Acta Astronautica 222 (2024) 52–68

Contents lists available at ScienceDirect

Acta Astronautica

journal homepage: www.elsevier.com/locate/actaastro

Research paper

Integrated architecture for navigation and attitude control of low-cost suborbital launch vehicles

Pedro dos Santos *, Paulo Oliveira

IDMEC, Instituto Superior Técnico, Universidade de Lisboa, Av. Rovisco Pais 1, 1049-001, Lisbon, Portugal

ARTICLE INFO

Keywords:
Suborbital launch vehicles
TVC
Attitude control
LQR
Navigation
Kinematic filters

ABSTRACT

This paper proposes an integrated architecture for navigation and attitude control of low-cost suborbital launch vehicles, propelled by a solid motor. Single-nozzle, two Degrees-of-Freedom (DoF) Thrust Vector Control (TVC) actuation is adopted. For architecture design purposes, a non-linear, unstable, 6 DoF model for the generic thrust-vector-controlled launcher dynamics and kinematics is deduced, and a linear state-space representation is proposed. The navigation system provides full-state estimates resorting to novel complementary kinematic filters, whose design allows to establish an explicit relation with steady-state Kalman filtering. A globally stable estimation solution is obtained, apart from the singularities arising from the use of Euler angles. The attitude control law is derived from the Linear Quadratic Regulator (LQR) using the state-space models for each linearization point of the reference trajectory, with an integral action (LQI) added to improve robustness and to provide null steady state attitude tracking error. A correction method is proposed to allow for pitch and yaw control in the presence of spinning motion, precluding the need of a supplementary roll control system. The control system is implemented through gain scheduling, resorting to an altitude-based linear parametric varying method. The architecture is implemented in a realistic simulation environment, composed by the 6 DoF non-linear model, the Navigation and Control solutions, and the environmental disturbances, to assess its performance through Monte Carlo simulations. The navigation system is able to provide accurate estimates of the state vector, while the control system satisfies attitude tracking performance and robustness to both external disturbances and model parametric uncertainties.

1. Introduction

During the last decades, suborbital launch vehicles endowed the scientific community with tools to perform a myriad of research studies [1,2]. Commonly denominated as sounding rockets, they provide long periods of microgravity conditions, allow to collect in-situ data across all atmospheric layers, and enable rapid Earth surveillance and monitoring [1–3]. Simultaneously, they can be instrumental as low-cost testing platforms to augment Technology Readiness Levels (TRL) of different systems and payloads, before their use in high risk, potentially crewed, orbital/space missions [3].

More recently, following the successful efforts of private companies, such as SpaceX, Blue Origin, and Rocket Lab, a growing number of both private corporations and national/international agencies, namely at European level, are investing towards a new generation of reusable micro and small launch vehicles [4], which motivates the development of suborbital platforms for technology demonstration purposes [5–7]. In addition, suborbital transportation and space tourism motivated a market increase which impacts the overall need for cost-effective, dedicated suborbital launchers [8].

To meet specific mission requirements, in terms of stability and trajectory tracking, launch vehicles must have a dedicated Guidance, Navigation, and Control (GN&C) system. This system is responsible for determining the trajectory to be followed and commanding the required attitude (or orientation) over time (Guidance), for estimating the state vector, composed by position, velocity, and attitude (Navigation), and for calculating the necessary actuation inputs to achieve the desired attitude (Control). In this paper, an integrated architecture for the navigation and attitude control of low-cost suborbital launchers is proposed. As for the actuation method, Thrust Vector Control (TVC), or thrust vectoring for short, is selected.

TVC is used by most launch vehicles and works by redirecting the thrust vector in order to create a control torque [9]. In this work, single gimballed nozzle actuation is considered, which is a suitable configuration for low-cost small suborbital launchers. However, it can only impact the pitch and yaw angles, whereas roll has to be controlled by an additional system, if needed. With respect to other actuation

* Corresponding author.
E-mail address: pedrodossantos31@tecnico.ulisboa.pt (P. dos Santos).

https://doi.org/10.1016/j.actaastro.2024.05.039
Received 18 October 2023; Received in revised form 11 April 2024; Accepted 28 May 2024
Available online 29 May 2024
0094-5765/

Nomenclature

Scalars

C_A, C_Y, C_N	Aerodynamic force coefficients
C_l, C_m, C_n	Aerodynamic moment coefficients
C_{l_p}, C_{m_q}, $C_{m_{\dot{\alpha}}}$, C_{n_r}, $C_{n_{\dot{\beta}}}$	Aerodynamic damping coefficients
C_{N_α}, C_{Y_β}	Aerodynamic stability derivatives
d	Fuselage diameter
g	Gravitational acceleration
J_l	Longitudinal inertia
J_t	Transverse inertia
l	Control moment arm
m	Mass
p, q, r	Body components of angular velocity
$\bar{q}$	Dynamic pressure
S	Fuselage cross-sectional area
$S.M$	Static stability margin
T	Thrust magnitude
u, v, w	Body components of linear velocity
V	Magnitude of the velocity vector
x_{cm}	Centre of mass location
x_{cp}	Centre of pressure location
α	Angle of attack
β	Sideslip angle
μ_p, μ_y	Pitch and yaw control inputs (nozzle deflections)
ϕ, θ, ψ	Roll, pitch, and yaw Euler angles

Vectors

$\mathbf{a}$	Acceleration
$\mathbf{b}_\omega$	Gyroscope bias
$\mathbf{f}$	Total external force
$\mathbf{g}$	Gravity
$\mathbf{m}$	Earth's magnetic field
$\mathbf{p}$	Position
$\mathbf{u}$	Control input
$\mathbf{v}$	Linear velocity
$\mathbf{x}$	State vector
$\mathbf{y}$	Output vector
$\boldsymbol{\lambda}$	Euler angles
$\boldsymbol{\omega}$	Angular velocity
$\boldsymbol{\tau}$	Total external moment

Matrices

$\mathbf{J}$	Inertia
$\overline{\mathbf{K}} = \begin{bmatrix} \mathbf{K} & \mathbf{K_i} \end{bmatrix}^T$	LQI gain matrix (LQR plus integrators)
$\mathbf{L}_{1\lambda}$, $\mathbf{L}_{2\lambda}$	Time-invariant ACF gains
$\mathbf{L}_{1p}$, $\mathbf{L}_{2p}$, $\mathbf{L}_{3p}$	Time-invariant PCF gains
$\mathbf{Q}(\lambda)$	Attitude kinematics matrix
$\mathbf{Q}$, $\mathbf{R}$	LQR tuning matrices
$\mathbf{R}(\lambda)$	Rotation matrix from body to inertial
$\mathbf{S}(^b\boldsymbol{\omega})$	Skew-symmetric matrix of the body angular velocities
$\boldsymbol{\Theta}$	Measurement noise covariance matrix
$\boldsymbol{\Xi}$	Process noise covariance matrix

Subscripts/Superscripts

b	Expressed in the body frame
c	Used for control
d	Desired value
i	Expressed in the inertial frame
lon	Associated with the longitudinal mode
lat	Associated with the lateral mode
r	Sensor reading
sl	At surface level
$\hat{}$	Estimate
0	Nominal value

Acronyms/Abbreviations

ACF	Attitude Complementary Filter
AD	Attitude Determination
DoF	Degrees-of-Freedom
GNSS	Global Navigation Satellite System
LQI	Linear Quadratic Integral
LQR	Linear Quadratic Regulator
MC	Monte Carlo
PCF	Position Complementary Filter
PID	Proportional-Integral-Derivative
TVC	Thrust Vector Control
UAS	Uniformly Asymptotically Stable

techniques, such as actively controlled fins, TVC allows for a wider range of operating conditions and provides better efficiency [10].

Solid motors are the most common propulsion technology in sub-orbital launch vehicles due to their Intercontinental Cruise Ballistic Missile (ICBM) heritage [11] and associated low production costs, which enables rapid and responsive launch missions [12]. Therefore, it has been selected as reference for this work. Contrarily to liquid or hybrid engines, solid motors do not possess throttle capability. This means that thrust cannot be controlled and, consequently, control authority is reduced.

The control system design tends to be very conservative in the aerospace industry [13]. Restricting the dynamic analysis to accommodate more sophisticated control design techniques risks the later realization that such restrictions would have to be lifted and would invalidate the control design [14]. Due to the highly non-linear dynamics and to the time-varying nature of the parameters, such as aerodynamic and inertial, the applicability of linear control techniques relies on the linearization of the system at several operating points. The design is then focused in each linear model and the resulting controller gains are changed during the flight through a technique called gain scheduling, as in [15].

Classic and linear control solutions, based on thrust vectoring, can be found in [9,16,17]. These include Proportional-Integrative-Derivative (PID) control and pole placement techniques, with time-varying gains. Although widely used, PID control has its limitations when it comes to model uncertainty robustness and external disturbances rejection.

Still in the linear domain, the use of optimal controllers, such as the Linear Quadratic Regulator (LQR), provides some degree of robustness and ensures a (sub-)optimal reference tracking solution for a given cost functional. In [18], the LQR is used to address the attitude control problem, and in [19] a LQG algorithm is proposed for state estimation and control, with both restricting the analysis and design to the pitch plane at a single operating point.

Non-linear techniques have also been proposed for launch vehicle control and estimation [20–22], and come with the advantage of ensuring a global solution, not dependant on the specific mission nor vehicle. However, these methods all have particular design characteristics which hinder the application of standardized, well-established, verification and validation procedures [9,14,16].

Although several solutions to the launcher control problem can be found in the literature, many fail to capture all the relevant dynamics and/or oversimplify the problem, while most assume full-state knowledge, creating a considerable gap between theoretical design and implementation. Hence, the main contribution of this paper is a robust architecture, which integrates both the navigation and control systems, that is computationally efficient and can be implemented in suborbital launchers relying on readily available low-cost components. It is suitable for vehicles using low-cost solid propulsion technology, off-the-shelf inertial navigation sensors, and simplified actuation methods. Furthermore, the design process considers the 6 DoF and the time-varying nature of the system, focusing on the entire trajectory rather than a single operating point, contrarily to what is found in the literature when using linear optimal control/estimation techniques. Preliminary work from the authors on the topic can be found in [23].

For the navigation system, novel complementary kinematic filters are proposed to fuse the sensor readings and obtain filtered, unbiased, full-state estimates. Making use of the Lyapunov transform concept, a relation is established between the derived time-varying filters and the time-invariant case. By exploiting this relation, the tuning of the complementary filters is performed for the time-invariant case relying on Kalman filtering theory, avoiding extensive tuning for each specific mission, and ensuring a globally stable solution apart from the singularities. This is an improvement with respect to classical methods which rely on the linearization of the plant for a given trajectory, and requires less computational effort than standard estimation solutions for non-linear systems, such as the Extended Kalman Filter (EKF).

As for the control system, LQR control is proposed with additional integral action (LQI) to improve robustness and provide zero attitude tracking error. The gains are obtained for different operating points of the reference trajectory using an original time-varying state-space representation, and are scheduled during flight with an altitude-based linear interpolation method. By considering the time-varying nature of the system, the implementation in a real scenario is facilitated. To prevent the additional complexity and cost of adding a roll control system, the architecture is designed to provide pitch and yaw control in the presence of uncontrolled spinning motion, up to a given limit.

This paper is organized as follows: some notation is detailed in Section 2. The physical model is shown in Section 3. The linear state-space representation is derived in Section 4. The proposed architecture is explained in Section 5. The navigation and control systems are detailed in Sections 6 and 7, respectively. Section 8 shows the implementation in simulation of the architecture, as well as the reference vehicle and mission used for validation. In Section 9, a linear domain analysis of the system follows, and in Section 10 the simulation results are presented and discussed. Finally, in Section 11, final remarks and conclusions are drawn.

2. Notation

Throughout this paper bold lowercase and bold uppercase symbols are used to represent vector and matrices, respectively. $\mathbf{I}$ and $\mathbf{0}$ respectively represent the identity and null matrices of appropriate dimensions and $\mathbf{D}(a_1, \dots, a_n)$ is a diagonal matrix of dimension n, with generic diagonal entries a_1 to a_n. Finally, superscript T is used to denote the transpose of a vector or matrix and superscript -1 to denote the inverse of a matrix.

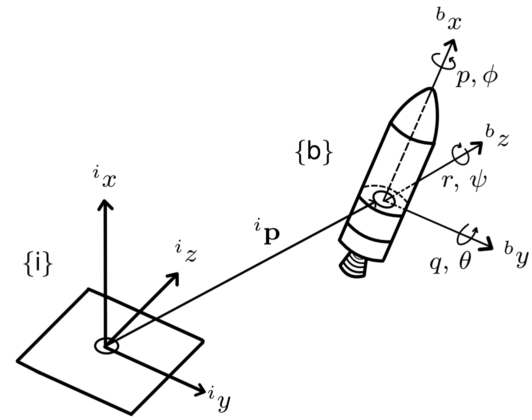


Fig. 1. Reference frames.

3. Physical model

In this section, the dynamics and kinematics of a generic launch vehicle with a single gimballed nozzle are provided. To derive the physical model some assumptions are used: the launch vehicle is assumed to be a rigid body; it is assumed to be axially symmetric, as well as the mass allocation; and the Earth's curvature and rotation are neglected. All these assumptions are often followed in the literature [13,22,24], and are considered valid for first stage design of the architecture, not compromising its overall structure when reproducing it in a real case scenario.

3.1. Reference frames

To describe the dynamics and kinematics of the vehicle, it is crucial to define the reference frames to be used. Two reference frames are used: a body-fixed one {b}, where the equations of motion are written; and an inertial, space-fixed one {i}, located at the launch site (see Fig. 1).

The coordinate transformation between both reference frames is defined using the Euler angles representation, $\lambda = \begin{bmatrix} \phi & \theta & \psi \end{bmatrix}^T$. With this representation, the transformation from {b} to {i} is obtained through a sequential rotation $\mathbf{R}(\lambda) = \mathbf{R}_z(\psi) \cdot \mathbf{R}_y(\theta) \cdot \mathbf{R}_x(\phi)$, where $\mathbf{R}(\lambda) \in SO(3)$ is given by [25]

$$\mathbf{R}(\lambda) = \begin{bmatrix} c_\theta c_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\theta s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{bmatrix},$$

in which s and c stand as abbreviations for the sine and cosine trigonometric functions. The inverse transform, from {i} to {b}, is defined by the transpose $\mathbf{R}^T(\lambda)$.

3.2. Dynamics and kinematics

Using Newton-Euler's equations for rigid body translational and rotational motion, the dynamics and kinematics of the launcher in the 6 DoF are obtained:

$$\begin{cases} {}^i\dot{\mathbf{p}} = \mathbf{R}(\lambda)\,{}^b\mathbf{v} \\ \dot{\mathbf{R}}(\lambda) = \mathbf{R}(\lambda)\,\mathbf{S}({}^b\boldsymbol{\omega}) \\ m\,{}^b\dot{\mathbf{v}} = -\mathbf{S}({}^b\boldsymbol{\omega})\,m\,{}^b\mathbf{v} + {}^b\mathbf{f} \\ \mathbf{J}\,{}^b\dot{\boldsymbol{\omega}} = -\mathbf{S}({}^b\boldsymbol{\omega})\,\mathbf{J}\,{}^b\boldsymbol{\omega} + {}^b\boldsymbol{\tau} \end{cases}, \tag{1}$$

where the first two equations detail the position and orientation kinematics and the last two detail the translational and rotational dynamics.

