

Rigid Body Attitude Control Based on a Manifold Representation of Direction Cosine Matrices

David Nakath, Joachim Clemens, Carsten Rachuy

Cognitive Neuroinformatics, University of Bremen,
Enrique-Schmidt-Str. 5, 28359 Bremen, Germany

E-mail: {dnakath,jclemens,rachuy}@informatik.uni-bremen.de

Abstract. Autonomous systems typically actively observe certain aspects of their surroundings, which makes them dependent on a suitable controller. However, building an attitude controller for three degrees of freedom is a challenging task, mainly due to singularities in the different parametrizations of the three dimensional rotation group $SO(3)$. Thus, we propose an attitude controller based on a manifold representation of direction cosine matrices: In state space, the attitude is globally and uniquely represented as a direction cosine matrix $R \in SO(3)$. However, differences in the state space, i.e., the attitude errors, are exposed to the controller in the vector space \mathbb{R}^3 . This is achieved by an operator, which integrates the matrix logarithm mapping from $SO(3)$ to $\mathfrak{so}(3)$ and the map from $\mathfrak{so}(3)$ to \mathbb{R}^3 . Based on this representation, we derive a proportional and derivative feedback controller, whose output has an upper bound to prevent actuator saturation. Additionally, the feedback is preprocessed by a particle filter to account for measurement and state transition noise. We evaluate our approach in a simulator in three different spacecraft maneuver scenarios: (i) stabilizing, (ii) rest-to-rest, and (iii) nadir-pointing. The controller exhibits stable behavior from initial attitudes near and far from the setpoint. Furthermore, it is able to stabilize a spacecraft and can be used for nadir-pointing maneuvers.

1. Introduction

Autonomous systems typically need to keep track of their position and attitude, i.e., their pose, as well as properties of their surroundings for successful operations. The gathered information is mutually beneficial, as – on the one hand – precise pose information improves the ability to describe the surroundings and – on the other hand – a precise description of the latter improves the ability to determine an exact pose. To this end, it is often useful for autonomous systems to actively observe certain aspects of their surroundings, which makes them dependent on a suitable controller.

This especially holds for space missions, where active observation of planetary objects can be used for absolute navigation [11, 18] as well as for nadir-pointing maneuvers for relative navigation in the vicinity of a celestial object, e.g., within a LIDAR-based SLAM approach [6, 20]. However, the scenario may also apply to airplanes, underwater vehicles [9], ground vehicles, and robotic systems, which have to inspect certain aspects of their surroundings by aligning some sensor to a point of interest.

The only global and unique representation of the set of all attitudes a rigid body might assume in three dimensions is the *direction cosine matrix* (DCM) [5]. Parameterizations like

Euler angles have the disadvantage of geometric and kinematic singularities as well as different conventions. The (modified) Rodriguez parameters, which are derived from unit quaternions, suffer from geometric singularities. Even the globally defined unit quaternions can lead to undesired rotations, which is known as controller unwinding [22, 2, 16]. The latter problem stems from the non-unique map between the unit quaternion space $\hat{\mathbb{H}}$ and the rotation space $SO(3)$ [2, 5].

As the computational power and available memory of autonomous systems constantly increases, it has become feasible to directly employ DCMs for pose representations. Thus, the drawbacks of the parametrized attitude representations can be omitted by running control algorithms directly on DCMs [10].

In this paper, we present a filtered proportional derivative saturation preventing rigid body controller. It represents the attitude in $SO(3)$ using a DCM, while exposing the differences between two attitudes in the vector space \mathbb{R}^3 employing the \boxplus -method [13]. The derivative term is bounded by a nonlinearity, to ensure a bounded control torque in each particular axis. The feedbacks are preprocessed by a particle filter to account for state and measurement noise. The tuning of the controller in the respective maneuver scenarios can be considered as another modest contribution.

This paper is structured as follows. First, in Sect. 2, the problems to be addressed and the corresponding mathematical and geometrical preliminaries will be presented. In Sect. 3, the spacecraft (s/c)-state and a particle filter-based (PF) estimation scheme will briefly be introduced. Based on these foundations, the controller will be derived in Sect. 4. Subsequently, it will be evaluated and discussed in Sect. 5. Finally, this paper concludes in Sect. 6.

2. Preliminaries

In the following, we will briefly cover the mathematical foundations of the derivation of the controller.

2.1. Attitude

Consider a rigid body representing a s/c.¹ An attitude is the rotation of the principal axes of some body-fixed frame F_b relative to some global inertial frame F_a , which is uniquely described by a DCM denoted by R_{ba} . The set of all attitudes is composed of all orthonormal rotation matrices with determinant equal to one, i.e., $\{R \in \mathbb{R}^{3 \times 3} | R^T R = 1, \det R = +1\}$. This set forms the special orthogonal group $SO(3)$ of rigid body rotations in 3D [17].

2.2. \boxplus -Method

We use a \boxplus -encapsulation of $SO(3)$ introduced by Hertzberg et al. [13], which is capable of integrating a higher dimensional state representation \mathcal{S} with a local lower dimensional state representation \mathbb{R}^n (cf. Fig. 1). This is achieved by the mappings

$$\begin{aligned} \boxplus : \mathcal{S} \times \mathbb{R}^n &\rightarrow \mathcal{S}, \\ \boxminus : \mathcal{S} \times \mathcal{S} &\rightarrow \mathbb{R}^n. \end{aligned} \tag{1}$$

The DCM-specific operators encapsulate the manifold by combining the matrix exponential (\boxplus -operator) or logarithm (\boxminus -operator) respectively with the isomorphism between the Lie-algebra $\mathfrak{so}(3)$ and \mathbb{R}^3 , in particular

$$\begin{aligned} x \boxplus \delta &= x \exp \delta, \\ y \boxminus x &= \log(x^{-1}y), \end{aligned} \tag{2}$$

¹ We will only take attitude control into account in this paper, as it can be treated independently of translation in our scenario.

