On One Method for Constructing a Programmed Trajectory in Problems of Stabilizing a Spacecraft

Alexey Lamotkin¹,*, Natalia Misura¹, and Evgenij Mityushov¹

¹Ural Federal University, 620002, Mira str., 19, Ekaterinburg, Russia

Abstract. A method is proposed for constructing a programmed trajectory in the problems of stabilizing a spacecraft in a fixed time. The method allows constructing a programmed motion, because of which the spacecraft is stabilized in rotational motion around the axis of dynamic symmetry, and this axis is oriented along the local vertical. The proposed algorithm is based on the possibility of representing the unit sphere of quaternions in four-dimensional space as a ball with a radius in three-dimensional space. This makes it possible to obtain a visual representation of the spacecraft rotation trajectory.

1 Introduction

Currently, small spacecraft are widely used to create multi-satellite orbital constellations. Such groups are used in remote sensing for monitoring emergency situations, as well as for educational and scientific purposes [1]. Due to the size of small spacecraft, there is no possibility of installing a thermoregulation system, which makes it necessary to apply various kinds of twists to it, which would provide uniform illumination of its surface by the Sun [2]. The authors consider one of the variants of such a spin, in which the axis of dynamic symmetry of the spacecraft is stabilized along the local vertical, and the spacecraft itself performs a slow rotation around this axis. Such a twist allows providing uniform illumination of the spacecraft surface, while maintaining optimal capabilities for remote sensing of the Earth [3].

The paper presents an algorithm for constructing a programmed motion, which allows one to perform the above stabilization from an arbitrary position of the spacecraft. To construct a programmed motion, the mapping of unit quaternions into a ball of radius c is used. The trajectory of rotation in this ball is described by a vector polynomial and satisfies the boundary conditions for the positions and angular velocities of the spacecraft. The resulting programmed trajectory is close to the trajectory of the Euler (flat) rotation, which endows it with some “optimality” properties.

At the end of the work, an example of stabilization of a spacecraft performing regular precession in the orbital coordinate system is given.

* Corresponding author: Alexey.Lamotkin@urfu.ru
2 Formulation of the problem

Consider a small spacecraft and associate with it a coordinate system $Ox'y'z'$, whose axes coincide with the main central axes of inertia. The axis $Oz'$ is the axis of the dynamic symmetry of this spacecraft. The angular position of the introduced coordinate system is determined relative to the orbital coordinate system $Ox'yz'$, with the axis $Ox$ directed in the direction of motion, the axis $Oz$ directed along the local vertical to the center of the Earth, and the axis $Oy$ of this coordinate system to the right.

Let at the initial moment of time

$$q(0) = q_0, \quad \Omega(0) = \Omega_0,$$

where $q$ is the quaternion specifying the angular position of the spacecraft, $\Omega$ is the vector of the spacecraft's angular velocity specified in the associated axes.

It is required to construct the programmed motion of the spacecraft that stabilizes the programmed rotation of the small spacecraft in the orbital coordinate system for a given fixed stabilization time $T$ in such a way that the axis of dynamic symmetry coincides with the local vertical, and the spacecraft itself performs slow rotation around this axis:

$$q(T) = q_r, \quad \Omega(T) = \Omega_r.$$

As you know, Euler (flat) rotations define geodesics on the surface of the unit sphere of quaternions $S^3: q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$, which makes them "optimal" in a sense. Therefore, we require that the desired trajectory be close to the Euler trajectory, which can be described using spherical linear interpolation of quaternions [4]:

$$q_c(t) = \frac{\sin\left(\arccos(q_0 \cdot q_r)\left(1 - \frac{t}{T}\right)\right)}{\sin\left(\arccos(q_0 \cdot q_r)\right)} \cdot q_0 + \frac{\sin\left(\arccos(q_0 \cdot q_r)\frac{t}{T}\right)}{\sin\left(\arccos(q_0 \cdot q_r)\right)} \cdot q_r.$$

The fulfillment of this requirement can be ensured by the presence of common intermediate points for the programmed trajectory $q(t)$ and the Euler trajectory $q_c(t)$:

$$q\left(\frac{iT}{n+1}\right) = q_c\left(\frac{iT}{n+1}\right), \quad i = 1, \ldots, n.$$

