72nd International Astronautical Congress (IAC), 25-29 October 2021.

IAC-21,C1,1,13,x65893

An Overview of Autonomous Optical Navigation for Deep-Space CubeSats

Eleonora Andreis[1], **Vittorio Franzese**[2], **Francesco Topputo**[3]

A new era of space exploitation is fast approaching. CubeSats have performed a revolution in the way satellites are deployed in interplanetary missions. The exploitation of standardized dimensions and Commercial-Off-The-Shelf components has boosted their utilization by reducing mission costs and development time. The cutting down on the space entry-price grants the democratization of interplanetary exploration. Yet, the flourishing growth of users in space will saturate the ground networks, hindering the traditional navigation through ground-based radiometric tracking. Miniaturized probes that can operate in complete autonomy from the ground represent the solution of this issue. From navigation perspective, a celestial triangulation algorithm fed by optical observations of planets can be exploited to retrieve the probe state. In this work, an autonomous optical navigation algorithm for interplanetary nano-spacecraft applications is developed. In particular, an Extended Kalman Filter featuring line-of-sight acquisitions of planets is adopted as state estimator. The solution accuracy is improved by correcting the planetary light-time and aberration effects and by exploiting the optimal beacons selection strategy. Moreover, an in-depth analysis concerning the numerical precision of the estimator is carried out. Finally, the navigation algorithm is tested on a platform comparable to a CubeSat computer to verify its sustainability and performances. The present work is framed within the EXTREMA project, awarded an ERC Consolidator Grant in 2019.

1 Introduction

CubeSats have triggered a revolution in the way satellites have been launched into space [1]. Two main characteristics have boosted their utilization and have expanded their application area: low costs and reduced development time. Although most of nanosatellites have been thus far launched into Low Earth Orbits (LEO), several CubeSats applications in deep space are foreseen [2]. Examples of innovative nanosats deep-space mission concept proposals are the Lunar Meteoroid Impacts Observer (LUMIO) [3], the Miniaturised Asteroid Remote Geophysical Observer (M-ARGO) [2][4], Juventas [5] and Milani [6] as part of Hera mission [7]. The break with the tradition obtained through the exploitation of standardized dimensions and Commercial-Off-The-Shelf (COTS) components allows space to become affordable not only to space agencies but also to small companies and universities. The obstacle of this flourishing growth resides in the saturation of the ground networks [8]. As state-of-the-art, ground-based radiometric tracking is employed for deep-space navigation; despite the high reliability, its adoption will become more and more unsustainable with the massive wave of nanosats to be controlled in space. Miniaturized probes that can perform guidance, navigation, and control operations without human instructions are the answer to this problem [9]. The computation shall switch from ground to onboard and miniaturized satellites shall be empowered to become stand-alone. This replacement will also cut the cost related to human employment that is currently independent of the dimensions and the mass of the spacecraft. In deep-space, the spacecraft must be aware of the external environment by determining, without ground support, its position in the Solar System. This could be possible by exploiting a celestial triangulation algorithm fed by planet positions vectors [10]. In this context, this work, framed within the ERC-funded EXTREMA (Engineering Extremely Rare Events in Astrodynamics for Deep-Space Mission in Autonomy) project [1], aims to develop an autonomous optical navigation approach suited for deep-space CubeSats. In particular, an Extended Kalman Filter (EKF) featuring line-of-sight acquisitions of planets is developed to estimate the probe state onboard, which simulates stand-alone navigation accomplished by an interplanetary CubeSat. An improvement of the solution accuracy is performed by correcting the planetary light-time and aberration effects and by exploiting the optimal beacons selection strategy to acquire the external observations. Moreover, the numerical precision of the estimator is improved through the implementation of factorization techniques and non-dimensionalization strategies. The navigation

[1]PhD Candidate, Department of Aerospace Science and Technology, Politecnico di Milano, Via La Masa 34, 20156, Milano, Italy, eleonora.andreis@polimi.it
[2]Postodoctoral Research Fellow, Department of Aerospace Science and Technology, Politecnico di Milano, Via La Masa 34, 20156, Milano, Italy, vittorio.franzese@polimi.it
[3]Full Professor, Department of Aerospace Science and Technology, Politecnico di Milano, Via La Masa 34, 20156, Milano, Italy, francesco.topputo@polimi.it

[1]`https://dart.polimi.it/extrema-erc/`, Last Visited on September 2021

algorithm is then tested on a platform comparable to a CubeSat onboard computer to verify its sustainability and performances. This work presents the results obtained for a deep-space CubeSat on an Earth-Mars transfer and discusses possible improvements in view of a subsequent hardware-in-the-loop simulation. The paper is structured as follows. The navigation problem is stated in Sec.2, where the celestial triangulation geometry (Sec.2.1), the optimal beacons selection strategy (Sec.2.2), and the corrections of the light-time effects (Sec.2.3) are described. The EKF is thus implemented in Sec.3. Apart from a standard filtering procedure (Sec.3.2), four alternative approaches are developed in Sec.3.3 to solve the numerical issues that can arise after the onboard deployment of the algorithm. At this point, the algorithm suitability needs to be tested by performing a simulation with a miniaturized processor-in-the-loop (PIL). The methodology followed for the deployment of the algorithm on a miniaturized processor, in this case a Raspberry Pi, is illustrated in Sec.3.4. Eventually, the results of the implementation are illustrated in Sec.4. In particular, the performances of the five filtering approaches are compared in terms of numerical stability and computational complexity in Sec.4.4. The validation of the best alternative is eventually performed on a Raspberry Pi in Sec. 4.6.