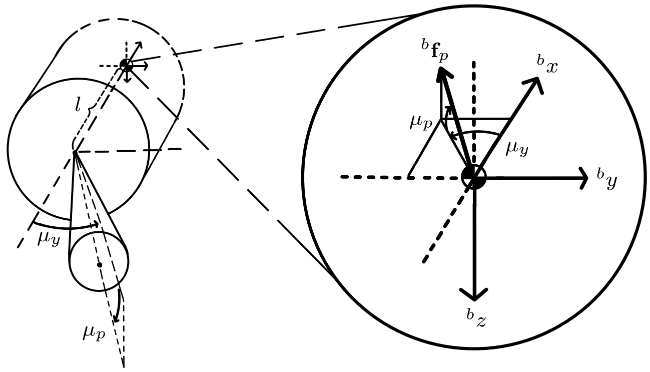


Fig. 2. Thrust vector decomposition in the body axes.

3.2.1. External forces and torques

The total external force can be decomposed as $^b\mathbf{f} = {}^b\mathbf{f}_g + {}^b\mathbf{f}_p + {}^b\mathbf{f}_a$, where $^b\mathbf{f}_g$ represents the gravity force, $^b\mathbf{f}_p$ the propulsive force, and $^b\mathbf{f}_a$ the aerodynamic force, all expressed in {b}. As for the external torque, it is given by $^b\boldsymbol{\tau} = {}^b\boldsymbol{\tau}_p + {}^b\boldsymbol{\tau}_a + {}^b\boldsymbol{\tau}_{rc}$, where $^b\boldsymbol{\tau}_p$ represents the propulsive control torque, $^b\boldsymbol{\tau}_a$ represents the aerodynamic torque, and $^b\boldsymbol{\tau}_{rc}$ is the reaction control torque provided by an additional system if present, all expressed in {b}.

Gravitational

Considering the Earth as a perfect sphere, and looking at the definition of the inertial frame {i}, the gravity force is simply

$$^b\mathbf{f}_g = \mathbf{R}^T(\lambda)\begin{pmatrix}-mg\\0\\0\end{pmatrix} = \begin{pmatrix}-mg\,c_\theta c_\psi\\ -mg\,(s_\phi s_\theta c_\psi - c_\phi s_\psi)\\ -mg\,(c_\phi s_\theta c_\psi + s_\phi s_\psi)\end{pmatrix}, \tag{2}$$

where g, the gravitational acceleration, varies with the altitude according to $g = g_{sl}\, R_E^2\,/\,(R_E + h)^2$, in which R_E is the mean Earth radius.

Propulsive

Considering ideal propulsion, and all its underlying assumptions, the thrust force produced by the motor is [10]

$$T = \underbrace{|\dot{m}|\cdot v_e}_{\text{Dynamic}} + \underbrace{(p_e - p_a)\cdot A_e}_{\text{Static}},$$

where $\dot{m}$ is the mass flow rate, v_e is the effective exhaust velocity, p_e is the nozzle exit pressure, p_a is the atmospheric pressure, and A_e is the nozzle exit area. Two separate contributions can be identified: the dynamic one, caused by the exhaust of the expanded combustion gases; and the static, caused by the pressure gradient between the nozzle exit and the atmosphere.

To obtain the resultant propulsive force and torque, the thrust vector has to be decomposed in the three body axes as illustrated in Fig. 2. According to it, the thrust vector is decomposed using the angles μ_p and μ_y, which are the control inputs, where μ_p is the gimbal angle that, on its own, produces a pitching moment, and μ_y is the one that produces a yawing moment. Using these angles, the propulsive force and torque in the body frame are, respectively, [13]

$$^b\mathbf{f}_p = \begin{pmatrix}T\,c_{\mu_p}c_{\mu_y}\\ -T\,c_{\mu_p}s_{\mu_y}\\ -T\,s_{\mu_p}\end{pmatrix} \text{ and } {}^b\boldsymbol{\tau}_p = \begin{pmatrix}0\\ -T\,s_{\mu_p}\,l\\ T\,c_{\mu_p}s_{\mu_y}\,l\end{pmatrix}, \tag{3}$$

where l, the control torque arm, corresponds to the distance between the nozzle gimbal point and the centre of mass of the rocket, x_{cm}, measured from the tip.

Aerodynamic

The aerodynamic force and moment, expressed in {b}, can be modelled as [14]

$$^b\mathbf{f}_a = \begin{pmatrix}-\bar{q}\,C_A\,S\\ \bar{q}\,C_Y\,S\\ -\bar{q}\,C_N\,S\end{pmatrix}, \quad {}^b\boldsymbol{\tau}_a = \begin{pmatrix}\bar{q}\,C_l\,S\,d\\ \bar{q}\,C_m\,S\,d\\ \bar{q}\,C_n\,S\,d\end{pmatrix}, \tag{4}$$

where C_A, C_Y, and C_N are, respectively, the axial, lateral, and normal aerodynamic force coefficients, and C_l, C_m, and C_n are, respectively, the rolling, pitching, and yawing aerodynamic moment coefficients.

The normal and lateral force coefficients can be determined using a linear relation with the aerodynamic angles of attack, α, and sideslip, β: $C_Y = C_{Y\beta}\,\beta$ and $C_N = C_{N\alpha}\,\alpha$, whose derivatives ($C_{Y\beta}$ and $C_{N\alpha}$) depend mainly on the angles themselves and Mach number. As for the axial force coefficient, C_A, in most applications, its dependency on the aerodynamic angles can be neglected and it is assumed to vary only with Mach number. The relevant velocity for aerodynamic computations is the one expressed in relation to the fluid composing the atmosphere, $^b\mathbf{v}_{\text{rel}} = \begin{bmatrix}u_{\text{rel}} & v_{\text{rel}} & w_{\text{rel}}\end{bmatrix}^T$. This is given by $^b\mathbf{v}_{\text{rel}} = {}^b\mathbf{v} - {}^b\mathbf{v}_w$, where $^b\mathbf{v}_w$ is the wind velocity vector expressed in {b}. The aerodynamic angles are then given by $\alpha = tan^{-1}(w_{\text{rel}}/u_{\text{rel}})$ and $\beta = sin^{-1}(v_{\text{rel}}/V_{\text{rel}})$, where V_{rel} is the norm of the relative velocity vector.

Regarding the moment coefficients, if the reference moment station is defined as the centre of pressure, and its location, x_{cp}, measured from the tip of the rocket, can be determined, the reference moments are zero and the moment coefficients take the form $C_l = C_{l_p}\,p\,d/(2V_{\text{rel}})$, $C_m = -C_N\,S.M + (C_{m_q} + C_{m_{\dot\alpha}})\,q\,d/(2V_{\text{rel}})$, and $C_n = -C_Y\,S.M + (C_{n_r} + C_{n_{\dot\beta}})\,r\,d/(2V_{\text{rel}})$, where the static stability margin, $S.M = (x_{cp} - x_{cm})/d$, intuitively appears.

3.2.2. Explicit dynamics and kinematics

The explicit dynamics and kinematics can be retrieved by substituting the total external force and torque in (1) by all the individual detailed components, (2), (3), and (4). Furthermore, following the axial symmetry assumption, the cross-products of inertia can be assumed as zero and the y and z terms can be assumed equal, resulting in a diagonal matrix, $\mathbf{J} = \mathbf{D}\,(J_l, J_t, J_t)$, yielding

$$\begin{cases}\dot{u} = -g\,c_\theta c_\psi - \frac{\bar{q}}{m}\,S\,C_A + \frac{T}{m}c_{\mu_p}\,c_{\mu_y} - q\,w + r\,v\\ \dot{v} = -g\,(s_\phi s_\theta c_\psi - c_\phi s_\psi) + \frac{\bar{q}}{m}SC_Y - \frac{T}{m}c_{\mu_p}s_{\mu_y} - r\,u + p\,w\\ \dot{w} = -g\,(c_\phi s_\theta c_\psi + s_\phi s_\psi) - \frac{\bar{q}}{m}\,S\,C_N - \frac{T}{m}\,s_{\mu_p} - p\,v + q\,u\\ \dot{p} = J_l^{-1}\,(\bar{q}\,S\,d\,C_l + \tau_{rc})\\ \dot{q} = J_t^{-1}\,(\bar{q}\,S\,d\,C_m - T\,s_{\mu_p}\,l - p\,r\,(J_l - J_t))\\ \dot{r} = J_t^{-1}\,(\bar{q}\,S\,d\,C_n + T\,c_{\mu_p}\,s_{\mu_y}\,l - p\,q\,(J_t - J_l))\\ \dot{\phi} = p + (q\,s_\phi + r\,c_\phi)\,t_\theta\\ \dot{\theta} = q\,c_\phi - r\,s_\phi\\ \dot{\psi} = \dfrac{q\,s_\phi + r\,c_\phi}{c_\theta}\end{cases}, \tag{5}$$

where the reaction control torque, $^b\boldsymbol{\tau}_{rc}$, if present, is assumed to only impact the roll axis, ^bx, with the component τ_{rc}. It is noted that by using the Euler angles representation a singularity arises for $\theta = \pm\frac{\pi}{2}$. However, the way the reference frames are defined prevents the rocket to reach this attitude inside the admissible range of operation (far from horizontal orientation).

4. Linearized physical model

Linear control and estimation techniques, such as the LQR and the Kalman filter, rely on mathematical representations of the linear systems under study. These representations are usually written in the state-space form. In this section, a generic state-space model for a thrust vector controlled launch vehicle is obtained by linearizing the already detailed explicit dynamics and kinematics in (5).

A widely used linearization technique consists in finding an equilibrium point of the system, in which the first-order derivatives of the states are null, and performing a Taylor series expansion, considering small perturbations around the equilibrium condition. However, rocket flight is dominated by highly varying conditions and parameters, such as mass and inertia, aerodynamic coefficients, dynamic pressure, and thrust, which make it impossible to find a so called trimming trajectory, for which equilibrium is reached with constant control inputs.

One viable alternative [13], is to linearize the system at multiple points, denominated as operating points, throughout a previously selected reference trajectory. The selected trajectory will impose the reference values for system states ($\mathbf{x}_0$) and inputs ($\mathbf{u}_0$), and the outcome is a linear time-varying system. Linear controllers can be designed for the state-space representations associated with each operating point and then scheduled during flight. Therefore, the operating points have to be selected so as to capture all the relevant dynamics of the system, preventing that the system destabilizes.

Firstly, the following variable transformations are defined: $\delta x = x - x_0$ and $\delta u = u - u_0$; where δx and δu are small perturbations around the reference values for each point. By using the variable transformation in the non-linear differential equations of the system (denoted by $\dot{x} = f(x,u)$), generically, we have that $\delta\dot{x} = f(x,u) - f(x_0,u_0) = f(x_0+\delta x, u_0+\delta u) - f(x_0,u_0)$. Using the Taylor series expansion of $f(x_0+\delta x, u_0+\delta u)$ around (x_0, u_0), and neglecting the higher-order terms, we obtain [26]

$$\delta\dot{x} = f(x_0,u_0) + \left.\frac{\partial f}{\partial x}\right|_{x_0,u_0} \cdot \delta x + \left.\frac{\partial f}{\partial u}\right|_{x_0,u_0} \cdot \delta u - f(x_0,u_0),$$

which simplifies to

$$\delta\dot{x} = \left.\frac{\partial f}{\partial x}\right|_{x_0,u_0} \cdot \delta x + \left.\frac{\partial f}{\partial u}\right|_{x_0,u_0} \cdot \delta u\,. \tag{6}$$

Expression (6) is then applied to all non-linear first order differential equations in (5), yet with further simplifications:

(i) the roll rate (p) is assumed to be null;
(ii) the roll angle (ϕ) is taken as a parameter rather than a state;
(iii) actuator dynamics are not included in the model;
(iv) the wind velocity is considered to be zero;
(v) system parameters are considered constant at each operating point (frozen parameters).

The first two simplifications are due to the fact that roll control is either not applied or achieved by an additional system, the third one makes the relative velocity vector equal to the linear velocity vector expressed in the body frame, and the final one removes the existent dependencies of the parameters on the state variables when computing the Taylor derivatives.

Considering a generic reference trajectory, the resultant state-space representation follows:

$$\boldsymbol{\delta}\mathbf{x} = \begin{bmatrix}\delta u & \delta v & \delta w & \delta q & \delta r & \delta\theta & \delta\psi\end{bmatrix}^T, \quad \boldsymbol{\delta}\mathbf{u} = \begin{bmatrix}\delta\mu_p & \delta\mu_y\end{bmatrix}^T, \tag{7a}$$

$$\boldsymbol{\delta}\dot{\mathbf{x}}(t) = \mathbf{A}(t)\cdot\boldsymbol{\delta}\mathbf{x}(t) + \mathbf{B}(t)\cdot\boldsymbol{\delta}\mathbf{u}(t)\,, \tag{7b}$$

$$\mathbf{A}(t) = \begin{bmatrix} 0 & r_0 & -q_0 & -w_0 & v_0 & a_{16} & a_{17} \\ -r_0 & a_{22} & 0 & 0 & -u_0 & a_{26} & a_{27} \\ a_{31} & 0 & a_{33} & u_0 & 0 & a_{36} & a_{37} \\ a_{41} & 0 & a_{43} & a_{44} & 0 & 0 & 0 \\ 0 & a_{52} & 0 & 0 & a_{55} & 0 & 0 \\ 0 & 0 & 0 & c_{\phi_0} & -s_{\phi_0} & 0 & 0 \\ 0 & 0 & 0 & s_{\phi_0}/c_{\theta_0} & a_{85} & a_{86} & 0 \end{bmatrix}, \tag{7c}$$

$$\mathbf{B}(t) = \begin{bmatrix} -\frac{T}{m}\, s_{\mu_{p_0}}\, c_{\mu_{y_0}} & -\frac{T}{m}\, c_{\mu_{p_0}}\, s_{\mu_{y_0}} \\ \frac{T}{m}\, s_{\mu_{p_0}}\, s_{\mu_{y_0}} & -\frac{T}{m}\, c_{\mu_{p_0}}\, c_{\mu_{y_0}} \\ -\frac{T}{m}\, c_{\mu_{p_0}} & 0 \\ -\frac{T\,l}{J_t}\, c_{\mu_{p_0}} & 0 \\ -\frac{T\,l}{J_t}\, s_{\mu_{p_0}}\, s_{\mu_{y_0}} & \frac{T\,l}{J_t}\, c_{\mu_{p_0}}\, c_{\mu_{y_0}} \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \tag{7d}$$

with

$$a_{16} = g\, s_{\theta_0}\, c_{\psi_0}$$
$$a_{17} = g\, c_{\theta_0}\, s_{\psi_0}$$
$$a_{22} = \frac{\bar{q}\, S\, C_{Y_\beta}}{m\left(1 - \frac{v_0^2}{{V_0}^2}\right)^{1/2} V_0}$$
$$a_{26} = -g\,(s_{\phi_0}\, c_{\theta_0}\, c_{\psi_0} - c_{\phi_0}\, s_{\psi_0})$$
$$a_{27} = g\,(s_{\phi_0}\, s_{\theta_0}\, s_{\psi_0} + c_{\phi_0}\, c_{\psi_0})$$
$$a_{31} = q_0 + \frac{\bar{q}\, S\, C_{N_\alpha}\, w_0}{m\,({u_0}^2 + {w_0}^2)}$$
$$a_{33} = -\frac{\bar{q}\, S\, C_{N_\alpha}\, u_0}{m\,({u_0}^2 + {w_0}^2)}$$
$$a_{36} = -g\, c_{\phi_0}\, c_{\theta_0}\, c_{\psi_0}$$
$$a_{37} = -g\,(-c_{\phi_0}\, s_{\theta_0}\, s_{\psi_0} + s_{\phi_0}\, c_{\psi_0})$$
$$a_{41} = \frac{\bar{q}\, S\, d\, S.M\, C_{N_\alpha}\, w_0}{J_t\,({u_0}^2 + {w_0}^2)}$$
$$a_{43} = -\frac{\bar{q}\, S\, d\, S.M\, C_{N_\alpha}\, u_0}{J_t\,({u_0}^2 + {w_0}^2)}$$
$$a_{44} = \frac{\bar{q}\, S\, d^2\,(C_{m_q} + C_{m_{\dot{\alpha}}})}{2\, J_t\, V_0}$$
$$a_{52} = -\frac{\bar{q}\, S\, d\, S.M\, C_{Y_\beta}}{J_t\, V_0\left(1 - \frac{v_0^2}{{V_0}^2}\right)^{1/2}}$$
$$a_{55} = \frac{\bar{q}\, S\, d^2\,(C_{n_r} + C_{n_{\dot{\beta}}})}{2\, J_t\, V_0}$$
$$a_{85} = \frac{c_{\phi_0}}{c_{\theta_0}}$$
$$a_{86} = \frac{(q_0\, s_{\phi_0} + r_0\, c_{\phi_0})\, s_{\theta_0}}{{c_{\theta_0}}^2}$$

where $\mathbf{A}(t)$ and $\mathbf{B}(t)$ are the state-space matrices given by the first-order Taylor derivatives in (6) with respect to system states and inputs, respectively, calculated at the operating points. Due to the aforementioned simplifications, p and ϕ are not states of the linear time-varying system, even though they are physical variables in the complete non-linear model.

