JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 46, No. 11, November 2023

Engineering Notes

Constrained Attitude Path Planning via Least-Squares Modified-Rodrigues-Parameters-Based Smoothing Curves

Riccardo Calaon* and Hanspeter Schaub†
University of Colorado Boulder, Boulder, Colorado 80303

https://doi.org/10.2514/1.G007324

I. 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, which causes the output control law to steer the spacecraft toward 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 [1–4]. Such approaches are straightforward to implement and computationally cheap, 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 [5–7]. In these cases, graph-searching algorithms such as depth-first [8], breadth-first [9], or A* [10] are applied to navigate such graphs. In different approaches, the attitude space sampling is not deterministic, but rather stochastic: e.g., Probabilistic RoadMaps [11] sacrifice the completeness of the attitude space mapping in exchange for faster execution times. These latter approaches do not suffer from 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. [12] and Calaon and Schaub [13] provide solutions based on attitude sampling, combined with some form of interpolation to obtain a smooth reference trajectory from a sequence of attitude waypoints. Other approaches, called metaheuristic, start from a baseline solution, which is further improved upon exploring the neighboring solution space. Differential evolution algorithms or particle swarm optimization are examples of such approaches. Their strength consists in the ability to refine the solution based on an optimality condition, while at the same time enforcing the constraints that such solution must not violate. Examples are found in Refs. [14–18].

This work is based on the modified Rodrigues parameters (MRPs) nonsingular attitude discretization presented in Ref. [13]. The focus of this newer contribution is on the second part of the problem, which is how to obtain a smooth reference trajectory from a sequence of attitude waypoints. Calaon and Schaub [13] use nonuniform rational basis spline (NURBS) 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 Ref. [13], and highlighted by simulations in Ref. [19]. This Note proposes a different type of NURBS curve that performs a least-squares (LS) fit of the waypoints and the desired angular rates, 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.

In this Note, the new LS approximating NURBS is derived mathematically. The two NURBS curves, the interpolating curve and the LS approximating curve, are then compared to one another in three different scenarios, to assess their performance with baseline A* solutions. Finally, the effort-based A* incorporating the two NURBS curves is applied to the same three scenarios, in order to benchmark the performance of the two versus different grid density levels.

II. Path Smoothing: LS Approximating NURBS

In path-planning applications a sequence of constraint-compliant waypoints is often not enough to provide an accurate reference, because it gives no insight on the required position, velocity, and acceleration as a function of time. For this reason, path-planning algorithms are often combined with some kind of smoothing techniques [20]. Moreover, Ref. [19] 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

Presented as Paper 22-541 at the AAS/AIAA Astrodynamics Specialist Conference, Charlotte, NC, August 7–11, 2022; received 25 October 2022; revision received 9 April 2023; accepted for publication 29 July 2023; published online 8 September 2023. Copyright © 2023 by Riccardo Calaon and Hanspeter Schaub. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission. All requests for copying and permission to reprint should be submitted to CCC at www.copyright.com; employ the eISSN 1533-3884 to initiate your request. See also AIAA Rights and Permissions www.aiaa.org/randp.

*Ph.D. Candidate, Ann and H.J. Smead Department of Aerospace Engineering Sciences, 3775 Discovery Drive. Student Member AIAA.

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

attitude becomes significant. On the other hand, Calaon et al. [19] highlighted how the reference trajectory obtained from NURBS interpolation of the waypoints can require a global control effort 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 Ref. [13].

This work is built on the same premises discussed in Secs. II and III of Ref. [13]: the unit MRP sphere is sampled with uniformly spaced points, each corresponding to an attitude with respect to the inertial frame. Constraint-incompliant attitude waypoints are removed from the grid, whereas the constraint-compliant attitude waypoints are connected to one another based on proximity considerations in order to build an undirected graph. Attention is paid to the fact that there is a discontinuity between points at opposite ends of the unit sphere, which describe the same attitude, but have a different MRP representation. Restricting the domain to the unit sphere introduces a discontinuity in the formulation, but removes the singularities and the problem of attitude unwinding [21]. Sampling the attitude space with a finer grid allows for a better representation of the obstacles, but it also increases the dimension of the graph that is to be searched. The assumption in this work is that the maneuver happens fast enough that the obstacles can be considered fixed. For a more detailed analysis of the attitude sampling and obstacle representation, the reader is directed to Ref. [13].

This section develops a new type of 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, Ref. [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 [22]. 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 Ref. [13], removes a step in the effort-based A* algorithm, ultimately improving its computational speed.

A. Properties of NURBS Curves

A NURBS 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 Runge's phenomenon, which arises when trying to fit such large sets of data with a single, high-order polynomial [23,24]. The degree of a NURBS curve 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 $\mathcal{C}^{p-1}$ in its entire domain; i.e., it is continuous and differentiable $p-1$ times. For the present application, the desire is to obtain a NURBS curve that is at least $\mathcal{C}^3$, 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

$$\boldsymbol{\sigma}(u) = \sum_{i=0}^{n} N_{i,p}(u)\boldsymbol{P}_i \tag{1}$$

where $u \in [0,1]$ is the dimensionless time parameter of the curve, $\boldsymbol{P}_i$ 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:

$$U = \{\underbrace{0,\ldots,0}_{p+1}, u_{p+1}, \ldots, u_{m-p-1}, \underbrace{1,\ldots,1}_{p+1}\} \tag{2}$$

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 $\boldsymbol{P}_0$ and $\boldsymbol{P}_n$. 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 [25]:

$$N_{i,0}(u) = \begin{cases} 1 & \text{if } u_i \le u < u_{i+1} \\ 0 & \text{otherwise} \end{cases}$$
$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u) \tag{3}$$

The derivative of a NURBS curve is obtained as

$$\boldsymbol{\sigma}'(u) = \sum_{i=0}^{n} N'_{i,p}(u)\boldsymbol{P}_i \tag{4}$$

where $\boldsymbol{\sigma}'(u) = (d\boldsymbol{\sigma}/du)$ 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 [23]

$$N'_{i,p}(u) = \frac{p}{u_{i+p} - u_i} N_{i,p-1}(u) - \frac{p}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u) \tag{5}$$

B. Improved Time Spacing

To build a NURBS curve, whether it is an interpolating curve or an LS approximating curve, it is necessary to define the time tags $\bar{u}_k$ for $k = 0, \ldots, q$, 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 application, Piegl and Tiller [23] suggest to compute the $\bar{u}_k$'s proportionally to a valid distance metric between two consecutive waypoints, defined in the space where such waypoints exist. In this Note, such metric is defined as

$$\begin{aligned} \hat{\theta}(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) &= \mathtt{min}\{\theta(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2), \theta(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2^S)\} \\ \theta(\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2) &= 4\arctan\left[\frac{(1-|\boldsymbol{\sigma}_2|^2)\boldsymbol{\sigma}_1 - (1-|\boldsymbol{\sigma}_1|^2)\boldsymbol{\sigma}_2 + 2\boldsymbol{\sigma}_1 \times \boldsymbol{\sigma}_2}{1 + |\boldsymbol{\sigma}_1|^2|\boldsymbol{\sigma}_2|^2 + 2\boldsymbol{\sigma}_1 \cdot \boldsymbol{\sigma}_2}\right] \end{aligned} \tag{6}$$

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 [26]. This distance metric is defined in order to account for the nonlinear relation between principal rotation angle and distance between attitude points in MRP space. Alternatively, this nonlinearity could be mitigated using higher-order, more-linear Rodrigues parameters [27,28]. Such representation, however, would present multiple singularities and introduce additional layers of complexities to the problem.