2 Statement of the Problem

2.1 Definition of the Problem Geometry

A CubeSat is assumed to cover an interplanetary trajectory. Its state is unknown. The probe position can be retrieved by solving the kinematic celestial triangulation problem [10]. This is possible only when two Line-of-Sight (LoS) directions associated to different planets are acquired simultaneously. Let the probe position $\boldsymbol{r}$ be

$$\boldsymbol{r} = \boldsymbol{r}_1 - \rho_1 \hat{\boldsymbol{\rho}}_1 = \boldsymbol{r}_2 - \rho_2 \hat{\boldsymbol{\rho}}_2 \tag{1}$$

where $\boldsymbol{r}_i$ denotes the position of the i-esimal planet with respect to the Sun in an inertial frame, ρ_i the position magnitude of the i-esimal planet with respect to the observer in an inertial frame and $\hat{\boldsymbol{\rho}}_i$ its direction. The planets' positions with respect to the Sun in the inertial reference frame are available from their ephemerides, $\hat{\boldsymbol{\rho}}_1$, $\hat{\boldsymbol{\rho}}_2$ can be measured onboard by the probe optical sensor, while the ranges ρ_1, ρ_2 are unknown. The scalar multiplication of Eq. 1 by $\hat{\boldsymbol{\rho}}_1$ and $\hat{\boldsymbol{\rho}}_2$ yields to a system of two equations which can be arranged in matrix form as

$$\underbrace{\begin{bmatrix} -1 & \hat{\boldsymbol{\rho}}_1^\top \hat{\boldsymbol{\rho}}_2 \\ -\hat{\boldsymbol{\rho}}_2^\top \hat{\boldsymbol{\rho}}_1 & 1 \end{bmatrix}}_{\boldsymbol{A}} \underbrace{\begin{bmatrix} \rho_1 \\ \rho_2 \end{bmatrix}}_{\boldsymbol{x}} = \underbrace{\begin{bmatrix} \hat{\boldsymbol{\rho}}_1^\top (\boldsymbol{r}_2 - \boldsymbol{r}_1) \\ \hat{\boldsymbol{\rho}}_2^\top (\boldsymbol{r}_2 - \boldsymbol{r}_1) \end{bmatrix}}_{\boldsymbol{b}} \tag{2}$$

described by the linear-algebra problem, $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$. The solution to Eq. 2 is determined as long as $\Delta(\boldsymbol{A}) = -1 + (\hat{\boldsymbol{\rho}}_1^\top \hat{\boldsymbol{\rho}}_2)(\hat{\boldsymbol{\rho}}_2^\top \hat{\boldsymbol{\rho}}_1) \neq 0$, where Δ is the determinant of $\boldsymbol{A}$. The expression of $\Delta(\boldsymbol{A})$ can be also written in function of γ, which describes the angle between the two beacons as seen by the observer (Fig. 1). Since $\cos\gamma = \hat{\boldsymbol{\rho}}_1^\top \hat{\boldsymbol{\rho}}_2$, $\Delta(\boldsymbol{A}) = -1 + \cos^2\gamma$. When $\Delta(\boldsymbol{A}) \neq 0$, the problem solution is $\boldsymbol{x} = \boldsymbol{A}^{-1}\boldsymbol{b}$, which plugged into Eq. 1 yields the probe inertial position. Whereas, when $\cos\gamma = \pm 1$, $\boldsymbol{A}$ is singular and the solution is undetermined. This occurs when the probe and the planets are in conjunction or opposition [11]. The method presented

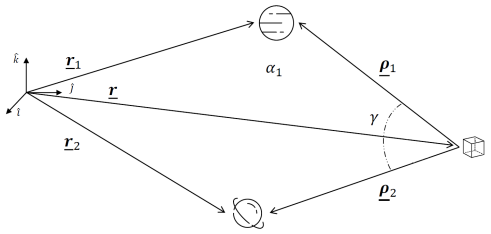

Fig. 1: Celestial triangulation problem.

so far solves for the spacecraft position at the same epoch of measurements acquisition. Dynamic methods, e.g., Kalman filtering, are however more accurate and can estimate the full state vector, including the spacecraft velocity. Moreover, it is possible to assume that only one beacon at a time can be observed by the CubeSat due to the narrow Field of View (FoV) of the onboard optical sensor. Thus, the kinematic celestial triangulation becomes not more applicable. For this reason, in this work, an Extended Kalman Filter (EKF) based on the celestial triangulation approach will be exploited to develop the autonomous navigation algorithm.

2.2 Optimal Beacons Selection Strategy

Once the celestial triangulation method is stated, the couple of planets to be tracked needs to be chosen. It has been proved that tracking the optimal selection of beacons yields the highest accuracy in the determination of the problem solution of Eq. 1 [11, 12]. In this work, the strategy adopted is the one described in [11], which is here implemented directly into the dynamic estimator. First, the planets observability is assessed by evaluating the associated Solar Aspect Angle (SAA) and apparent magnitude. Then, the optimal beacon selection strategy is applied to only the planets whose parameters respect the threshold values imposed by the optical sensor. The optimal pair is found by minimizing the figure of merit $\mathcal{J}$, which is evaluated for each pair of traceable planets.

