JSST Journal of Space Sience and Technology

Vol. 16 / Special Issue / 2023 / (No. 58)
Print ISSN: 2008-4560 / Online ISSN: 2423-4516
https://doi.org/10.22034/jsst.2023.1421

OPEN ACCESS

Pages: 51-64 / Research Paper / Received: 07 October 2022 / Revised: 20 December 2022 / Accepted: 01 February 2023

Journal Homepage: https://jsst.ias.ir

Attitude Control of Nanosatellites Using Optimal Quadratic Linear and Nonlinear Methods

Arash Abarghooei[1*], Hassan Salarieh[2] and Pedram Hosseiniakram[3]

1. Ph.D. Candidate, Mechanical Engineering Department, Sharif University of Technology, Tehran, Iran

2. Professor, Mechanical Engineering Department, Sharif University of Technology, Tehran, Iran

3. M.Sc., Amirkabir University of Technology, Tehran, Iran

*Corresponding Author's E-mail: a.abarghuee@gmail.com

Abstract

Linear algorithms are the most widely used method for satellite attitude control using reaction wheels because of their simplicity and low computational cost. The first part of the paper introduces different attitude determination and control algorithms, and reviews resources that utilized optimal linear and nonlinear control methods (such as LQR and SDRE). Next, dynamic equations for the control of the satellite using reaction wheels have been extracted, then the satellite controller has been designed by using optimal linear and nonlinear methods, which are robust against noise and disturbance, as an alternative for the PD controller. Finally, the designed control algorithms have been implemented for different satellite pointing scenarios, and by simulating these methods in MATLAB software, their performance has been studied and compared.

Keywords: Nanosatellite, Satellite control algorithms, Reaction wheel, Optimal control, LQR, SDRE

Introduction

The satellite attitude determination and control subsystem (ADCS) consists of three main parts including monitoring, estimation and control system. The overall workflow begins with the monitoring system. Based on the information collected from all satellite sensors, and predefined tasks, the ACDS performs the necessary commands to begin and end a control algorithm.

The control system has two main tasks including control of angular velocity and pointing toward the desired target. Nanoscale satellites generally use actuators such as reaction wheels and magnetometers for this purpose.

The dynamic system of satellites is nonlinear and Quaternion Proportional Derivative (PD) feedback is commonly used to control reaction wheels [1-3]. Although some research proposed model-free optimal control algorithms [4], most of the optimization algorithms are used to adjust PD coefficients; and the linear quadratic method is one of the most widely used methods for this purpose [5].

In this method, in order to calculate the optimal control coefficients, a cost function is defined as (1).

$$J = \int_0^\infty (X^T Q X + U^T R U) dt \qquad (1)$$

In this equation, Q and R are real and symmetric matrices. R is a definite positive matrix (all its eigenvalues are positive) and Q is a positive semi-definite matrix (all its eigenvalues are non-negative). Matrix Q represents the weight and cost of the tracking error or deviation from the desired condition and R represents the weight and importance of energy consumption for the control actuator. These two matrices are determined according to the nature of the system, performance requirements, cost constraints as well as the maximum actuators' capacity. The Linear–Quadratic Regulator (LQR) method is a suitable approach to attitude control of the satellite in order to optimize energy consumption,

How to cite this article:
A. Abarghooei, H. Salarieh, P. Hosseiniakram, "Attitude Control of Nanosatellites Using Optimal Quadratic Linear and Nonlinear Methods," *Journal of Space Science and Technology*, Vol. 16, Special Issue, pp. 51-64, 2023, https://doi.org/10.22034/jsst.2023.1421.

control time and effort, and obtain the optimal gain matrix. In this method, the goal is to find a linear control command in the form of $U = -KX$ so that the mentioned cost function becomes minimized. The Q and R matrices can be achieved by compromising between performance requirements and cost constraints.

The structure of the State-Dependent Riccati Equation (SDRE) is quite similar to the LQR controller and using Q and R coefficients, the algebraic Riccati equation is formed and the control coefficient is calculated from its solution. The difference between SDRE and LQR controls is that SDRE control is used for nonlinear time-variant systems. In these systems, because of the variability of the state matrix, A, and input matrix, B, and the need for linearizing the system around the new operating point, the Riccati equation is continuously formed in the control operation, and the coefficient matrix is updated consequently. In other words, in this controller, the control coefficient is nonlinear and depends on the state variables directly.

The LQR method is suitable for systems that have an accurate model and ideal sensors and actuators. In practice, however, the effects of sensor noise and disturbances on actuators always affect control performance. In the LQG[1] method, by combining the controller and estimator, the effects of system input noise and measurement noise are considered.

LQG algorithm consists of two parts: 1. determining the control signal by estimating the state of the outputs, and 2. applying the optimal control law based on the dual principle. Each random signal in a linear system is Gaussian and is determined by its mean and variance. The Kalman filter provides a non-biased estimate with minimal variance. Combining this optimal estimate with optimal control creates an optimal closed-loop system. But estimation and control can be solved separately to avoid over-complicating of the combined problem.

In the article [6], Arefkhani and his colleagues evaluated the magnetic attitude controller using PD and LQR rules and used a three-axial satellite control simulator based on air bearing to evaluate the control performance. For this purpose, they used Eulerian angles and their derivatives as states and obtained A and B matrices using linearized equations. Then calculated control coefficients by solving the algebraic Riccati equation.

In another paper [7], Miri designed an optimal control for satellite attitude maneuvers with the control law of the second-order linear regulator. In this work, Eulerian angles and their derivatives were used as system state variables. Also, neglecting the multiplication terms, the linearization around the equilibrium point was done and the equations were simplified. Also, by performing various simulations, it was been shown that to achieve higher convergence speed by adjusting the coefficients Q and R, energy consumption is significantly increased. Also, by comparing the results with another reference in which more than three reaction wheels are used, it was shown that satellite control with more than three reaction wheels is far from optimality. Without managing the speed of the reaction wheels, their final values will not be zero at the end of the maneuver and energy consumption will continue.

In the article [8], Nikkhah used a single-frame gyroscopic actuator to attitude control a satellite in a 300 km orbit, and various control methods such as LQR and LQG are used to stabilize the attitude, and quaternion feedback is utilized for orbital maneuvering.

In another article [9], a satellite is controlled by LQR and SDRE controllers, and the effect of reaction wheels failure is investigated. In this paper, first, the satellite dynamic model is extracted based on Eulerian angles, then by linearization around the Nadir point using the Taylor series, the LQR controller with a fixed gain is designed.

In general, it is concluded that the LQR and SDRE controllers are able to stabilize the satellite if the reaction wheel in the x-direction (corresponding to the angle θ) does not fail, and it is claimed that the SDRE nonlinear controller results are better than the LQR linear controller.

In another article [10], Rodrigues modified parameters are used to express the kinematics of the satellite and the satellite control is done by designing the SDRE controller. singularity at a 360-degree angle is the main problem with using Rodrigues parameters but this article claims it does not occur in the satellite's range of motion. In order to solve the Riccati equation, Rodrigues parameters and angular velocities are considered state variables, and it was shown that matrix B is not dependent on the state in this case.

In the article [11], using the SDRE control method, satellite control with reactive wheel actuators and gas thrusters is discussed and a 3D INPE[2]simulator is used for simulations. This simulator has an air bearing equipped with three reaction wheels and gas thrusters, and three angular velocity sensors such as gyroscopes are mounted on its three coordinate directions. It is also shown that the SDRE controller can be practically implemented on satellite hardware.

The AAUsat satellite of the University of Aalborg uses an optimal LQR controller for pointing. In the report [12], it is mentioned that the reason for using the LQR method instead of a PD controller is the limitation of energy consumption in the satellite.

In the present paper, after extracting the dynamic equations of the satellite, three types of linear, and nonlinear time-invariant optimal controllers, including optimal gain PD, LQR state feedback and state-dependent Riccati state feedback will be designed and by performing various simulations, the performance of these controllers will be compared with each other. Then with the combination of quaternion and quadratic feedback

1. Linear Quadratic Gaussian

2. National Institute for Space Research

controllers, the most optimal controller for each pointing maneuver will be proposed.