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

$$S(t) = \int_0^t \|\boldsymbol{\omega}\| \, \mathrm{d}\tau \tag{7}$$

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

$$\begin{cases} S_{k+1} = S_k + \dot{S}_k(t_{k+1} - t_k) + \dfrac{1}{2}\ddot{S}_k(t_{k+1} - t_k)^2 \\ \dot{S}_{k+1} = \dot{S}_k + \ddot{S}_k(t_{k+1} - t_k) \end{cases} \tag{8}$$

where $S_k = S(t_k)$. Eliminating $\ddot{S}_k$ from Eq. (8) yields the expression

$$S_{k+1} - S_k = \frac{\dot{S}_{k+1} + \dot{S}_k}{2}(t_{k+1} - t_k) \tag{9}$$

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:

$$t_{k+1} = t_k + k\frac{\hat{\theta}(\boldsymbol{\sigma}_k, \boldsymbol{\sigma}_{k+1})}{\omega_k + \omega_{k+1}} \tag{10}$$

with $k = 2$. The constant $k = 2$ is a direct consequence of assuming that $S(t)$ is a second-order polynomial, as Eq. (8) implicitly states; i.e., the angular rate norm $\|\boldsymbol{\omega}\|$ is a piecewise-linear function. This approximation holds quite well for the intervals between internal waypoints, where the angular rate norm nominally does not vary. However, when the NURBS curve is required to quickly ramp up and down from/to zero at the endpoints, a better result is obtained assuming a polynomial of higher-order p for $S(t)$ in the first and last interval. With zero endpoint angular rates, this gives $k = p/(p-1)$. Using a fourth-order NURBS for $\boldsymbol{\sigma}(t)$ suggests the choice for fourth-order polynomial for the endpoint time intervals, which gives $k = 4/3$. With zero endpoint rates, and constant central angular rate norm $\|\boldsymbol{\omega}\| = \omega^*$, the improved algorithm to calculate the time spacing between the waypoints is

$$\begin{aligned} t_0 &= 0 \\ t_1 &= t_0 + \frac{4}{3}\frac{\hat{\theta}(\boldsymbol{\sigma}_0, \boldsymbol{\sigma}_1)}{\omega^*} \\ t_{k+1} &= t_k + \frac{\hat{\theta}(\boldsymbol{\sigma}_k, \boldsymbol{\sigma}_{k+1})}{\omega^*} \quad \text{for } k = 2, \ldots, q-2 \\ t_q &= t_{q-1} + \frac{4}{3}\frac{\hat{\theta}(\boldsymbol{\sigma}_{q-1}, \boldsymbol{\sigma}_q)}{\omega^*} \end{aligned} \tag{11}$$

The dimensionless time tags $\bar{u}_k$'s can be obtaining normalizing the t_k's by t_q, i.e., the time tag corresponding to the final waypoint:

$$\bar{u}_k = \frac{t_k}{t_q} \tag{12}$$

The linear relation between dimensional time t and dimensionless time u allows to define a mapping between the derivatives of $\boldsymbol{\sigma}$ with respect to one or the other:

$$\frac{d\boldsymbol{\sigma}}{du} = \frac{dt}{du}\cdot\frac{d\boldsymbol{\sigma}}{dt} = t_q \qquad \frac{d\boldsymbol{\sigma}}{dt}\frac{d^2\boldsymbol{\sigma}}{du^2} = \frac{d^2t}{du^2}\cdot\frac{d\boldsymbol{\sigma}}{dt} + \left(\frac{dt}{du}\right)^2\cdot\frac{d^2\boldsymbol{\sigma}}{dt^2} = t_q^2\frac{d^2\boldsymbol{\sigma}}{dt^2} \tag{13}$$

C. LS Waypoint Approximation

This subsection outlines the procedure to obtain an LS approximating NURBS curve from a set of $q+1$ attitude waypoints. As previously mentioned, the idea that motivated the use of an 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 two known pieces of information that the reference trajectory must match exactly. 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 $\boldsymbol{\omega}_0$ and $\boldsymbol{\omega}_q$ according to the MRP kinematic equation:

$$\dot{\boldsymbol{\sigma}} = \frac{1}{4}[B(\boldsymbol{\sigma})]\boldsymbol{\omega} \tag{14}$$

As opposed to the interpolating NURBS, in the LS approximation the number of control points $\boldsymbol{P}_i$ for $i = 0, \ldots, n$ can arbitrarily be chosen by the user.

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

$$\begin{bmatrix} N_{0,p}(0) & 0 & 0 & 0 \\ N'_{0,p}(0) & N'_{1,p}(0) & 0 & 0 \\ 0 & 0 & N'_{n-1,p}(1) & N'_{n,p}(1) \\ 0 & 0 & 0 & N_{n,p}(1) \end{bmatrix} \begin{Bmatrix} \boldsymbol{P}_0 \\ \boldsymbol{P}_1 \\ \boldsymbol{P}_{n-1} \\ \boldsymbol{P}_n \end{Bmatrix} = \begin{Bmatrix} \boldsymbol{\sigma}_0 \\ \boldsymbol{\sigma}'_0 \\ \boldsymbol{\sigma}'_q \\ \boldsymbol{\sigma}_q \end{Bmatrix} \tag{15}$$

Note that it is possible to define also the endpoint accelerations of the NURBS curve, by adding two additional control points. In such case, Eq. (15) becomes a 6 × 6 linear system. This approach often proves not beneficial, because in order to match the multiple endpoint constraints, the resulting NURBS often presents significant oscillations. The remaining $\boldsymbol{P}_i$ for $i = 2, \ldots, n-2$ are determined using an LS approach. Before getting into the actual LS fit, the following quantities are defined:

$$\boldsymbol{\rho}_k = \boldsymbol{\sigma}_k - N_{0,p}(\bar{u}_k)\boldsymbol{P}_0 - N_{1,p}(\bar{u}_k)\boldsymbol{P}_1 - N_{n-1,p}(\bar{u}_k)\boldsymbol{P}_{n-1} - N_{n,p}(\bar{u}_k)\boldsymbol{P}_n \quad \text{for } k = 1, \ldots, q-1 \tag{16}$$

The $\boldsymbol{\rho}_k$ terms are used to remove the dependence of the waypoints from the four control points that have already been precomputed. 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 Eq. (1), it is possible to set up an error function that is the sum of the quadratic errors and only depends on the remaining $\boldsymbol{P}_i$ for $i = 2, \ldots, n-2$:

$$f = \sum_{k=1}^{q-1}\|\boldsymbol{\sigma}_k - \boldsymbol{\sigma}(\bar{u}_k)\|^2 = \sum_{k=1}^{q-1}\left\|\boldsymbol{\rho}_k - \sum_{i=2}^{n-2}N_{i,p}(\bar{u}_k)\boldsymbol{P}_i\right\|^2 \tag{17}$$

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

$$\sum_{i=2}^{n-2}\left(\sum_{k=1}^{q-1}N_{l,p}(\bar{u}_k)N_{i,p}(\bar{u}_k)\right)\boldsymbol{P}_i = \sum_{k=1}^{q-1}N_{l,p}(\bar{u}_k)\boldsymbol{\rho}_k \quad \text{for } l = 2, \ldots, n-2 \tag{18}$$

For more details on this derivation the reader is referred to Ref. [23]. Defining $[\boldsymbol{N}]$ the $(q-1)\times(n-3)$ matrix of scalars,

$$[\boldsymbol{N}] = \begin{bmatrix} N_{2,p}(\bar{u}_1) & \ldots & N_{n-2,p}(\bar{u}_1) \\ \vdots & \ddots & \vdots \\ N_{2,p}(\bar{u}_{q-1}) & \ldots & N_{n-2,p}(\bar{u}_{q-1}) \end{bmatrix} \tag{19}$$

it is possible to rewrite Eq. (18) in the matrix form:

$$([\boldsymbol{N}]^T[\boldsymbol{N}])\boldsymbol{P} = [\boldsymbol{N}]^T\boldsymbol{\rho} \quad \text{with } \boldsymbol{P} = \begin{Bmatrix} \boldsymbol{P}_2 \\ \vdots \\ \boldsymbol{P}_{n-2} \end{Bmatrix} \quad \text{and}$$
$$\boldsymbol{\rho} = \begin{Bmatrix} \boldsymbol{\rho}_1 \\ \vdots \\ \boldsymbol{\rho}_{q-1} \end{Bmatrix} \tag{20}$$

Equation (20) represents three underdetermined systems, because the control points $\boldsymbol{P}_i$ and the terms $\boldsymbol{\rho}_k$ are three-dimensional. The system can be solved choosing the minimum norm solution, which yields the vector $\boldsymbol{P}$ for which the NURBS curve gives the LS approximation of the $q-1$ intermediate waypoints:

$$\boldsymbol{P} = ([\boldsymbol{N}]^T[\boldsymbol{W}][\boldsymbol{N}])^{-1}\ [\boldsymbol{W}][\boldsymbol{N}]^T\boldsymbol{\rho} \tag{21}$$

where $[\boldsymbol{W}]$ is a square, diagonal, positive semidefinite matrix of size $(q-1)$ containing the weights associated with each waypoint. If $[\boldsymbol{W}]$ is the identity, all the waypoints are weighed equally. Equation (21) only works if the matrix $([\boldsymbol{N}]^T[\boldsymbol{W}][\boldsymbol{N}])$ is full rank. It is important to note that the elements that compose the matrix $[\boldsymbol{N}]$ are strongly dependent from the knot vector U defined in Eq. (2). De Boor showed that when every knot span in U contains at least one $\bar{u}_k$, the matrix $([\boldsymbol{N}]^T[\boldsymbol{W}][\boldsymbol{N}])$ is positive definite and well-defined [29]. The following algorithm for choosing the internal knots ensures that this is true [23]:

$$d = \frac{q+1}{n-p+1} \qquad i = \mathtt{int}(jd) \quad \alpha = jd - i$$
$$u_{p+j} = (1-\alpha)\bar{u}_{i-1} + \alpha\bar{u}_i \quad \text{for} \quad j = 1, \dots, n-p \tag{22}$$

In this Note, the algorithm presented in Eq. (22) is slightly modified such that if $u_{p+1} < \bar{u}_1$, then u_{p+1} is set equal to $\bar{u}_1$, and if $u_n > \bar{u}_{q-1}$, then u_n is set equal to $\bar{u}_{q-1}$, to ensure that the first and last polynomial arcs cover at least the distance between the first two and last two waypoints, respectively. This has been observed to stabilize the NURBS function in those intervals, and avoiding overshoots.