$\mathcal{J}$ is defined as

$$\mathcal{J} = \sigma_{\text{str}}^2 \frac{1+\cos\gamma^2}{\sin\gamma^4} \boldsymbol{d}^\top \Big(\boldsymbol{L}_i + \boldsymbol{L}_j \Big) \boldsymbol{d} \tag{3}$$

where $\boldsymbol{L}_i = \boldsymbol{I}_{3x3} - \hat{\boldsymbol{\rho}}_i \hat{\boldsymbol{\rho}}_i^\top$ and $\boldsymbol{L}_j = \boldsymbol{I}_{3x3} - \hat{\boldsymbol{\rho}}_j \hat{\boldsymbol{\rho}}_j^\top$; $\hat{\boldsymbol{\rho}}_i$ and $\hat{\boldsymbol{\rho}}_j$ are the unitary LoS vectors to the i-th and j-th planet, respectively. Moreover, σ_{str} is the standard deviation of the LoS angular error from the star tracker, $\boldsymbol{d} = \boldsymbol{r}_i - \boldsymbol{r}_j$ and $\gamma = \text{acos}(\hat{\boldsymbol{\rho}}_i \hat{\boldsymbol{\rho}}_j)$. The complete derivation can be found in [11].

2.3 Light Effects

In deep space, the navigation problem becomes more complex due to the large distance between the observer and the beacon. In this context, two issues arise, namely 1) the light-time effect and 2) the light-aberration effect [10]. If corrections are neglected, the planet apparent position in the star tracker FoV will appear as shifted to the real one. In particular, this error has a consistent impact on the estimation of the probe state in deep space [13]. Indeed, the estimation is more and more affected by the celestial aberration and by the light-time effect when the probe velocity and distance to the observed object increase, respectively.

2.3.1 Light-time Effect

Due to the finite speed of light and the enormous distance between the satellite and the beacon, the time difference between the emission and the reception of the luminous signal is significant. The light received by the imager at epoch t has been emitted by the observed planet at a previous epoch τ, thus $\tau < t$. This effect, called light-time, yields a systematic error when neglected [14]. Let us see Fig.2; the definition of the spacecraft position

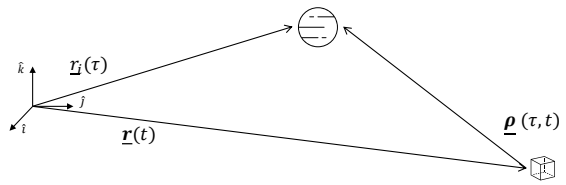

Fig. 2: Ligh-time effect.

$\boldsymbol{r}$ becomes

$$\boldsymbol{r}(t) = \boldsymbol{r}_i(\tau) - \boldsymbol{\rho}(t,\tau) \tag{4}$$

To address correctly this problem, the evaluation of epoch τ is fundamental. In this work, τ is estimated directly by the filter by introducing it as part of the state vector. Once estimated, τ is exploited to retrieve $\boldsymbol{r}_i(\tau)$, which represents the ephemerides of the observed beacon. As the light-time effect correction is performed internally, it is necessary to find an expression for the time derivative of τ.
Note that $\rho = ||\boldsymbol{r}_i - \boldsymbol{r}||$, which is also equal to $c(t-\tau)$. Enforcing this equality yields to the following constraint functional

$$\mathcal{L} = c(t-\tau) - ||\boldsymbol{r}_i(\tau) - \boldsymbol{r}(t)|| = 0 \tag{5}$$

By virtue of the implicit function theorem, the differentiation of $\mathcal{L}$ for on-board time is

$$\frac{\mathrm{d}\mathcal{L}}{\mathrm{d}t} = \frac{\partial \mathcal{L}}{\partial t} + \frac{\partial \mathcal{L}}{\partial \tau}\frac{\mathrm{d}\tau}{\mathrm{d}t} = 0 \tag{6}$$

from which the derivative of τ is achieved

$$\frac{\mathrm{d}\tau}{\mathrm{d}t} = -\frac{\partial \mathcal{L}/\partial t}{\partial \mathcal{L}/\partial \tau} = \frac{c + \hat{\boldsymbol{\rho}}^\top \boldsymbol{v}}{c + \hat{\boldsymbol{\rho}}^\top \boldsymbol{v}_i} \tag{7}$$

with $\boldsymbol{v}$ and $\boldsymbol{v}_i$ being the probe and i-th planet velocities, respectively.

2.3.2 Celestial Aberration

The celestial aberration matters whenever the observer velocity to the target is not negligible, as in the case of a probe moving in the Solar System. To account for this, a correction is performed every time a new observation is acquired, and before the estimation is measurement-updated. The procedure followed for the correction is the one described in [10]. With reference to Fig. 3, let

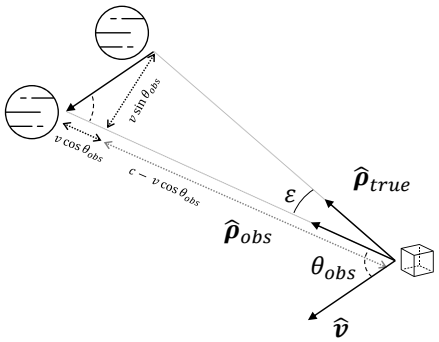

Fig. 3: Light-aberration effect.