Extraction of Dynamic Equations

The attitude of a rigid body is usually expressed by Euler or quaternion angles. Using quaternions simplifies equations and solves singularity problems, but increases the state variables by one and needs attention to maintain the unit length of the quaternion vector. In both kinematic methods, the orientation rate of change is present with the angular velocities of the object.

If the satellite attitude relative to the orbital coordinate system is shown by quaternion vector $q = (q_1, q_2, q_3, q_4)^T$ and angular velocity $\omega_{b/o}^{b} = (\omega_1, \omega_2, \omega_3)^T$ in the body coordinate, The state equations (quaternion changes with respect to angular velocities) are written as (2) [13]:

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 & \omega_3 & -\omega_2 & \omega_1 \\ -\omega_3 & 0 & \omega_1 & \omega_2 \\ \omega_2 & -\omega_1 & 0 & \omega_3 \\ -\omega_1 & -\omega_2 & -\omega_3 & 0 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \tag{2}$$

In some resources, when using the LQR control to avoid the singularity of the Riccati equation, the satellite attitude equation is also expressed in the form of (3) [14].

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} q_4 & -q_3 & q_2 \\ q_3 & q_4 & -q_1 \\ -q_2 & q_1 & q_4 \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} \tag{3}$$

The advantage of using this method is that the q_4 term is removed from the state variables and the quaternion error norm in the controller should approach zero.

By knowing the moment of inertia matrix I and the applied torques T, the rotation dynamic equations are obtained. The moment of inertia matrix is usually expressed in the body system. The angular velocity can be measured relative to the inertial device or orbital device. The attitude dynamic equation of a rigid satellite in space in the body system is expressed as (4).

$$I\dot{\omega}_{b/i}^{b} = -\omega_{b/i}^{b} \times I\omega_{b/i}^{b} + T^{b} \tag{4}$$

Assuming a rigid body for the satellite, the rotation dynamic equations in the main system are expressed as (5) (in which the non-diagonal expressions of the moment of inertia matrix can be neglected):

$$\begin{aligned} \dot{\omega}_1 &= \frac{1}{I_1}(T_1 - (I_3 - I_2)\omega_3\omega_2) \\ \dot{\omega}_2 &= \frac{1}{I_2}(T_2 - (I_1 - I_3)\omega_1\omega_3) \\ \dot{\omega}_3 &= \frac{1}{I_3}(T_3 - (I_2 - I_2)\omega_2\omega_1) \end{aligned} \tag{5}$$

In the above equation I_1, I_2, I_3 are the main inertia moments of the satellite and $\omega = (\omega_1, \omega_2, \omega_3)^T$ is the angular velocity vector in the body system. Also, $T = (T_1, T_2, T_3)^T$ is the torque vector on the satellite which includes control torques, definite external torques (such as gravity gradient and drag torque) and random torques (unpredictable disturbances). It is assumed that the control torques are available, the inertia matrix of the satellite is diagonal, and the effect of magnitum of reaction wheels is negligible. Gravitational and drag torques solar pressure force and torque due to the changes in the magnetic field can be determined as a function of satellite state variables. Disturbance torques are also considered as random inputs to the system. Thus, quaternion and angular velocity can be expressed as vectors of state variables to express the system state equations, as (6) [2]:

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix} q \\ \omega \end{pmatrix} = \begin{pmatrix} \frac{1}{2}\begin{bmatrix} q_4 & -q_3 & q_2 \\ q_3 & q_4 & -q_1 \\ -q_2 & q_1 & q_4 \end{bmatrix} \\ \frac{1}{I_1}(T_1 - (I_3 - I_2)\omega_3\omega_2) \\ \frac{1}{I_2}(T_2 - (I_1 - I_3)\omega_1\omega_3) \\ \frac{1}{I_3}(T_3 - (I_2 - I_2)\omega_2\omega_1) \end{pmatrix} \tag{6}$$

By choosing the state vector as $[q_1, q_2, q_3, \omega_x, \omega_y, \omega_z]^T$, the state function of the satellite system becomes in the form of (7):

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{\omega}_x \\ \dot{\omega}_y \\ \dot{\omega}_z \end{bmatrix} = A(X)\begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} + \mathrm{B}\begin{bmatrix} T_{rw}^{1} \\ T_{rw}^{2} \\ T_{rw}^{3} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{I_x}T_{dis}^{x} \\ \frac{1}{I_y}T_{dis}^{y} \\ \frac{1}{I_z}T_{dsi}^{z} \end{bmatrix} \tag{7}$$

Therefore, matrices A(X) and B are expressed as (8) and (9).

$$A(X) = \begin{bmatrix} 0_{3\times3} & \begin{matrix} \frac{1}{2}q_4 & -\frac{1}{2}q_3 & \frac{1}{2}q_2 \\ \frac{1}{2}q_3 & \frac{1}{2}q_4 & -\frac{1}{2}q_1 \\ -\frac{1}{2}q_2 & \frac{1}{2}q_1 & \frac{1}{2}q_4 \end{matrix} \\ 0_{3\times3} & \begin{matrix} 0 & -\frac{I_3}{I_1}\omega_3 & \frac{I_2}{I_1}\omega_2 \\ \frac{I_3}{I_2}\omega_3 & 0 & -\frac{I_1}{I_2}\omega_1 \\ -\frac{I_2}{I_3}\omega_2 & \frac{I_1}{I_3}\omega_1 & 0 \end{matrix} \end{bmatrix} \tag{8}$$

$$B = \begin{bmatrix} & 0_{3\times3} & \\ {I_1}^{-1} & 0 & 0 \\ 0 & {I_2}^{-1} & 0 \\ 0 & 0 & {I_3}^{-1} \end{bmatrix} \tag{9}$$

Controller Design

This paper designs and simulates the implementation of three types of controllers and compares their performance. The first controller is the PD Quaternion Feedback Controller. The control law that generates the torque of the reaction wheels is calculated from (10)[2].

$$U = -K_p \delta q_{1:3} - K_d(\omega - \omega_{des}) \tag{10}$$

In this equation, the expression $\delta q_{1:3}$, which is a quaternion error, is calculated by (11):

$$\delta q_{1:3} = q_{des} q^T = \begin{bmatrix} q_4^{des} & q_3^{des} & -q_2^{des} & -q_1^{des} \\ -q_3^{des} & q_4^{des} & q_1^{des} & -q_2^{des} \\ q_2^{des} & -q_1^{des} & q_4^{des} & -q_3^{des} \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \tag{11}$$

In this equation, q_{des} is the desired quaternion. In equation (10), K_p and K_d coefficients are obtained by trial and error, and a quadratic controller can be used to achieve optimal coefficients.

Quadratic Controller Design

For state equation in the form $X\dot{(t)} = A(t)X(t) + B(t)U(t)$ the cost function is considered as (12):

$$J = \frac{1}{2}X(t_f)^T H X(t_f) + \frac{1}{2}\int_{t_0}^{t_f} \{(X(t) - X_d(t))^T Q(t)(X(t) - X_d(t)) + U^T R(t) U\} dt \tag{12}$$

Where the matrices R, H, and Q are real and symmetric. H and Q are semi-definite positive and R is positive definite. H, which is the final spatial error coefficient, indicates the portion of the final error in the cost function J. A higher H results in a path in which the final error rate is lower. Q also indicates the portion of spatial error along the path. R represents the portion of actuators in optimization. By determining these matrices, the importance of each error or operator can be tuned.