Figure 1 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 n, for each MRP component σ_j. It can be observed that for a smaller number of control points the approximation fits the waypoints more loosely. 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, because the required angular rate norm is not accounted for in this LS approximation.

D. LS 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 Eq. (20) 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 $\|\boldsymbol{\omega}\| = \omega^*$. 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

$$g = \sum_{k=1}^{q-1} (\|\boldsymbol{\sigma}'_k\|^2 - \|\boldsymbol{\sigma}'(\bar{u}_k)\|^2)^2$$
$$= \sum_{k=1}^{q-1} \left[\|\boldsymbol{\sigma}'_k\|^2 - \sum_{j=1}^{3} \left(\sum_{i=2}^{n-2} N'_{i,p}(\bar{u}_k)\boldsymbol{P}_{i,j} \right)^2 \right]^2 \tag{23}$$

However, solving for the control points $\boldsymbol{P}_i$ that minimize the error function in Eq. (23) is nontrivial, because, due to the quadratic dependence of the terms in square brackets from the $\boldsymbol{P}_i$, 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 [13], 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 $\boldsymbol{\sigma}'_k$ is provided as follows. As a first step, such derivatives are computed using central finite differences:

$$\hat{\boldsymbol{\sigma}}'_k = \frac{\bar{u}_{k+1} - \bar{u}_k}{\bar{u}_{k+1} - \bar{u}_{k-1}} \cdot \frac{\boldsymbol{\sigma}_k - \boldsymbol{\sigma}_{k-1}}{\bar{u}_k - \bar{u}_{k-1}} + \frac{\bar{u}_k - \bar{u}_{k-1}}{\bar{u}_{k+1} - \bar{u}_{k-1}} \cdot \frac{\boldsymbol{\sigma}_{k+1} - \boldsymbol{\sigma}_k}{\bar{u}_{k+1} - \bar{u}_k}$$
$$\text{for } k = 1, \dots, q-1 \tag{24}$$

From Eq. (14) it is possible to derive the following relation between the angular rate norm and the MRP derivative norm:

$$\|\dot{\boldsymbol{\sigma}}\| = \frac{1+\sigma^2}{4}\|\boldsymbol{\omega}\| \tag{25}$$

Knowing the desired angular rate norm ω^*, 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:

$$\boldsymbol{\sigma}'_k = \frac{\hat{\boldsymbol{\sigma}}'_k}{\|\hat{\boldsymbol{\sigma}}'_k\|} \cdot \frac{1 + \|\boldsymbol{\sigma}_k\|^2}{4}\omega^* \cdot t_q \quad \text{for } k = 1, \dots, q-1 \tag{26}$$

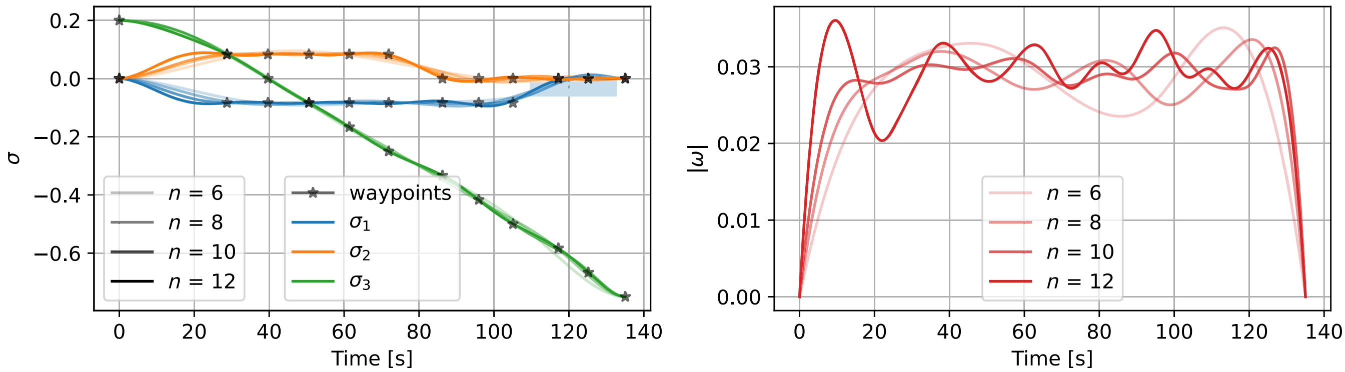

Fig. 1 LS approximating NURBS, attitude-based, for varying numbers of control points *n*.

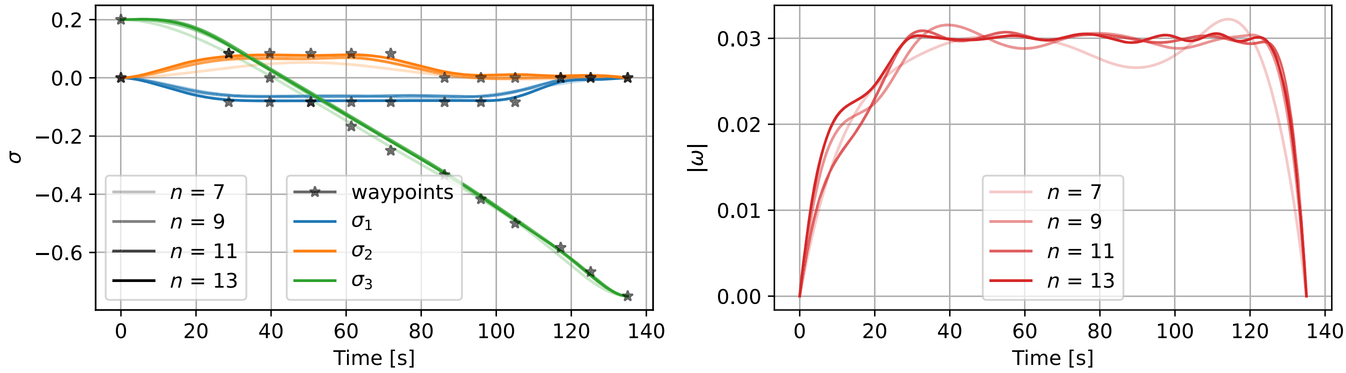

Fig. 2   LS approximating NURBS, attitude- and angular rate-based, for varying numbers of control points *n*.