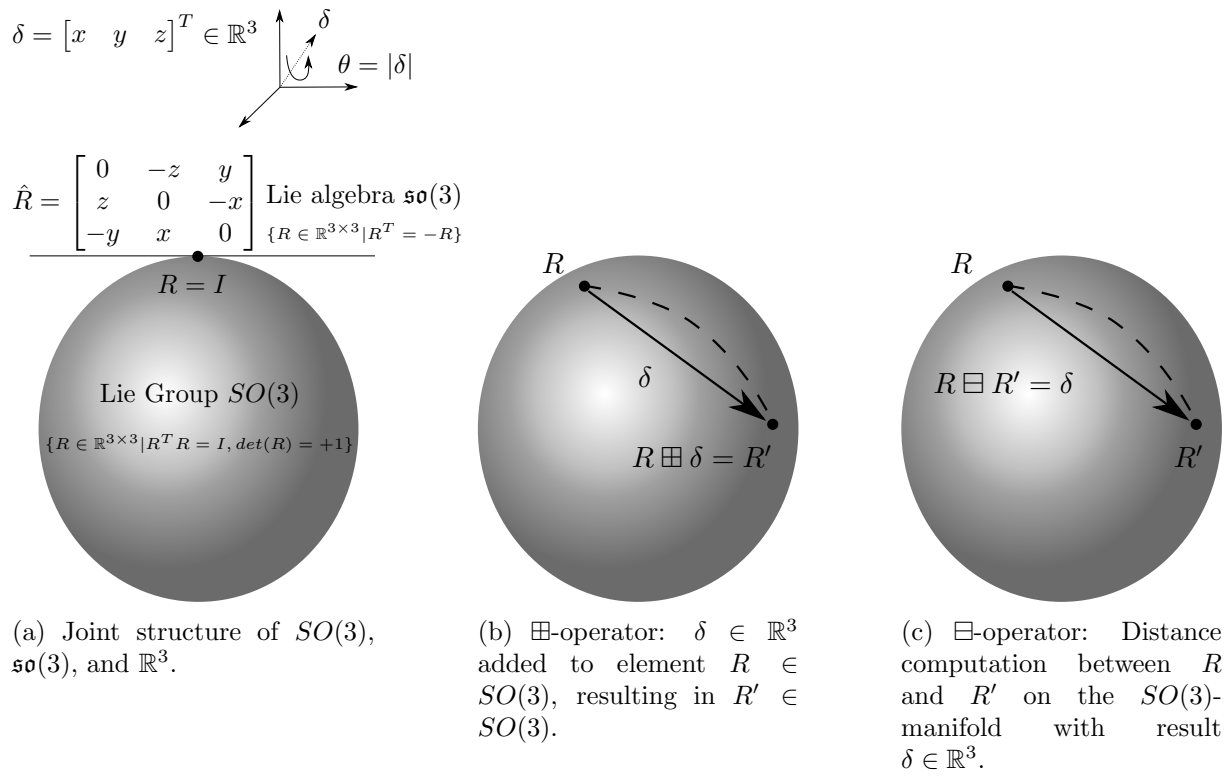


Figure 1: (a) Joint structure of $SO(3)$, $\mathfrak{so}(3)$, and \mathbb{R}^3 with corresponding encapsulating operators in (b) and (c). Dashed lines indicate the geodesics on the manifold, while solid arrows indicate the corresponding vectors in \mathbb{R}^3 . Following [13], the \boxplus -principles are shown using the unit sphere S^2 to simplify visualization.

with $x, y \in SO(3)$ and $\delta \in \mathbb{R}^3$. In the case of 3D rotations, the matrix exponential can be computed in closed-form by the Rodriguez-formula [13, 17, 14]. Combined with the hat map from \mathbb{R}^3 to $\mathfrak{so}(3)$ this results in

$$\exp \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \cos \theta + cx^2 & -sz + cxy & sy + cxz \\ sz + cxy & \cos \theta + cy^2 & -sx + cyz \\ -sy + cxz & sx + cyz & \cos \theta + cz^2 \end{bmatrix}, \quad (3)$$

where $\theta = \sqrt{x^2 + y^2 + z^2}$, $s = \frac{\sin \theta}{\theta}$, and $c = \frac{1 - \cos \theta}{\theta^2}$. The inverse mapping is a combination of the matrix logarithm and the hat map from $\mathfrak{so}(3)$ to \mathbb{R}^3 , which is computed according to

$$\log x = \frac{\theta}{2 \sin \theta} \begin{bmatrix} x_{32} - x_{23} \\ x_{13} - x_{31} \\ x_{21} - x_{12} \end{bmatrix}, \quad (4)$$

where $\theta = \arccos \frac{\text{tr } x - 1}{2}$ and $|\theta| < \pi$. Thus, the global state is represented uniquely in $SO(3)$, while state differences δ , i.e., state transitions or errors, are uniquely represented in \mathbb{R}^3 for angles $< \pi$.

2.3. Dynamic Setpoint Calculation

To have an autonomous system looking at some observable, we have to rotate the sensor normal into the desired view direction. The rotation of a unit vector A to some desired unit vector B

(both defined in F_a) can be described as a 2D rotation on a plane with the normal vector $A \times B$. The corresponding rotation matrix is given by

$$G = \begin{bmatrix} \cos \phi & -\sin \phi & 0 \\ \sin \phi & \cos \phi & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} A \cdot B & -\|A \times B\| & 0 \\ \|A \times B\| & A \cdot B & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (5)$$

where ϕ is the rotation angle. However, to apply the rotation, we have to temporarily change the basis with

$$F = \left[A \quad \frac{B - (A \cdot B)A}{\|B - (A \cdot B)A\|} \quad B \times A \right]^{-1}. \quad (6)$$

The final rotation matrix R_e is obtained by a transformation from the inertial frame F_a to the rotation frame, followed by a rotation about G , and a transformation back to the inertial frame. Thus, we get

$$R_e = F^{-1}GF. \quad (7)$$

If we now assume, for the sake of simplicity, that we want to orient a sensor whose sensor normal points along the z_{ba} axis of R_{ba} , we have that $A = z_{ba}$ and $B = z_{da}$. Thus, we can calculate a rotation matrix R_e following the procedure described above. The resulting R_e can also be applied to another rotation matrix, as $SO(3)$ is closed under matrix multiplication. Hence, we can obtain the desired attitude R_{da} , i.e., where z_{ba} is rotated to z_{da} , by rotating the current attitude R_{ba} according to

$$R_{da} = R_e R_{ba}. \quad (8)$$

In order to obtain a suitable representation for the controller, the rotation from R_{ba} onto R_{da} can be retrieved in vector space using Eq. (2). This yields

$$\delta = R_{ba} \boxminus R_{da}, \quad (9)$$

where the \boxminus -operator ensures that δ represents the geodesic, i.e., the shortest rotation on $SO(3)$ [13]. The approach described above will be used for time-dependent setpoint (SP) calculation $R_{da}(t)$ within the nadir-pointing maneuver (cf. Sect. 5).