Now the cumulative goal function of the system is obtained as (13). It should be noted that the added term that expresses dynamics is equal to zero.

$$H = \frac{1}{2}X(t_f)^T H X(t_f) + \frac{1}{2}\int_{t_0}^{t_f} \left\{ \begin{matrix} (⟦X(t) - X_d(t))⟧^T Q(t)(X(t) - X_d(t)) + \\ U^T R(t) U + \\ P^T(A(t)X(t) + B(t)U(t) - X\dot{(t)}) \end{matrix} \right\} dt \tag{13}$$

Where P is the Lagrangian coefficient for the dynamic Euler equations for spatial variables and control inputs are obtained as (14) and (15). Here * index represents the optimal values:

$$\frac{\partial \mathrm{H}}{\partial \mathrm{x}} - \frac{d}{dt}\left(\frac{\partial H}{\partial \dot{x}}\right) = 0 \rightarrow \dot{p}^*(t) = -\frac{\partial \mathrm{H}}{\partial \mathrm{x}} = -Q(t)x^*(t) - A^T(t)p^*(t) \tag{14}$$

$$\frac{\partial \mathrm{H}}{\partial \mathrm{u}} - \frac{d}{dt}\left(\frac{\partial H}{\partial \dot{u}}\right) = 0 \rightarrow \frac{\partial \mathrm{H}}{\partial \mathrm{u}} = R(t)u^*(t) + B^T(t)p^*(t) = 0 \Rightarrow u^*(t) = -R^{-1}(t)B^T(t)p^*(t) \tag{15}$$

By replacing the obtained $u^*(t)$ in the dynamic equation of the system for $\dot{x}^*(t)$ and solving the obtained equation with initial conditions as (16) and (17), the value of u^* is obtained algebraically as (18)

$$x^*(t=0) = x^*(t_0) \tag{16}$$

$$p^*(t_f) = Hx^*(t_f) \tag{17}$$

$$u^*(t) = -R^{-1}(t)B^T(t)K(t)x^*(t) \tag{18}$$

To find the K value we need to solve the below equation, which expresses the Riccati differential equation [15].

$$\dot{K}(t) = -Q(t) - K(t)A(t) - A^T(t)K(t) + K(t)B(t)R^{-1}(t)B^T(t)K(t) \tag{19}$$

The SDRE technique for the finite-horizon optimal regulator problem basically consists of representing any given dynamic system in the form of a state-dependent coefficient (SDC) and then solving the SDRE at small time intervals during the given finite-horizon period from the initial time to the final time. Therefore, to solve the optimal control problem in the SDRE method, the Riccati equation is solved at each moment, and in this finite time, the value of *K(t)* is considered constant and therefore $\dot{K}(t)$ is equal to zero [16]. Then we can drive Riccati algebra as (20) and the control law is expressed as (21).

$$Q + KA + A^T K - KBR^{-1}B^T K = 0 \tag{20}$$

$$u^* = -R^{-1}B^T K x^* \tag{21}$$

Methods for Solving the Riccati Equation

Since in an SDRE controller, the control coefficients must be recalculated at each time step by updating the state variables, it is necessary to solve the discussed Riccati equation frequently to obtain the P matrix. For this purpose, a fast algorithm is needed to solve this equation.

There are many different methods to solve the algebraic Riccati equation [17, 18]. Some of these methods are analytical and some are recursive and numerical.

Using recursive methods makes the task more complicated due to the need for an initial guess. Among the algebraic methods, the Schur method is used in most resources, so in this paper, this method is used to solve the Riccati equation depending on the state variables.

Optimal Control Simulation

In order to assess the performance of the designed controllers, a satellite attitude and transition motion simulator has been implemented in MATLAB software. In this paper, the performance of the design controllers for Nadir pointing is compared. For each pointing mode, a function is defined to calculate the desired quaternion according to the position and attitude of the satellite. Therefore, in addition to solving the attitude motion equations of the satellite, the transitional motion equations of the satellite are also solved using the Runge-Kutta numerical solution method.

In order to implement the LQR linear controller, it is necessary to linearize the dynamic equation around the nadir point, then by solving the Riccati equation obtained from the linearized matrix A, the gain matrix K can be calculated. Taylor series is used to linearize matrix A, and thus matrix A is simplified to the form (22).

$$Aa \approx \begin{bmatrix} 0_{3\times3} & & I_3 & \\ & 0 & 0 & 0 \\ 0_{3\times3} & 0 & 0 & -\frac{I_x}{I_y}0.0011 \\ & 0 & \frac{I_x}{I_z}0.0011 & 0 \end{bmatrix} \quad (22)$$

By setting the amount of moment of inertia of a nanosatellite as (23), matrices A and B are obtained, which can be seen in (24) and (25):

$$I \approx \begin{bmatrix} 0.17 & 0 & 0 \\ 0 & 0.19 & 0 \\ 0 & 0 & 0.13 \end{bmatrix} (kg.m^2) \quad (23)$$

$$A \approx \begin{bmatrix} 0_{3\times3} & & I_3 & \\ & 0 & 0 & 0 \\ 0_{3\times3} & 0 & 0 & -0.00098 \\ & 0 & 0.00144 & 0 \end{bmatrix} \quad (24)$$

$$B \approx \begin{bmatrix} & 0_{3\times3} & \\ 5.88 & 0 & 0 \\ 0 & 5.26 & 0 \\ 0 & 0 & 7.69 \end{bmatrix} \quad (25)$$

By selecting the Q and R matrices, the Riccati equation can be solved and the control gain matrix K can be calculated.

As mentioned earlier, Q and R must be real and symmetric matrices (R is a definite positive matrix and Q is a semi-definite positive matrix).

According to the cost function J, matrix Q determines the error portion of the state variables and matrix R determines the activity portion of the actuators. The higher the Q/R ratio, the faster the convergence of the state variables occurs, but a higher R/Q ratio causes a lower energy consumption and a more distance from the saturation situation. A general method for selecting Q and R matrices is to choose them diagonally as (26):

$$Q = \begin{bmatrix} q_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & q_n \end{bmatrix} \quad R = \rho \begin{bmatrix} r_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & r_m \end{bmatrix} \quad (26)$$
$$q_i = \frac{1}{t_{si}(x_{imax})^2}, \quad r_i = \frac{1}{(u_{imax})^2}, \quad \rho > 0$$

Here t_{si} is the optimal settling time for the i^{th} state variable, x_{imax} is the absolute maximum value of the i^{th} state variable, u_{imax} is the absolute maximum value of the i^{th} input, and ρ is an arbitrary controller coefficient selected by trial and error.

Here the first three state variables are quaternions, the maximum value of each is equal to 1, and the next three state variables are the satellite's angular velocity, which is 30 degrees per second just after release but after the settling mode is less than 0.3 degrees per second. The maximum torque of the reaction wheels is also considered to be equal to their saturation limit (about 0.0002 Nm). Given the settling time of 200 seconds for the Nadir pointing, the Q and R matrices are expressed in (27).

$$Q = 0.005 \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3.65 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 3.65 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3.65 \end{bmatrix}$$
$$R = \rho * 2.5 * 10^7 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (27)$$

Finally, To find the control gain we can use either the *lqr()* function in Matlab, or solve the Riccati equation using the Schur method [19] in which we need to form Schur decomposition using the *schur()* syntax in Matlab.

In order to make a correct comparison of PD, LQR, and SDRE controllers, the coefficients are selected in such a way that the settling time of all three controllers is the same, then the energy consumption is compared. As the PD controller does not work well without fine-tuning, it is not comparable to optimal controllers. To solve this problem, we obtain the optimal coefficients for the PD controller with the help of the LQR controller.

As explained before, in the LQR controller, the control coefficients matrix is obtained by solving the Riccati equation, and the control action, which is the torque of actuators, is calculated using the quaternion error and the angular velocity error according to Equation (28).

$$\begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} = \begin{bmatrix} k_{11} & k_{12} & k_{13} & k_{14} & k_{15} & k_{16} \\ k_{21} & k_{22} & k_{23} & k_{24} & k_{25} & k_{26} \\ k_{31} & k_{32} & k_{33} & k_{34} & k_{35} & k_{36} \end{bmatrix} \begin{bmatrix} \delta q \\ \omega - \omega_{des} \end{bmatrix}$$
$$\begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} = [K_q \quad K_\omega] \begin{bmatrix} \delta q \\ \omega - \omega_{des} \end{bmatrix}$$
$$K_q = norm \begin{bmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{bmatrix}$$
$$, K_\omega = norm \begin{bmatrix} k_{14} & k_{15} & k_{16} \\ k_{24} & k_{25} & k_{26} \\ k_{34} & k_{35} & k_{36} \end{bmatrix} \quad (28)$$

In (28), the coefficients K_q and K_ω are the optimal coefficients for the PD controller, which were considered as the average (norm) coefficients of the first and the second part of the K matrix. Now we apply these coefficients to the PD controller.