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

$$\boldsymbol{\rho}'_k = \boldsymbol{\sigma}'_k - N'_{0,p}(\bar{u}_k)\boldsymbol{P}_0 - N'_{1,p}(\bar{u}_k)\boldsymbol{P}_1 - N'_{n-1,p}(\bar{u}_k)\boldsymbol{P}_{n-1} - N'_{n,p}(\bar{u}_k)\boldsymbol{P}_n \quad \text{for } k = 1,\ldots,q-1 \tag{27}$$

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:

$$\begin{aligned} f &= \sum_{k=1}^{q-1}(\|\boldsymbol{\sigma}_k - \boldsymbol{\sigma}(\bar{u}_k)\|^2 + \|\boldsymbol{\sigma}'_k - \boldsymbol{\sigma}(\bar{u}_k)'\|^2) \\ &= \sum_{k=1}^{q-1}\left(\left\|\boldsymbol{\rho}_k - \sum_{i=2}^{n-2} N_{i,p}(\bar{u}_k)\boldsymbol{P}_i\right\|^2 + \left\|\boldsymbol{\rho}'_k - \sum_{i=2}^{n-2} N'_{i,p}(\bar{u}_k)\boldsymbol{P}_i\right\|^2\right) \end{aligned} \tag{28}$$

Applying the same procedure to minimize f with respect to $\boldsymbol{P}_i$ gives the same minimum norm solution as in Eq. (21), only in this case the matrices that appear in the equation take the form:

$$[\boldsymbol{N}] = \begin{bmatrix} N_{2,p}(\bar{u}_1) & \ldots & N_{n-2,p}(\bar{u}_1) \\ \vdots & \ddots & \vdots \\ N_{2,p}(\bar{u}_{q-1}) & \ldots & N_{n-2,p}(\bar{u}_{q-1}) \\ N'_{2,p}(\bar{u}_1) & \ldots & N'_{n-2,p}(\bar{u}_1) \\ \vdots & \ddots & \vdots \\ N'_{2,p}(\bar{u}_{q-1}) & \ldots & N'_{n-2,p}(\bar{u}_{q-1}) \end{bmatrix},$$

$$\boldsymbol{P} = \begin{Bmatrix} \boldsymbol{P}_2 \\ \vdots \\ \boldsymbol{P}_{n-2} \end{Bmatrix}, \quad \boldsymbol{\rho} = \begin{Bmatrix} \boldsymbol{\rho}_1 \\ \vdots \\ \boldsymbol{\rho}_{q-1} \\ \boldsymbol{\rho}'_1 \\ \vdots \\ \boldsymbol{\rho}'_{q-1} \end{Bmatrix} \tag{29}$$

with $[\boldsymbol{N}]$ being a $(2q-2)\times(n-3)$ matrix and $\boldsymbol{\rho}$ being a $(2q-2)$ vector, while $\boldsymbol{P}$ remains an $(n-3)$ vector. The weight matrix $[\boldsymbol{W}]$ 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 $\boldsymbol{\omega}$ is a function of both $\boldsymbol{\sigma}$ and $\dot{\boldsymbol{\sigma}}$: this means that if the error on $\boldsymbol{\sigma}$ is large, the desired ω^* will not be achieved even if the error on $\dot{\boldsymbol{\sigma}}$ 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 2 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 Fig. 1, but still recover the distribution of the waypoints faithfully. The reduced 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.

The LS approach allows to also model the accelerations at each intermediate waypoint and add them to the error function analogously. However, in such case the resulting function attempts at satisfying three different requirements, on attitude, angular rates, and accelerations, therefore losing tracking accuracy on all three. If the desire is to reduce the angular accelerations, especially in the on-ramping and off-ramping phases, the best choice is to reduce the nominal angular rate ω^*, which ultimately results in a longer total maneuver time.

III.   Performance Study of NURBS Interpolation Versus 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. In fact, the interpolating NURBS causes the trajectory to wiggle more between the waypoints, which causes 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. For the LS approximating NURBS, $n+1$ control points are used, where $n = q+2$, i.e., one control point per waypoint plus two control points for the endpoint derivatives constraints. This is the same number of control points required by the interpolating NURBS. 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 Eq. (6) as the algorithm’s cost function: such trajectories are, therefore, not effort-optimal, but this is not relevant for the comparisons that this sections aims to make. Lastly, the computational cost of the two NURBS is also compared.

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

$$
{}^{\mathcal{B}}[\boldsymbol{I}] = \begin{bmatrix} 6.67 & 0 & 0 \\ 0 & 41.87 & 0 \\ 0 & 0 & 41.87 \end{bmatrix} \cdot 10^{-3}\ \mathrm{kg}\cdot\mathrm{m}^2 \tag{30}
$$

The control effort is estimated, like in Ref. [13], as the integral over the trajectory of the norm of the instantaneous control torque:

$$
U = \int_0^T \|\boldsymbol{L}\|\,\mathrm{d}t \quad \text{with} \quad \boldsymbol{L} = {}^{\mathcal{B}}[\boldsymbol{I}]^{\mathcal{B}}\dot{\boldsymbol{\omega}}_{\mathcal{B}\mathcal{N}} + {}^{\mathcal{B}}\boldsymbol{\omega}_{\mathcal{B}\mathcal{N}} \times {}^{\mathcal{B}}[\boldsymbol{I}]^{\mathcal{B}}\boldsymbol{\omega}_{\mathcal{B}\mathcal{N}} \tag{31}
$$

The control effort in Eq. (31) relies on the assumption that the whole spacecraft can be treated as a rigid body. This is an approximation, because the reaction wheels that actuate the spacecraft contribute to the inertia and angular momentum of the system, and introduce gyroscopic effects that also contribute to the control effort. A more refined, although more computationally demanding, path-planning algorithms is presented in Ref. [30], where the dynamics of the reaction wheels is factored into the cost function. All the scenarios presented in this section feature rest-to-rest maneuvers with a target angular rate norm $\omega^* = 0.03\ \mathrm{rad/s}$.

A. Scenario 1: Eigenaxis Rotation

The first scenario is very simple, as it features an eigenaxis rotation about the $\hat{\boldsymbol{b}}_3$ axis. Only one sensitive instrument with a field of view of 20 deg is oriented along the $\hat{\boldsymbol{b}}_1$ axis, and one keep-out zone is present along the inertial direction ${}^{\mathcal{N}}\hat{\boldsymbol{s}} = [-1, 0, 0]$. The initial and final attitudes are, respectively, $\boldsymbol{\sigma}_0 = [0, 0, 0.1]$ and $\boldsymbol{\sigma}_q = [0, 0, -0.75]$. Figure 3 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 Fig. 4, 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 $\boldsymbol{\sigma}_3$ axis. For all values of N the angular rate norm is approximated very well, as it stays very close to the target value. It can be noticed that the on-ramping at the beginning of the maneuver is milder than the off-ramping. This can happen due to two main reasons: the first is that the sampling is uniform in MRP space, but not in attitude space, due to the nonlinearity in the MRP formulation. As a result, neighboring points farther away from the origin describe tighter maneuvers in terms of principal rotation angle. As a result, the NURBS curve at the end of the maneuver has less time to off-ramp from the average angular rate ω^*. The second reason is that initial and final attitude points do not, in general, belong to the uniformly spaced grid, and could therefore be located closer or farther to the next node than the average node distance. This can further emphasize the first phenomenon, if this happens when one of the endpoints is far away from the origin. As a result of these combined effects, the fast off-ramping can cause oscillations in the curve near the point where the velocity starts to decrease. Figure 5 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.

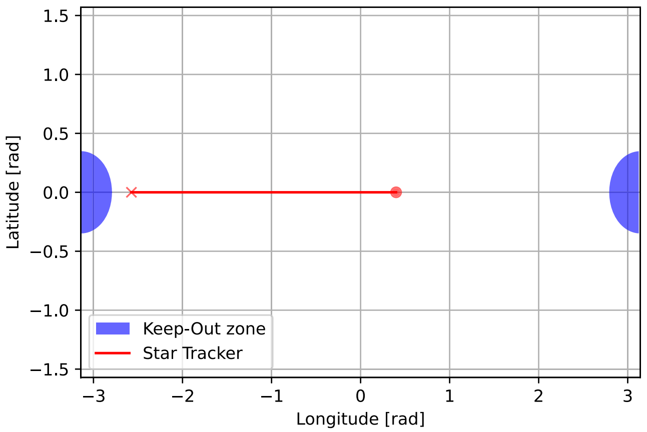

Fig. 3 Boresight plot: scenario 1.