3 Building a program trajectory

To construct a programmed trajectory in the four-dimensional space of quaternions, which satisfies the listed conditions, we will use its image in three-dimensional space. As noted in [5], the set of unit quaternions defining the orientation can be associated with the points of the ball with the radius $\pi$ of the three-dimensional space, taking this into account the problem of finding the program trajectory $q(t)$ can be replaced by the problem of finding its three-dimensional image $r(t)$ in the ball with the radius $\pi$. Connection between the coordinates of the points of the ball and components of the quaternion was established in [6]:

$$
\[
q_0(t) = \cos \frac{\sqrt{x_1^2(t) + x_2^2(t) + x_3^2(t)}}{2}, \\
q_1(t) = \frac{x_1(t)}{\sqrt{x_1^2(t) + x_2^2(t) + x_3^2(t)}} \sin \frac{\sqrt{x_1^2(t) + x_2^2(t) + x_3^2(t)}}{2}, \quad k = 1, 2, 3. 
\]

from where

\[
x_1(t) = \frac{2q_1(t)\arccos q_0}{\sqrt{1 - q_0^2}}, \quad k = 1, 2, 3.
\]

We will look for the image of the program trajectory \( r(t) \) in the form of a polynomial of degree \( n + 3 \), since this is the minimum degree of a polynomial that allows all of the above conditions to be satisfied:

\[
r = \sum_{k=0}^{n+3} a_k t^k. 
\]

In this case, the following conditions must be met:

\[
r \left( \frac{iT}{n+1} \right) = \\
\begin{cases} 
2q_{i,1} \left( \frac{iT}{n+1} \right) \arccos \frac{iT}{n+1} \left( \frac{iT}{n+1} \right) \\
\sqrt{1 - \left( \frac{iT}{n+1} \right)^2} \\
\end{cases}, \quad i = 0, \ldots, n + 1,
\]

\[
r(0) = \\
\begin{cases} 
2q_{0,1} \left( \frac{\arccos q_0(0)}{1 - (q_0(0))^2} \right) \left( \frac{1}{1 - (q_0(0))^2} \right) \dot{q}_0(0) + \frac{1}{2} \frac{\arccos q_0(0)}{\sqrt{1 - (q_0(0))^2}} \dot{q}_1(0) \\
2q_{0,2} \left( \frac{\arccos q_0(0)}{1 - (q_0(0))^2} \right) \left( \frac{1}{1 - (q_0(0))^2} \right) \dot{q}_0(0) + \frac{1}{2} \frac{\arccos q_0(0)}{\sqrt{1 - (q_0(0))^2}} \dot{q}_2(0) \\
2q_{0,3} \left( \frac{\arccos q_0(0)}{1 - (q_0(0))^2} \right) \left( \frac{1}{1 - (q_0(0))^2} \right) \dot{q}_0(0) + \frac{1}{2} \frac{\arccos q_0(0)}{\sqrt{1 - (q_0(0))^2}} \dot{q}_3(0)
\end{cases},
\]
\[
\mathbf{r}(T) = \begin{cases}
2q_{x}(T)\left(\frac{q_{o}(T)\arccos q_{o}(T)}{1-(q_{o}(T))^2} - \frac{1}{1-(q_{o}(T))^2}\right)\dot{q}_{o}(T) + \frac{1}{2\sqrt{1-(q_{o}(T))^2}} \dot{q}_{1}(T), \\
2q_{y}(T)\left(\frac{q_{o}(T)\arccos q_{o}(T)}{1-(q_{o}(T))^2} - \frac{1}{1-(q_{o}(T))^2}\right)\dot{q}_{o}(T) + \frac{1}{2\sqrt{1-(q_{o}(T))^2}} \dot{q}_{2}(T), \\
2q_{z}(0)\left(\frac{q_{o}(T)\arccos q_{o}(T)}{1-(q_{o}(T))^2} - \frac{1}{1-(q_{o}(T))^2}\right)\dot{q}_{o}(T) + \frac{1}{2\sqrt{1-(q_{o}(T))^2}} \dot{q}_{3}(T),
\end{cases}
\]

where

\[
\dot{q}_{o} = -\frac{1}{2}(q_{o}(t)\Omega_{1}(t) + q_{x}(t)\Omega_{2}(t) + q_{z}(t)\Omega_{3}(t)), \\
\dot{q}_{1} = \frac{1}{2}(q_{o}(t)\Omega_{1}(t) - q_{x}(t)\Omega_{2}(t) + q_{z}(t)\Omega_{3}(t)), \\
\dot{q}_{2} = \frac{1}{2}(q_{o}(t)\Omega_{2}(t) - q_{x}(t)\Omega_{1}(t) + q_{z}(t)\Omega_{3}(t)), \\
\dot{q}_{3} = \frac{1}{2}(q_{o}(t)\Omega_{3}(t) - q_{x}(t)\Omega_{1}(t) + q_{z}(t)\Omega_{2}(t)).
\]