First, we compare the behaviour of the controllers for a major maneuver with large initial angles and angular velocities error (180 degrees tilt and angular velocities of 30 degrees per second in each direction) in order to point to the Nadir:

- *PD* controller with optimal gain: $K_p=0.002$ & $K_d=0.1738$

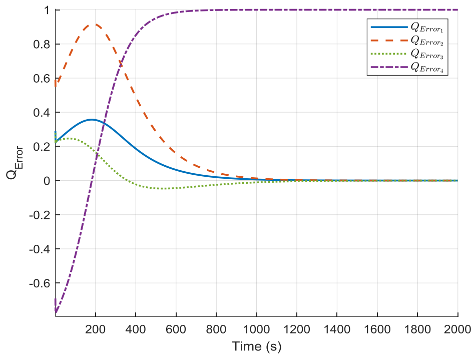

Fig 1. Quaternion error changes for Nadir pointing maneuver with a large initial error by the PD controller

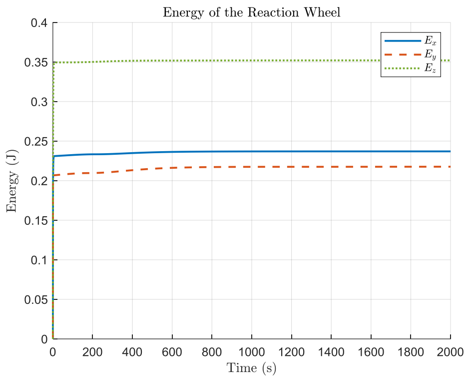

Fig 2. Energy changes, for Nadir pointing maneuver with a large initial error by the PD controller

- *LQR* controller with convergence coefficient of $\rho=5\times10^{-5}$:

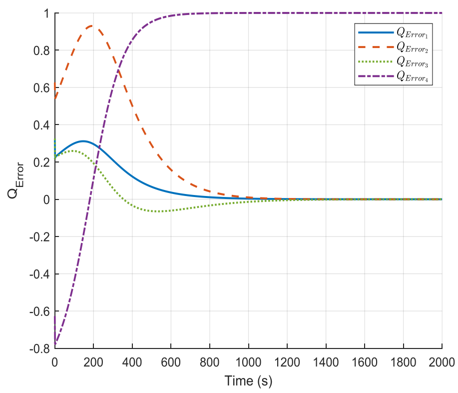

Fig 3. Quaternion error changes, for Nadir pointing maneuver with a large initial error by the LQR controller

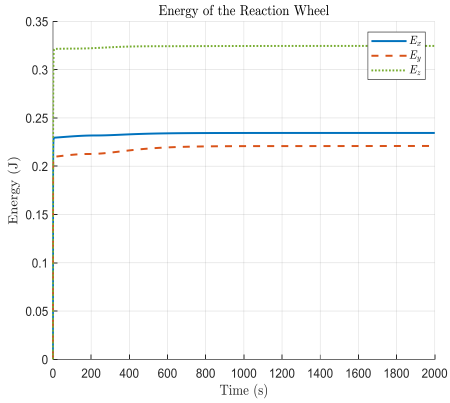

Fig 4. Energy changes, for Nadir pointing maneuver with a large initial error by the LQR controller

- SDRE controller with convergence coefficient of $\rho=5\times10^{-5}$:

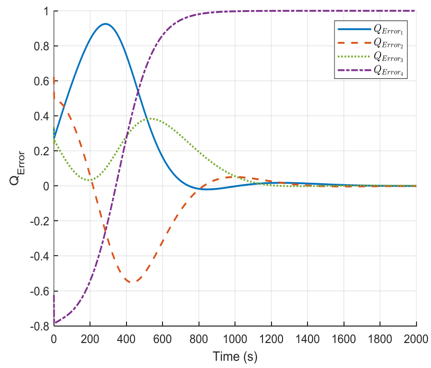

Fig 5. Quaternion error changes for Nadir pointing maneuver with a large initial error by the SDRE controller

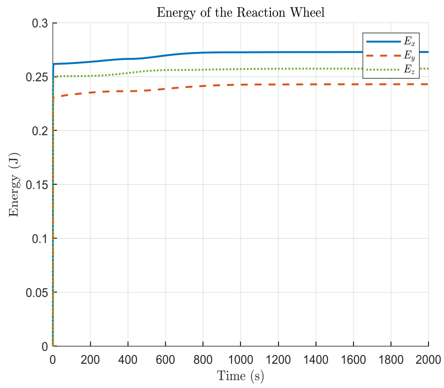

Fig 6. Energy changes, for Nadir pointing maneuver with a large initial error by the SDRE controller

As can be seen in the figures, for the mentioned maneuver, the energy consumption of the SDRE controller is slightly less than LQR and the energy consumption of the LQR controller is also slightly better than the energy consumption of the PD controller with optimal coefficients.

In the next step, the behaviour of the controllers for small maneuvers around the equilibrium point (less than 60 degrees tilt and angular velocity of 0.2 degrees per second in each direction) is compared:

- *PD* controller with optimal gain: $K_p=0.002$ & $K_d=0.1738$

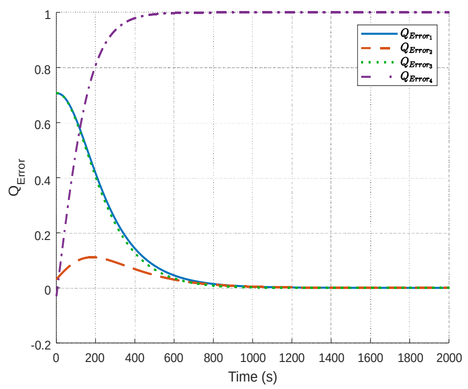

Fig 7. Quaternion error changes, for small maneuver in Nadir pointing by the PD controller

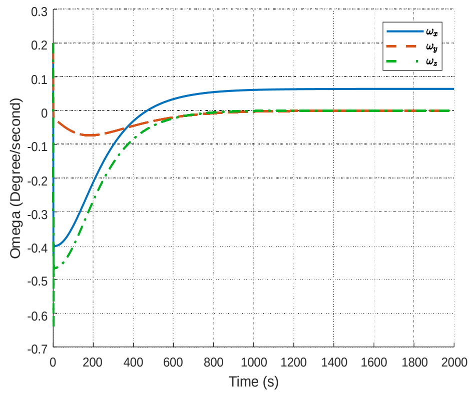

Fig 8. Angular velocity changes, for small maneuver in Nadir pointing by the PD controller

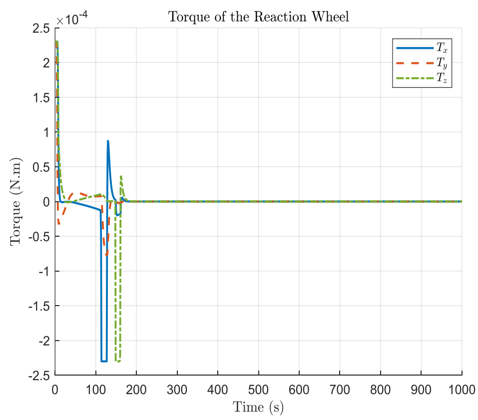

Fig 9. Torque of reaction wheels, for small maneuver in Nadir pointing by the PD controller

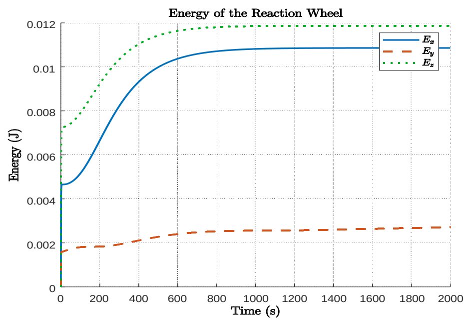

Fig 10. Energy changes, for small maneuver in Nadir pointing by the PD controller

- *LQR* controller with convergence coefficient of $\rho=5\times10^{-5}$:

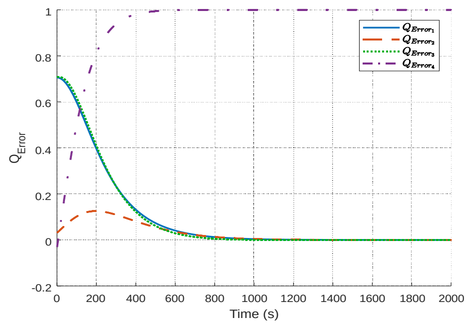

Fig 11. Quaternion error changes, for small maneuvering in pointing on the Nadir by the LQR controller

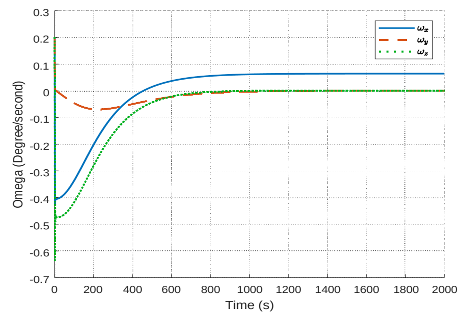

Fig 12. Angular velocity changes, for small maneuver in Nadir pointing by the LQR controller

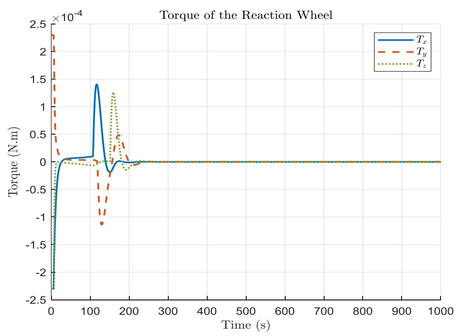

Fig 13. Torque of reaction wheels, for small maneuver in Nadir pointing by the LQR controller

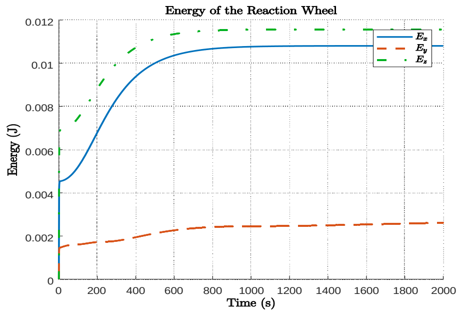

Fig 14. Energy changes, for small maneuver in Nadir pointing by the LQR controller

- SDRE controller with convergence coefficient of $\rho=3\times10^{-5}$:

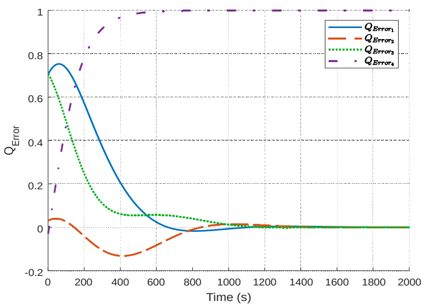

Fig 15. Quaternion error changes, for small maneuver in Nadir pointing by the SDRE controller

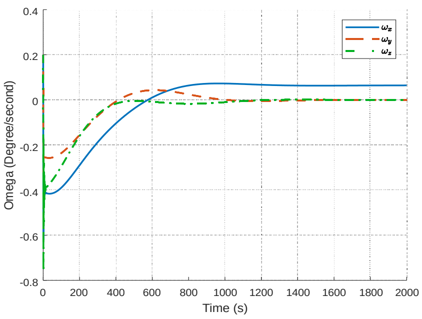

Fig 16. Angular velocity changes, for small maneuver in Nadir pointing by the SDRE controller

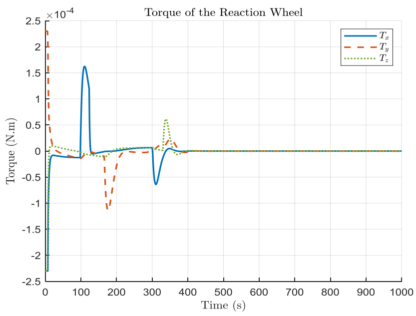

Fig 17. Torque of reaction wheels, for small maneuver in Nadir pointing by the SDRE controller

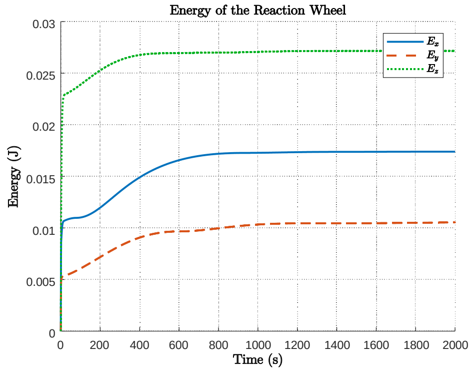

Fig 18. Energy changes, for small maneuver in Nadir pointing by the SDRE controller

In contrast to the previous case, for small maneuvers, the energy consumption of the PD controller with optimal coefficients and LQR was lower than SDRE.

Optimal Quaternion Feedback Control

The mentioned PD controller has proof of stability and always convergence. But this controller does not guarantee the shortest path to the final orientation. This problem occurs when the error of the fourth component in the quaternion is negative. But in the Quaternion feedback controller, one term is added to the PD controller based on the sign of the fourth quaternion element, which increases the convergence speed and converts the control law to (29) [13]:

$$L = -k_p sign(\delta q_4)\delta q_{1:3} - k_d(\omega - \omega_{des}) \quad (29)$$

In this equation, δq_4 is calculated from (30):

$$\delta q_4 = q^T q_{des} \quad (30)$$

In the LQR controller, the term $sign(\delta q_4)$ can also be used to improve the convergence speed and therefore the control equation can be defined as equation (31):

$$U = -K \begin{bmatrix} sign(\delta q_4)\delta q \\ \omega - \omega_{des} \end{bmatrix} \quad (31)$$

Here, matrix K is the optimal coefficients matrix. Now we can compare the performance of the new controller with a conventional LQR controller with the same control coefficients for Q and R. Simulations have been performed for Nadir pointing with initial conditions of [0.5 0 0 0.867] for quaternion and [0.2 0.2 0.2] for angular velocity. The saturation limit has also been considered for the reaction wheels.

- LQR controller simulation without using the $sign(\delta q_4)$ term:

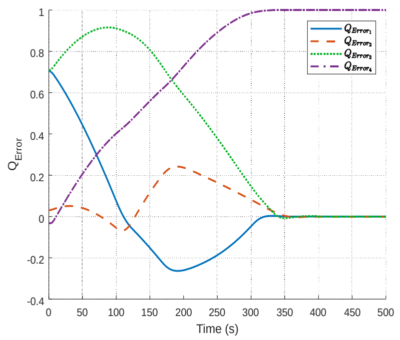

Fig 19. Quaternion error changes, in the LQR controller without using the sign term in the Nadir pointing

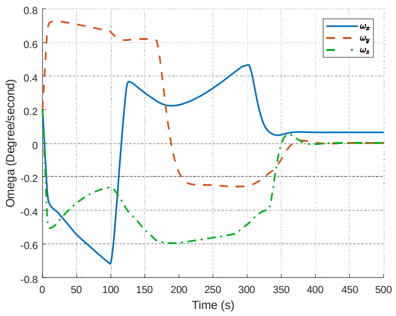

Fig 20. Changes in angular velocity, in the LQR controller without using the sign term in the Nadir pointing

- LQR controller simulation using $sign(\delta q_4)$ term:

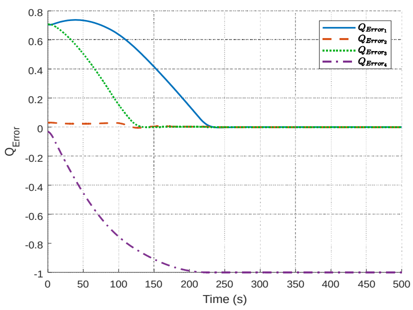

Fig 21. Quaternion error changes, in the LQR controller with the sign term in the Nadir pointing

