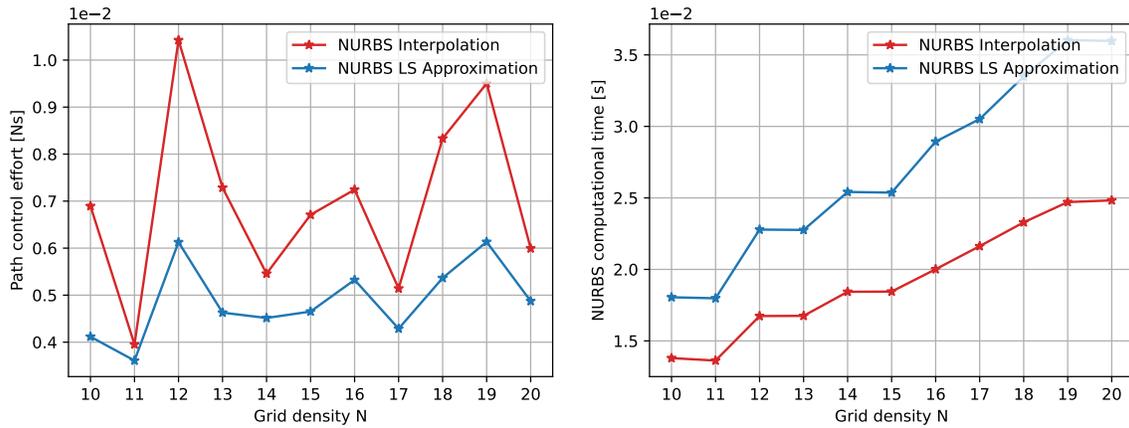
The third scenario is the most articulated, as it features multiple instruments. There is still a sensitive camera aligned along the  ${}^B\hat{b}_1$  axis, and two sun sensors, along the  $\hat{b}_2$  and  $\hat{b}_3$  axes respectively, each with a field of view of 70 deg. The only celestial object is the Sun, located along the  ${}^N\hat{s}_3 = [1, 0, 0]$  inertial direction. The constraints for this problem consist of maneuvering such that the Sun remains out of the field of view of the camera, but within the field of view of at least one of the sun sensors. The initial and final attitude are, respectively,  $\sigma_0 = [0, -0.25, -0.25]$  and  $\sigma_q = [0.4, 0.4, 0.3]$ . Figure 11 shows the trajectories of the three boresights in inertial space: all



**Figure 11:** Boresights plot, scenario 3



**Figure 12:** Attitude  $\sigma$  and angular rate  $\|\omega\|$  for varying grid densities  $N$ ; scenario 3



**Figure 13:** Control effort and computational time of interpolating NURBS vs LS approximating NURBS, for varying grid densities  $N$ ; scenario 3

three are compliant, in all their parts, for all grid densities  $N$ . The sun sensors switch, since at the initial attitude only sun sensor 1 sees the Sun, while sun sensor 2 sees the Sun once the final pose is

reached. Similar considerations apply for Figure 12 as to the previous scenario: the target angular rate norm is approximated fairly well, and the more so when the grid density is higher. Relative to Figure 13, the control effort displays an irregular behavior for both NURBS as  $N$  varies, but again the LS approximation outperforms the interpolation for each grid density. Regarding computational time, the same considerations apply as in the previous two scenarios.