5. Architecture

To achieve a stable solution with accurate reference tracking for the pitch and yaw angles of a naturally unstable launcher, the integrated architecture in Fig. 3, comprising both the navigation and control systems, is proposed. The selected techniques to tackle the problems at hand are LQR control with integrative components (LQI) and complementary kinematic filtering with a close relation to Kalman filtering theory.

The navigation system relies on a set of on-board sensors, which measure relevant quantities associated with rocket flight, $\mathbf{y}$, and an estimator, based on complementary kinematic filters, which provides estimates on the state vector, $\hat{\mathbf{x}}$, given the measurements, $\mathbf{y}_r$, by filtering the noise and correcting the bias. The subset of the estimated state vector, $\hat{\mathbf{x}}$, used for feedback control is represented by $\hat{\mathbf{x}}_c$.

The control system is divided in two major blocks: feedforward control and LQI feedback control. Feedforward control consists in the pre-determined values for the system inputs, $\mathbf{u}_0$, that allow the vehicle to follow the reference trajectory under nominal conditions, i.e, without disturbances and model uncertainties. On the other hand, feedback LQI control is responsible for ensuring stability and accurate reference tracking (θ_d and ψ_d) in a real flight scenario.

Feedback control is implemented in the perturbation domain, meaning that the reference values of the states used for feedback, $\mathbf{x}_{c_0}$, are needed to retrieve the perturbed states according to $\boldsymbol{\delta}\hat{\mathbf{x}}_c = \hat{\mathbf{x}}_c - \mathbf{x}_{c_0}$. It acts on the perturbed states using the optimal gains calculated for each operating point through the use of the LQI control law and the respective state-space representation. To ensure a smooth time evolution in the control inputs, linear interpolation is used to schedule

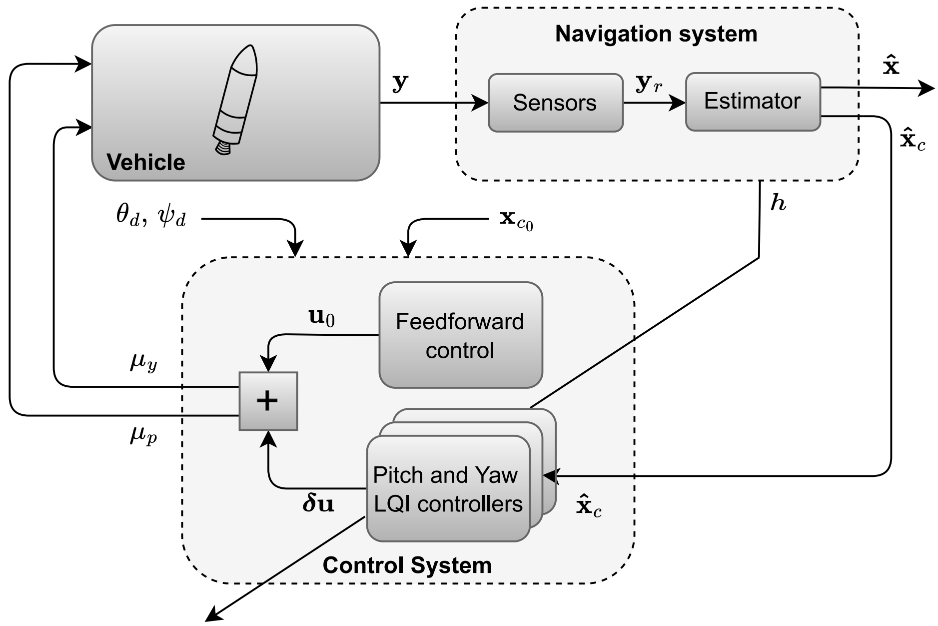


Fig. 3. System architecture.

the gains. The variable selected to interpolate the gains is the altitude, h, to avoid potential mismatches resulting from delays that could occur in a time-based interpolation. The scheduled controllers are represented in Fig. 3 by multiple block layers. By summing the feedforward and feedback control values, respectively $\mathbf{u}_0$ and $\boldsymbol{\delta}\mathbf{u}$, the control inputs, μ_p and μ_y, are obtained.

6. Navigation

As mentioned, the navigation system relies both on measurements from on-board sensors and an estimator. In this section, the selected sensor suite is detailed and the estimator is derived.

6.1. Sensor suite

To design a navigation system, it is necessary to select the on-board sensor suite. Sensors either provide a direct measurement on the required state variables or on other quantities that can then be used to estimate them. For launch vehicles, and taking into account the state variables to be estimated – position, linear and angular velocities, and Euler angles – it is common to use an Inertial Measurement Unit (IMU) combined with a Global Navigation Satellite System (GNSS) receiver. Barometers and magnetometers are also standard sensors installed on-board to provide measurements.

The IMU is composed by 3-axis accelerometers and gyroscopes. An accelerometer supplies a measure of the system's acceleration and can be used to determine the vehicle's velocity by integration. To do so, it is necessary to know the initial condition. Over time, the velocity measurement will drift from the true value due to the inherent noise and bias properties of the accelerometer. By combining the 3-axis accelerometers, a measurement on the linear acceleration vector in the body frame is obtained, $\mathbf{a}_r \in \mathbb{R}^3$. A gyroscope provides a measurement of the system's angular rate expressed in the body frame. The angular rate measurements, $\boldsymbol{\omega}_r \in \mathbb{R}^3$, can be integrated to determine an estimate of the system's attitude. Once again, the calculated attitude drifts boundlessly from the true attitude of the system due to the inherent noise and bias properties of the gyroscope.

If the 3-axis accelerometer is assumed to be measuring gravity alone, it is possible to calculate the pitch and yaw angles from the direction of the gravity vector. However, since the accelerometer is assumed to be measuring gravity alone, any added dynamic motion causes an error in the calculation of the system's pitch and yaw. A magnetometer can be used to obtain an estimate on the roll angle by comparing the measurement of the magnetic field surrounding the system to Earth's magnetic field. Hence, the two vector observations, gravity and magnetic, can be used to obtain an attitude solution, $\lambda_r = \begin{bmatrix}\phi_r & \theta_r & \psi_r\end{bmatrix}^T$.

A GNSS is a satellite configuration, or constellation, that provides coded satellite signals which are processed by a GNSS receiver inside the vehicle to calculate position, velocity, and time. In this paper, the position measurements by the GNSS receiver are assumed to be already translated into the Cartesian coordinates of the inertial frame and, in combination with the indirect altitude measurements from the barometer, yield the position readings available for the estimator, $\mathbf{p}_r \in \mathbb{R}^3$,

6.2. Estimator design

The estimator is based on the concept of complementary kinematic filtering, as presented in [27]. Relying only on the kinematics of the vehicle in attitude and position, two complementary filters can be used to obtain an estimate on the full state vector, with the guarantee of having an uniformly asymptotically stable tracking error around the origin. Furthermore, through the use of appropriate variable transformations, the gains can be computed for the time-invariant case resorting to the steady-state Kalman filter. In real time, the estimator gains are then recovered with expressions which have explicit dependency on the time-invariant gains.

In this work, a continuous time version of the kinematic filters proposed in [27] is derived, with the inclusion of gravity vector estimation to improve the attitude determination from vector observations by avoiding the aforementioned issue of corruption by the dynamic acceleration component.

6.2.1. Estimator architecture

Figure 4 details the estimator architecture, which is composed by the two complementary kinematic filters and an additional algorithm to determine the attitude from vector observations.

The Attitude Determination (AD) algorithm takes the magnetic field measurements, $\mathbf{m}_r$, the value of the Earth magnetic field in the inertial frame, ${}^i\mathbf{m}$, the estimated gravity vector in the body frame, ${}^b\hat{\mathbf{g}}$, and the

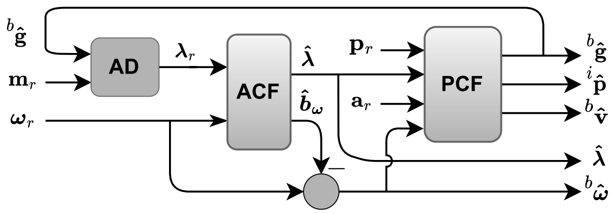


Fig. 4. Estimator architecture.

gravitational acceleration in the inertial frame, $^i\mathbf{g} = \begin{bmatrix} -g & 0 & 0 \end{bmatrix}^T$, to obtain an indirect measurement on the Euler angles, λ_r, essentially serving as a magneto-pendular sensor.

The first kinematic filter is the Attitude Complementary Filter (ACF), which uses the Euler angles readings, λ_r, from AD, and the measured angular rates from the gyroscope, $\boldsymbol{\omega}_r$, to provide a filtered attitude estimate, $\hat{\lambda}$, and an estimate on the bias of the three angular rates, $\mathbf{b}_\omega \in \mathbb{R}^3$, to correct the signal from the sensor.

The second one is the Position Complementary Filter (PCF), which merges the position readings, $\mathbf{p}_r$, obtained through the combination of the GNSS receiver and the altimeter, with the acceleration measurements from the accelerometer, $\mathbf{a}_r$, to provide an estimate on the inertial position, $^i\hat{\mathbf{p}}$, linear velocity vector, $^b\hat{\mathbf{v}}$, and gravity vector in the body frame $^b\hat{\mathbf{g}}$. For this filter, the accelerometer is assumed to have been previously calibrated to allow a completely observable estimation of the gravity vector induced component. For reference, [28] presents a method for offline accelerometer calibration.

6.2.2. Kalman filter

The Kalman filter is a widely used observer to tackle the estimation problem for linear dynamic systems [29]. When both the process and measurement associated with the estimated state are corrupted by random, independent, zero mean Gaussian white noise, the solution provided by the Kalman filter is statistically optimal with respect to a quadratic function of the estimation error. For this reason, it is also referred to as Linear Quadratic Estimator (LQE), and represents the dual of the LQR to the estimation problem.

In continuous time, the random process and observation are given by

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}\,\mathbf{x} + \mathbf{B}\,\mathbf{u} + \mathbf{G}\,\mathbf{w} \\ \mathbf{y} = \mathbf{C}\,\mathbf{x} + \mathbf{v} \end{cases},$$

where all terms are time dependant, $\mathbf{w}$ is the process noise (associated with the model), $\mathbf{v}$ is the measurement noise (associated with the sensors) and $\mathbf{G}$ is the process noise coupling matrix. These random noises are represented by the covariance matrices $\boldsymbol{\Xi} \succeq 0$ and $\boldsymbol{\Theta} \succ 0$ for the process and measurement noise, respectively.

Given the defined process, observation and noise properties, the Kalman filter is capable of providing an optimal state estimation according to the differential equation

$$\dot{\hat{\mathbf{x}}} = \mathbf{A}\,\hat{\mathbf{x}} + \mathbf{B}\,\mathbf{u} + \mathbf{L}\,(\mathbf{y} - \mathbf{C}\,\hat{\mathbf{x}}),$$

in which $\hat{\mathbf{x}}$ is the state estimate and $\mathbf{L}$ is the Kalman gain. Given an initial condition $\hat{\mathbf{x}}(t_0)$, the state estimate derivative $\dot{\hat{\mathbf{x}}}$ is recursively propagated by correcting the process with the state estimation error $(\mathbf{y} - \mathbf{C}\,\hat{\mathbf{x}})$ multiplied by the Kalman gain. The Kalman gain is given by

$$\mathbf{L} = \mathbf{P}\,\mathbf{C}^T\,\boldsymbol{\Theta}^{-1},$$

where $\mathbf{P}$ is the solution to the matrix Riccati differential equation

$$\dot{\mathbf{P}} = \mathbf{A}\,\mathbf{P} + \mathbf{P}\,\mathbf{A}^T + \mathbf{G}\,\boldsymbol{\Xi}\,\mathbf{G}^T - \mathbf{P}\,\mathbf{C}^T\,\mathbf{R}^{-1}\,\mathbf{C}\,\mathbf{P}. \tag{8}$$

If the process is time-varying, this equation has to be continuously propagated. However, for the steady-state case, $\dot{\mathbf{P}}$ is zero and (8) simplifies to the Algebraic Riccati Equation (ARE). For the ARE to have a unique positive definite solution $\mathbf{P}$, it is a sufficient condition that the pair $(\mathbf{A}, \mathbf{C})$ is observable.