2.4. Attitude Kinematics and Dynamics

Applying this geometric framework to attitude kinematics, we let $\omega \in \mathbb{R}^3$ be the angular velocity relative to F_a given in F_b , then the time rate of change of the rigid body attitude is

$$\dot{R}_{ba} = R_{ba} \hat{\omega}. \quad (10)$$

Here, $\hat{\omega}$ is a skew symmetric matrix, as it is an element of the Lie algebra $\mathfrak{so}(3)$ (cf. Fig. 1a), and \dot{R}_{ba} maps the body coordinates of a point to the spatial velocity of that point. The attitude dynamics are the rate of change of angular velocity expressed in the body frame given by

$$I\dot{\omega} = I\omega \times \omega + u, \quad (11)$$

where I is the inertia tensor and u is the sum of external moments applied to the body given in F_b . Finally, throughout this paper we assume that the s/c is fully actuated regarding attitude, i.e., angular moments can be applied on all three axes.

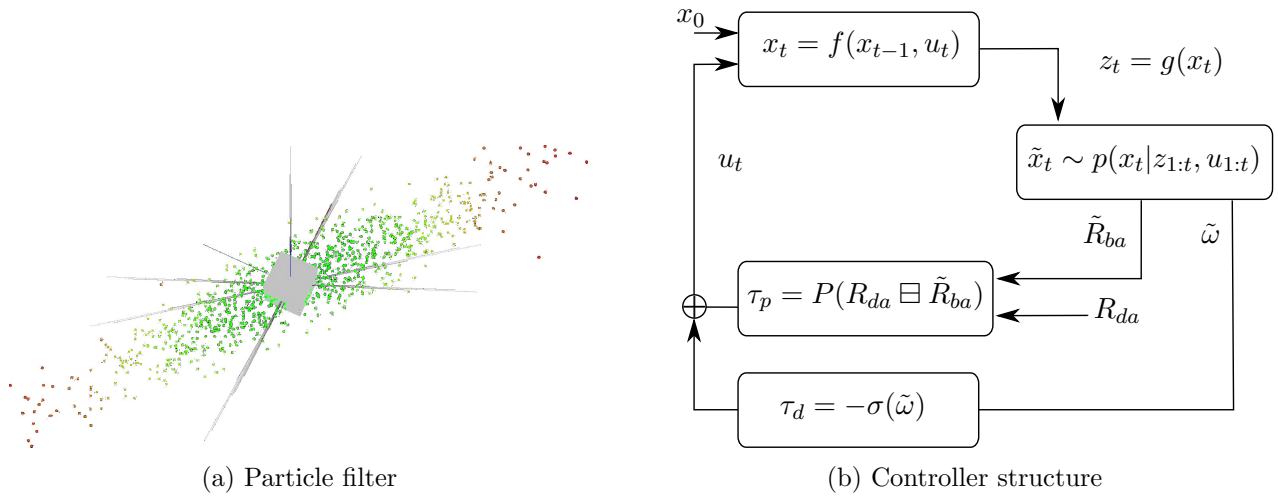


Figure 2: (a) shows the s/c visualization with the corresponding particle distribution, where green and red represents a high and low weight respectively. (b) depicts the control diagram of the filtered pd controller.

3. Particle Filtered Measurements

The controller proposed in the next section relies on measurements in the closed loop. To account for noise in state transition as well as sensor measurements we use a particle filter (PF) [8, 21] for state estimation (cf. Fig. 2a). This type of filter is chosen, as it integrates well with the SLAM approach [6] to be used for relative navigation in the vicinity of asteroids. The basic concepts and integration of attitude sensors (star tracker and 3-axis gyroscope), which are relevant in the context of this paper, are described in the following. Further details on our quaternion-based approach including all sensor models can be found in [11, 18]. The state space is defined as $\mathcal{S} = SE(3) \times \mathbb{R}^6$, where $SE(3) = \mathbb{R}^3 \times SO(3)$ represents the space of position and attitude. The remaining \mathbb{R}^6 is the space of the corresponding first derivatives, i.e., the linear and angular velocities. The PF estimates a probability distribution of the current state $x_t \in \mathcal{S}$ of the following form

$$\tilde{x}_t \sim p(x_t | z_{1:t}, u_{1:t}), \quad (12)$$

where $\tilde{x}_t \in \mathcal{S}$ is the estimated state, $z_{1:t}$ are all preceding measurements and $u_{1:t}$ are all preceding controls. This probability density function is approximated using a set of particles, where each particle represents a state hypothesis. The particle set is updated recursively over time. In each step, the particles are predicted from the previous to the current time step by sampling from the proposal distribution. This is done by integrating the dynamic model of the s/c, which depends on the previous state estimate and the most recent control u_t , using a fourth order Runge-Kutta integrator and applying additive normally-distributed noise with zero mean. Subsequently, the distribution is corrected using importance resampling. The importance measure is obtained by weighting the particles according to the measurement likelihood given the sampled state. Finally, new particles are sampled with a probability proportional to the weights.

A star tracker is used for determining the attitude R_{ba} of the s/c. It operates by measuring angles to stars (see Fig. 3b) and comparing these to known star positions indexed in a star catalog, e.g., the *HIPPARCOS* [19]. For simplicity, we assume that the resulting attitude measurement z_s corresponds to the true attitude R_{ba} and additive normally-distributed noise with zero mean. Hence, the measurement model is given by

$$z_s = R_{ba} \boxplus \varepsilon_s, \quad (13)$$

with $\varepsilon_s \sim \mathcal{N}(0, \Sigma_s)$, where Σ_s is covariance of the noise.

Furthermore, we use the 3-axis gyroscope integrated in an inertial measurement unit (IMU) to obtain the angular velocity ω (see Fig. 3c) of the s/c. The corresponding measurement z_g is also assumed to be affected by normally-distributed noise with zero mean, which results in

$$z_g = \omega + \varepsilon_g, \quad (14)$$

with $\varepsilon_g \sim \mathcal{N}(0, \Sigma_g)$ and Σ_g being the noise covariance matrix.

In order to retrieve a single estimate from the set of particles as an input for our controller, the posterior distribution is approximated using a normal distribution. This basically corresponds to a weighted average for the mean and calculating the covariance with respect to the estimated mean using the \boxminus -operator (see [18] for further details). This avoids computationally expensive least squares estimations [12]. The controller proposed in the following section will rely on the angular velocity $\tilde{\omega}$ and the attitude \tilde{R}_{ba} estimated by the PF as part of \tilde{x}_t .