θ_{obs} be the angle between the observed LoS $\hat{\boldsymbol{\rho}}_{\text{obs}}$ and the estimated unitary velocity vector of the probe $\hat{\boldsymbol{v}}$; i.e.,

$$\tan\theta_{\text{obs}} = \frac{||\hat{\boldsymbol{\rho}}_{\text{obs}} \hat{\boldsymbol{v}}||}{\hat{\boldsymbol{\rho}}_{\text{obs}}^\top \hat{\boldsymbol{v}}} \tag{8}$$

Then, the aberration angle ε is computed as

$$\tan\varepsilon = \frac{(v/c)\sin\theta_{\text{obs}}}{1-(v/c)\cos\theta_{\text{obs}}} \tag{9}$$

and the true-corrected LoS $\hat{\boldsymbol{\rho}}_{\text{true}}$ is retrieved such that

$$\hat{\boldsymbol{\rho}}_{\text{true}} = \frac{\hat{\boldsymbol{\rho}}_{\text{obs}} \sin\theta_{\text{true}} - \hat{\boldsymbol{v}} \sin\varepsilon}{\sin\theta_{\text{obs}}} \tag{10}$$

with $\theta_{\text{true}} = \theta_{\text{obs}} + \varepsilon$.

3 Filtering Approach

In this section dynamic estimators featuring LoS acquisition of planets are developed. First, the system state space is described. Then, moving from the EKF standard filtering scheme, alternative estimators are presented.

3.1 System State Space

Let $\boldsymbol{x}$ be the process state, thus

$$\boldsymbol{x}(t) = [\boldsymbol{r}(t), \boldsymbol{v}(t), \boldsymbol{\tau}, \boldsymbol{\eta}(t)]^\top \tag{11}$$

where $\boldsymbol{r}$ and $\boldsymbol{v}$ are the inertial probe position and velocity, respectively; $\boldsymbol{\tau} = [\tau_1, \tau_2, \ldots, \tau_8]^\top$ represents the epoch at which the light departs from each of the eight planets; $\boldsymbol{\eta} = [\boldsymbol{\eta}_{\text{R}}, \boldsymbol{\eta}_{\text{SRP}}]^\top$ is a vector of Gauss–Markow (GM) processes, which accounts for two unmodeled terms, namely 1) a 3-dimensional residual accelerations $\boldsymbol{\eta}_{\text{R}}$ and 2) the stochastic component of the Solar Radiation Pressure (SRP) $\boldsymbol{\eta}_{\text{SRP}}$. The system dynamics is described by the vector field $\boldsymbol{f}$, which embeds the deterministic part, and process white noise $\boldsymbol{w}$:

$$\boldsymbol{f} := \begin{bmatrix} \boldsymbol{v} \\ -\mu_{\text{Sun}} \dfrac{\boldsymbol{r}}{r^3} + C_{\text{R}} \dfrac{P_0 R_0^2}{c} \dfrac{A_{\text{s}}}{m_{\text{s}}} \dfrac{\boldsymbol{r}}{r^3} \\ \boldsymbol{\psi} \\ -\xi \boldsymbol{\eta}_{\text{R}} \\ -\xi \boldsymbol{\eta}_{\text{SRP}} \end{bmatrix} \tag{12}$$

with $\boldsymbol{\psi} = [\psi_1, \psi_2, ..., \psi_8]^\top$ such that

$$\psi_i = \frac{c + \hat{\boldsymbol{\rho}}_i^\top \boldsymbol{v}}{c + \hat{\boldsymbol{\rho}}_i^\top \boldsymbol{v}_i} \qquad i = 1, \ldots, 8 \tag{13}$$

In particular, C_{R} is the reflection coefficient, P_0 the solar power, R_0 the Sun radius, A_{s} the cross section area of the probe, and m_{s} its mass. Since the spacecraft is on an interplanetary transfer, third-body perturbations are not taken into account [4]. In the Langevin equations

$$\dot{\boldsymbol{\eta}}_{\text{R}} = -\xi \boldsymbol{\eta}_{\text{R}} + \boldsymbol{w}_{\text{R}} \qquad \dot{\boldsymbol{\eta}}_{\text{SRP}} = -\xi \boldsymbol{\eta}_{\text{SRP}} + \boldsymbol{w}_{\text{SRP}} \tag{14}$$

the parameter ξ defines the reciprocal of the correlation time, and $\boldsymbol{w}_{\text{R}}$ and $\boldsymbol{w}_{\text{SRP}}$ the process noises of the GM parameters with σ_{R} and σ_{SRP} standard deviations, respectively [15]. The measurement model is defined by the equation $\boldsymbol{h}$ and white noise $\boldsymbol{\nu}$. In particular, the relative direction of the navigation beacon as seen by the probe $\hat{\boldsymbol{\rho}}$ is defined by two angles that represent the planet location in the camera FoV: the Azimuth (Az) and the Elevation (El). The measurement model equation is so defined as

$$\boldsymbol{h} = \begin{bmatrix} Az \\ El \end{bmatrix} = \begin{bmatrix} \arctan\left(\dfrac{\hat{\rho}_y}{\hat{\rho}_x}\right) \\ \arcsin(\hat{\rho}_z) \end{bmatrix} \tag{15}$$

where $\hat{\rho}_x$, $\hat{\rho}_y$ and $\hat{\rho}_z$ are the components of the LoS unit vector. At a given epoch t_k, the observed vector $\boldsymbol{y}_{\text{obs}}$ is a 2-dimensional vector, as only one planet at a time is tracked. It is modeled starting from $\boldsymbol{h}$ as