B. 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 ${}^{\mathcal{N}}\hat{\boldsymbol{s}}_2 = [0, -0.981, -0.196]$ and ${}^{\mathcal{N}}\hat{\boldsymbol{s}}_3 = [0.958, 0, 0.287]$. Figure 6 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 7 shows that, even in this case, the angular rate is approximated quite well, although the plateau around $\omega^* = 0.03\ \mathrm{rad/s}$ 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 Eqs. (11) and (26) 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, Fig. 8 compares the performances of the two NURBS curves. In this scenario, the

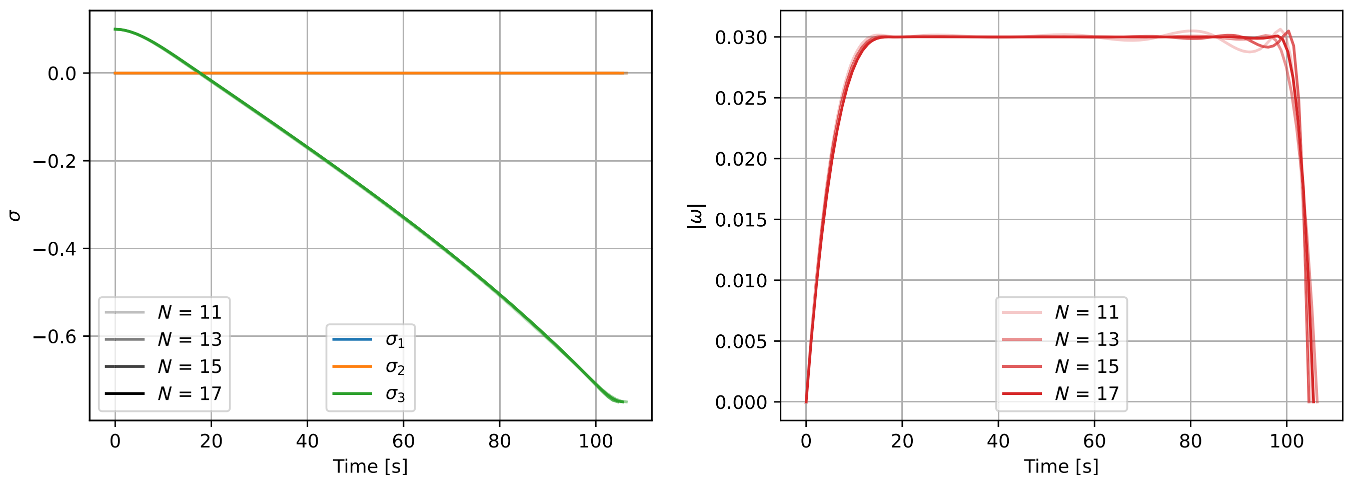

Fig. 4 Attitude σ and angular rate $\|\omega\|$ for varying grid densities N: scenario 1.

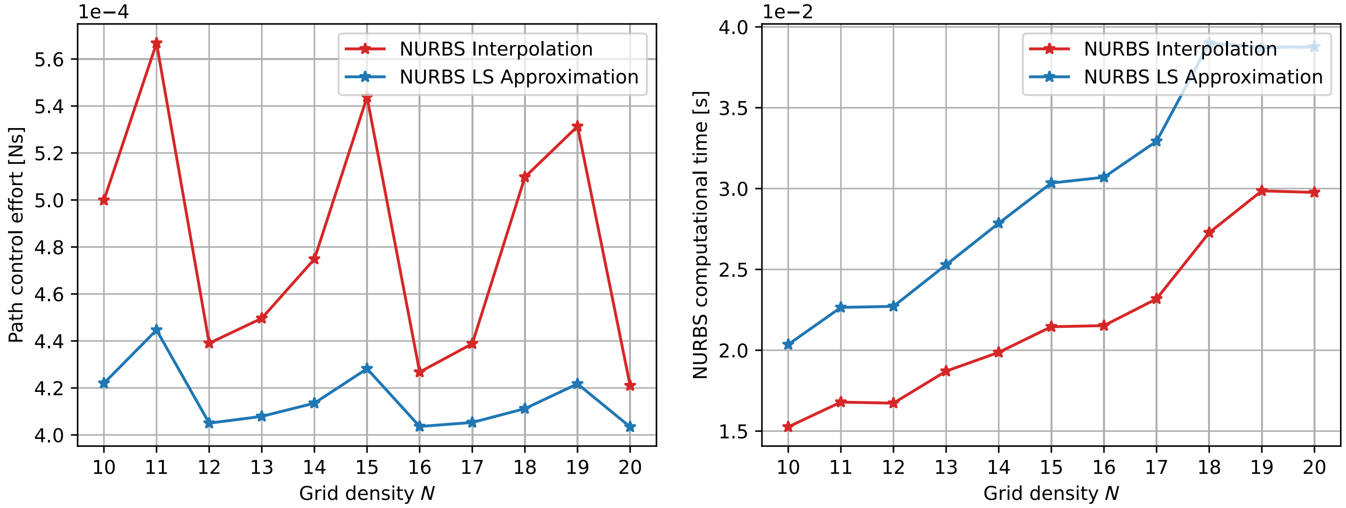

Fig. 5 Control effort and computational time of interpolating NURBS vs LS approximating NURBS, for varying grid densities N: scenario 1.

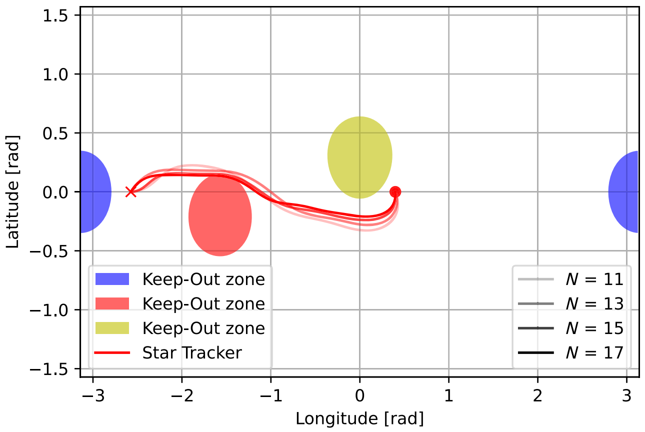

Fig. 6 Boresights plot: scenario 2.

interpolating NURBS is associated 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.

C. 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 $\hat{\boldsymbol{b}}_1$ axis, and two sun sensors, along the $\hat{\boldsymbol{b}}_2$ and $\hat{\boldsymbol{b}}_3$ axes, respectively, each with a field of view of 70 deg. The only celestial object is the sun, located along the ${}^{\mathcal{N}}\hat{s}_3 = [1, 0, 0]$ 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 attitudes are, respectively, $\boldsymbol{\sigma}_0 = [0, -0.25, -0.25]$ and $\boldsymbol{\sigma}_q = [0.4, 0.4, 0.3]$. Figure 9 shows the trajectories of the three boresights in inertial space: all 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 Fig. 10 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 Fig. 11, 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.

IV. 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 Ref. [13]. As this version of the A* algorithm searches the graph, it uses intermediate paths

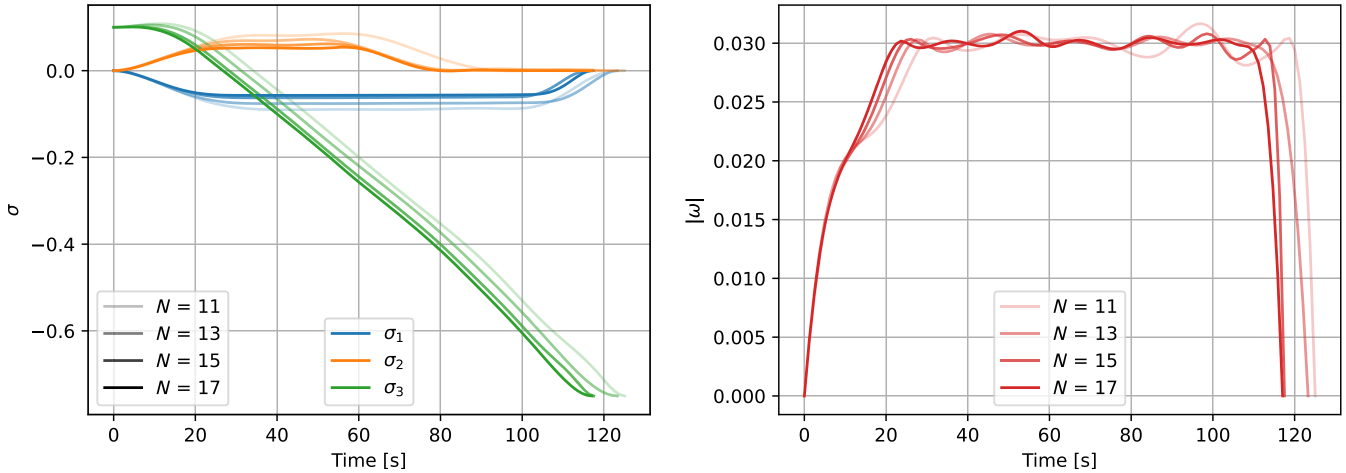

Fig. 7 Attitude σ and angular rate $\|\omega\|$ for varying grid densities N: scenario 2.