6.2.3. AD

By using two non-collinear vector observations, expressed in both reference frames, it is possible to get an indirect measurement on the rotation matrix, $\mathbf{R}_r^T$, that transforms vectors from the inertial to the body frame and, consequently, the attitude of the vehicle expressed through the Euler angles [30]. The AD algorithm provides the Euler angles readings, λ_r, by implementing this technique, resorting to the magnetic field measurements, which are expressed in the body frame, $\mathbf{m}_r$, the value of the Earth magnetic field in the inertial frame, $^i\mathbf{m}$, the gravity vector in the body frame as estimated by the PCF, $^b\hat{\mathbf{g}}$, and the gravitational acceleration in the inertial frame, $^i\mathbf{g}$. By using the generic structure for this algorithm [31], the estimated rotation matrix for this case is

$$\mathbf{R}_r^T = \mathbf{r_1} \cdot \mathbf{s_1}^T + \mathbf{r_2} \cdot \mathbf{s_2}^T + \mathbf{r_3} \cdot \mathbf{s_3}^T, \tag{9}$$

where

$$\mathbf{r_1} = \frac{^b\hat{\mathbf{g}}}{\|\,^b\hat{\mathbf{g}}\|}, \quad \mathbf{r_2} = \frac{\mathbf{r_1} \times \mathbf{m_r}}{\mathbf{r_1} \times \mathbf{m_r}}, \quad \mathbf{r_3} = \mathbf{r_1} \times \mathbf{r_2},$$

$$\mathbf{s_1} = \frac{^i\mathbf{g}}{\|\,^i\mathbf{g}\|}, \quad \mathbf{s_2} = \frac{\mathbf{s_1} \times {}^i\mathbf{m}}{\|\mathbf{s_1} \times {}^i\mathbf{m}\|}, \quad \mathbf{s_3} = \mathbf{s_1} \times \mathbf{s_2}.$$

from which the Euler angle readings can be extracted, as described in [32]. Note that there is a biased consideration when choosing the vectors, with a preference being given to the gravity vector which serves as an anchor in the computation of the rotation matrix. However, this selection is not arbitrary as the gravity vector estimates will serve as an inclinometer, mostly influencing the pitch and yaw measurements, while the magnetic readings mostly influence the roll measurements — considering the vertical attitude to be the nominal condition.

6.2.4. ACF

For the ACF, the attitude kinematics, expressed in Euler angles, are rewritten as

$$\dot{\lambda} = \mathbf{Q}(\lambda)\,^b\boldsymbol{\omega}, \quad \mathbf{Q}(\lambda) = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta \end{bmatrix}. \tag{10}$$

Furthermore, it is assumed that the gyroscope readings are corrupted by zero mean Gaussian white-noise, $\mathbf{w}_\omega \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Xi}_\omega)$, and have a given bias, $\mathbf{b}_\omega$, which is assumed to be slowly time-varying and taken as constant for the process model, yielding

$$\boldsymbol{\omega}_r = {}^b\boldsymbol{\omega} + \mathbf{b}_\omega + \mathbf{w}_\omega, \quad \dot{\mathbf{b}}_\omega = 0. \tag{11}$$

By combining the attitude kinematics (10) and (11), the following representation of the process model in state-space form is obtained:

$$\begin{bmatrix} \dot{\lambda} \\ \dot{\mathbf{b}}_\omega \end{bmatrix} = \begin{bmatrix} \mathbf{0} & -\mathbf{Q}(\lambda) \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \lambda \\ \mathbf{b}_\omega \end{bmatrix} + \begin{bmatrix} \mathbf{Q}(\lambda) \\ \mathbf{0} \end{bmatrix} \boldsymbol{\omega}_r + \begin{bmatrix} -\mathbf{Q}(\lambda) \\ \mathbf{0} \end{bmatrix} \mathbf{w}_\omega. \tag{12}$$

Given the attitude kinematics (12), the following nonlinear feedback system is proposed as the ACF:

$$\begin{aligned} \begin{bmatrix} \dot{\hat{\lambda}} \\ \dot{\hat{\mathbf{b}}}_\omega \end{bmatrix} &= \begin{bmatrix} \mathbf{0} & -\mathbf{Q}(\lambda_r) \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \hat{\lambda} \\ \hat{\mathbf{b}}_\omega \end{bmatrix} + \begin{bmatrix} \mathbf{Q}(\lambda_r) \\ \mathbf{0} \end{bmatrix} \boldsymbol{\omega}_r \\ &+ \begin{bmatrix} \mathbf{Q}(\lambda_r)\left(\mathbf{L}_{1\lambda} + \dot{\mathbf{Q}}^{-1}(\lambda_r)\,\mathbf{Q}(\lambda_r)\right) \\ \mathbf{L}_{2\lambda} \end{bmatrix} (\mathbf{y}_\lambda - \hat{\mathbf{y}}_\lambda), \end{aligned} \tag{13a}$$

$$\mathbf{y}_\lambda = \mathbf{Q}^{-1}(\lambda_r)\,\lambda_r + \mathbf{v}_\lambda, \quad \hat{\mathbf{y}}_\lambda = \mathbf{Q}^{-1}(\lambda_r)\,\hat{\lambda}, \tag{13b}$$

where $\mathbf{y}_\lambda$ is the vector of observed Euler angles (from AD) transformed to the space of angular rate and corrupted by Gaussian white observation noise, $\boldsymbol{v}_\lambda \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Theta}_\lambda)$, and $\mathbf{L}_{1\lambda}, \mathbf{L}_{2\lambda} \in M(3,3)$ are constant feedback gain matrices.

In order to obtain the constant feedback gain matrices and prove the stability of the filter, let us consider the following time-invariant version of the system (12):

$$\begin{bmatrix} \dot{\mathbf{x}}_\lambda \\ \dot{\mathbf{x}}_{\mathbf{b}_\omega} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & -\mathbf{I} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{x}_\lambda \\ \mathbf{x}_{\mathbf{b}_\omega} \end{bmatrix} + \begin{bmatrix} \mathbf{I} \\ \mathbf{0} \end{bmatrix} \omega_r + \begin{bmatrix} -\mathbf{I} \\ \mathbf{0} \end{bmatrix} \mathbf{w}_\omega, \quad \mathbf{y}_x = \mathbf{x}_\lambda + \mathbf{v}_\lambda \tag{14}$$

which is equivalent to simplifying the attitude kinematics to the vertical case, $\lambda = [0\ 0\ 0]^T$.

Theorem 1. *Let $\mathbf{L}_{1\lambda}$ and $\mathbf{L}_{2\lambda}$ be the steady-state kalman gains for the system (14) and assume that the pitch angle is bounded, $|\theta| < \pi/2$. Then, the ACF (13) is uniformly asymptotically stable (UAS).*

Proof. By defining the estimation errors as $\tilde{\lambda} = \lambda - \hat{\lambda}$, $\tilde{\mathbf{b}}_\omega = \mathbf{b}_\omega - \hat{\mathbf{b}}_\omega$, the system describing the ACF (13) can be represented in terms of the estimation error dynamics:

$$\begin{bmatrix} \dot{\tilde{\lambda}} \\ \dot{\tilde{\mathbf{b}}}_\omega \end{bmatrix} = \begin{bmatrix} -\mathbf{Q}(\lambda_r)\left(\mathbf{L}_{1\lambda} + \dot{\mathbf{Q}}^{-1}(\lambda_r)\mathbf{Q}(\lambda_r)\right)\mathbf{Q}^{-1}(\lambda_r) & -\mathbf{Q}(\lambda_r) \\ -\mathbf{L}_{2\lambda}\mathbf{Q}^{-1}(\lambda_r) & \mathbf{0} \end{bmatrix} \begin{bmatrix} \tilde{\lambda} \\ \tilde{\mathbf{b}}_\omega \end{bmatrix} \tag{15}$$

By definition, the filter is said to be UAS if the origin of the system (15) is UAS.

As for the auxiliary time-invariant system (14), it is straightforward to infer that it is observable, hence, a steady-state Kalman filter applied to it, yields UAS error dynamics. Upon this realization, if there is a well-defined Lyapunov transform between the estimation error dynamics of both systems ((13) and the time-invariant kalman filter), then the ACF (13) is also UAS. The Lyapunov transform

$$\begin{bmatrix} \tilde{\lambda} \\ \tilde{\mathbf{b}}_\omega \end{bmatrix} = \mathbf{T} \begin{bmatrix} \tilde{\mathbf{x}}_\lambda \\ \tilde{\mathbf{x}}_{\mathbf{b}_\omega} \end{bmatrix}, \quad \mathbf{T} = \begin{bmatrix} \mathbf{Q}(\lambda_r) & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \tag{16}$$

is well defined [33] in the specified domain and, through algebraic manipulation not here presented, when applied to the error dynamics of the time-invariant filter leads to the error dynamics of the ACF (13). Hence, the ACF is UAS. □

With this, we have proven that the ACF is stable apart from the singularities in the attitude representation ($|\theta| = \pi/2$), while having a structure that simplifies its design. The time-invariant gains, $\mathbf{L}_{1\lambda}$ and $\mathbf{L}_{2\lambda}$, are determined by applying the steady-state Kalman filter to the auxiliary time-invariant system (14), with the design freedom being present in the noise covariance matrices, which can be tuned to obtained the desired performance. Although stable, the performance of the ACF will not be optimal when the attitude differs from the time-invariant condition, $\lambda = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$.

6.2.5. PCF

For the PCF, the process model proposed in [34] for navigation with calibrated accelerometer is used, which was proven to be uniformly completely observable. The position kinematics are rewritten in the body frame, yielding

$${}^b\dot{\mathbf{p}} = {}^b\mathbf{v} - \mathbf{S}({}^b\omega)\,{}^b\mathbf{p}, \tag{17a}$$

$${}^b\dot{\mathbf{v}} = {}^b\mathbf{a} - \mathbf{S}({}^b\omega)\,{}^b\mathbf{v}. \tag{17b}$$

Additionally, by considering the gravitational acceleration as locally constant in the inertial frame, the gravity vector time derivative expressed in the body frame is

$${}^b\dot{\mathbf{g}} = -\mathbf{S}({}^b\omega)\,{}^b\mathbf{g}. \tag{18}$$

The acceleration measurements are considered to be corrupted by zero mean Gaussian white noise, $\mathbf{w}_\mathbf{a} \sim \mathcal{N}(\mathbf{0}, \Xi_a)$, and the accelerometer is assumed to be calibrated, resulting in the following measurement equation:

$$\mathbf{a}_r = {}^b\mathbf{a} - {}^b\mathbf{g} + \mathbf{w}_\mathbf{a}. \tag{19}$$

It is noted that the accelerometer measures the inertial acceleration as well as the gravity vector, both expressed in the body frame.

By combining the position kinematics (17), the gravity vector time derivative (18), and the accelerometer Eq. (19), the process model for this filter can be written in state-space form:

$$\begin{bmatrix} {}^b\dot{\mathbf{p}} \\ {}^b\dot{\mathbf{v}} \\ {}^b\dot{\mathbf{g}} \end{bmatrix} = \begin{bmatrix} -\mathbf{S}({}^b\omega) & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & -\mathbf{S}({}^b\omega) & \mathbf{I} \\ \mathbf{0} & \mathbf{0} & -\mathbf{S}({}^b\omega) \end{bmatrix} \begin{bmatrix} {}^b\mathbf{p} \\ {}^b\mathbf{v} \\ {}^b\mathbf{g} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \\ \mathbf{0} \end{bmatrix} \mathbf{a}_r + \begin{bmatrix} \mathbf{0} \\ -\mathbf{I} \\ \mathbf{0} \end{bmatrix} \mathbf{w}_\mathbf{a}. \tag{20}$$

Given this model (20), the following nonlinear feedback system is proposed as the PCF:

$$\begin{bmatrix} {}^b\dot{\hat{\mathbf{p}}} \\ {}^b\dot{\hat{\mathbf{v}}} \\ {}^b\dot{\hat{\mathbf{g}}} \end{bmatrix} = \begin{bmatrix} -\mathbf{S}({}^b\hat{\omega}) & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & -\mathbf{S}({}^b\hat{\omega}) & \mathbf{I} \\ \mathbf{0} & \mathbf{0} & -\mathbf{S}({}^b\hat{\omega}) \end{bmatrix} \begin{bmatrix} {}^b\hat{\mathbf{p}} \\ {}^b\hat{\mathbf{v}} \\ {}^b\hat{\mathbf{g}} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \\ \mathbf{0} \end{bmatrix} \mathbf{a}_r + \begin{bmatrix} \mathbf{R}^T(\hat{\lambda})\,\mathbf{L}_{1p} \\ \mathbf{R}^T(\hat{\lambda})\,\mathbf{L}_{2p} \\ \mathbf{R}^T(\hat{\lambda})\,\mathbf{L}_{3p} \end{bmatrix} (\mathbf{y}_p - \hat{\mathbf{y}}_p), \tag{21a}$$

$$\mathbf{y}_p = \mathbf{p}_r + \mathbf{v}_\mathbf{p}, \quad \hat{\mathbf{y}}_p = {}^i\hat{\mathbf{p}} = \mathbf{R}(\hat{\lambda})\,{}^b\hat{\mathbf{p}}, \tag{21b}$$

where $\mathbf{y}_p$ is the vector of position observations, obtained from the GNSS receiver and the altimeter, assumed to be corrupted by zero mean Gaussian white observation noise, $\mathbf{v}_\mathbf{p} \sim \mathcal{N}(\mathbf{0}, \Theta_p)$, and $\mathbf{L}_{1p}$, $\mathbf{L}_{2p}$, $\mathbf{L}_{3p} \in M(3,3)$ are constant feedback gain matrices. It is important to note that, in the process model, the skew-symmetric matrices are computed using the estimated angular velocity vector from the ACF (removed bias). Furthermore, the feedback component contains the necessary transformation of the observations from the inertial to the body frame, with the rotation matrix, $\mathbf{R}^T(\hat{\lambda})$, being computed using the estimated Euler angles from the ACF.

Similarly to what was done for the ACF, to obtain the constant feedback gain matrices and prove the stability of the filter, let us consider the following time-invariant version of the system (20):

$$\begin{bmatrix} \dot{\mathbf{x}}_\mathbf{p} \\ \dot{\mathbf{x}}_\mathbf{v} \\ \dot{\mathbf{x}}_\mathbf{g} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{x}_\mathbf{p} \\ \mathbf{x}_\mathbf{v} \\ \mathbf{x}_\mathbf{g} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \\ \mathbf{0} \end{bmatrix} \mathbf{a}_r + \begin{bmatrix} \mathbf{0} \\ -\mathbf{I} \\ \mathbf{0} \end{bmatrix} \mathbf{w}_\mathbf{a}, \quad \mathbf{y}_x = \mathbf{x}_\mathbf{p} + \mathbf{v}_\mathbf{p}, \tag{22}$$

which is equivalent to taking the angular velocity as zero.

Theorem 2. *Let $\mathbf{L}_{1p}$, $\mathbf{L}_{2p}$, and $\mathbf{L}_{3p}$ be the steady-state kalman gains for the system (22). Then, the PCF (21) is uniformly asymptotically stable (UAS).*

Proof. By defining the estimation errors as ${}^b\tilde{\mathbf{p}} = {}^b\mathbf{p} - {}^b\hat{\mathbf{p}}$, ${}^b\tilde{\mathbf{v}} = {}^b\mathbf{v} - {}^b\hat{\mathbf{v}}$, and ${}^b\tilde{\mathbf{g}} = {}^b\mathbf{g} - {}^b\hat{\mathbf{g}}$, the system describing the PCF (21) can be represented in terms of the estimation error dynamics:

$$\begin{bmatrix} {}^b\dot{\tilde{\mathbf{p}}} \\ {}^b\dot{\tilde{\mathbf{v}}} \\ {}^b\dot{\tilde{\mathbf{g}}} \end{bmatrix} = \begin{bmatrix} -\mathbf{S}({}^b\hat{\omega}) - \mathbf{R}^T(\hat{\lambda})\,\mathbf{L}_{1p}\mathbf{R}(\hat{\lambda}) & \mathbf{I} & \mathbf{0} \\ -\mathbf{R}^T(\hat{\lambda})\,\mathbf{L}_{2p}\mathbf{R}(\hat{\lambda}) & \mathbf{S}({}^b\hat{\omega}) & \mathbf{I} \\ -\mathbf{R}^T(\hat{\lambda})\,\mathbf{L}_{3p}\mathbf{R}(\hat{\lambda}) & \mathbf{0} & \mathbf{S}({}^b\hat{\omega}) \end{bmatrix} \begin{bmatrix} {}^b\tilde{\mathbf{p}} \\ {}^b\tilde{\mathbf{v}} \\ {}^b\tilde{\mathbf{g}} \end{bmatrix} \tag{23}$$

By definition, the filter is said to be UAS if the origin of the system (23) is UAS.