4. Filtered Proportional Derivative Controller

The proportional derivative controller (cf. Fig. 2b) computes the current control u_t as a sum of the proportional τ_p and the derivative τ_d control torque according to

$$u_t = \tau_p + \tau_d. \quad (15)$$

In [4], a feedback controller is presented, which uses the matrix logarithm mapped into vector space to expose the geodesic, i.e., the shortest path on $SO(3)$, as the error term. This approach has been adopted by [1, 15], who additionally show the stability of proportional derivative controllers using such an error term. Analogously, the proportional control term can be computed according to a weighted error function based on the \boxminus -operator, resulting in

$$\tau_p = P \delta = P(R_{da} \boxminus \tilde{R}_{ba}), \quad (16)$$

where P is the proportional gain matrix. The error function calculates the difference $\delta \in \mathbb{R}^3$ between the desired $R_{da} \in SO(3)$ and the current estimated attitude $\tilde{R}_{ba} \in SO(3)$.

To implement an actuator saturation prevention mechanism [3, 7] based on the proportional term, we observe that δ is guaranteed to never exceed $|\pi|$ in its components. This is asserted, as the error function in Eq. (16) returns the vector representation of the geodesic, whose magnitude can never be greater than π . We initially set $Q = \text{diag}(\bar{u}_1/\pi, \bar{u}_2/\pi, \bar{u}_3/\pi)$, where \bar{u}_i indicates the maximum allowed torque per axis, to ensure a bounded maximum value on every axis. For tuning purposes, this value can be scaled with the factor $K = \text{diag}([0, 1], [0, 1], [0, 1])$ (cf. Sect. 5). Thus, we let $P = KQ$, where $K_{ii} = 0$ puts all weight on the derivative term, and $K_{ii} = 1$ puts all weight on the proportional term of the respective axis i .

The derivative term is the negative output of the nonlinearity σ , which acts upon the estimated angular velocity $\tilde{\omega}$

$$\tau_d = -\sigma(\tilde{\omega}). \quad (17)$$

The nonlinearity applies upper and lower bounds to the estimated angular velocity $\tilde{\omega}$ [10]

$$\sigma(\tilde{\omega}) = [\sigma_1(\tilde{\omega}_1), \sigma_2(\tilde{\omega}_2), \sigma_3(\tilde{\omega}_3)]^T \quad (18)$$

$$\sigma_i(\tilde{\omega}_i) = \begin{cases} \bar{\omega}_i & \tilde{\omega}_i > \bar{\omega}_i \\ \tilde{\omega}_i & -\bar{\omega}_i \leq \tilde{\omega}_i \leq \bar{\omega}_i, \\ -\bar{\omega}_i & \tilde{\omega}_i < -\bar{\omega}_i, \end{cases} \quad i = 1, 2, 3 \quad (19)$$

where the bounding term $\bar{\omega}_i$, $i = 1, 2, 3$ is computed as the difference between the maximum allowed torque and the maximal value the scaled error term might assume, i.e., $\bar{\omega}_i = \bar{u}_i - P_{ii}\pi$. The indirect dependence on the choice of K ensures that the tuned control u_t never exceeds the maximum allowed torque \bar{u} .

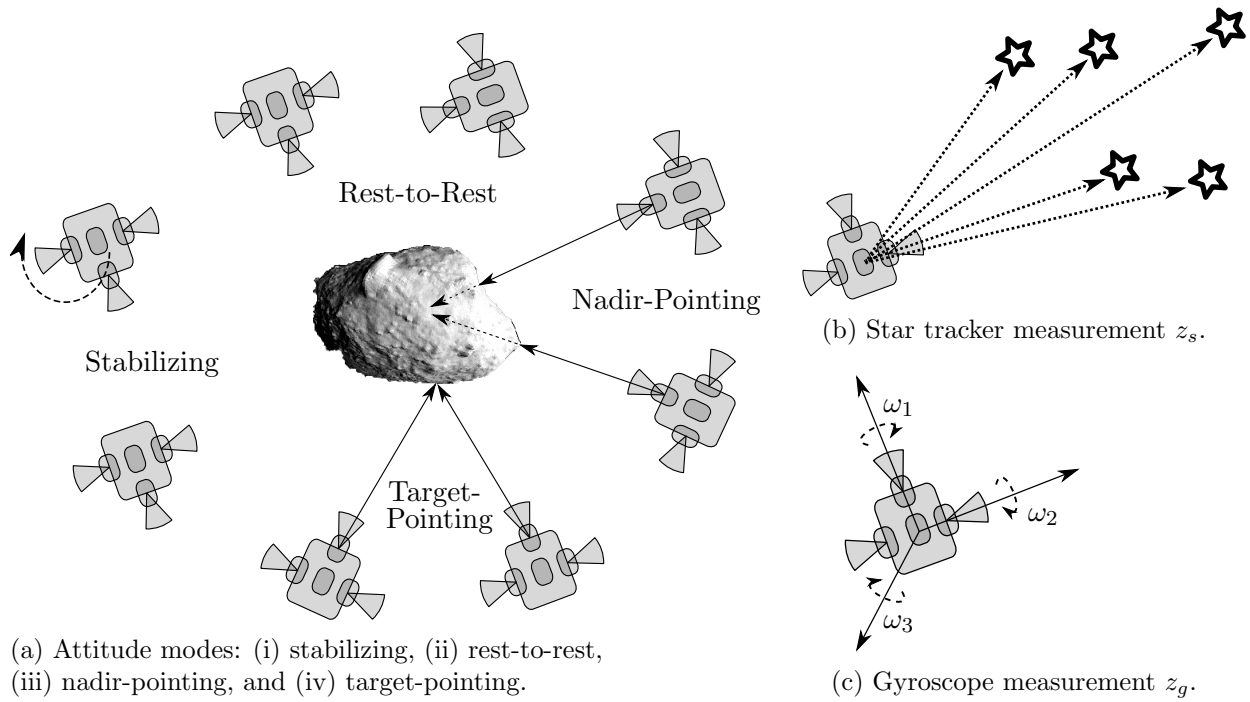


Figure 3: (a) s/c-attitude modes exemplified around the Itokawa asteroid, (b) gyroscope measurement, and (c) star tracker measurement geometry.