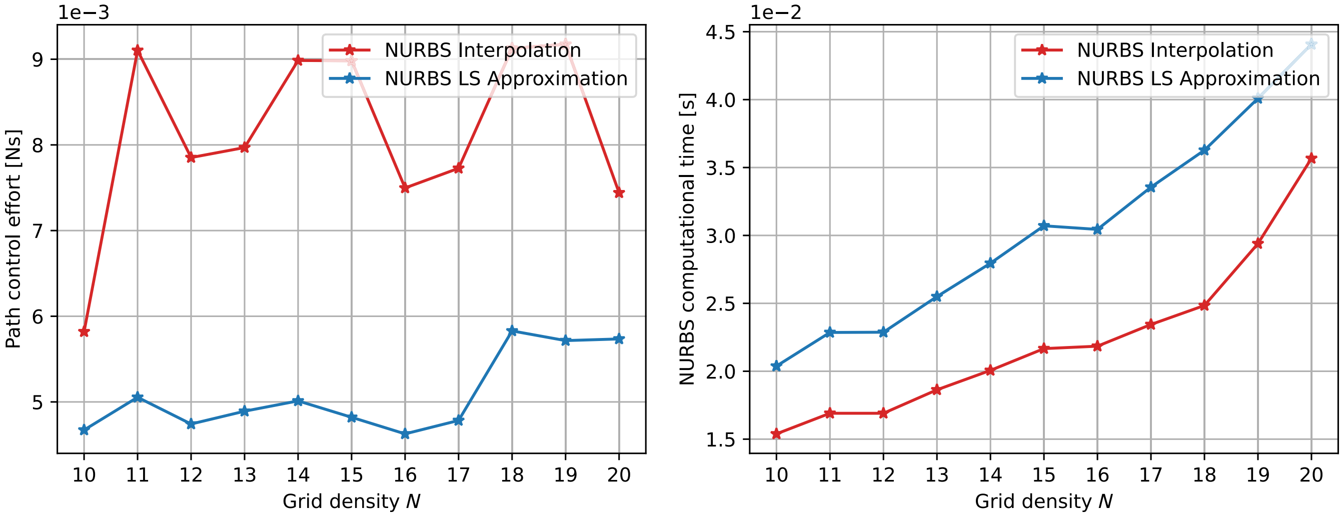

Fig. 8 Control effort and computational time of interpolating NURBS vs LS approximating NURBS, for varying grid densities N: scenario 2.

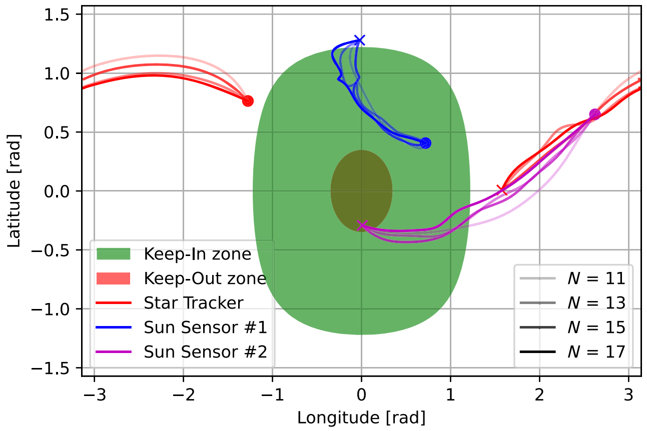

Fig. 9 Boresights plot: scenario 3.

consisting of the previously explored nodes ($\boldsymbol{\sigma}_0, \ldots, \boldsymbol{\sigma}_{n-1}$), the current open node ($\boldsymbol{\sigma}_n$), and the goal node ($\boldsymbol{\sigma}_q$) as a baseline for an intermediate trajectory computed via NURBS curves. The control effort integral in Eq. (31), evaluated along these intermediate trajectories, is used as 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 $\boldsymbol{\sigma}_0$ to the current open node $\boldsymbol{\sigma}_n$; the heuristic is an estimate of the cost required to track the trajectory from the current open node to the goal node $\boldsymbol{\sigma}_q$. 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 [10]; i.e., the cost of the final path to goal is higher or equal to the priority function $p(n)$. This gives

$$p(n) = g(n) + h(n) = \int_0^{t_n} \|\boldsymbol{L}\| \,\mathrm{d}t + \int_{t_n}^{t_q} \|\boldsymbol{L}\| \,\mathrm{d}t = \int_0^{t_q} \|\boldsymbol{L}\| \,\mathrm{d}t \tag{32}$$

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 12 shows that the control effort of the effort-optimal solutions, for both types of NURBS, matches the results shown in Fig. 5. 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. 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 13 and 14 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

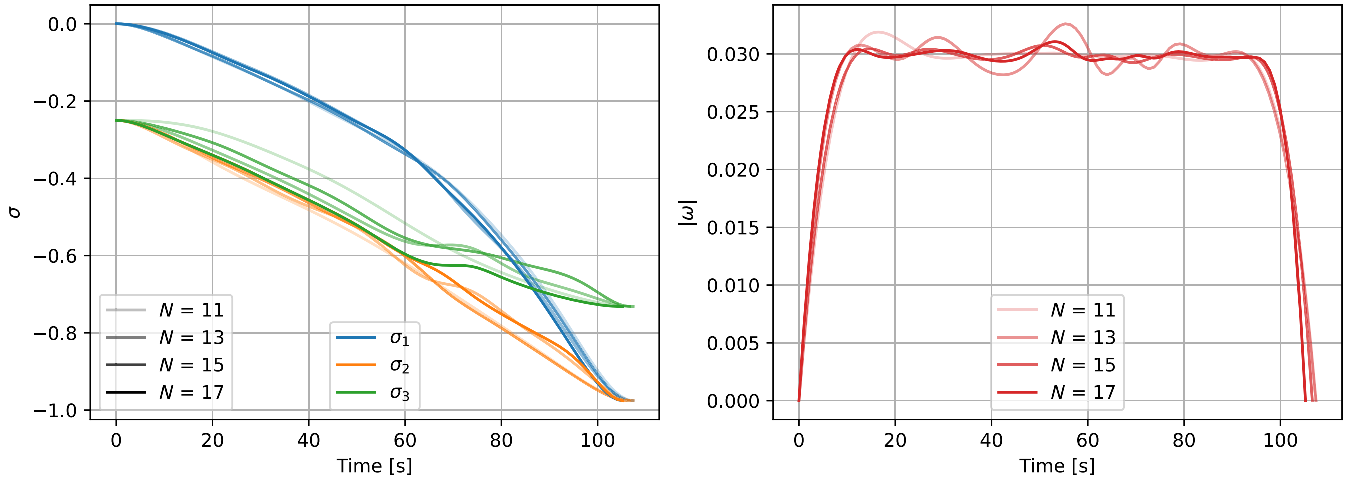

Fig. 10 Attitude σ and angular rate $\|\omega\|$ for varying grid densities N: scenario 3.

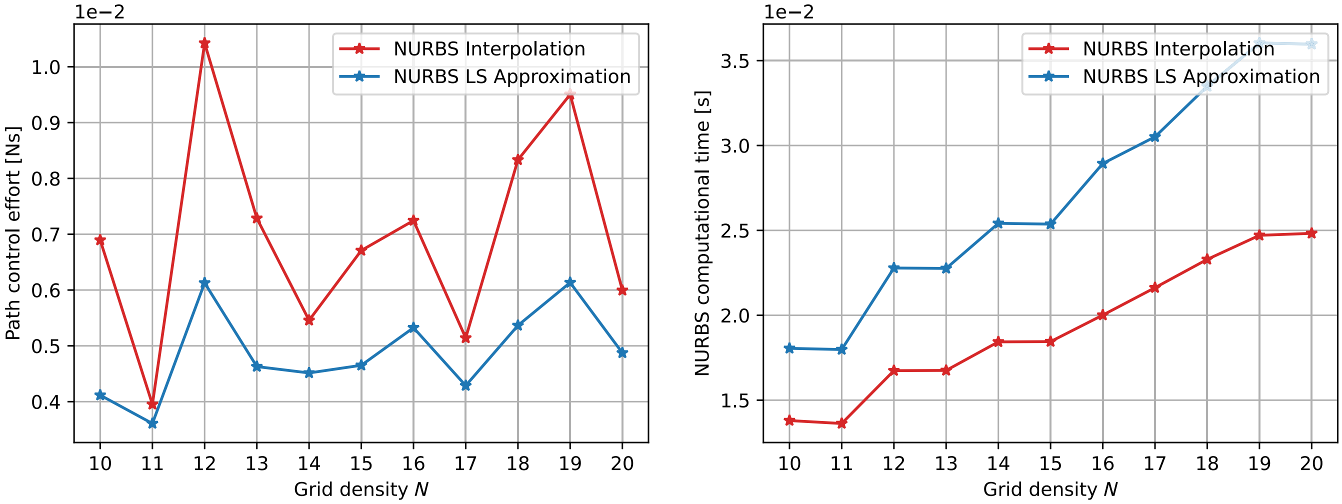

Fig. 11 Control effort and computational time of interpolating NURBS vs LS approximating NURBS, for varying grid densities N: scenario 3.