As for the auxiliary time-invariant system (22), it is straightforward to infer that it is observable, hence, a steady-state Kalman filter applied to it, yields UAS error dynamics. Upon this realization, if there is a well-defined Lyapunov transform between the estimation error dynamics of both systems ((21) and the time-invariant kalman filter), then the PCF (21) is also UAS. The Lyapunov transform

$$\begin{bmatrix} {}^b\tilde{\mathbf{p}} \\ {}^b\tilde{\mathbf{v}} \\ {}^b\tilde{\mathbf{g}} \end{bmatrix} = \mathbf{T} \begin{bmatrix} \tilde{\mathbf{x}}_\mathbf{p} \\ \tilde{\mathbf{x}}_\mathbf{v} \\ \tilde{\mathbf{x}}_\mathbf{g} \end{bmatrix}, \quad \mathbf{T} = \begin{bmatrix} \mathbf{R}(\hat{\lambda}) & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}(\hat{\lambda}) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{R}(\hat{\lambda}) \end{bmatrix} \tag{24}$$

is well defined [33] and, through algebraic manipulation not here presented, when applied to the error dynamics of the time-invariant filter leads to the error dynamics of the PCF (21). Hence, the PCF is UAS. □

As done for the ACF, the PCF was proven to be stable and the constant feedback gain matrices are obtained by applying the steady-state Kalman filter to the time-invariant version of the system (22), with the error covariance matrices as design freedom.

7. Control

In this section, the feedforward and feedback control components are described. For feedback control, an LQR with integrative action (LQI) is proposed and then particularized into a decoupled version.

7.1. Feedforward control

Given the natural instability of the system, and its time-varying nature, finding the time evolution of the nominal control inputs, $\mathbf{u}_0$, that places the vehicle in the desired trajectory can pose a difficult task. A first approach could be to solve the non-linear differential equations of the system (5) over time such that the attitude reference is correctly followed. However, this is a mathematically complex problem that would require a numerical solution.

A more practical strategy is to rely on a simulation model, based on the detailed physical model (5), and use a controller that stabilizes the plant and ensures that the reference trajectory is followed in simulation. The resulting actuation values can then be stored to later use in real-time as feedforward control. As long as the model is sufficiently accurate and the varying parameters are approximately known, this approach can be valid.

Since the simulated flight is disturbance-free and no uncertainties are added to the model, a simple PID controller per degree of freedom (pitch and yaw), with constant gains, can achieve this task.

7.2. Feedback control

Feedback control uses a subset of the state estimates from the navigation system, $\hat{\mathbf{x}}_c$, to stabilize the plant and provide reference tracking of the desired pitch and yaw angles, θ_d and ψ_d. Given the nature of the TVC actuation, trying to control the linear velocities would conflict with the attitude control, specially for non-zero attitude references, therefore, $\mathbf{x}_c = \begin{bmatrix} q & r & \theta & \psi \end{bmatrix}^T$.

7.2.1. LQR

The LQR is an optimal controller for linear systems that finds the gain matrix $\mathbf{K}$ in the linear control law $\mathbf{u} = -\mathbf{K}\,\mathbf{x}$, which minimizes the quadratic cost functional

$$J = \int_t^T \left[\, \mathbf{x}^T(\tau)\,\mathbf{Q}\,\mathbf{x}(\tau) + \mathbf{u}^T(\tau)\,\mathbf{R}\,\mathbf{u}(\tau) \,\right] d\tau,$$

where $\mathbf{Q} \succeq 0$ and $\mathbf{R} \succ 0$. In the cost functional J, the quadratic form $\mathbf{x}^T\mathbf{Q}x$ represents a penalty on the deviation of the state $\mathbf{x}$ from the origin, and the term $\mathbf{u}^T\mathbf{R}u$ represents the cost of control, making $\mathbf{Q}$ and $\mathbf{R}$ the tuning parameters for the resulting controller.

It can be shown [35] that for the infinite-horizon, or steady-state, version ($T = \infty$), the solution to this optimization problem, which guarantees closed-loop asymptotic stability, is the constant gain matrix

$$\mathbf{K} = \mathbf{R}^{-1}\,\mathbf{B}^T\,\mathbf{M},$$

where $\mathbf{M}$ is the solution to the ARE,

$$\mathbf{M}\,\mathbf{A} + \mathbf{A}^T\,\mathbf{M} - \mathbf{M}\,\mathbf{B}\,\mathbf{R}^{-1}\,\mathbf{B}^T\,\mathbf{M} + \mathbf{Q} = \mathbf{0}\,. \tag{25}$$

In order for the ARE (25) to have a unique, positive definite solution $\mathbf{M}$, it is a sufficient condition that the system defined by the pair ($\mathbf{A}$, $\mathbf{B}$) is controllable.

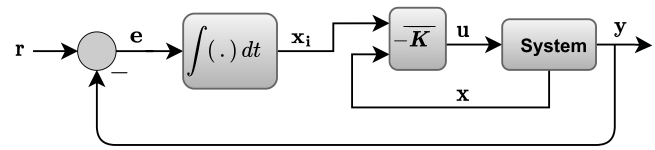


Fig. 5. Generic LQI control scheme.

7.2.2. LQR with integrative component (LQI)

The LQR feedback control law, applied to the system under study, would ideally drive the states in the perturbation domain to zero, ensuring that the nominal values throughout the trajectory would be followed. However, it does not guarantee a zero tracking error for non-zero attitude references, $\lambda_d = \begin{bmatrix} \theta_d & \psi_d \end{bmatrix}^T$. In order to minimize reference tracking error, and to increase the robustness of the controller to model uncertainties and external disturbances, an integrative component that acts on the attitude tracking error is added.

To obtain this controller using the already detailed LQR technique, it is only necessary to modify the state-space matrices when calculating the ARE. Generically, the closed-loop control with LQI follows the scheme in Fig. 5. Let the difference between the reference signal, $\mathbf{r}$, and the output of the system, $\mathbf{y}$, (the tracking error) be the time derivative of the state-space variables that result from adding the integrative component, $\mathbf{x_i}$. The state-space representation of the resulting regulator is obtained by combining the open-loop state-space representation with the feedback law, yielding

$$\dot{\mathbf{z}} = \left(\begin{bmatrix} \mathbf{A} & \mathbf{0} \\ -\mathbf{C} & \mathbf{0} \end{bmatrix} - \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix} \overline{\mathbf{K}} \right) \mathbf{z} + \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix} \mathbf{r},$$

where $\mathbf{z} = \begin{bmatrix} \mathbf{x} & \mathbf{x_i} \end{bmatrix}^T$ is the augmented state vector and $\mathbf{C}$ is the output matrix that selects the output of the system, i.e, the states for reference tracking, from the original state vector ($\mathbf{y} = \mathbf{C}\,\mathbf{x}$). The optimal gain is $\overline{\mathbf{K}} = \begin{bmatrix} \mathbf{K} & \mathbf{K_i} \end{bmatrix}^T$, where $\mathbf{K}$ is the original gain matrix for the state variables, and $\mathbf{K_i}$ is the gain matrix for the integrative components, and can be obtained by solving the ARE using the rearranged state-space matrices

$$\overline{\mathbf{A}} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ -\mathbf{C} & \mathbf{0} \end{bmatrix}, \quad \overline{\mathbf{B}} = \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix}.$$

Since the system under study is time-varying, the ARE has to be solved for models coming from each linearization point, resulting in a set of gain matrices to be selected, or scheduled, throughout the flight. Moreover, it is important to note that the state-space representation obtained is expressed in the perturbation domain. The augmented state-vector is

$$\boldsymbol{\delta}\mathbf{z} = \begin{bmatrix} \delta u & \delta v & \delta w & \delta q & \delta r & \delta\theta & \delta\psi & \delta\theta_i & \delta\psi_i \end{bmatrix}^T,$$

where $\delta\theta_i$ and $\delta\psi_i$ are the states associated with the integrative components. The $\mathbf{C}$ matrix is given by

$$\mathbf{C} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

in order to select $\delta\theta$ and $\delta\psi$ as the variables for reference tracking.

Given the order of the augmented system and the number of inputs, each gain matrix $\overline{\mathbf{K}}$ will be of dimension 2 by 9, however, since partial feedback is used, $\boldsymbol{\delta}\mathbf{z}_c = \begin{bmatrix} \delta q & \delta r & \delta\theta & \delta\psi & \delta\theta_i & \delta\psi_i \end{bmatrix}^T$, the columns associated with the linear velocities are removed, yielding a 2 by 6 matrix, with $\mathbf{K}$ being 2 by 4 and $\mathbf{K_i}$ being 2 by 2. The implementation of the resultant pitch and yaw controller (seen in the architecture diagram of Fig. 3) is detailed in Fig. 6. The integral component acts on the tracking error for the pitch and yaw angles, which are a subset of $\mathbf{x}_c$ represented by $\lambda_c = \begin{bmatrix} \theta & \psi \end{bmatrix}^T$. The attitude command λ_d is given in absolute pitch and yaw values, meaning that it has to be transformed into a relative command with respect to the values for the reference trajectory according to $\delta\lambda_d = \lambda_d - \lambda_{c_0}$.

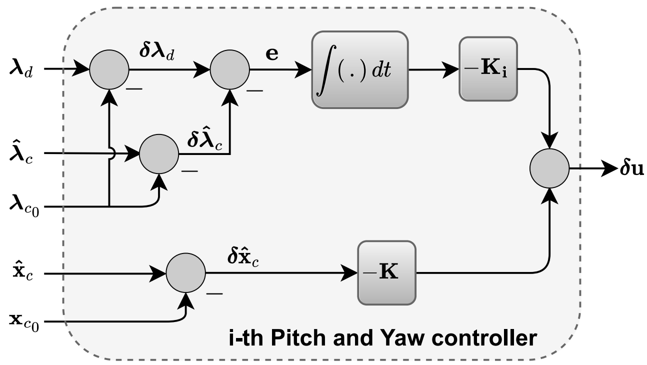


Fig. 6. Pitch and yaw LQI controller.

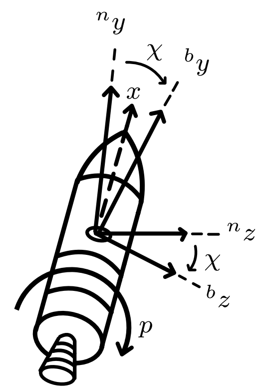


Fig. 7. Non-spinning frame {n}.

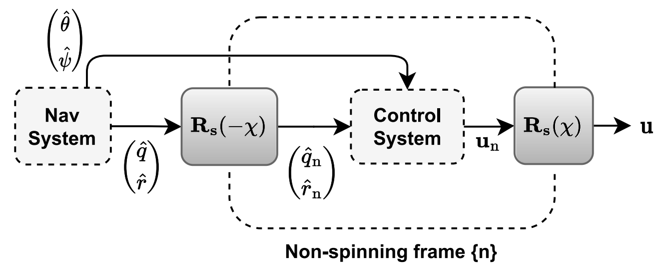


Fig. 8. Spin correction for the control system.

Once again, it is important to recall that the gain matrices $\mathbf{K}$ and $\mathbf{K}_\mathrm{i}$ are obtained for each operating point and are scheduled throughout the flight via linear interpolation with respect to altitude.

7.3. Decoupled control and spin correction

Considering the state-space representation of the system (7), it is possible to identify the conditions under which it can be separated into two decoupled modes: the lateral, composed by the state vector $\mathbf{x}_\mathrm{lat} = \begin{bmatrix} v & r & \psi \end{bmatrix}^T$ and the input μ_y; and the longitudinal, composed by the state vector $\mathbf{x}_\mathrm{lon} = \begin{bmatrix} u & w & q & \theta \end{bmatrix}^T$ and the input μ_p. Besides the assumption of a null roll rate, $p = 0$, a condition that allows for decoupling is to consider a reference trajectory restricted to one plane, for instance, the pitch plane. By doing so, the nominal values of the lateral states and input are zero and a decoupled state-space representation is easily derived from (7).

Using the decoupled state-space representation, the gains for the longitudinal and lateral modes, $\overline{\mathbf{K}}_\mathrm{lon}$ and $\overline{\mathbf{K}}_\mathrm{lat}$, can be obtained by applying the previously detailed LQI control law, particularized to the state-space matrices associated with each individual mode. The implementation of the resultant control system is equivalent to the one in Fig. 6, but now each control input is calculated separately using the gains, state estimates, and references for each mode, yielding two decoupled scheduled controllers.

7.3.1. Spin correction

The derived control system relies on the assumption that the roll rate, or spinning motion, is null ($p = 0$). This can be valid if an additional roll control system is used, for instance through reaction control devices. However, it cannot be guaranteed that spinning motion does not occur, given that such system can be designed to limit and not eliminate spin, or that disturbances may cause its appearance. Furthermore, it is a possibility only to have pitch and yaw control and use spinning motion for passive stabilization through the gyroscopic effect. In this way, it is important to consider the possibility of a non-zero roll rate, p, and add the necessary corrections to the system so that it can still perform under that condition. In this work, we decided to correct the actuation given by the original control law, as opposed to rewrite the linearized dynamics including the roll rate p and derive a new control law.

Firstly, an additional frame of reference is defined: the non-spinning frame {n}. This frame of reference is attached to the body but it does not rotate with respect to the x-axis, which is the spinning axis of rotation in the original body frame {b}. This is the frame where the states used in feedback, $\mathbf{x}_c$, and the control inputs, $\mathbf{u}$, will be defined according to the original control law. With the appearance of spinning motion, the body frame will rotate with respect to the non-spinning frame, with the instant angle of rotation being represented by χ, as depicted in Fig. 7. In most scenarios, χ will be very similar to ϕ, since for small angles the roll angle approximately coincides with the x-axis body rotation.

The appearance of the angle χ means that the TVC actuation is rotated, as well as the measurements of the pitch and yaw angular velocities, both expressed in {b}. Therefore, the estimates of the angular velocities, $\hat q$ and $\hat r$, have to be translated from {b} to {n} before passing to the control system, and the input vector computed in the non-spinning frame, $\mathbf{u}_\mathrm{n}$, has to be translated to {b}, according to scheme in Fig. 8. These translations are simply given by a positive or negative instantaneous rotation of χ around the x-axis:

$$\mathbf{R}_\mathrm{s}(\chi) = \begin{bmatrix} c_\chi & -s_\chi \\ s_\chi & c_\chi \end{bmatrix}, \quad \mathbf{R}_\mathrm{s}(-\chi) = \mathbf{R}_\mathrm{s}{}^T(\chi),$$

yielding,

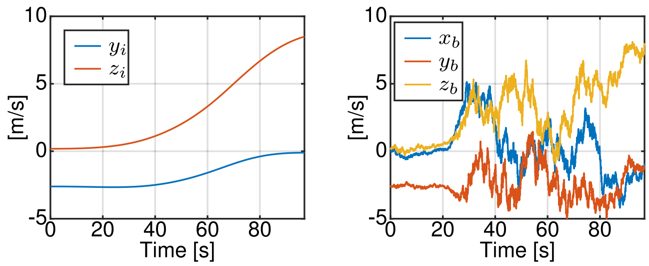
$$\begin{pmatrix} \hat q_\mathrm{n} \\ \hat r_\mathrm{n} \end{pmatrix} = \begin{pmatrix} \hat q\, c_\chi + \hat r\, s_\chi \\ -\hat q\, s_\chi + \hat r\, c_\chi \end{pmatrix},$$

$$\begin{pmatrix} \mu_p \\ \mu_y \end{pmatrix} = \begin{pmatrix} \mu_{p_\mathrm{n}}\, c_\chi - \mu_{y_\mathrm{n}}\, s_\chi \\ \mu_{p_\mathrm{n}}\, s_\chi + \mu_{y_\mathrm{n}}\, c_\chi \end{pmatrix}.$$

With this correction method, both the coupled dynamics caused by the spinning motion and the potential lack of axial symmetry are disregarded by the control system. Therefore, its validity has to be verified for the vehicle under study, taking into account the maximum expected spin rate.