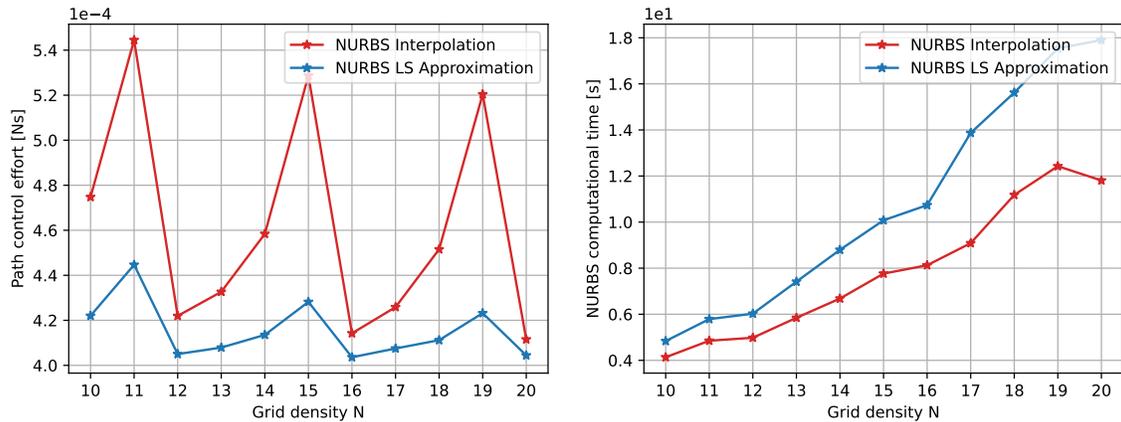
## PERFORMANCE STUDY OF EFFORT-BASED GRAPH SEARCH ALGORITHM

This last section shows the results computed using the effort-based A\* graph-search algorithm, outlined in Reference 13. As this version of the A\* algorithm searches the graph, it uses intermediate paths consisting of the previously explored nodes  $(\sigma_0, \dots, \sigma_{n-1})$ , the current open node  $(\sigma_n)$  and the goal node  $(\sigma_q)$  as a baseline for an intermediate trajectory computed via NURBS curves. The control effort integral in Equation (36), evaluated along these intermediate trajectories, is used as the the priority function  $p(n)$  to explore the graph, with the objective of finding the sequence of waypoints that yields the optimal trajectory in terms of required control effort. Such priority function is the sum of two terms, the cost to current node  $g(n)$  and the heuristic  $h(n)$ . The cost to current node is associated to the control effort required to track the trajectory from the starting node  $\sigma_0$  to the current open node  $\sigma_n$ ; the heuristic is an estimate of the cost required to track the trajectory from the current open node to the goal node  $\sigma_q$ . This heuristic is computed assuming that the goal node can be reached from the open node  $n$  through a straight path in MRP space, without obstacles. This is to ensure that the heuristic is optimistic,<sup>10</sup> i.e., the cost of the final path to goal is higher or equal to the priority function  $p(n)$ . This gives:

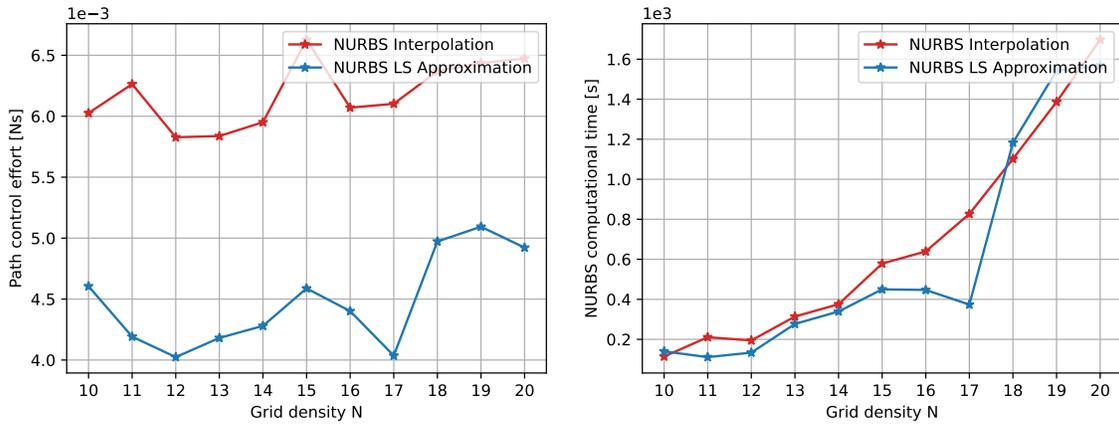
$$p(n) = g(n) + h(n) = \int_0^{t_n} \|\mathbf{L}\| dt + \int_{t_n}^{t_q} \|\mathbf{L}\| dt = \int_0^{t_q} \|\mathbf{L}\| dt. \quad (37)$$

The effort-based A\* is run for the three scenarios presented in the previous sections, for both the interpolating NURBS and the LS approximating NURBS.