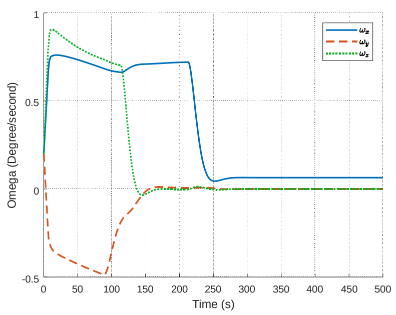

Fig 22. Changes in angular velocity, in the LQR controller with the sign term in the Nadir pointing

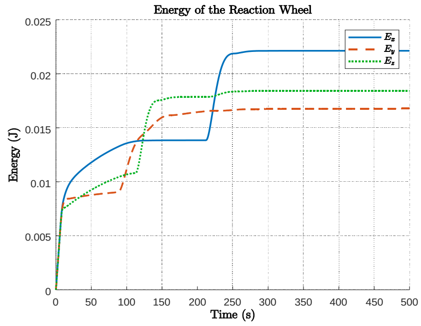

Fig 23. Energy consumption changes in the LQR controller using the sign term

As can be seen, the maneuvering speed has significantly increased since the convergence time has increased from 400 seconds for the conventional LQR controller to 250 seconds for the new controller.
Now we can inspect energy consumption in two controllers:

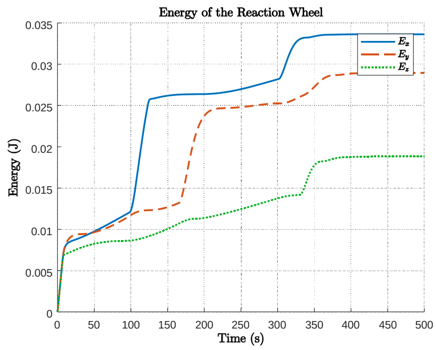

Fig 24. Energy consumption changes in the LQR controller without using the sign term

As can be seen, energy consumption has decreased by up to 30% in the new controller.

In the case of the SDRE controller, it is also possible to use the $sign(\delta q_4)$ term. Figures 25 to 28 show the performance of the SDRE controller with and without the $sign(\delta q_4)$:

- *SDRE* controller simulation without using the $sign(\delta q_4)$ term:

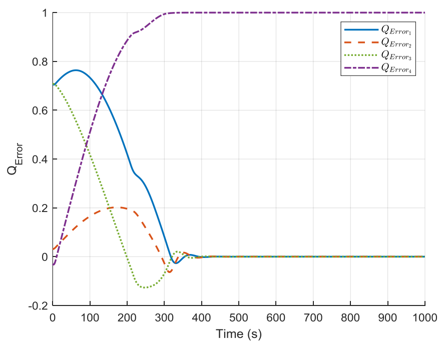

Fig.25. Quaternion erro changes, in the SDRE controller without using the sign term in the Nadir pointing

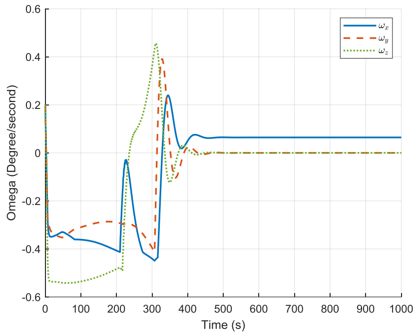

Fig 26. Angular velocity changes, in the SDRE controller without using the sign term in the Nadir pointing

- *SDRE controller simulation with the $sign(\delta q_4)$ term:*

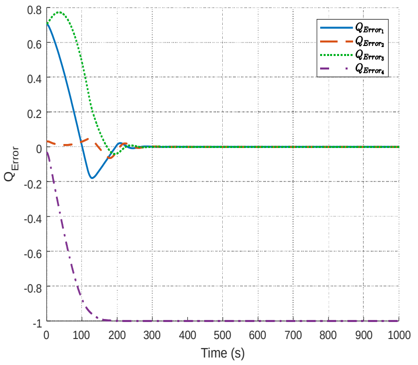

Fig 27. Quaternion error changes, in the SDRE controller with the sign term in the Nadir pointing

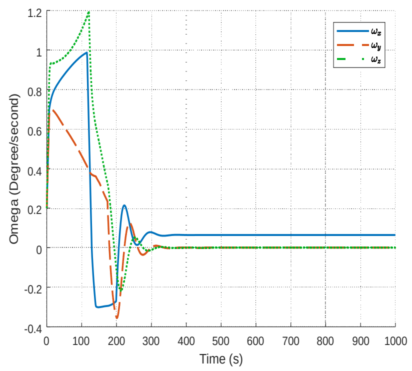

Figure 28. Angular velocity changes, in the SDRE controller with the sign term in the Nadir pointing

As can be seen, similar to the LQR controller, the maneuvering speed has been significantly increased and the convergence time has improved from 450 seconds for the conventional SDRE controller to 300 seconds for the new controller.

Now we can take a look at two controllers in terms of energy consumption:

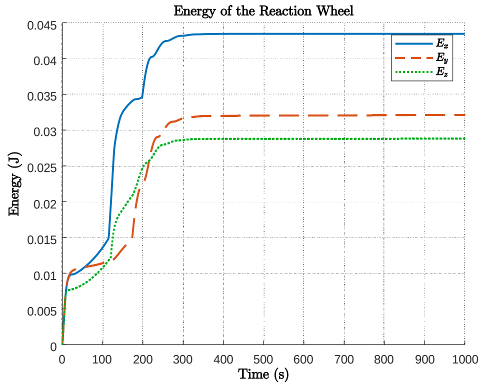

Fig 29. Changes in power consumption, in the SDRE controller, using the sign term

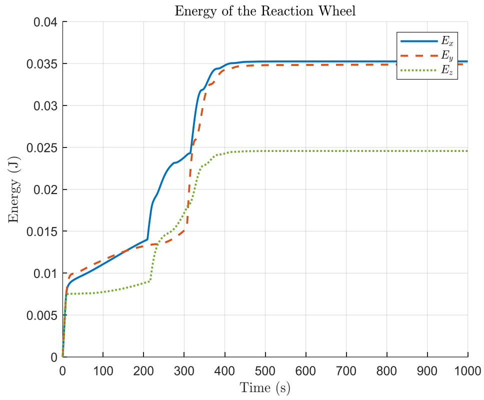

Fig 30. Changes in power consumption, in the SDRE controller without using the sign term

Unexpectedly, energy consumption has not improved by using the $sign(\delta q_4)$ term. It can be inferred that due to the nonlinear nature and the possibility of adjusting coefficients at each time step, the SDRE controller does not need to use the sign term and this term has no effect on energy consumption.

Control Implementation with LQG Observer

In real satellite conditions, an estimator is used to determine the state variables. So far in all simulations, dynamic equations and state variables were calculated by the designed satellite simulator software, but in a real problem, the state variables are estimated from the information obtained by sensors. The best satellite estimation algorithm is the Kalman filter. In this algorithm, in addition to estimating the state variables from different sensory data, measured data is filtered and noise and bias effects are reduced. In the present problem, an Extended Kalman Filter algorithm (EKF) is used to determine the attitude of the satellite.

In a real case, sensor noise always affects the performance of the estimator and controller. The number, type, effects of noise, and bias of sensors can be set in the utilized satellite simulation software[1]. The implemented Kalman algorithm uses magnetometer and gyroscope data. The effect of noise and bias of the mentioned sensors is also considered according to the sensors' datasheet.