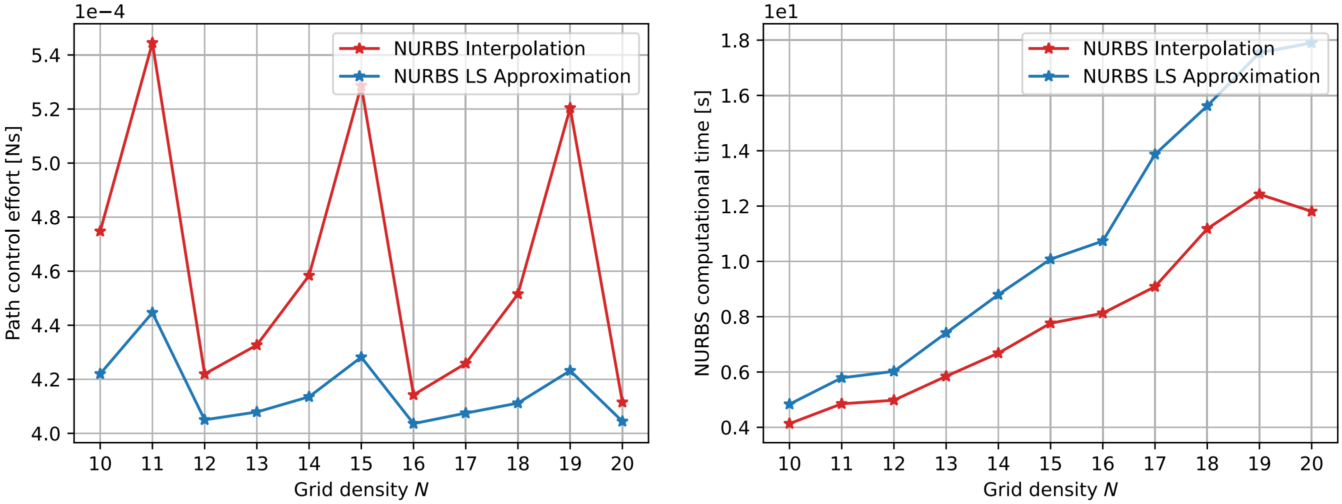

Fig. 12 Control effort and computational time of effort-based A*, NURBS comparison: scenario 1.

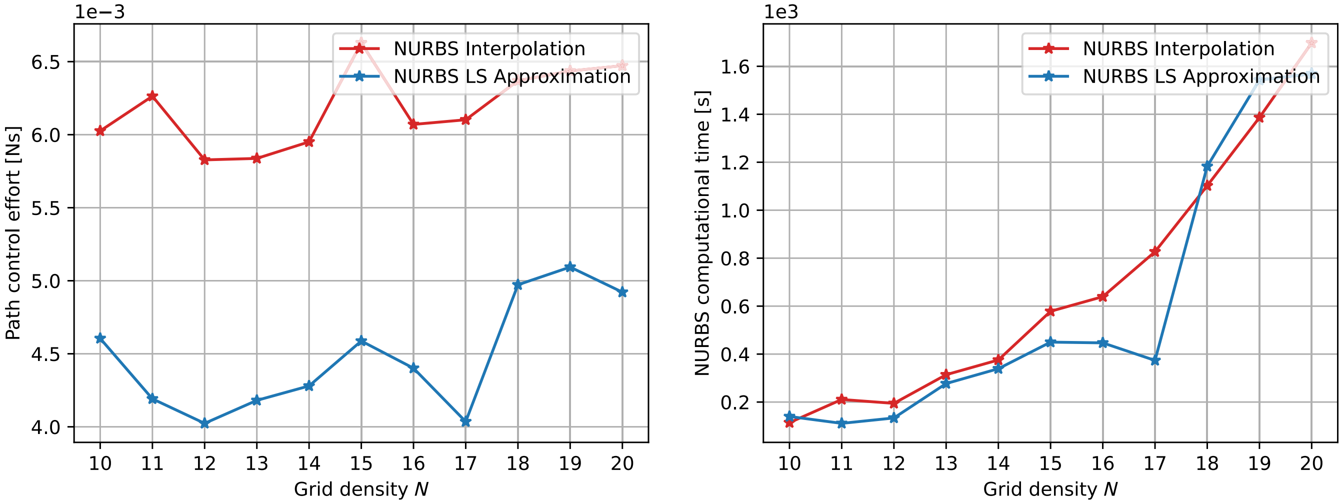

Fig. 13 Control effort and computational time of effort-based A*, NURBS comparison: scenario 2.

consistently outperforms the interpolating curve, always yielding less costly solutions.

The effort-based A* algorithm, whose performance is described in Figs. 12–14, 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 min to compute the optimal solution. However, the code used in this work is not implemented to optimize for runtime. Future work will address the improvement of runtime an memory utilization to assess the suitability for onboard utilization.

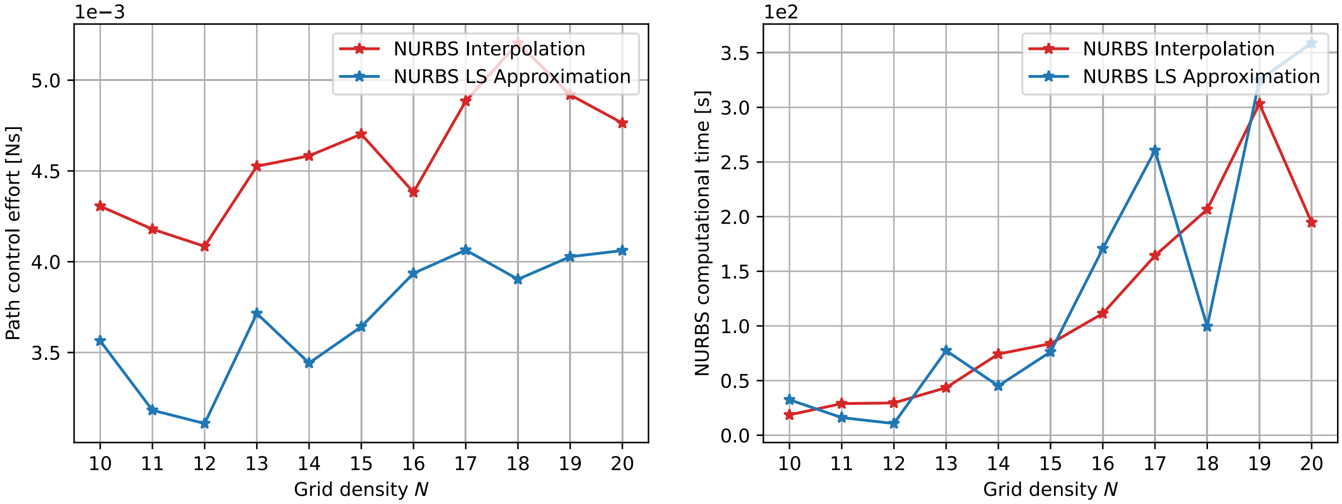

Fig. 14 Control effort and computational time of effort-based A*, NURBS comparison: scenario 3.

Interestingly, Figs. 13 and 14 show that the computational times for the two different NURBS are comparable. This is not intuitive if looking at the results shown in Figs. 4, 7, and 10, where, when the path is precomputed by a metric-based version of A*, the LS approximating NURBS seems to consistently require a larger computational time than the interpolating NURBS. The reason why the effort-based A* with LS approximation is not consistently more time consuming is that, in the effort-based implementation, the priority function $p(n)$ outputs a smaller, more "optimistic" estimate of the total path cost. This allows the effort-based A* algorithm combined with the LS approach to converge to the final solution more efficiently [10], 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 $10 \le N \le 15$, 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 reduces the chances of violating the constraints. On the other hand, the computational time increases significantly, and the computed trajectories, even the effort-optimal ones, are more costly in terms of control torque. It is important to state that these considerations are case specific. In a different scenario, with the same grid density, but where the obstacles occupy a very large portion of the attitude space, convergence is accelerated by the reduced number of compliant sampled waypoints. Ultimately, the appropriate grid size can be further characterized by exploring node density in correlation with the size of the constraint-incompliant attitude space.