$$\boldsymbol{y}_{\text{obs}} = \bar{\boldsymbol{y}}_{\text{obs}} + \boldsymbol{\nu} = \underbrace{\boldsymbol{h}(\boldsymbol{x}_k) + \boldsymbol{\Delta}_{\text{aber}}}_{\bar{\boldsymbol{y}}_{\text{obs}}} + \underbrace{3\sigma_{\text{str}} \boldsymbol{k}}_{\boldsymbol{\nu}} \tag{16}$$

where $\bar{\boldsymbol{y}}_{\text{obs}}$ is the unperturbed observed vector and $\boldsymbol{\nu}$ the measurement white noise. More specifically, $\boldsymbol{x}_k$ is the probe state at t_k, $\boldsymbol{\Delta}_{\text{aber}}$ the vector that introduces the celestial aberration error , σ_{str} the standard deviation of the LoS angular error from the star tracker (considered equal for both the azimuth and the elevation), and $\boldsymbol{k}$ a white noise vector with values within $[-1; 1]$. The measurement noise has zero mean and is uncorrelated to $\boldsymbol{w}$. An algorithmic overview of the system state space is reported in Table 1.

Tab. 1: System State Space of the Extended Kalman Filter.

System State Space
$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}(t), t) + \boldsymbol{w}$ $\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k) + \boldsymbol{\nu}_k$

3.2 Standard EKF Scheme

In the standard EKF, the batch EKF filter is exploited for measurement-update [16], whereas the state error covariance matrix $\boldsymbol{P}$ is employed to determine the state uncertainty. Tables 2 and 3 summarise the procedure adopted by the standard EKF scheme. In particular, Table 2 describes the approach exploited for the propagation steps. At t_k, $\boldsymbol{x}_{p_k}$ represents the predicted state vector with error covariance matrix $\boldsymbol{P}_{p_k}$, $\boldsymbol{\Phi}_k$ the state transition matrix (STM), $\boldsymbol{Q}$ the covariance matrix of the process noise, and $\boldsymbol{F}$ the Jacobian matrix of dynamics $\boldsymbol{f}$. On the other hand, Table 3 describes the procedure adopted inside the correction block. At t_k, $\boldsymbol{K}_k$ represents the Kalman gain, $\boldsymbol{x}_{c_k}$ the corrected state vector

Tab. 2: Scheme of the Propagation Block of the Extended Kalman Filter.

Propagation Block	
$\boldsymbol{x}_{p_k} = \boldsymbol{x}_{c_{k-1}} + \int_{t_{k-1}}^{t_k} \boldsymbol{f}(\boldsymbol{x}(t), t)\mathrm{d}t$	$\boldsymbol{x}_{c_0} = E[\boldsymbol{x}_0]$
$\boldsymbol{\Phi}_k = \boldsymbol{\Phi}_{k-1} + \int_{t_{k-1}}^{t_k} \boldsymbol{F}\boldsymbol{\Phi}\mathrm{d}t$	$\boldsymbol{\Phi}_0 = \boldsymbol{I}$
$\boldsymbol{S}_k = \boldsymbol{S}_{k-1} + \int_{t_{k-1}}^{t_k} \boldsymbol{\Phi}\boldsymbol{Q}\boldsymbol{\Phi}^T\mathrm{d}t$	$\boldsymbol{S}_0 = \boldsymbol{Q}$
$\boldsymbol{P}_{p_k} = \boldsymbol{\Phi}_k \boldsymbol{P}_{c_k} \boldsymbol{\Phi}_k^T + \boldsymbol{S}_k$	$\boldsymbol{P}_{c_0} = E[\boldsymbol{x}_0 \boldsymbol{x}_0^T]$

Tab. 3: Scheme of the Correction Block of the Extended Kalman Filter.

Correction Block
$\boldsymbol{K}_k = \boldsymbol{P}_{p_k}\boldsymbol{H}_k^\top (\boldsymbol{H}_k \boldsymbol{P}_{p_k} \boldsymbol{H}_k^\top + \boldsymbol{R}_k)^{-1}$
$\boldsymbol{x}_{c_k} = \boldsymbol{x}_{p_k} + \boldsymbol{K}_k[\boldsymbol{y}_k - \boldsymbol{h}(\boldsymbol{x}_{p_k})]$
$\boldsymbol{P}_{c_k} = (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H}_k)\boldsymbol{P}_{p_k}(\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H}_k)^\top + \boldsymbol{K}_k\boldsymbol{R}_k\boldsymbol{K}_k^\top$

with error covariance matrix $\boldsymbol{P}_{c_k}$, $\boldsymbol{R}_k$ the covariance matrix of the measurement noise, $\boldsymbol{H}_k$ the Jacobian matrix of measurement model $\boldsymbol{h}$, and $\boldsymbol{y}_k$ the observed vector.

3.3 Alternative EKF Schemes

The goal of this work consists of developing an autonomous navigation algorithm which has to be suitable to be adopted onboard a CubeSat. To achieve this, the numerical stability of the filter needs to be investigated to avoid the rise of numerical issues after the upcoming deployment on hardware [15]. To tackle this problem, alternative formulations of the standard filtering strategy are applied to both the time and the measurement update steps, which will result in the implementation of four additional alternative filtering schemes. Differently from the standard scheme, three of the new approaches exploit the sequential measurement update method. This one carries out the correction of the estimation without matrix inversion by processing the measurements once at a time. This helps saving computational time and memory, and is a desirable formulation for embedded systems with no matrix libraries [16]. Moreover, factorization methods, a non-dimensional strategy, or a combination of these two are exploited for the solution time-update.