5. Evaluation

We conducted the evaluation in a custom written software, which simulates the solar system bodies as well as rigid body kinematics and dynamics using a fourth order Runge-Kutta integrator. The attitude propagation is based on Eqs. (10) and (11), where the inertia tensor is $I = \text{diag}(I_1, I_2, I_3)$, with $I_1 = 3305.49 \text{ kgm}^2$, $I_2 = 4653.63 \text{ kgm}^2$, and $I_3 = 4814.78 \text{ kgm}^2$. For evaluation, we chose an orbit around an asteroid with the gravitational parameter $2.342520648 \text{ m}^3 \text{ s}^{-2}$, which is orbiting with a semi-major axis of 3 AU around the Sun. The orbital elements defining the orbit of the s/c around the asteroid are: semi-major axis $0.00000000235 \text{ AU} \approx 0.34 \text{ km}$, inclination 0.0° , eccentricity 0.1 , ascending node 0.0° , periapsis 114.6° , and longitude 57.3° . The simulation operated with a timedelta $dt = 60 \text{ s}$ per step. The initial attitude in the evaluation is $R_{ba}(0) = I$ in the **T1/T2** scenarios and varied in the **T3** scenario. Of the possible maneuvers shown in Fig. 3a, we evaluate the following ones under different conditions:

- T1** *Rest-to-rest*: From a resting state with $\omega(0) = 0$, we try to reach a constant setpoint R_{da} .
- T1A** The setpoint (SP) is close to the initial state $R_{da} = I \boxplus [3^\circ, 3^\circ, 3^\circ]^T$, and
- T1B** the setpoint is exactly 180° rotated with respect to the initial state $R_{da} = \text{diag}(-1, -1, 1)$.
- T2** *Stabilization*: From an initial angular velocity $\omega(0) = [\pi/2, \pi/2, \pi/2]^T$, we try to reach a constant setpoint R_{da} .
- T2A** Again, the setpoint is close to the initial state $R_{da} = I \boxplus [3^\circ, 3^\circ, 3^\circ]^T$, and
- T2B** the setpoint is exactly 180° rotated with respect to the initial state $R_{da} = \text{diag}(-1, -1, 1)$.
- T3** *Nadir-pointing*: From a resting state $\omega(0) = 0$, we try to follow a dynamic setpoint $R_{da}(t)$, i.e., we want to orient our sensors towards the center of mass of the celestial object to be observed. As the setpoint is determined by the position of the s/c and the asteroid, we now change the initial attitude of the s/c to achieve the desired initial position deltas.

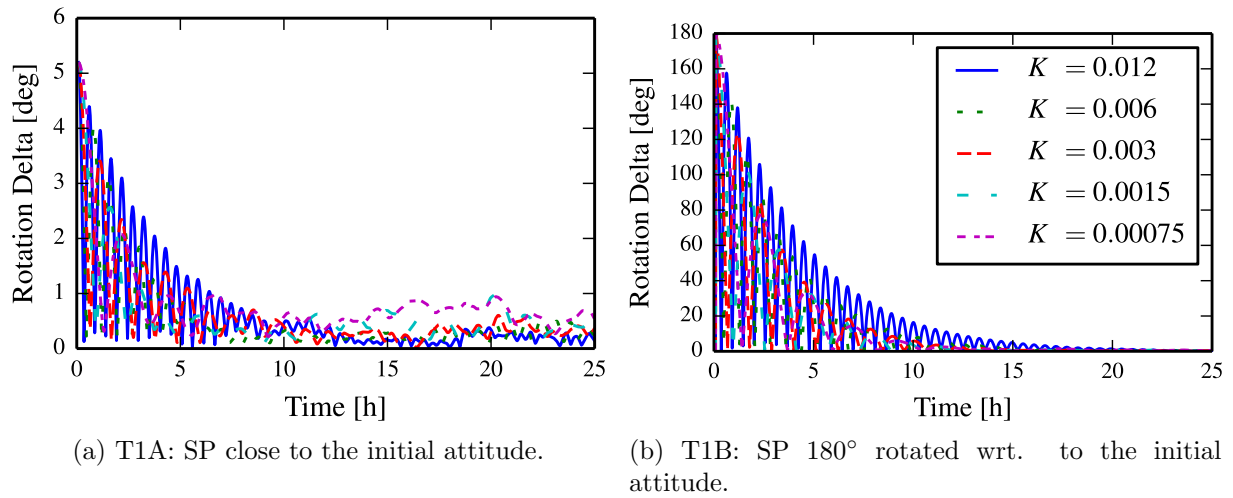


Figure 4: T1A/B: 2-norm of rotation deltas for rest-to-rest maneuvers. (Note the different scalings of the Y-axes.)

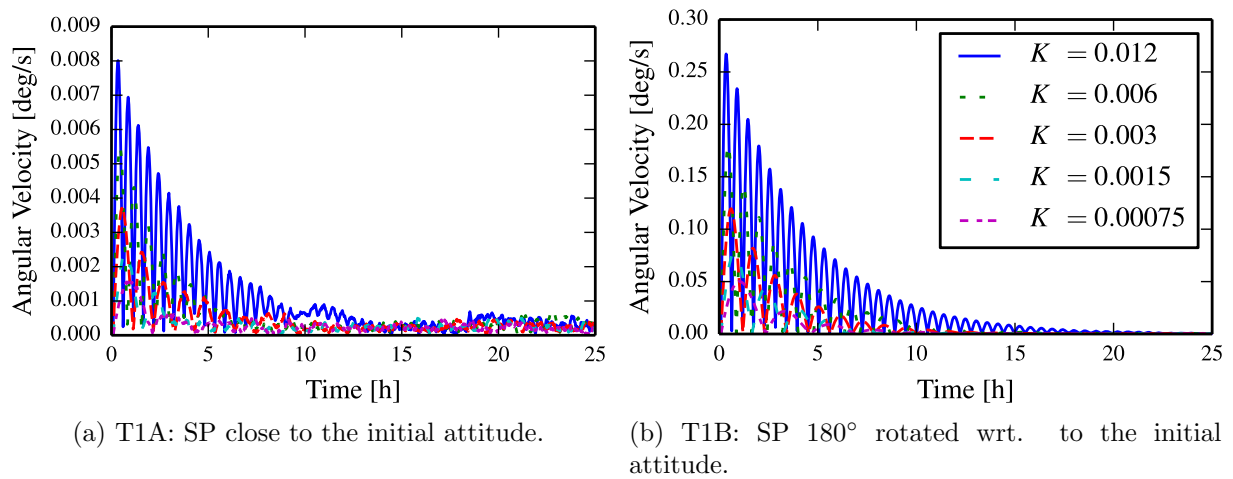


Figure 5: T1A/B: 2-norm of angular velocities for rest-to-rest maneuvers. (Note the different scalings of the Y-axes.)