V. Conclusions

This Note 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 an LS minimization technique. This contribution shows how the LS 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 standalone run, requires a larger computational time, due to the larger number of calculations it needs to perform to invert larger matrices. However, within the effort-optimal A* graph-search algorithm, which runs multiple NURBS calculations sequentially, the two NURBS approaches are comparable in terms of computational time. Lastly, this Note shows how the grid density N affects the computational time of the algorithm and the ultimate path cost associated with the computed trajectories.

References

[1] McInnes, C. R., "Large Angle Slew Maneuvers with Autonomous Sun Vector Avoidance," *Journal of Guidance, Control, and Dynamics*, Vol. 17, No. 4, 1994, pp. 875–877.
https://doi.org/10.2514/3.21283

[2] Diaz Ramos, M., and Schaub, H., "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.
https://doi.org/10.2514/1.G002873

[3] Lee, U., and Mesbahi, M., "Spacecraft Reorientation in Presence of Attitude Constraints via Logarithmic Barrier Potentials," *Proceedings of the 2011 American Control Conference*, IEEE, New York, 2011, pp. 450–455.
https://doi.org/10.1109/acc.2011.5991284

[4] Lee, U., and Mesbahi, M., "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.
https://doi.org/10.1109/TAES.2014.120240

[5] Kjellberg, H. C., and Lightsey, E. G., "Discretized Constrained Attitude Pathfinding and Control for Satellites," *Journal of Guidance, Control, and Dynamics*, Vol. 36, No. 5, 2013, pp. 1301–1309.
https://doi.org/10.2514/1.60189

[6] Kjellberg, H. C., and Lightsey, E. G., "Discretized Quaternion Constrained Attitude Pathfinding," *Journal of Guidance, Control, and Dynamics*, Vol. 39, No. 3, 2015, pp. 713–718.
https://doi.org/10.2514/1.G001063

[7] Tanygin, S., "Fast Autonomous Three-Axis Constrained Attitude Pathfinding and Visualization for Boresight Alignment," *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 2, 2017, pp. 358–370.
https://doi.org/10.2514/1.G001801

[8] Tarjan, R., "Depth-First Search and Linear Graph Algorithms," *SIAM Journal on Computing*, Vol. 1, No. 2, 1972, pp. 146–160.
https://doi.org/10.1137/0201010

[9] Lee, C. Y., "An Algorithm for Path Connections and Its Applications," *IRE Transactions on Electronic Computers*, Vol. EC-10, No. 3, 1961, pp. 346–365.
https://doi.org/10.1109/tec.1961.5219222

[10] Hart, P. E., Nilsson, N. J., and Raphael, B., "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.
https://doi.org/10.1109/TSSC.1968.300136

[11] Frazzoli, E., Dahleh, M., Feron, E., and Kornfeld, R., "A Randomized Attitude Slew Planning Algorithm for Autonomous Spacecraft," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, AIAA Paper 2001-4155, 2001.
https://doi.org/10.2514/6.2001-4155

[12] Tan, X., Berkane, S., and Dimarogonas, D. V., "Constrained Attitude Maneuvers on SO(3): Rotation Space Sampling, Planning and Low-Level Control," *Automatica*, Vol. 112, Feb. 2020, Paper 108659.
https://doi.org/10.1016/j.automatica.2019.108659

[13] Calaon, R., and Schaub, H., "Constrained Attitude Maneuvering via Modified-Rodrigues-Parameter-Based Motion Planning Algorithms,"

Journal of Spacecraft and Rockets, Vol. 59, No. 4, 2022, pp. 1342–1356. https://doi.org/10.2514/1.A35294

[14] Xiaojun, C., Hutao, C., Pingyuan, C., and Rui, X., "Large Angular Autonomous Attitude Maneuver of Deep Spacecraft Using Pseudospectral Method," *3rd International Symposium on Systems and Control in Aeronautics and Astronautics*, IEEE, New York, 2010, pp. 1510–1514. https://doi.org/10.1109/ISSCAA.2010.5632498

[15] Melton, R. G., "Maximum-Likelihood Estimation Optimizer for Constrained, Time-Optimal Satellite Reorientation," *Acta Astronautica*, Vol. 103, Oct. 2014, pp. 185–192. https://doi.org/10.1016/j.actaastro.2014.06.032

[16] Melton, R. G., "Hybrid Methods for Determining Time-Optimal, Constrained Spacecraft Reorientation Maneuvers," *Acta Astronautica*, Vol. 94, No. 1, 2014, pp. 294–301. https://doi.org/10.1016/j.actaastro.2013.05.007

[17] Spiller, D., Ansalone, L., and Curti, F., "Particle Swarm Optimization for Time-Optimal Spacecraft Reorientation with Keep-Out Cones," *Journal of Guidance, Control, and Dynamics*, Vol. 39, No. 2, 2016, pp. 312–325. https://doi.org/10.2514/1.G001228

[18] Spiller, D., Melton, R. G., and Curti, F., "Inverse Dynamics Particle Swarm Optimization Applied to Constrained Minimum-Time Maneuvers Using Reaction Wheels," *Aerospace Science and Technology*, Vol. 75, April 2018, pp. 1–12. https://doi.org/10.1016/j.ast.2017.12.038

[19] Calaon, R., Schaub, H., and Trowbridge, M. A., "Basilisk-Based Benchmark Analysis of Different Constrained Attitude Dynamics Planners," *Journal of Aerospace Information Systems*, Vol. 20, No. 2, 2023, pp. 60–69. https://doi.org/10.2514/1.I011109

[20] Ravankar, A., Ravankar, A. A., Kobayashi, Y., Hoshino, Y., and Peng, C.-C., "Path Smoothing Techniques in Robot Navigation: State-of-the-Art, Current and Future Challenges," *Sensors*, Vol. 18, No. 9, 2018, Paper 3170. https://doi.org/10.3390/s18093170

[21] Hu, Q., Chi, B., and Akella, M. R., "Anti-Unwinding Attitude Control of Spacecraft with Forbidden Pointing Constraints," *Journal of Guidance, Control, and Dynamics*, Vol. 42, No. 4, 2019, pp. 822–835. https://doi.org/10.2514/1.G003606

[22] Schaub, H., and Piggott, S., "Speed-Constrained Three-Axes Attitude Control Using Kinematic Steering," *Acta Astronautica*, Vol. 147, June 2018, pp. 1–8. https://doi.org/10.1016/j.actaastro.2018.03.022

[23] Piegl, L., and Tiller, W., *The NURBS Book*, Springer Science & Business Media, Berlin, 1996, Chap. 9. https://doi.org/10.1007/978-3-642-59223-2

[24] Fornberg, B., and Zuev, J., "The Runge Phenomenon and Spatially Variable Shape Parameters in RBF Interpolation," *Computers & Mathematics with Applications*, Vol. 54, No. 3, 2007, pp. 379–398. https://doi.org/10.1016/j.camwa.2007.01.028

[25] De Boor, C., "On Calculating with B-Splines," *Journal of Approximation Theory*, Vol. 6, No. 1, 1972, pp. 50–62. https://doi.org/10.1016/0021-9045(72)90080-9

[26] Schaub, H., and Junkins, J. L., *Analytical Mechanics of Space Systems*, 4th ed., AIAA, Reston, VA, 2018, Chap. 3. https://doi.org/10.2514/4.105210

[27] Tsiotras, P., Junkins, J. L., and Schaub, H., "Higher-Order Cayley Transforms with Applications to Attitude Representations," *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 3, 1997, pp. 528–534. https://doi.org/10.2514/2.4072

[28] Tanygin, S., "Attitude Parameterizations as Higher-Dimensional Map Projections," *Journal of Guidance, Control, and Dynamics*, Vol. 35, No. 1, 2012, pp. 13–24. https://doi.org/10.2514/1.54085

[29] De Boor, C., and De Boor, C., *A Practical Guide to Splines*, Vol. 27, Springer–Verlag, New York, 1978, Chap. 14.

[30] Calaon, R., and Schaub, H., "Optimal Actuator-Based Attitude Maneuvering of Constrained Spacecraft via Motion Planning Algorithms," *International Astronautical Congress*, International Astronautical Federation (IAF) Paper IAC-22,C1,2,5,x68843, Paris, France, 2022.