First, the PD controller with the optimal coefficients obtained from the previous section is used to simulate the Nadir pointing:

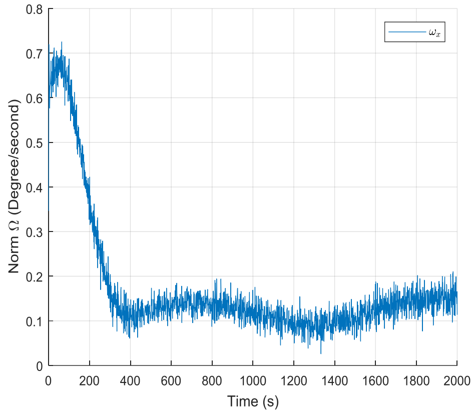

Fig 31. Smooth angular velocity changes for the PD controller with optimal coefficients along with the observer in the Nadir pointing

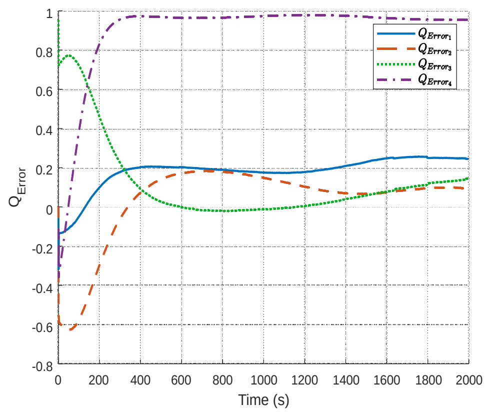

Fig 32. Quaternion error changes for the PD controller with optimal coefficients along with the observer in the Nadir pointing

As can be seen, after 2000 seconds the controller has not yet succeeded in pointing with the required accuracy and the permanent angle error is still high.

Figure 30 shows the estimator error in this case:

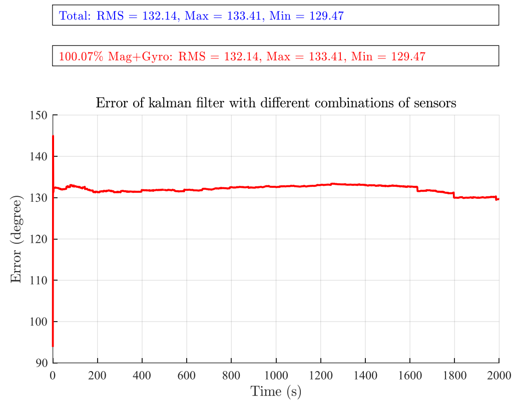

Fig 33. Changes in estimator error during PD control operation with optimal coefficients in the absence of a solar sensor in the Nadir pointing

The considerable estimation error is caused by the high magnetometer and gyro sensors' noise.

Now the same maneuver is done with a normal LQR controller. The results are similar to the PD controller:

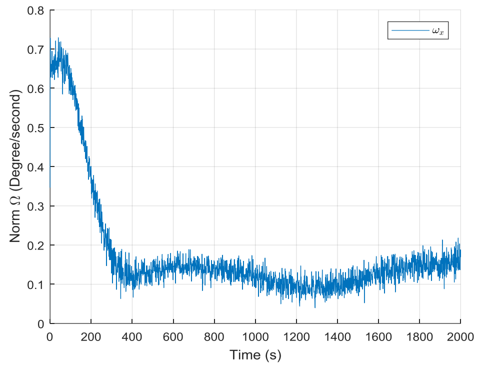

Fig 34. Smooth angular velocity changes for a typical LQR controller with an observer in the Nadir pointing

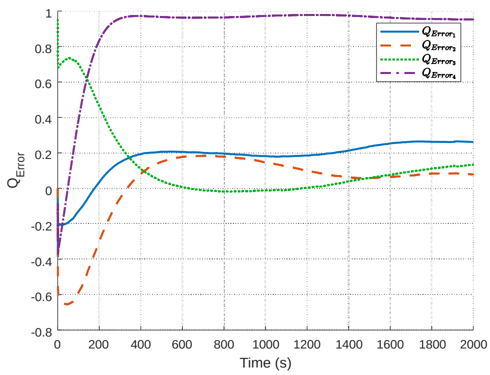

Fig 35. Quaternion error changes for a typical LQR controller with an observer in the Nadir pointing

1. Sharif Nano Satellite Simulator

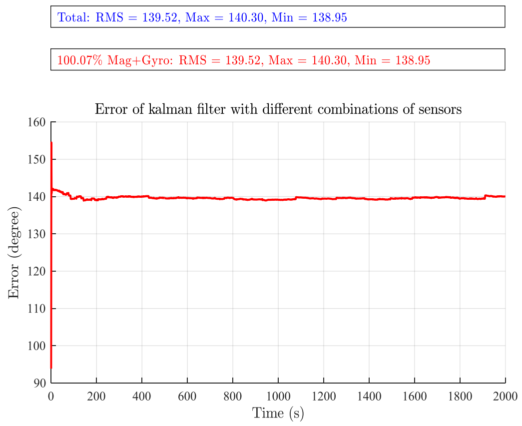

Fig 36. Estimation error changes during normal LQR control operation in the absence of a solar sensor in the Nadir pointing

However, the estimator error has improved a bit.

Now we can evaluate the LQR controller with the $sign(\delta q_4)$ coefficient:

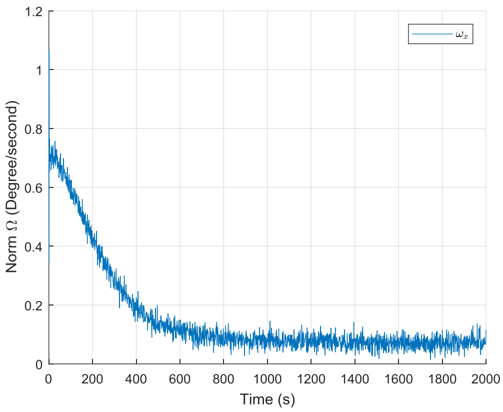

Fig 37. Smooth angular velocity changes for the LQR controller, including the sign term with the observer in the Nadir pointing

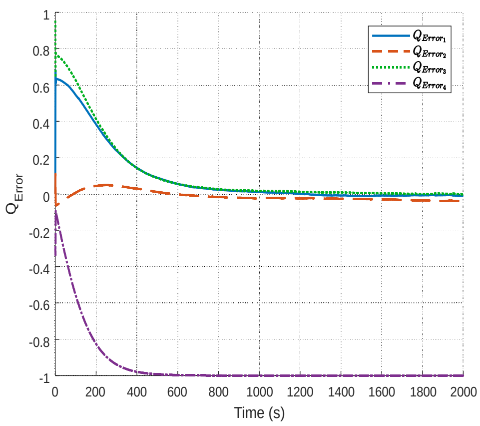

Fig 38. Quaternion error changes for the LQR controller, including the sign term with the observer in the Nadir pointing

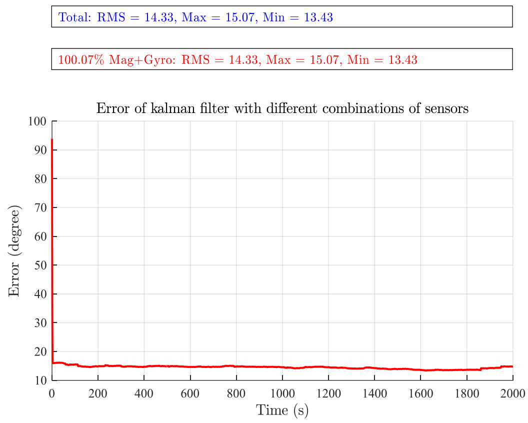

Fig 39. Changes in estimator error during LQR control operation including sign term in the absence of a solar sensor in the Nadir pointing

As can be seen, the convergence occurred in less than 1000 seconds, which indicates the optimal performance of the LQR controller including the $sign(\delta q_4)$ term.The SDRE controller performance is also examined below:

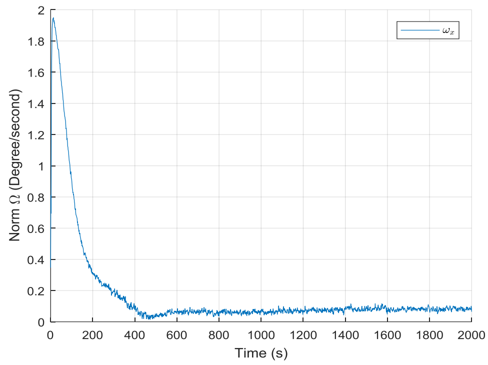

Fig 40. Smooth angular velocity changes for the SDRE controller with the observer in the Nadir pointing