8. Implementation in simulation

To test the proposed architecture, a realistic simulation environment, composed by the 6 DoF non-linear model, the integrated architecture, and the environmental properties, was implemented in Matlab&Simulink®. Additionally, a reference vehicle and trajectory had to be selected. In this section, the simulation environment, the reference vehicle and trajectory, and the implementation details of the architecture are presented.



(a) Average wind in inertial frame (b) Total wind in body frame

Fig. 9. Average horizontal wind and total wind.

8.1. Simulation environment

The simulation environment follows the structure of the architecture previously shown in Fig. 3, with additional components for the generation of the model parameters and environmental properties.

The environmental properties are generated by the atmospheric, wind, and gravitational models. The 1976 U.S standard atmosphere model [36] was implemented, which describes the evolution of temperature and pressure with altitude using average annual values, from which density and speed of sound are derived. Wind is introduced through the summation of the average horizontal wind components from the U.S Naval Research Laboratory horizontal wind model with a stochastic component (wind gusts) added from the Dryden model (Fig. 9), both available as Simulink blocks. Finally, the gravitational model is implemented according to the equations in Section 3.2.

Several varying model parameters have to be computed during simulation. The ideal thrust force and mass flow rate are taken as pre-calculated inputs, and the static, atmospheric pressure-dependant thrust component is added during the simulation. The aerodynamic properties, i.e., the aerodynamic coefficients and derivatives, and centre of pressure location, are stored in look-up tables and are selected according to the instant values of the aerodynamic angles and Mach number. The mass properties are also computed during the simulation, including the mass, inertia, and centre of mass, which vary due to the propellant consumption.

The block corresponding to the rocket model in Fig. 3 is responsible for computing the non-linear equations of motion, as presented in Section 3.2, using the time-varying model parameters and environmental properties. It is important to note that some assumptions were used when deriving the model and, although considered valid for design, can have an impact on the expected performance, obtained in simulation, when in a real case scenario. Elastic modes might be excited by the control action if the associated frequencies are similar, causing undesired oscillatory behaviour; asymmetries may cause the centre of mass to be dislocated from the x-axis of the body, which imposes additional effort on the control action; and non-linear aerodynamic effects may cause unexpected behaviour, as well as unaccounted effects caused by the rotation of the Earth, such as the Coriolis acceleration.

8.2. Reference vehicle

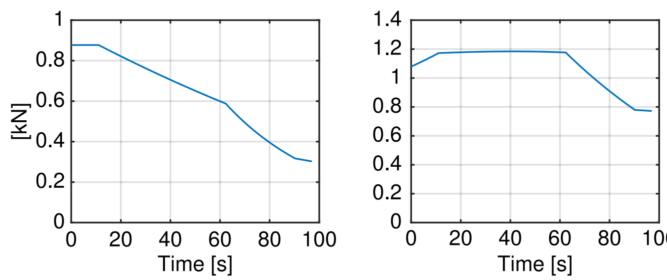
The reference vehicle was obtained through a preliminary design of a low-cost, solid motor rocket to serve as a testing platform for TVC technology. The vehicle is designed to have a burning phase coinciding with the full duration of the climb, so that TVC can be used to control its attitude until close to the apogee. It is also required for the terminal velocity to be inside a safe range to allow the correct activation of a recovery system. To meet these design requirements, the thrust produced by the motor is adjusted by iteratively testing different solid motor parameters, and the flight for a vertical undisturbed trajectory is simulated resorting to the simulation model. Tables 1 and 2 respectively present the main vehicle characteristics and the simulation results, while Fig. 10 details the ideal thrust and thrust-to-weight ratio.

Table 1
Main vehicle characteristics.

Total mass	82.9 kg
Dry mass	40.0 kg
Length	3.57 m
Max diameter	24 cm

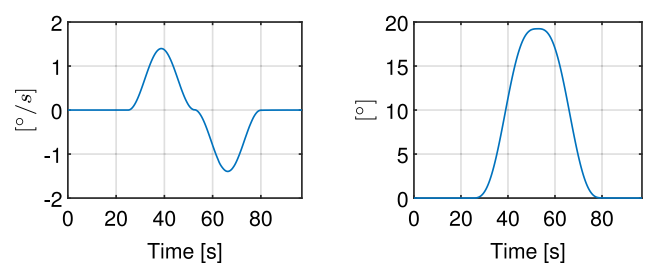
Table 2
Vertical trajectory parameters.

Apogee	4945 m
Max velocity	82 m/s
Max acceleration	1.7 m/s^2
Time to apogee	100 s



(a) Ideal thrust (b) Thrust-to-weight ratio

Fig. 10. Ideal thrust and thrust-to-weight ratio.



(a) Reference pitch rate (q_0) (b) Reference pitch angle (θ_0)

Fig. 11. Reference pitch rate (q_0) and angle (θ) over time.

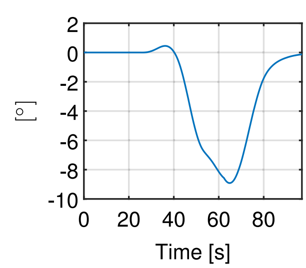


Fig. 12. Nominal pitch control input (μ_{p_0}).

8.3. Reference trajectory

Regarding the attitude reference that defines the reference trajectory, a varying pitch trajectory, in which the controller restricts the motion to the pitch plane (yaw equal to zero) and makes the vehicle deviate from the vertical to later recover it, is selected. In this way, the apogee is reached further away from the launch site, increasing safety, and an overall demanding scenario is presented to the system. Figure 11 shows the reference pitch rate and angle over time. The feedforward control inputs are computed as stated in Section 7.1, yielding the nominal actuation present in Fig. 12. The PID gains were set to $k_p = -10$, $k_i = -20$, and $k_d = -5$.

Table 3
Covariance and gain matrices of the filters.

	Cov. matrices	Filter gains
ACF	$\Xi_\omega = \sigma_\omega^2\mathbf{I}$ $\Xi_b = 4\times10^{-11}\mathbf{I}$ $\Theta_\lambda = 10^{-7}\mathbf{I}$	$\mathbf{L}_{1\lambda} = 1.93\mathbf{I}$ $\mathbf{L}_{2\lambda} = -0.02\mathbf{I}$
PCF	$\Xi_p = 10^{-2}\mathbf{I}$ $\Xi_a = \sigma_a^2\mathbf{I}$ $\Xi_g = 10^{-2}\mathbf{I}$ $\Theta_p = \mathbf{D}(\sigma_h{}^2, \sigma_p{}^2, \sigma_p{}^2)$	$\mathbf{L}_{1_p} = \mathbf{D}(0.94, 0.54, 0.54)$ $\mathbf{L}_{2_p} = \mathbf{D}(0.44, 0.15, 0.15)$ $\mathbf{L}_{3_p} = \mathbf{D}(0.1, 0.02, 0.02)$

8.4. Architecture implementation and parameters

In this section, the implementation of the architecture in simulation is discussed, and the parameters, obtained after tuning, are presented. These include the gains for both the navigation and control systems, as well as the models used to represent the on-board sensors and the actuators' dynamics.

8.4.1. Navigation system

The implementation of the navigation system follows the scheme in Fig. 4. Besides the components detailed on the scheme, the sensors are represented by adding noise to the exact value of the state variables. More specifically, zero-mean additive white Guassian noise is added to the measurements, sampled at 100 Hz. The assumed standard deviations for each sensor were: $\sigma_a = 0.014\,\mathrm{m\,s^{-2}}$ for the accelerometer, $\sigma_\omega = 0.035\,°/\mathrm{s}$ for the rate gyro, $\sigma_h = 1\,\mathrm{m}$ for the altimeter, $\sigma_p = 5\,\mathrm{m}$ for the GNSS receiver, and $\sigma_m = 140\,\mathrm{nT}$ for the magnetometer. Additionally, the rate gyro bias is set as $\mathbf{b}_\omega = \begin{bmatrix}-0.1 & 0.2 & 0.1\end{bmatrix}^T$ in degrees per second, and is driven by zero-mean white Gaussian noise to simulate the slowly time-varying nature. The noise intensity can be fine-tuned according to the change rate of the bias.

Considering the noise properties described above, the noise covariance matrices of both complementary filters were tuned and the time-invariant gain matrices were obtained, with all values shown in Table 3. While some weights have a direct correspondence with the noise covariances of the on-board sensors, others act as tuning knobs, which can be adjusted by analysing simulation results or by using real flight data. A noteworthy mention is the one associated with the rate gyro bias, Ξ_b: although assumed constant for the process model ($\dot{\mathbf{b}}_\omega = 0$), a small covariance value is used on the filter design so that a small correspondent gain, $\mathbf{L}_{2\lambda}$, allows the system to track slowly time-varying bias. This property makes it necessary to have an initial calibration period before the start of the mission so that there is time for the bias estimates to converge to the true values.

8.4.2. Control system

The implementation of the control system follows the schemes in Figs. 6 and 8, with an additional component to perform the altitude based gain scheduling.

The design degree of freedom is the selection of the tuning matrices $\mathbf{Q}$ and $\mathbf{R}$. First of all, setting all non-diagonal entries to zero, and only focusing on the diagonal ones, allows for a more intuitive matrix selection given by the "penalty" method [35]. According to this method, the diagonal entries of the $\mathbf{Q}$ matrix will determine the relative importance of the state variables in terms of origin tracking performance, while the diagonal entries of the $\mathbf{R}$ matrix allow to directly adjust the control effort for each input. Therefore, the weighting matrices have the following generic format, $\mathbf{Q} = \mathbf{D}(0, 0, 0, q_q, q_r, q_\theta, q_\psi, q_{\theta_i}, q_{\psi_i})$, $\mathbf{R} = \mathbf{D}(r_{\mu_p}, r_{\mu_y})$, where the terms associated with the linear velocities in the $\mathbf{Q}$ matrix are set to zero since those variables are not used for feedback control. The matrix entries can differ between operating points and are iteratively adjusted by analysing the closed-loop poles and the step response of the system in the linear domain.

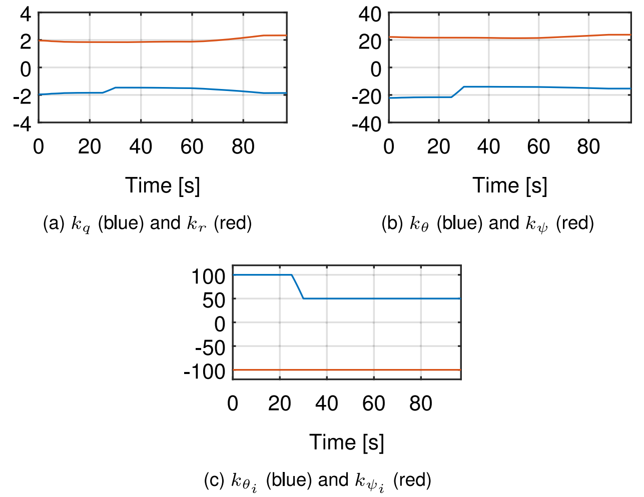


Fig. 13. Controller gains over time.

The decoupled version of the controller is implemented, yielding two separate gain matrices, one for each mode: $\overline{\mathbf{K}}_{\mathrm{lon}} = \begin{bmatrix}k_q & k_\theta & k_{\theta_i}\end{bmatrix}^T$ and $\overline{\mathbf{K}}_{\mathrm{lat}} = \begin{bmatrix}k_r & k_\psi & k_{\psi_i}\end{bmatrix}^T$. The altitude-based gain scheduling is performed through the linear interpolation Simulink block, resulting in the time evolution of the control gains, throughout the nominal trajectory, depicted in Fig. 13, obtained after tuning the $\mathbf{Q}$ and $\mathbf{R}$ matrices. The gains remain approximately constant given that the tuning matrices were left constant for all operating points, except for the ones associated with the longitudinal mode during the varying pitch section, which were tuned in order to reduce the control effort and avoid saturation.

The actuator dynamics are modelled using a first-order transfer function for each input (μ_p and μ_y), considering a servo-actuated system. The transfer function is

$$\frac{\mu_{resp}}{\mu} = \frac{1}{\tau s + 1},$$

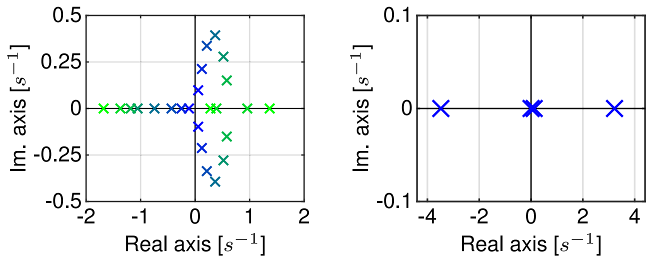
where μ_{resp} is the actuator angular response and τ is the time constant. Additionally, servo motors typically have a maximum rotation speed, which is modelled by a rate limiter block in Simulink. The time constant and maximum rotation speed were set to 0.02 s and 1 full rotation per second, respectively, considering a standard high grade servo motor.

9. Linear domain analysis

Using the linear representation of the system (7) and the reference values of its states, inputs, and parameters, it is possible to derive both the open-loop and closed-loop stability and response in the linear domain. For a time-varying system, determining the location of the poles throughout the reference trajectory does not provide a mathematical stability proof, however, the study is carried out to understand the behaviour of the system throughout the flight. Given the symmetry of the vehicle, and the fact that the reference trajectory is inside the pitch plane, the study is performed for the longitudinal mode.

9.1. Open-loop stability

Figure 14 details the pole evolution (from blue to green) during the initial vertical section (up to 25 s) and the poles at $t = 60\,\mathrm{s}$, which exemplifies the distribution type during the varying pitch section. By evaluating the location of the open-loop poles some conclusions can be made. Firstly, the system is naturally unstable, which was expected due to negative static stability margin caused by the absence of aerodynamic fins. Secondly, the system displays natural unstable oscillatory



(a) Up to $t = 25$ s (b) At $t = 60$ s

Fig. 14. Open-loop poles. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

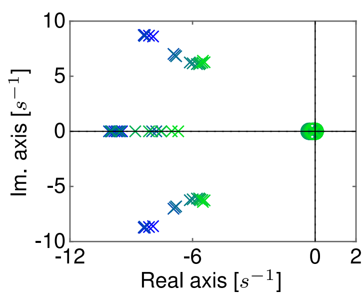


Fig. 15. Closed-loop poles and zeros.