3.3.1 Method A

Method A employs Potter's equation for processing the measurements, and the Householder algorithm for the square root matrix time propagation [16]. The core procedure of Method A consists of the complete replacement of the matrix $\boldsymbol{P}$ with its square-root matrix $\mathcal{S}$, which is numerically more stable due to the halving of the conditioning number.

3.3.2 Method B

Method B implements the UDU filtering scheme, in which the upper triangular matrix $\boldsymbol{U}$ and the diagonal matrix $\boldsymbol{D}$ are propagated. The observations are then processed employing a sequential algorithm [16].

3.3.3 Method C

In Method C, the non-dimensionalization procedure is applied to the model equations. This is possible by virtue of the Buckingham theorem, which states that the validity of the physics laws does not depend on a particular unit system [17]. As first step, a dimensional analysis of the variables of the estimator is performed. For this specific case, the dimensions of the components of the state vector are described only through two fundamental physical quantities: the length and the time. Thus, only a position $\bar{r}$ and an epoch $\bar{t}$ parameter are needed to non-dimensionalize the whole filtering procedure. The dimensionless state vector $\bar{\boldsymbol{x}}$ is defined as

$$\bar{\boldsymbol{x}} = \begin{bmatrix} \boldsymbol{r}/\bar{r} \\ \boldsymbol{v}/(\bar{r}\bar{t}^{-1}) \\ \boldsymbol{\tau}/\bar{t} \\ \boldsymbol{\eta}_{\mathrm{R}}/(\bar{r}\bar{t}^{-2}) \\ \boldsymbol{\eta}_{\mathrm{SRP}}/(\bar{r}\bar{t}^{-2}) \end{bmatrix} \tag{17}$$

and the same argument is applied for evaluating the dimensionless equation of motion $\bar{\boldsymbol{f}}$, the Jacobian matrices $\bar{\boldsymbol{F}}$ and $\bar{\boldsymbol{H}}$, and the covariance matrices $\bar{\boldsymbol{Q}}$, $\bar{\boldsymbol{R}}$ and $\bar{\boldsymbol{P}}_0$.
In detail, Method C adopts the standard filtering strategy with dimensionless model equations.

3.3.4 Method D

Method D adopts the non-dimensionalization strategy as well. In particular, it can be addressed as simply the non-dimensional version of Method A.
Tables 4 and 5 report the algorithms used for the implementation of the aforementioned techniques.

Tab. 4: Measurement update algorithms for alternative filtering techniques.

Method	Measurement update algorithm
A	Potter's equation (Sequential)
B	Sequential scheme
C	Non-dimens. standard (batch) scheme
D	Non-dimens. Potter's equation

Tab. 5: Time update algorithms for alternative filtering techniques.

Method	Time update algorithm
A	Householder algorithm (Prop. of $\mathcal{S}$)
B	Propagation of $\boldsymbol{U}$ and $\boldsymbol{D}$
C	Non-dimensional propagation of $\boldsymbol{P}$
D	Non-dimensional Householder algorithm

3.4 Deployment on the Raspberry Pi

At this point, the performance of the estimators needs to be tested on a platform comparable to a COTS miniaturized processor in terms of computational power and size. For this aim, a Raspberry Pi [1] is adopted, whose main specifics are given in Table 6.

Tab. 6: Technical specification of the Raspberry Pi 4 model B

Processor	RAM	OS
ARM Cortex-A7	2GB	Raspbian

The code deployment on the Raspberry Pi is performed by exploiting the *Matlab Support Package for Raspberry Pi Hardware* [2] and *Matlab Coder* [3]. Firstly, these apps generate automatically an equivalent C++ code starting from a `.m` file by creating a connection between MATLAB and the platform. Then, the coder deploys the algorithm as an executable file on the Raspberry Pi, on which it is run as a standalone application.

4 Case Study

In this work, the CubeSat performs an interplanetary transfer towards Mars. An arbitrary navigation procedure is adopted to recover the probe state by tracking the external planets.

4.1 Navigation Strategy

Figure 4 shows the navigation cycle adopted over the probe flying window. Starting from t_0, the spacecraft tracks the first planet of the optimal pair for 1 hour with a measurement frequency of 0.01 Hz. Subsequently, it performs a slew maneuver of 30 minutes during which no external observations are acquired. Then, the measurements of the second planet are gathered for another hour, and, finally, the state is propagated for 5 days. This procedure is repeated recursively 42 times, where each navigation leg lasts 5 days 2 hours and 30 minutes.

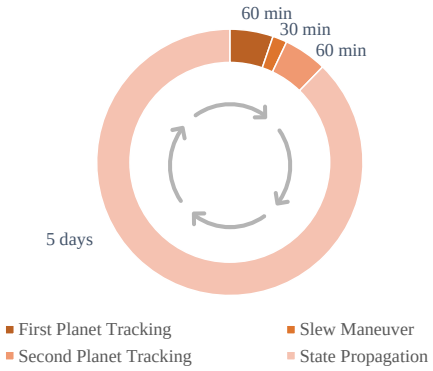

Fig. 4: Outline of the navigation cycle.

4.2 Settings

Figure 5 represents the probe nominal trajectory over the flying window. At t_0, the probe nominal state is defined by $\tilde{\boldsymbol{r}}_0 = [-3.97,\ 148,\ 3.23] \times 10^6$ km and $\tilde{\boldsymbol{v}}_0 = [-32.67,\ 0.87,\ 1.01]$ km/s described in the J2000 ecliptic reference frame. Whereas, the initial state uncertainties are reported in Table 7.