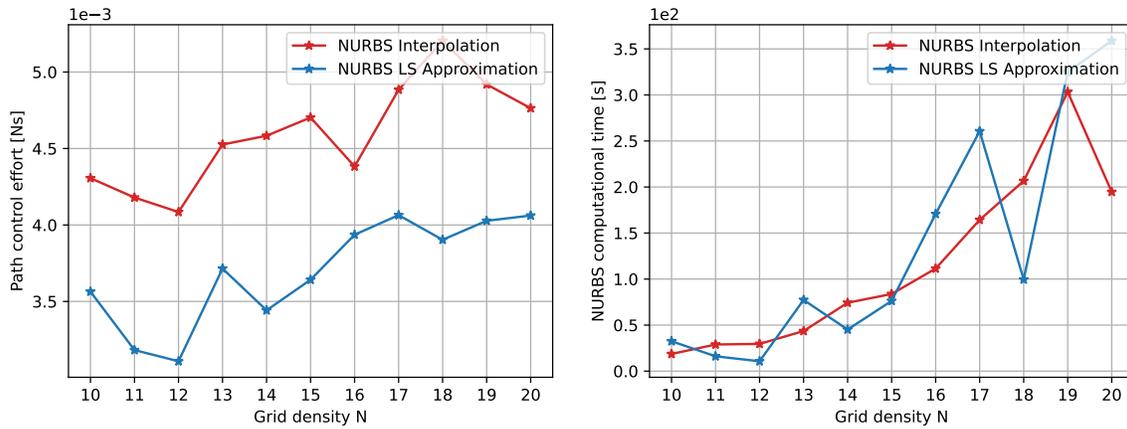
Figure 14 shows that the control effort of the effort-optimal solutions, for both types of NURBS, match the results shown in Figure 7. This is expected, because for the eigenaxis rotation in scenario 1, the control strategy defaults to a bang-bang type with an effortless coasting arc in the middle.



**Figure 14:** Control effort and computational time of effort-based A\*, NURBS comparison; scenario 1



**Figure 15:** Control effort and computational time of effort-based A\*, NURBS comparison; scenario 2



**Figure 16:** Control effort and computational time of effort-based A\*, NURBS comparison; scenario 3

Any other trajectory that deviates from that would likely be more costly in terms of control effort, due to the insurgence of gyroscopic terms and torque components along the other two axes.

Figures 15 and 16 show that the effort-based solutions, for both NURBS curves, provide better trajectory in terms of control effort than those in the previous section, as expected. While the results improve for both types of NURBS, the LS approximating curve consistently outperforms the interpolating curve, always yielding less costly solutions.

The effort-based A\* algorithm, whose performance is described in Figures 14 to 16, is implemented in Python and run on a Macbook Pro with an M1 Pro chip. These plots show how computationally demanding the effort-based algorithm is, where for high grid densities it can take up to 15-20 minutes to compute the optimal solution. However, an implementation of the same algorithm in C++ could significantly reduce the computational time from the order of minutes to the order of seconds. Interestingly, Figures 15 and 16 show that the computational times for the two different NURBS are comparable. This is not intuitive, if looking at the results shown in Figures 6, 9 and 12

where, when the path is pre-computed by a metric-based version of  $A^*$ , the LS approximating NURBS seems to consistently require a larger computational time than the interpolating NURBS. In the effort-based implementation, however, the priority function  $p(n)$  computed from the LS approximating NURBS likely produces a more "optimistic" estimate of the total path cost. This allows the effort-based algorithm combined with the LS approach to converge to the final solution more efficiently,<sup>10</sup> i.e., exploring a smaller number of nodes.

This benchmark analysis also showed, as it was expected, that increasing the grid density  $N$  yields longer computational times for the effort-based  $A^*$ . However, in the range  $10 \leq N \leq 15$ , such increase is still quite acceptable. The grid density  $N$  is, in this problem, a tradeoff parameter: increasing  $N$  gives a better representation of the attitude space and less of a chance of violating the constraints. On the other hand, the computational time increases significantly and the computer trajectories, even the effort-optimal ones, are more costly in terms of control torque.

## CONCLUSIONS

This paper proposes a new solution to the constrained attitude maneuvering problem using a newly designed type of NURBS curve. The new curve matches the endpoint constraints of initial and final attitude and angular rates, while it uses MRP guidance waypoints as a baseline. The difference between the trajectory and the waypoints, and the velocity along the trajectory and the desired angular rates, are minimized via a least-squares minimization technique. This paper shows how the Least Squares approximating NURBS consistently outperforms the interpolating NURBS in terms of required control torque, for all sequences of constraint-compliant waypoints used as baselines. The LS approximating NURBS, in a single run, requires a slightly longer computational time, due to the larger number of calculations it needs to perform. However, within the effort-optimal  $A^*$  graph-search algorithm, which runs multiple NURBS calculations in sequence, the two NURBS approaches are comparable in terms of computational time. This paper also shows how the grid density  $N$  affects the computational time of the algorithm and the ultimate path cost associated with the computed trajectories, allowing to identify the range  $N \in [10, 15]$  as a good tradeoff between attitude sampling accuracy and computational tractability.