Table 4
Closed-loop step response parameters.

Op. point	Rise time (s)	Settling time (s)	Overshoot (%)
$t = 5$ s	0.27	0.45	0.57
$t = 35$ s	0.34	0.57	0.12
$t = 65$ s	0.33	0.53	1.76
$t = 95$ s	0.37	0.61	0.80

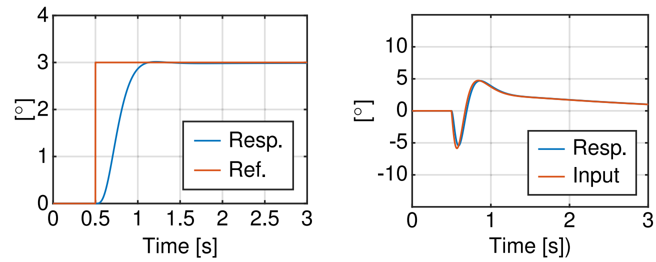
behaviour during the first seconds, after which all poles are located in the real axis. Finally, it is concluded that the velocity of the vehicle is a driving factor for the response of the system: at higher velocities the system is seen to have higher magnitude poles and hence faster dynamics.

9.2. Closed loop stability and response

By closing the loop with the derived control law, the closed-loop poles and zeros are obtained for all operating points of the reference trajectory. Figure 15 displays the ones associated with the longitudinal mode. The control law stabilizes all operating points, placing the closed-loop poles in the left-hand side of the complex plane. The pole-zero cancellation of the poles and zeros approximately located at the origin is noted. For each operating point, the relevant poles correspond to a pair of stable conjugated complex poles and a stable real pole. The complex poles are expected to cause oscillatory behaviour in the response of the system, nonetheless, it was the ideal compromise found between limiting oscillations while keeping a fast settling time. To exemplify this, Fig. 16 displays the response to a step request of 3 degrees in pitch angle, and the associated control input variation, at $t = 60$ s. Table 4 details some key parameters of the closed-loop system step response in the linear domain for distinct operating points. A fast response with limited overshoot is obtained for all operational regimes.

10. Simulation results

In this section, the results obtained using the proposed architecture in the simulation environment are presented. Given the stochastic



(a) Pitch angle ($\delta\theta$) response (b) Actuation ($\delta\mu_p$)

Fig. 16. Response to a 3° step in pitch angle.

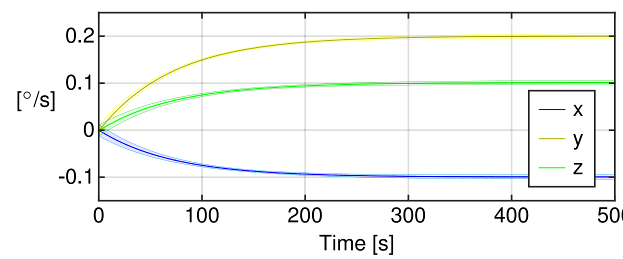


Fig. 17. Rate gyro bias estimation (initial calibration).

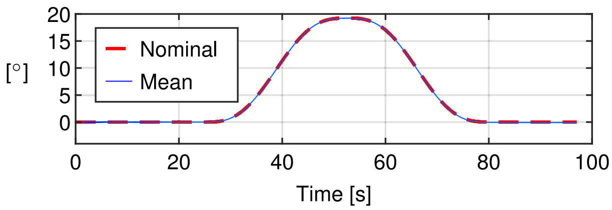
nature of the system, caused by the noise from the sensors and external disturbances (wind gusts), Monte Carlo (MC) simulations were performed by varying the noise seeds associated with the stochastic components in each run and by sampling the initial state of the navigation system. The performance of the navigation and control systems is detailed, both together and individually, with the MC simulations results being presented. Additionally, the robustness of the overall architecture to parametric uncertainties was tested by introducing randomness and uncertainty in some model parameters in a further MC study.

10.1. Navigation system

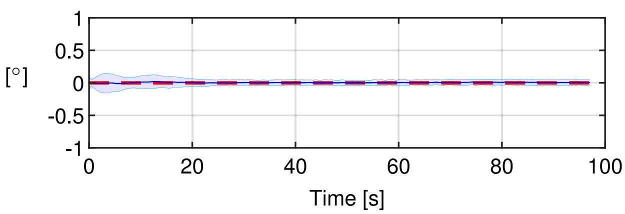
Firstly, the initial calibration period was simulated over 100 MC runs. During this period, the vehicle is standing vertically on the launch pad while the navigation system estimates the approximately constant rate gyro bias. In each run, the state vector is sampled using Gaussian distributions to add a given uncertainty to the initial estimates. The following distributions were used: $\hat{\lambda}(t_0) \sim \mathcal{N}(\mathbf{0}, 0.1^2)$, in degrees; $\hat{\mathbf{b}}_\omega(t_0) \sim \mathcal{N}(\mathbf{0}, 0.01^2)$, in degrees; ${}^b\hat{\mathbf{p}}(t_0) \sim \mathcal{N}(\mathbf{0}, 1)$, in meters; ${}^b\hat{\mathbf{v}}(t_0) \sim \mathcal{N}(\mathbf{0}, 0.1^2)$, in meters per second; and ${}^b\hat{\mathbf{g}}(t_0) \sim \mathcal{N}\left(\begin{bmatrix}-g_{sl} & 0 & 0\end{bmatrix}^T, 0.01^2\right)$, in meters per square second. Figure 17 displays the evolution of the rate gyro bias on the three axis in terms of the MC mean and standard deviation intervals. It is possible to conclude that the bias estimates converge to the true values after a relatively short time period. By inspection, the time $t = 300$ s is selected as the end of the calibration period, after which launch is initiated. The MC standard deviation calculated at this instant, averaged for the 3 axes, is used to correctly propagate the uncertainty for the following part of the simulation. This is done by initializing the bias estimates with the normal distribution $\hat{\mathbf{b}}_{\omega_0} \sim \mathcal{N}\left(\begin{bmatrix}-0.1 & 0.2 & 0.1\end{bmatrix}^T, 0.0035^2\right)$, in degrees. The remaining state variables are sampled once again using the previous distributions. In the following sections, the MC results for the estimation performed by each complementary filter are shown for the nominal scenario, where no disturbances are present and only the feedforward control input is required.

Table 5
ACF estimation Monte Carlo results (expressed in degrees).

		θ	ψ	b_{ω_y}	b_{ω_z}
ACF	$\overline{rmse}$	0.0654	0.0567	0.0028	0.0021
	σ_{rmse}	0.0208	0.0169	0.0012	0.0013
AD	$\overline{rmse}$	0.0714	0.0799	n/a	n/a
	σ_{rmse}	0.0196	0.0124	n/a	n/a



(a) Pitch angle



(b) Yaw angle

Fig. 18. ACF pitch and yaw angles estimation.

10.1.1. ACF

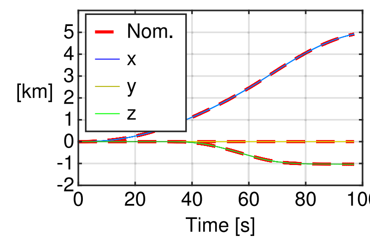
The complete MC results for the ACF estimation are present in Table 5 for 100 MC runs, where the average and standard deviation of the root mean square error between the estimated and true values for all runs ($\overline{rmse}$ and σ_{rmse}) are displayed. The root mean square error is shown for the pitch and yaw angles estimation in degrees, as well as for the pitch and yaw rate bias, b_{ω_y} and b_{ω_z}, estimation in degrees per second. For comparison, the pitch and yaw determination performance by the AD algorithm, which relies solely on the aiding sensors, is also shown. The average root mean square errors are reduced, as well as its standard deviation for all runs, making it possible to infer that AD is able to provide accurate observations on the attitude of the vehicle, and that the ACF further improves those estimates, while correcting the bias of the gyroscope. Figure 18(a) visually displays the estimation results for the pitch angle in terms of the MC mean with $\pm 1\sigma$ boundaries, while Fig. 18(b) presents the equivalent results for the yaw angle.

10.1.2. PCF

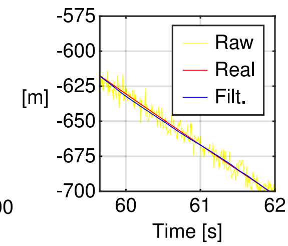
Regarding the PCF, it was also possible to verify its correct functioning by analysing the position, velocity and gravity vector estimates. The complete MC results for the PCF estimation are present in Table 6, once again for 100 MC runs and in terms of the mean and standard deviation of the root mean square estimation error, using SI units. The filter is able to reject the noise from the position measurements while maintaining good accuracy with respect to the true value. The root mean square error on the position estimates is one order of magnitude below the standard deviation assumed for the altimeter and GNSS receiver noises. Accurate estimates on the velocity and gravity vectors are also obtained, both quantities which are not available as a measurement to the filter. Figure 19 details the position estimation by the PCF in terms of the MC mean with the $\pm 1\sigma$ bounds (a), and a zoomed section of the crossrange position (${}^{i}z$) estimation (b) to illustrate the filtering performed by the PCF by comparison with the noisy measurements. Figure 20 details the estimation of the velocity and gravity vectors in the body frame, also by representing the MC mean with the $\pm 1\sigma$ boundaries. Due to the small standard deviations, the shaded intervals representing the $\pm 1\sigma$ boundaries are hardly visible in the figures.

Table 6
PCF estimation Monte Carlo results (SI units).

	${}^{i}x$	${}^{i}y$	${}^{i}z$	u	v	w
$\overline{rmse}$	0.18	0.69	0.71	0.09	0.10	0.9
σ_{rmse}	0.028	0.050	0.047	0.035	0.034	0.032
		${}^{b}g_x$	${}^{b}g_y$	${}^{b}g_z$		
$\overline{rmse}$		0.02	0.01	0.01		
σ_{rmse}		0.006	0.003	0.003		

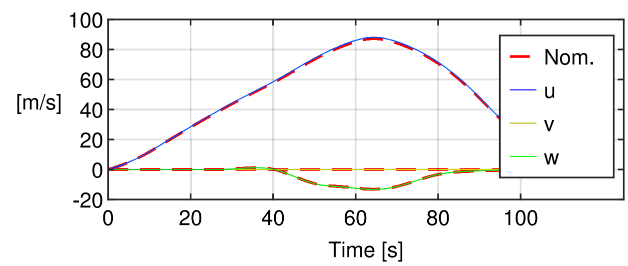


(a) Position

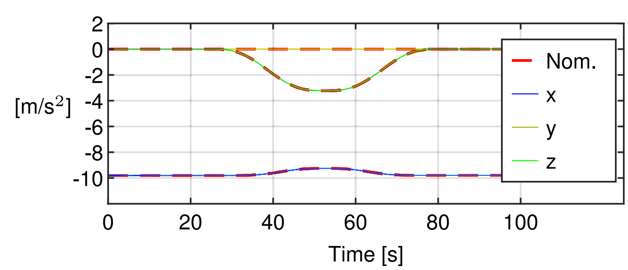


(b) Zoom in ${}^{i}z$ estimation

Fig. 19. PCF position estimation.



(a) Linear velocity



(b) Gravity vector

Fig. 20. PCF velocity and gravity estimation.

10.2. Control system

The control system was tested by adding stochastic wind as external disturbance, according to the previously detailed model. For this part, the seeds associated with the wind gusts generation were the additional parameters to be randomly changed in each MC run. The system was tested for two distinct cases: assuming exact full-state knowledge, and using the state estimates provided by the navigation system in the presence of sensor noise, which represents the complete architecture. For each scenario, 100 MC runs were completed.

Table 7 presents the detailed MC results in terms of the average over all runs of the root mean square of the tracking errors and actuation signals, with the correspondent standard deviations also being displayed. Moreover, the results are shown both when using the derived LQI control law, and when using a PID controller per degree of freedom (pitch and yaw) for comparison. It is concluded that, with the implementation of the control system, the complete architecture was able to reject the external wind perturbation while maintaining both

Table 7
Pitch and yaw tracking error and control effort Monte Carlo results (expressed in degrees).

	Exact state			
	LQI		PID	
	$\bar{x}$	σ	$\bar{x}$	σ
θ_{rmse}	0.017	0.001	0.026	0.0028
ψ_{rmse}	0.007	0.0004	0.016	0.0018
$\delta\mu_{p_{rms}}$	0.64	0.089	1.53	0.227
$\delta\mu_{y_{rms}}$	0.47	0.081	0.48	0.077
	Estimated state			
	LQI		PID	
	$\bar{x}$	σ	$\bar{x}$	σ
θ_{rmse}	0.073	0.024	0.072	0.019
ψ_{rmse}	0.060	0.020	0.059	0.017
$\delta\mu_{p_{rms}}$	0.65	0.080	1.55	0.24
$\delta\mu_{y_{rms}}$	0.51	0.081	1.00	0.20

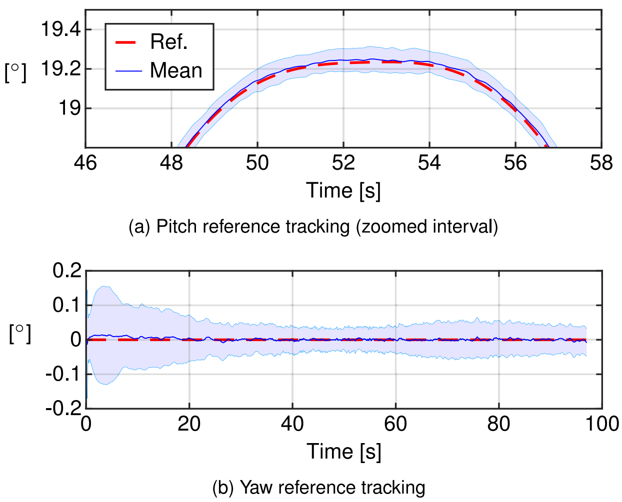


Fig. 21. Attitude reference tracking.

Table 8
Monte Carlo results for different roll rates (expressed in degrees).

p	0.1 Hz		0.5 Hz		1 Hz	
	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ
θ_{rmse}	0.054	0.015	0.077	0.014	0.12	0.016
ψ_{rmse}	0.056	0.015	0.13	0.009	0.25	0.009
$\delta\mu_{p_{rms}}$	0.92	0.13	1.11	0.20	1.45	0.25
$\delta\mu_{y_{rms}}$	0.94	0.11	1.30	0.12	1.93	0.14

satisfactory attitude tracking performance and reduced control effort. Additionally, it is noted that LQI control provides similar attitude tracking performance to PID control, although with significantly less control effort, and that there is a decrease in performance when measurement noise and bias are added. In fact, the attitude tracking performance is mostly limited/imposed by the estimation accuracy.

Figure 21 presents the simulation results for the pitch (a) and yaw (b) angles reference tracking, with the navigation system included in the loop. It depicts the average evolution of both angles for all MC runs, as well as the $\pm\sigma$ intervals. For the pitch angle, a zoomed interval where the maximum pitch occurs is shown. Figure 22 details the actuation by the TVC system for a single MC run to exemplify the typical actuation profile.