T3A Here, the initial state is modified to differ from the first setpoint setting $R_{ba}(0) = R_{da}(0) \boxplus [3^\circ, 3^\circ, 3^\circ]^T$, and

T3B the initial state is exactly 180° rotated with respect to the initial setpoint $R_{ba}(0) = R_{da}(0) \text{diag}(-1, -1, 1)$.

We evaluated these scenarios with different gain scaling factors $K_{ii} = 0.00075, 0.0015, 0.003, 0.006, 0.012$, applied uniformly over all axes i . For simplicity, the factor will just be denoted K in the plots in Figs. 4–10. The rotation deltas in Figs. 4–10 are computed in every epoch t as the 2-norm of the current attitude error in vector space

$$\Delta R(t) = \|R_{ba}(t) \boxminus R_{da}(t)\|. \quad (20)$$

Thus, the error states the angle, which has to be carried out about a unique rotation axis to rotate $R_{ba}(t)$ onto $R_{da}(t)$. Similarly, the angular velocities and the control torques are shown in the plots as the 2-norm over all axes.

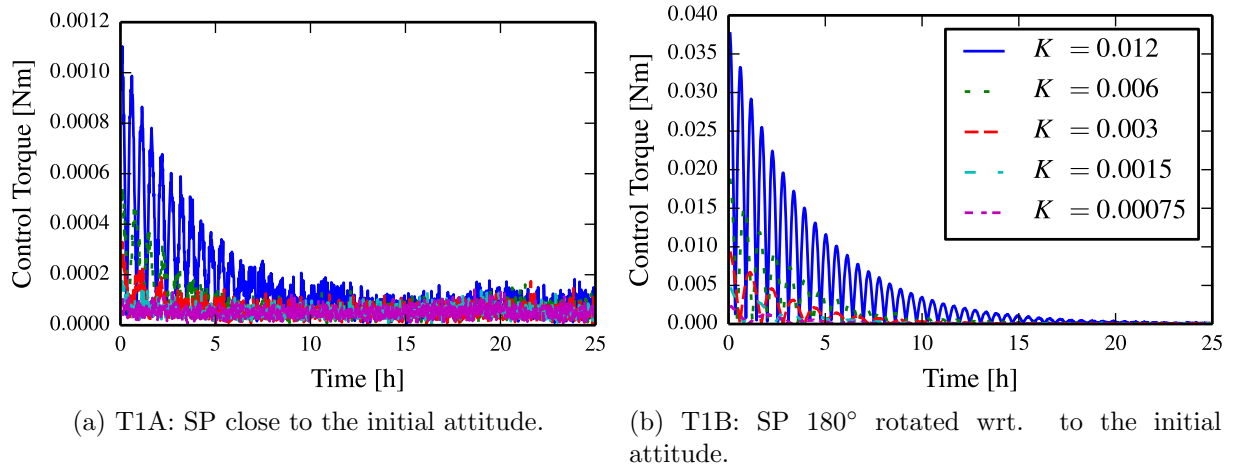


Figure 6: T1A/B: 2-norm of spent control torques for rest-to-rest maneuvers. (Note the different scalings of the Y-axes.)

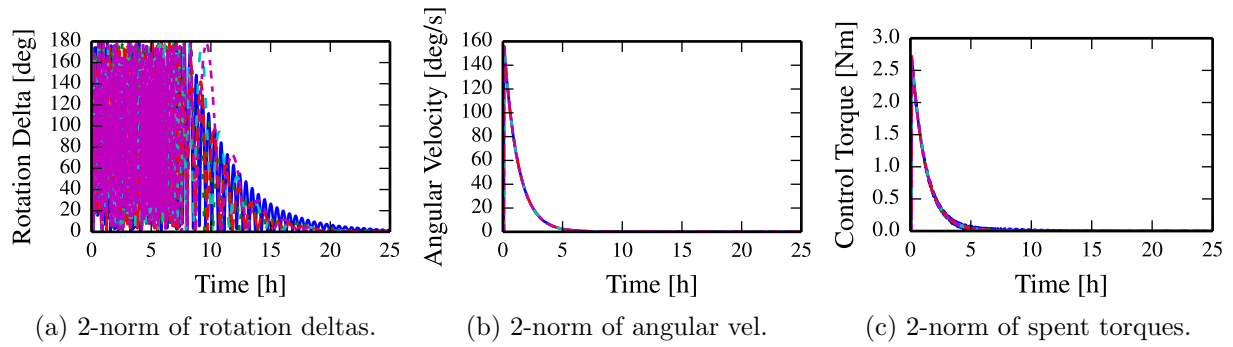


Figure 7: T2A: Stabilizing with a setpoint close to the initial attitude. (Note that the error is bounded by 180°.)

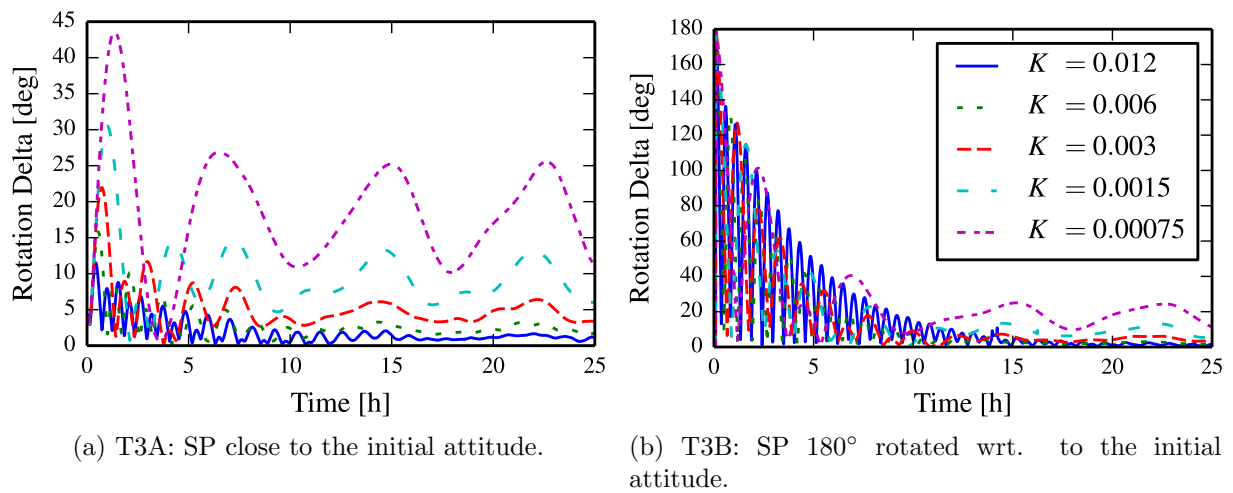


Figure 8: T3A/B: 2-norm of rotation deltas for nadir-pointing maneuvers. (Note the different scalings of the Y-axes.)