## REFERENCES

- [1] C. R. McInnes, "Large angle slew maneuvers with autonomous sun vector avoidance," *Journal of guidance, control, and dynamics*, Vol. 17, No. 4, 1994, pp. 875–877, 10.2514/3.21283.
- [2] M. Diaz Ramos and H. Schaub, "Kinematic steering law for conically constrained torque-limited spacecraft attitude control," *Journal of Guidance, Control, and Dynamics*, Vol. 41, No. 9, 2018, pp. 1990–2001, 10.2514/1.g002873.
- [3] U. Lee and M. Mesbahi, "Spacecraft reorientation in presence of attitude constraints via logarithmic barrier potentials," *Proceedings of the 2011 American Control Conference*, IEEE, 2011, pp. 450–455, 10.1109/acc.2011.5991284.
- [4] U. Lee and M. Mesbahi, "Feedback control for spacecraft reorientation under attitude constraints via convex potentials," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 50, No. 4, 2014, pp. 2578–2592, 10.1109/taes.2014.120240.
- [5] H. C. Kjellberg and E. G. Lightsey, "Discretized constrained attitude pathfinding and control for satellites," *Journal of Guidance, Control, and Dynamics*, Vol. 36, No. 5, 2013, pp. 1301–1309, 10.2514/1.60189.
- [6] H. C. Kjellberg and E. G. Lightsey, "Discretized quaternion constrained attitude pathfinding," *Journal of Guidance, Control, and Dynamics*, Vol. 39, No. 3, 2015, pp. 713–718, 10.2514/1.g001063.
- [7] S. Tanygin, "Fast autonomous three-axis constrained attitude pathfinding and visualization for bore-sight alignment," *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 2, 2017, pp. 358–370, 10.2514/1.g001801.

- [8] R. Tarjan, "Depth-first search and linear graph algorithms," *SIAM journal on computing*, Vol. 1, No. 2, 1972, pp. 146–160, 10.1137/0201010.
- [9] C. Y. Lee, "An algorithm for path connections and its applications," *IRE transactions on electronic computers*, Vol. EC-10, No. 3, 1961, pp. 346–365, 10.1109/tec.1961.5219222.
- [10] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE transactions on Systems Science and Cybernetics*, Vol. 4, No. 2, 1968, pp. 100–107, 10.1109/tssc.1968.300136.
- [11] E. Frazzoli, M. Dahleh, E. Feron, and R. Kornfeld, *A randomized attitude slew planning algorithm for autonomous spacecraft*, Vol. 4155. AIAA Reston, VA, 2001, 10.2514/6.2001-4155.
- [12] X. Tan, S. Berkane, and D. V. Dimarogonas, "Constrained attitude maneuvers on SO(3): Rotation space sampling, planning and low-level control," *Automatica*, Vol. 112, 2020, p. 108659, 10.1016/j.automatica.2019.108659.
- [13] R. Calaon and H. Schaub, "Constrained Attitude Maneuvering via Modified-Rodrigues-Parameter-Based Motion Planning Algorithms," *Journal of Spacecraft and Rockets*, 2022, pp. 1–15, 10.2514/1.A35294.
- [14] R. Calaon, M. Trowbridge, and H. Schaub, "A Basilisk-Based Benchmark Analysis Of Different Constrained Attitude Dynamics Planners," *AIAA SciTech Forum*, San Diego, CA, Jan. 3–7 2022, 10.2514/6.2022-2277.
- [15] H. Schaub and J. L. Junkins, *Analytical Mechanics of Space Systems*. AIAA, 4 ed., 2018, 10.2514/4.105210.
- [16] H. Schaub and S. Piggott, "Speed-constrained three-axes attitude control using kinematic steering," *Acta Astronautica*, Vol. 147, 2018, pp. 1–8, 10.1016/j.actaastro.2018.03.022.
- [17] L. Piegl and W. Tiller, *The NURBS book*. Springer Science & Business Media, 1996, 10.1007/978-3-642-59223-2.
- [18] B. Fornberg and J. Zuev, "The Runge phenomenon and spatially variable shape parameters in RBF interpolation," *Computers & Mathematics with Applications*, Vol. 54, No. 3, 2007, pp. 379–398, 10.1016/j.camwa.2007.01.028.
- [19] C. De Boor, "On calculating with B-splines," *Journal of Approximation theory*, Vol. 6, No. 1, 1972, pp. 50–62, 10.1016/0021-9045(72)90080-9.
- [20] C. De Boor and C. De Boor, *A practical guide to splines*, Vol. 27. Springer-Verlag New York, 1978.