The step response is also analysed to determine if the system is able to track deviation requests from the reference condition. Figure 23 illustrates the results for the same instant and request as shown for the linear domain (Fig. 16). It is possible to verify that the step response

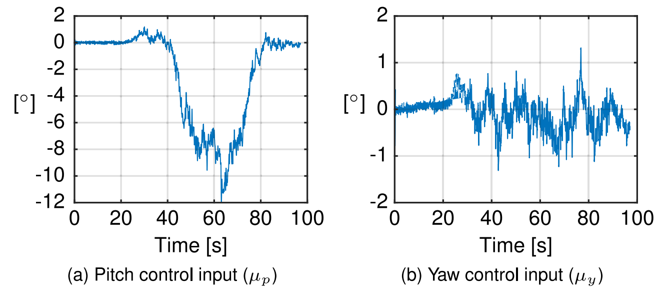


Fig. 22. Actuation profile example.

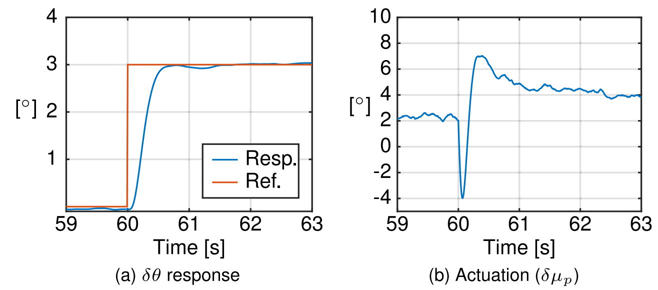


Fig. 23. Response to a 3° step in $\delta\theta$.

performance is similar to the one found for the linear domain, apart from disturbance/noise induced irregularities.

10.3. Spinning motion impact

So far, the simulation results were presented for the case in which there is no spinning motion. When spinning motion is present, the system must track the same pitch and yaw reference angles by relying on the correction of the actuation signals explained in Section 7.3.1. Table 8 details the values obtained for the previously used performance metrics when the vehicle possesses different values of roll rate in the body frame (p), once again conducting a total of 100 MC runs for each case. Looking at the results, it is seen that the attitude tracking performance gradually degrades with the increase of the roll rate in the body frame, and that the control effort increases. Nonetheless, we conclude that the system is still able to correctly function in the presence of spinning motion. Through simulation, it was verified that above the approximate rate of 1 Hz (1 full rotation per second) the vehicle starts to display unstable coupled oscillatory behaviour towards the end of the flight, when the control authority is at its minimum. It is noteworthy that for 0.1 Hz the attitude tracking performance is superior to the one shown in the absence of spinning motion. This fact is attributed to the increase in natural stability and external disturbance rejection provided by the gyroscopic effect, which, until a given roll rate value, surpasses the negative effect caused by the adjustment effort imposed on the actuators and by the cross-coupling of the pitch and yaw axes.

10.4. Robustness analysis

Finally, an analysis was performed to determine the robustness of the architecture to model uncertainties. Several system parameters, including mass, inertia, thrust, centre of mass position, and aerodynamic coefficients, were altered in each run inside admissible ranges in terms of percentage of the nominal value. The multiplication factors were sampled according to Gaussian distributions with unitary mean and the following $\pm 3\sigma$ bounds: $3\sigma_T = 0.05$; $3\sigma_{cm} = 3\sigma_{J_y} = 3\sigma_{m_{dry}} = 0.1$; and $3\sigma_{C_A} = 3\sigma_{C_N} = 0.2$, where J_y is the inertia in the y-axis and m_{dry} is the

Table 9
Monte Carlo robustness analysis results (expressed in degrees).

	LQI		PID	
	$\bar{x}$	σ	$\bar{x}$	σ
θ_{rmse}	0.054	0.013	0.065	0.014
ψ_{rmse}	0.055	0.013	0.073	0.011
$\delta\mu_{p_{rms}}$	0.98	0.17	2.53	0.56
$\delta\mu_{y_{rms}}$	0.94	0.16	1.32	0.20

dry mass. The inertia and lateral aerodynamic coefficient selected for the study are the ones associated with the pitch plane, which allows to verify if the system is able to perform when the assumed symmetry is broken.

A total of 500 MC runs were conducted assuming a roll rate in the body frame of 0.1 Hz. The robustness when using a PID is also studied for comparison. Table 9 presents the results. The system is concluded to be robust to model parametric uncertainties as it was able to stabilize the plant and track the attitude reference while keeping a similar performance to the nominal case. In comparison, the PID attitude tracking performance is slightly worse and the associated control effort is significantly larger. This indicates that the designed control system is more robust than its classical PID counterpart.

11. Conclusions

With the conclusion of this work, it is possible to state that the primary goal has been achieved: the successful design of an integrated architecture for attitude control and navigation, applicable to low-cost suborbital launch vehicles. Initially, both a non-linear model for the dynamics and kinematics of a generic thrust-vector-controlled launch vehicle and an original linear state-space representation were derived, which served as foundation for the architecture design. Subsequently, the proposed architecture, comprising the navigation and control systems, was completely detailed by presenting its overall structure and each of its individual components.

The navigation system relies on readily available components, providing accurate state estimates by removing measurement noise and bias. Its structure is derived starting directly from the available on-board measurements, facilitating a future implementation. Namely, Euler angle determinations are obtained from vector observations, with the particularity of using an estimate on the gravity vector instead of relying on accelerometer readings, as commonly done in the literature, potentially corrupted with a dynamic component. Furthermore, the structure of the time-varying complementary kinematic filters simplifies the tuning procedure by establishing an explicit relation with the time-invariant case, and ensures an UAS estimation solution apart from the singularities. However, solely relying on kinematic models yields non-optimal performance in the sense that the dynamics of the problem are not considered. The time-invariant design can be based on the well-established linear Kalman filtering theory, enabling the application of standardized requirements.

The control system, based on the scheduling of pre-calculated gains with an LQI control law, ensured satisfactory attitude reference tracking performance and robustness to model uncertainties. A method for correcting the actuation in the presence of spinning motion, which allows the implementation of the architecture in a vehicle with limited, or even without, roll control capability, was also proposed. In future work, an extension of the control system to non-linear techniques, which allow for global, trajectory-independent solutions, will be exploited with the goal of comparing the performance of both formulations.

The integrated architecture was intensively tested in simulation through Monte Carlo analysis, using a simulation model implemented in Simulink, yielding an overall satisfactory performance. The simulation model includes the proposed architecture, the derived non-linear model for the vehicle dynamics and kinematics, models for generating the environmental conditions, and the time-varying model parameters. As future work, the simulation environment shall be improved by including phenomena yet to be modelled, such as elastic modes, non-linear aerodynamic effects, the curvature and rotation of the Earth, and body asymmetries, to further verify the system. A final validation of the proposed architecture shall occur through the use of small-scale rocket prototypes, before its implementation in a real suborbital launch vehicle.

CRediT authorship contribution statement

Pedro dos Santos: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Paulo Oliveira:** Conceptualization, Funding acquisition, Methodology, Project administration, Resources, Supervision, Validation, Visualization, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors acknowledge the Portuguese Foundation for Science and Technology (FCT) for its financial support via the projects LAETA Base Funding (DOI: 10.54499/UIDB/50022/2020) and CAPTURE (DOI: 10.54499/PTDC/EEI-AUT/1732/2020). Pedro dos Santos holds a PhD scholarship from FCT (2023.00268.BD).

References

[1] NASA Sounding Rockets: Anual Report 2021, Technical Report, National Aeronautics and Space Administration (NASA), 2021.
[2] G. Seibert, The History of Sounding Rockets and Their Contribution to European Space Research, Technical Report, European Space Agency (ESA), 2006.
[3] T. Noga, R. Puri, Microgravity, atmosphere sounding, astronomy, technology validation - an overview of suborbital rockets' missions and payloads, Int. J. Space Sci. Eng. 6 (2) (2020) 179–208, http://dx.doi.org/10.1504/IJSPACESE.2020.110365.
[4] Small satellite launch market trends: What's to come in 2021–2025?, 2021, URL: https://orbitaltoday.com/2021/05/12/small-satellite-launch-market-trends-whats-to-come-in-2021-2025/. (Accessed 6 February 2023) [Online].
[5] Ariane next, a vision for the next generation of Ariane Launchers, Acta Astronaut. 170 (2020) 735–749, http://dx.doi.org/10.1016/j.actaastro.2020.02.003.
[6] P. Simplício, A. Marcos, S. Bennani, Guidance of reusable launchers: Improving descent and landing performance, J. Guid. Control Dyn. 42 (10) (2019) 2206–2219, http://dx.doi.org/10.2514/1.G004155.
[7] E. Dumont, S. Ishimoto, P. Tatiossian, J. Klevanski, B. Reimann, T. Ecker, L. Witte, J. Riehmer, M. Sagliano, S. Giagkozoglou Vincenzino, I. Petkov, W. Rotärmel, R. Schwarz, D. Seelbinder, M. Markgraf, J. Sommer, D. Pfau, H. Martens, CALLISTO: a demonstrator for reusable launcher key technologies, Trans. JSASS Aerosp. Tech. Jpn 19 (1) (2021) 106–115, http://dx.doi.org/10.2322/tastj.19.106.
[8] Sub-orbital transportation and space tourism market, 2021, URL: https://www.emergenresearch.com/industry-report/sub-orbital-transportation-and-space-tourism-market. (Accessed 6 February 2023) [Online].
[9] J.W. Jang, A. Alaniz, R. Hall, N. Bedrossian, C. Hall, M. Jackson, Design of launch vehicle flight control systems using ascent vehicle stability analysis tool, in: The 2011 AIAA Guidance, Navigation, and Control Conference, Portland, Oregan, 8–11 August, 2011, http://dx.doi.org/10.2514/6.2011-6652.
[10] G.P. Sutton, O. Biblarz, Rocket Propulsion Elements, ninth ed., John Willey & Sons, New Jersey, 2017.
[11] E. Kulu, Small launchers - 2021 industry survey and market analysis, in: The 72nd International Astronautical Congress, Dubai, United Arab Emirates, 25–29 October, 2021, https://www.newspace.im/assets/Small-Launchers-2021_Erik-Kulu_IAC2021.pdf.
[12] Q. Zhang, Z. Xu, Autonomous ascent guidance with multiple terminal constraints for all-solid launch vehicles, Aerosp. Sci. Technol. 97 (2020) http://dx.doi.org/10.1016/j.ast.2019.105633.

[13] A. Tewari, Advanced Control of Aircraft, Spacecraft and Rockets, John Willey & sons, West Sussex, 2011.

[14] T. Barrows, J. Orr, Dynamics and Simulation of Flexible Rockets, Academic Press, London, 2021.

[15] B. Lu, D. Falde, E. Iriarte, E. Besnard, Switching robust control for a nanosatellite launch vehicle, Aerosp. Sci. Technol. 42 (2015) 259–266, http://dx.doi.org/10.1016/j.ast.2015.01.019.

[16] B. Wie, W. Du, M. Whorton, Analysis and design of launch vehicle flight control systems, in: The 2008 AIAA Guidance, Navigation, and Control Conference & Exhibit, Honululu, Hawaii, 18–21 August, 2008, http://dx.doi.org/10.2514/6.2008-6291.

[17] Y. Jenie, I. Suarjaya, R. Poetro, Falcon 9 rocket launch modeling and simulation with thrust vectoring control and scheduling, in: The 6th IEEE Asian Conference on Defence Technology, Bali, Indonesia, 13–15 November 2019, 2019, http://dx.doi.org/10.1109/ACDT47198.2019.9072837.

[18] L. Sopegno, P. Livreri, M. Stefanovic, K.P. Valavanis, Linear quadratic regulator: A simple thrust vector control system for rockets, in: The 30th Mediterranean Conference on Control and Automation (MED), Vouliagmeni, Greece, 28 June – 01 July, 2022, http://dx.doi.org/10.1109/MED54222.2022.9837125.

[19] A. Kisabo, A. Adebimpe, S. Samuel, Pitch control of a rocket with a novel LQG/LTR control algorithm, J. Aircr. Spacecr. Technol. 3 (1) (2019) 24–37, http://dx.doi.org/10.3844/jastsp.2019.24.37.

[20] E. Mooji, Nonlinear robust control and observation for aeroelastic launch vehicles with propellant slosh in a turbulent atmosphere, in: The 2023 AIAA Scitech Forum, 23–27 January, 2023, http://dx.doi.org/10.2514/6.2023-1999.

[21] F. Celani, Global and robust attitude control of a launch vehicle in exoatmospheric flight, Aerosp. Sci. Technol. 74 (2018) 22–36, http://dx.doi.org/10.1016/j.ast.2017.12.016.

[22] J. Guadagnini, M. Lavagna, P. Rosa, Model predictive control for reusable space launcher guidance improvement, Acta Astronaut. 193 (2022) 767–778, http://dx.doi.org/10.1016/j.actaastro.2021.10.014.

[23] P. dos Santos, P. Oliveira, ADCS design for a sounding rocket with thrust vectoring, Aerotec. Missili Spaz. 102 (3) (2023) 257–270, http://dx.doi.org/10.1007/s42496-023-00161-w.

[24] P. Simplício, A. Marcos, S. Bennani, Launcher flight control design using robust wind disturbance, Acta Astronaut. 186 (2021) 303–318, http://dx.doi.org/10.1016/j.actaastro.2021.05.044.

[25] W. Du, B. Wie, M. Whorton, Dynamic modeling and flight control simulation of a large flexible launch vehicle, in: The 2008 AIAA Guidance, Navigation, and Control Conference & Exhibit, Honululu, Hawaii, 18–21 August, 2008, http://dx.doi.org/10.2514/6.2008-6620.

[26] J.A. Farrel, M.M. Polycarpou, Adaptive Approximation Based Control, John Willey & Sons, New Jersey, 2006.

[27] J. Vasconcelos, B. Cardeira, C. Silvestre, P. Oliveira, P. Batista, Discrete-time complementary filters for attitude and position estimation: Design, analysis and experimental validation, IEEE Trans. Control Syst. Technol. 19 (1) (2011) 181–198, http://dx.doi.org/10.1109/TCST.2010.2040619.

[28] P. Batista, C. Silvestre, P. Oliveira, B. Cardeira, Accelerometer calibration and dynamic bias and gravity estimation: Analysis, design, and experimental evaluation, IEEE Trans. Control. Syst. Technol. 19 (5) (2011) 1128–1137, http://dx.doi.org/10.1109/TCST.2010.2076321.

[29] D. Simon, Optimal State Estimation, John Wiley & Sons, New Jersey, 2006.

[30] M. Suster, S. Oh, Three-axis attitude determination from vector observations, J. Guid. Control Dyn. 4 (1) (1981) 70–77, http://dx.doi.org/10.2514/3.19717.

[31] I. Bar-Itzhack, R. Harman, Optimized TRIAD algorithm for attitude determination, J. Guid. Control Dyn. 20 (1) (1997) 208–211, http://dx.doi.org/10.2514/2.4025.

[32] G. Slabaugh, Computing Euler angles from a rotation matrix, 1999.

[33] W.J. Rugh, Linear System Theory, Prentice-Hall, New Jersey, 1996.

[34] P. Batista, C. Silvestre, P. Oliveira, On the observability of linear motion quantities in navigation systems, Systems Control Lett. 60 (2) (2011) 101–110, http://dx.doi.org/10.1016/j.sysconle.2010.11.002.

[35] B. Friedland, Control System Design: An Introduction to State-Space Methods, Dover Publications, New York, 2005.

[36] U.S. Standard Atmosphere, 1976, Technical Report, National Aeronautics and Space Administration (NASA), 1976.