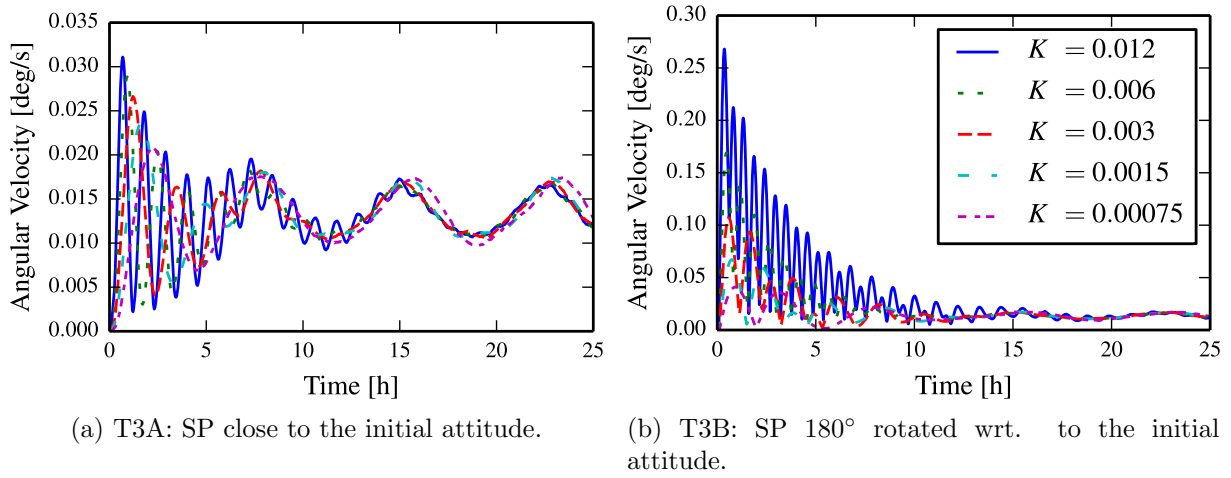


Figure 9: T3A/B: 2-norm of angular velocities for nadir-pointing maneuvers. (Note the different scalings of the Y-axes.)

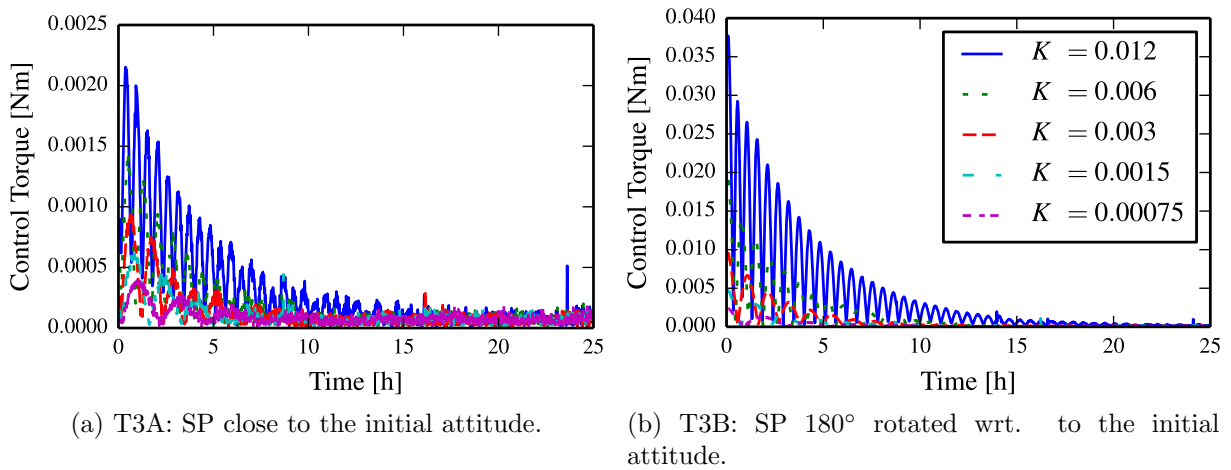


Figure 10: T3A/B: 2-norm of spent control torques for nadir-pointing maneuvers. (Note the different scalings of the Y-axes.)

In the first scenario T1A, the controller is able to reach the setpoint within a range $< 1^\circ$ in roughly 6h (cf. Fig. 4). The residual error is due to the state transition and measurement noise, which cannot be entirely smoothed by the filter. As expected, a small K is not able to keep track of the setpoint, while a higher K shows better results at the cost of more spent torque (cf. Fig. 6). In the second scenario T1B, the controller shows its robustness by stabilizing at the setpoint in about 15h (cf. Fig. 5). In T1A and T1B, the gain factor has a similar proportional impact on the control torque although the error starts from a much higher level (cf. Fig. 6).

The scenarios T2A and T2B show that the controller successfully stabilizes the s/c from tumbling after 20h (cf. Fig. 7). Starting from a high error the gain factor K is not that important anymore. The controller neatly decelerates over all axes and decreases the spent torque regardless of the gain factor (cf. Figs. 7b and 7c). We do not show plots of T2B here, as the initial pose does not have a high impact compared to the high initial angular velocity. Thus, the results are qualitatively equal to those of T2A. The gain factor $K = 0.006$ seems to be preferable over $K = 0.012$ for the T1 / T2 scenarios, as it yields a comparable performance while applying

significantly less control torque.