$\sigma_{\boldsymbol{r}}$ [km]	$\sigma_{\boldsymbol{v}}$ $[\frac{km}{s}]$	σ_τ [s]	σ_{SRP} $[\frac{km}{s^2}]$	σ_R $[\frac{km}{s^2}]$
10^4	10^{-1}	$3 \cdot 10^{-2}$	10^{-12}	10^{-12}

Tab. 7: Accuracy of the state components at t_0

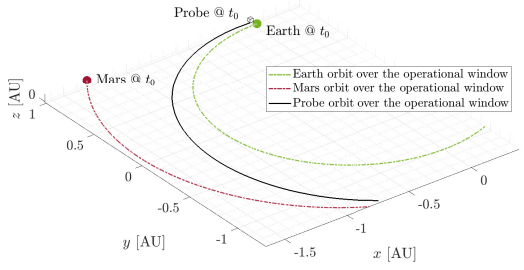

Fig. 5: Trajectory of the probe, Mars, and Earth starting at $t_0 = 10580$ [mjd2000].

It is assumed the onboard optical sensor is able to track planets with relative magnitude lower than 6 and SAA

[1] `https://www.raspberrypi.org/products/raspberry-pi-4-model-b/specifications/`, Last Visited on Sept. 2021

[2] `https://it.mathworks.com/help/supportpkg/raspberrypiio/`, Last Visited on Sept. 2021

[3] `https://it.mathworks.com/products/matlab-coder.html`, Last Visited on Sept. 2021

greater than 35°. Figure 6 shows the profile of the planets SAA and apparent magnitude over the probe trajectory. It can be noticed that Mercury, Venus, Uranus, and Neptune are never available to be tracked. The SAA of Mercury and Venus is always lower then 35°due to the Sun vicinity, whereas Uranus and Neptune are always too faint to be visible by the probe. The 3σ standard de-

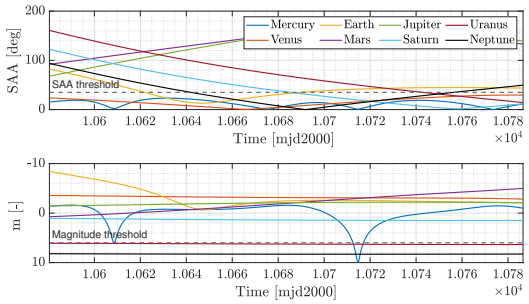

Fig. 6: Profile of the planets' SAA and apparent magnitude over the probe trajectory.

viation of the instrument measurement error is 15 arcsec. Table 8 summarises the features of the onboard optical sensor exploited for this analysis.

m_{limit} [-]	SAA_{limit} [°]	$3\sigma_{str}$ [arcsec]
6	35	15

Tab. 8: Features of the onboard optical sensor

Moreover, the probe parameters needed to model the SRP acceleration are reported in Table 9.

C_R [-]	m_s [kg]	A_s [m^2]
1.3	20	1

Tab. 9: Numerical values of the probe parameters to model the SRP

4.3 Optimal Beacons Selection

At the beginning of each navigation cycle the optimal couple of planets is selected. Since the time slot reserved to observation lasts only about three hours, it is reasonable to assume that the selected pair of planets preserves its optimality throughout the navigation cycle. Figure 7 shows the profile of $\mathcal{J}$ and γ of all the possible couples of planets. The black dots represent the optimal combination for each navigation cycle. It is possible to notice that the optimal selection is always feature by the lowest value of $\mathcal{J}$, accordingly to its definition.

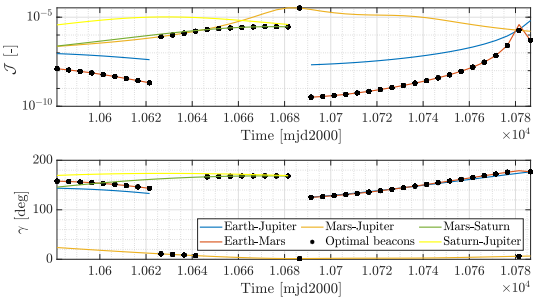

Fig. 7: $\mathcal{J}$ and γ trends over the operational window for the combinations of available planets. The optimal selection is represented with black dots.

4.4 Estimator Selection

The standard and alternative filtering schemes are applied to the test case. To select the approach most suited to the deployment, two drivers are exploited: 1) the computational time (CPU) and 2) the maximum conditioning number of the state error covariance matrix ($\mathcal{K}$). Table 10 shows the performances of the five filtering approaches in terms of these two parameters.

Tab. 10: Performances of the alternative filtering procedures

Method	CPU time [s]	Max $\mathcal{K}$ [-]
Standard	77	10^{33}
A	79	10^{16}
B	80	10^{32}
C	64	10^{25}
D	69	10^{12}

To carry out the trade-off, it is assigned a weight of 0.5 to both drivers. Moreover, marks from 1 to 5 are employed, where 1 represents the poorest alternative while 5 the best one. The criteria matrix is so developed.

Tab. 11: Criteria matrix

Method	CPU time [s]	Max $\mathcal{K}$ [-]	Total Score
Standard	3	1	2
A	2	4	3
B	1	2	1.5
C	5	3	4
D	4	5	4.5

As it can be seen, method D has the highest total score, and it so selected as the best suited method to be deployed on a miniaturized processor.