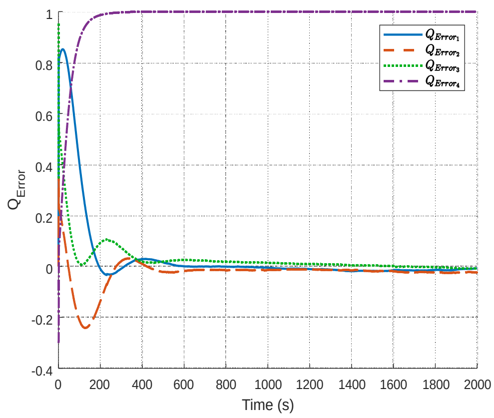

Fig 41. Quaternion error changes for the SDRE controller with the observer in the Nadir pointing

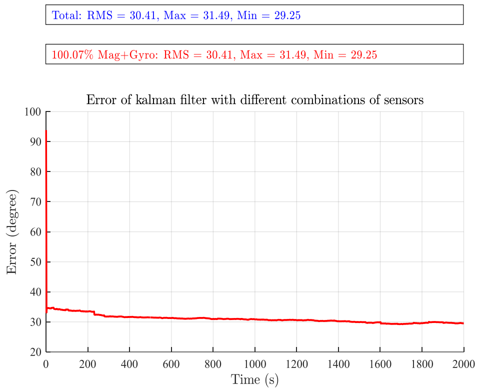

Fig 42. Changes in estimator error during SDRE control operation in the absence of a solar sensor in the Nadir pointing

As can be seen, the convergence speed of the SDRE controller is higher than other controllers.

Conclusion

In this paper, a PD controller with optimal coefficients, as well as LQR, and SDRE controllers were used to control the attitude of a nanosatellite to perform various maneuvers pointing toward Nadir.

In the first step, PD, LQR, and SDRE controllers were investigated without considering the estimator and by solving the differential equations of the satellite dynamics. In order to fine-tune the PD controller and make it comparable with the two other controllers, proportional and derivative coefficients were extracted with the help of the LQR controller. To compare the performance of the controllers, the coefficients of all three were adjusted so that the convergence time remains in the same range, and then the energy consumption of the actuators in each controller was compared.

Results of the simulations showed that for extreme maneuvers, the energy consumption for the SDRE nonlinear controller was better than the LQR and PD linear controllers. But for small maneuvers around the Nadir, the PD and LQR linear controllers perform better than the SDRE nonlinear controller.

In the second part of the simulations, the controller is used along with a Kalman filter estimator and the noise and bias of the sensors are also added to the system. This time, the controller receives its input from the estimator. Therefore, the simulation conditions become much more realistic.

In our problem, an Extended Kalman filter is used as an estimator, which uses magnetometers, and gyroscope sensors to determine the attitude of the satellite. After simulations, it was shown that the convergence speed of nonlinear SDRE and LQR with quaternion feedback controllers is much higher than PD and LQR linear controllers. Meanwhile, the SDRE controller has the best performance and lowest power consumption.

In a major maneuver, i.e., a maneuver with considerable initial error, the SDRE control has better performance. Because in this case, the linearization performed for the LQR controller is not accurate. But for small maneuvers, the LQR controller combined with quaternion feedback works well. It was also observed that the SDRE nonlinear controller performed much better than the linear controllers when using the estimator with the presence of noise.

Nevertheless, the disadvantage of the SDRE controller and what limits its usage is the high computational cost. By defining different scenarios and determining control strategies by a supervisor, the type of controller, or even the control coefficients can be changed based on different conditions and maneuvers, and the SDRE controller can be utilized only in some required conditions to achieve the best performance, the highest convergence speed and the lowest energy consumption.

Conflict of Interests

No conflict of interest has been expressed by the authors.

References

[1] P. Kumar, V. S. Dwivedi, D. K. Giri, and A. K. Ghosh, “Quaternion Based Optimal Controller for Momentum Biased Nadir Pointing Satellite”, *in 2020 IEEE Aerospace Conference*, pp. 1-10, 2020, doi: 10.1109/AERO47225.2020.9172570 .

[2] Y. Yang, *Spacecraft modeling, attitude determination, and control: quaternion-based approach*. CRC Press, 2019.

[3] A. Aghalari, and J. Tayebi, "Designing and Implemntation of PID and Feed back Quaternion Control Strategies for Three Axis Satellite Simulator Equipped With Control Moment Gyros Actiatprs," *Journal of Space Science and Technology* , Vol. 9, No.1, pp. 13-23, 2016.(in Persian)

[4] F. Taji Hervi, and A. Novinzadeh, "Designing Spacecraft Attitude Control Using Model-free Optimal Control Theory," *Journal of Space Science and Technology*, Vol. 10, No.3, pp. 41-57, 2017.(in Persian)

[5] I. Ofodile, H. Ehrpais, A. Slavinskis, and G. Anbarjafari, “Stabilised LQR control and optimised spin rate control for nanosatellites,” *In 9th International Conference on Recent Advances in Space Technologies (RAST)*, IEEE pp. 715-722, 2019, doi: 10.1109/RAST.2019.8767850

[6] H. Arefkhani, S. M. M. Dehghan, and A. H. Tavakoli, "Evaluation of Magnetic Attitude control with Air-Bearing simulator," *Journal of Space Science and Technology* , Vol 9, No.2, pp. 47-60, 2016. (in Persian)

[7] S. H. Miri, S. M. Mirshams, and A. Nikkhah, "Spacecraft Optimal Attitude Control by means of Reaction Wheels," *Journal of Space Science and Technology*, Vol 2, No.5, pp. 35-42, 2010. (in Persian)

[8] A. Nikkhah, J. Tayebi, and J. Roshanian, "Attitude Stabilization and Manauvring of Agility Nanosatellite with Control Moment Gyros", *Journal of Space Science and Technology*, Vol.7, No.2, pp. 1-9, 2014 (in Persian).

[9] C. A. de Carvalho Junior, and A. Fenili, "Nonlinear Atitude Control of Artificial Satellites Considering Failure in Momentum Wheels", *In 22nd International Congress of Mechanical Engineering (COBEM)*, Rome, 2013.

[10] R. O. Doruk, "SDRE Based Attitude Control Using Modified Rodriguez Parameters," *arXiv preprint* arXiv:1103.5444, 2011, doi: https://doi.org/10.48550/arXiv.1103.5444

[11] L. C. de Souza, and V. M. Arena, "Design of satellite attitude control algorithm based on the SDRE method using gas jets and reaction wheels," *Journal of Engineering*, vol. 2013, 2013, doi: https://doi.org/10.1155/2013/318072.

[12] B. Ø. Andresen, C. Grøn, R. H. Knudsen, and et al. ,"Attitude control system for AAUSAT-II," *Institute of Electronic Systems*, Aalborg University, 2005.

[13] F. L. Markley and J. L. Crassidis, *Fundamentals of spacecraft attitude determination and control*, Springer, 2014, doi: https://doi.org/10.1007/978-1-4939-0802-8

[14] Y. Yang, "Quaternion-based lqr spacecraft control design is a robust pole assignment design, " *Journal of Aerospace Engineering*, Vol 27, No.1, pp.168-176, 2014, doi: https://doi.org/10.1061/(ASCE)AS.1943-5525.0000232.

[15] S. Bittanti, A. J. Laub, and J. C. Willems, *The Riccati Equation*, Springer Science & Business Media, 2012.

[16] D. S. Naidu, S. Paul, and C. R. Rieger, "A simplified SDRE technique for regulation in optimal control systems," *In 2019 IEEE International Conference on Electro Information Technology (EIT),* IEEE, pp. 327-332, 2019, doi: 10.1109/EIT.2019.8834201 .

[17] B. D. Anderson, and J. B. Moore, *Optimal control: linear quadratic methods*, Courier Corporation, 2007.

[18] A. Bunse-Gerstner, "Computational solution of the algebraic Riccati equation", *Journal of The Society of Instrument and Control Engineers*, Vol 35, No.8, pp. 632-639, 1996, doi: https://doi.org/10.11499/sicejl1962.35.8_632.

[19] A. J. Laub, "A Schur Method for Solving Algebraic Riccati Equations," Massachusetts Institute of Technology, Laboratory for Information and Decision Systems, LIDS Report number 859, 1978. doi: 10.1109/TAC.1979.1102178 .