In the nadir-pointing scenarios T3A and T3B, the gain factor becomes important again (cf. Figs. 8 and 9). It is obvious that a stiff controller is needed for such a demanding maneuver, as the weaker K are not able to reach the setpoint, neither in condition A (cf. Fig. 8a) nor in condition B (cf. Fig. 8b). In this scenario, we furthermore note a constant residual angular velocity, which is essential to follow the dynamic setpoint (cf. Figs. 9a and 9b). The controllers with a higher gain are able to apply higher control torques (cf. Fig. 10), and thus are able to follow the moving setpoint. Hence, for scenario T3 a gain factor of $K = 0.012$ seems to be preferable over $K = 0.006$ as it (i) approaches the dynamic setpoint faster and (ii) is able to maintain a slightly lower residual error.

6. Conclusion

This paper presented a proportional derivative controller working on a manifold encapsulation of direction cosine matrices. The controller uses particle filtered angular velocity and attitude estimates to compute control torques. The proportional term is calculated from the weighted error between two attitudes on the manifold $SO(3)$ which is mapped to the vector space \mathbb{R}^3 . The derivative term is bounded by a nonlinearity, which depends on the proportional gain to prevent actuator saturation. In a numerical evaluation, we examined its behavior in different maneuvers: (i) stabilizing, (ii) rest-to-rest, and (iii) nadir-pointing. The controller exhibits stable behavior near the setpoint and from instability-prone initial attitudes. Furthermore, it is able to stabilize a s/c rotating about all three axes and can be used for nadir-pointing maneuvers, which involve a dynamic setpoint.

For future work, a combination with an information driven selection [18] of the setpoint with the control approach presented here is of high interest. Furthermore, the controller can be combined with a SLAM-approach [6] to examine possible landing sites by actively reducing uncertainty. This would result in a highly dynamic setpoint, which is not only determined by decisions as well as the motion of the autonomous system, but also by the motion of the observable itself. However, achieving this would allow autonomous systems to actively select dynamic observables and directly come up with a corresponding control.

Acknowledgments

This work was supported by the German Aerospace Center (DLR) with financial means of the German Federal Ministry for Economic Affairs and Energy (BMWi), project “KaNaRiA” (grant No. 50 NA 1318) and project “EnEx-CAUSE” (grant No. 50 NA 1505).

References

- [1] S Bharadwaj, M Osipchuk, K D Mease, and F C Park. Geometry and inverse optimality in global attitude stabilization. *Journal of Guidance, Control, and Dynamics*, 21(6):930–939, 1998.
- [2] S P Bhat and D S Bernstein. A topological obstruction to continuous global stabilization of rotational motion and the unwinding phenomenon. *Systems & Control Letters*, 39(1):63–70, 2000.
- [3] J D Bošković, S Li, and R K Mehra. Robust adaptive variable structure control of spacecraft under control input saturation. *Journal of Guidance, Control, and Dynamics*, 24(1):14–22, 2001.
- [4] F Bullo and R M Murray. Proportional derivative (pd) control on the euclidean group. In *European Control Conference*, volume 2, pages 1091–1097, 1995.
- [5] N A Chaturvedi, A K Sanyal, and H N McClamroch. Rigid-body attitude control. *IEEE Control Systems*, 31(3):30–51, 2011.
- [6] J Clemens, T Reineking, and T Kluth. An evidential approach to SLAM, path planning, and active exploration. *International Journal of Approximate Reasoning*, 73:1–26, 2016.

- [7] A H J De Ruiter. Adaptive spacecraft attitude control with actuator saturation. *Journal of Guidance, Control, and Dynamics*, 33(5):1692–1696, 2010.
- [8] A Doucet, N de Freitas, and N Gordon. *Sequential Monte Carlo methods in practice*. Springer Science & Business Media, 2001.
- [9] N Fairfield, G Kantor, and D Wettergreen. Real-time SLAM with octree evidence grids for exploration in underwater tunnels. *Journal of Field Robotics*, 24(1):3–22, 2007.
- [10] J R Forbes. Direction-cosine-matrix-based attitude control subject to actuator saturation. *IET Control Theory & Applications*, 9(11):1653–1661, 2015.
- [11] G González Peytaví, J Clemens, D Nakath, A Probst, R Förstner, K Schill, and B Eissfeller. Autonomous orbit navigation for a mission to the asteroid main belt. In *Proceedings of the 66th International Astronautical Congress, Jerusalem, Israel*. International Astronautical Federation, 2015.
- [12] R Hartley, J Trumpf, Y Dai, and H Li. Rotation averaging. *International Journal of Computer Vision*, 103(3):267–305, 2013.
- [13] C Hertzberg, R Wagner, U Frese, and L Schröder. Integrating generic sensor fusion algorithms with sound state representations through encapsulation of manifolds. *Information Fusion*, 14(1):57–77, 2013.
- [14] A Iserles, H Z Munthe-Kaas, S P Nørsett, and A Zanna. Lie-group methods. *Acta Numerica 2000*, 9:215–365, 2000.
- [15] T Lee. Exponential stability of an attitude tracking control system on $SO(3)$ for large-angle rotational maneuvers. *Systems & Control Letters*, 61(1):231–237, 2012.
- [16] C G Mayhew, R G Sanfelice, and A R Teel. On quaternion-based attitude control and the unwinding phenomenon. In *Proceedings of the 2011 American Control Conference*, pages 299–304. IEEE, 2011.
- [17] R M Murray, Z Li, and S S Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1994.
- [18] D Nakath, C Rachuy, J Clemens, and K Schill. Optimal rotation sequences for active perception. In *Proc. SPIE*, volume 9872, pages 987204-1–987204-13, 2016.
- [19] M A C Perryman, L Lindegren, J Kovalevsky, E Hoeg, U Bastian, P L Bernacca, M Crézé, F Donati, M Grenon, M Grewing, et al. The HIPPARCOS catalogue. *Astronomy and Astrophysics*, 323, 1997.
- [20] A Probst, G González Peytaví, D Nakath, A Schattel, C Rachuy, P Lange, J Clemens, M Echim, V Schwarting, A Srinivas, K Gadzicki, R Förstner, B Eissfeller, K Schill, C Büskens, and G Zachmann. KaNaRiA: Identifying the challenges for cognitive autonomous navigation and guidance for missions to small planetary bodies. In *Proceedings of the 66th International Astronautical Congress, Jerusalem, Israel*. International Astronautical Federation, 2015.
- [21] S Thrun, W Burgard, and D Fox. *Probabilistic Robotics*. MIT Press, Cambridge, MA, 2005.
- [22] B Wie and P M Barba. Quaternion feedback for spacecraft large angle maneuvers. *Journal of Guidance, Control, and Dynamics*, 8(3):360–365, 1985.