Therefore, we can write the system

\[
\begin{align*}
\mathbf{a}_{i} &= \mathbf{r}(0), \\
\sum_{k=0}^{n+3} \mathbf{a}_{i}kT^{k-1} &= \mathbf{r}(T), \\
\mathbf{a}_{0} &= \mathbf{r}(0), \\
\sum_{k=0}^{n+3} \mathbf{a}_{i}\left(\frac{iT}{n+1}\right)^{k-1} &= \mathbf{r}\left(\frac{iT}{n+1}\right), \quad i = 1,\ldots,n+1,
\end{align*}
\]

from which the coefficients of the image (2) are uniquely determined. After that, using relations (1), you can determine the programmed trajectory \(q(t)\). The law of variation of the angular velocity can be found from the relations:

\[
\begin{align*}
\Omega_{1} &= 2\left(\dot{q}_{x}q_{o} - \dot{q}_{o}q_{x} - \dot{q}_{z}q_{2} + \dot{q}_{z}q_{3}\right), \\
\Omega_{2} &= 2\left(\dot{q}_{z}q_{o} - \dot{q}_{o}q_{z} - \dot{q}_{x}q_{3} + \dot{q}_{x}q_{1}\right), \\
\Omega_{3} &= 2\left(\dot{q}_{x}q_{o} - \dot{q}_{o}q_{x} - \dot{q}_{z}q_{1} + \dot{q}_{z}q_{2}\right).
\end{align*}
\]

The programmed control, realizing (3) can be found from the dynamic Euler equations.

### 4 Example

Let us consider the stabilization of a small spacecraft making a regular precession in the orbital reference frame:

\[
q_{0} = (0,0,-0.131,0.991), \quad \Omega_{0} = (0,-0.831,3.86),
\]
after stabilization through $T = 10$ s, it moves to the position at which the associated axes coincide with the orbital coordinate system, and continues to slowly rotate around the axis of dynamic symmetry:

$$q_r = (0.999999, 0, 0, 0), \quad \Omega_r = (0, 0, -0.2).$$

Figure 1 shows the graphs of changes in the quaternion components in the case of motion along the Euler trajectory $q_e(t)$ and the programmed trajectory $q(t)$ (for $n = 1, 5, 20$). It can be seen from these graphs that with an increase, the program trajectory approaches the Euler.

![Graphs of changes in quaternion components](image1.png)

(a) (b) (c) (d)

**Fig. 1.** a) Changing in the quaternion components $q_e(t)$, (b) Changing in the quaternion $q(t)$ components at $n = 1$, (c) Changing in the quaternion $q(t)$ components at $n = 5$, (d) Changing in the quaternion $q(t)$ components at $n = 20$.

### 5 Conclusion

The paper presents a method for constructing a programmed trajectory for stabilization of a small spacecraft, as a result of which the spacecraft is stabilized in rotational motion around the axis of dynamic symmetry, and this axis is oriented along the local vertical. The resulting trajectory can be, with a sufficient degree of accuracy, approximated to the trajectory of a uniform Euler rotation, which endows it with a certain “optimality” property.

The mapping of quaternions into three-dimensional space, used in the method, makes it possible to obtain a visual representation of the resulting trajectories, which can be useful in solving problems of controlling the turn of a rigid body.
References

1. I. N. Ganswindt *Small spacecraft – new directions in space activities*, International research journal, 12(78), pp. 82-91 (2018)

2. S. Yu. Chebukov *Investigation of the biaxial dynamics of spinning a satellite in the orbital plane*. Ph.D. thesis, Keldysh Institute of Applied Mathematics (1998)

3. A. Yu. Aleksandrov and A. A. Tikhonova *Electrodynamic stabilization of satellite programmed rotation in the orbital coordinate system*, Saint Petersburg University Bulletin (series 1) 2, pp. 79-90 (2012)

4. K. Shoemake *Animating rotation with quaternion curves*, Proceedings of the 12th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH85 (1985)

5. V. I. Arnold *Geometry of complex numbers, quaternions and spin* (MCCME, Moscow, 2002)

6. A. E Lamotkin, N. E. Misyura and E. A Mityushov *Designing the program trajectory for steering a spacecraft under arbitrary boundary conditions*, IOP Conference Series: Materials Science and Engineering, (2020)