4.5 Performances

The performances of Method D fed by the optimal beacon selection strategy are shown in this section. In Fig. 8 the position error profiles and the covariance bounds are represented with solid and dashed lines, respectively. At the end of the flying window, the filter estimated the probe position with a 3σ accuracy of the order of 700 km. Figure 9 shows the velocity error profiles. The

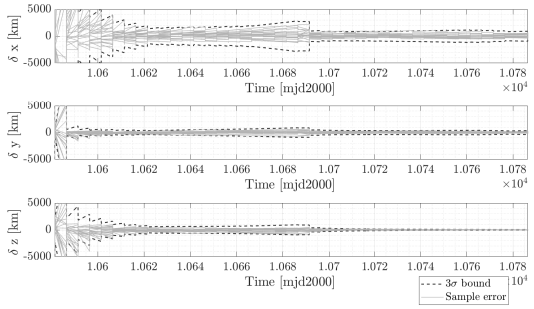

Fig. 8: Position error profiles with covariance bounds. Twenty samples are shown.

samples error profile is displayed with grey solid lines, whereas the dashed ones define the 3σ filter covariance bounds. At the end of the flying window, the 3σ error of the probe velocity is about 0.09 m/s. It can be noticed in Fig. 8 how the position error greatly decreases at the beginning of the 25^{th} navigation cycle, when the couple Earth–Mars is tracked. This can be explained by observing the profile of $\mathcal{J}$, whose value is subjected to the same rapid degrowth when this couple is selected as the optimal one. The enhancement of the estima-

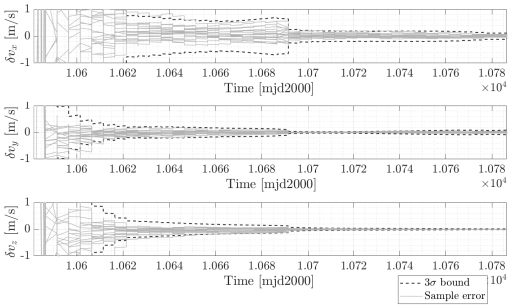

Fig. 9: Velocity error profiles with covariance bounds. Twenty samples are shown.

tion accuracy thanks to the exploitation of the optimal beacons selection is shown in Fig. 10. If the beacons are not optimally selected for each navigation leg, only one combination of planets, i.e., Mars–Jupiter, can be used throughout the navigation. Indeed, this one is the only available over the entire flying window. Figure 10 compares the estimator performances when the optimal beacons selection strategy is adopted against the case in which only the couple Mars–Jupiter is tracked. In general, the filter that uses the optimal selection has always the lowest 3σ covariance bound values, demonstrating thus the improvement in the estimation.

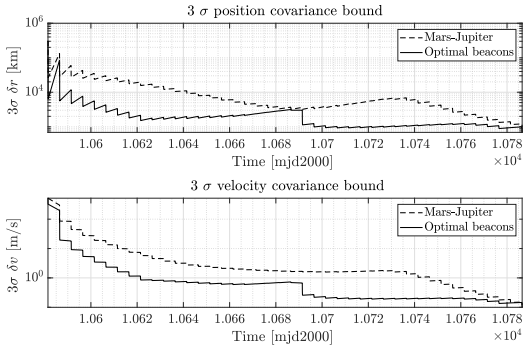

Fig. 10: Comparison of the covariance bounds with and without the exploitation of the optimal beacons selection strategy.

Moreover, Figure 10 shows how the profile of the position 3σ bound, when the optimal selection strategy is adopted, matches perfectly with the profile of the associated $\mathcal{J}$ values represented by the black dots in Fig. 7. At the beginning of the 10^{th} navigation cycle, when the couple Mars–Jupiter is selected for the first time, the 3σ covariance bound starts to worsen with respect to the previous estimation, and this is reflected on the selection of an optimal couple with greater $\mathcal{J}$ value. This behaviour continues until the couple Earth–Mars is chosen. At this point, the associated $\mathcal{J}$ is characterized by a value considerably lower, and this brings the estimation to greatly improve. For what concerns the velocity 3σ bound, the matching with $\mathcal{J}$ is not guaranteed by the optimization procedure since $\mathcal{J}$ is derived through the analysis of the kinematic celestial triangulation. Nevertheless, it is possible to notice that it is still respected, and also for the velocity the optimal selection allows to reach the best results in the estimation accuracy.

4.6 Performances with a Processor-in-the-Loop

Finally, Method D is deployed on a miniaturized processor to test its suitability to be adopted onboard CubeSats. A Raspberry Pi is chosen, whose computational capabilities are representative of a CubeSat onboard computer. The procedure exploited for the deployment is the one described in Sec. 3.4.

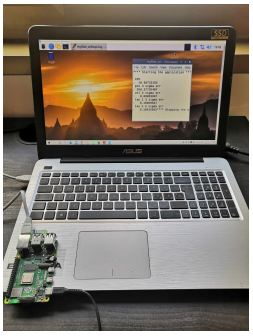

Fig. 11: Testing of the algorithm performances on a Raspberry-Pi.

The average CPU time required to run the filter over the operational window for 20 simulations is about 13 s, whereas the estimation accuracy is equal to the one obtained without a PIL simulation.

5 Conclusion

This work has tackled the autonomous navigation problem for deep-space CubeSats. Different strategies have been adopted to enhance the solution accuracy and the estimator numerical stability. In particular, the solution accuracy has been improved thanks to the exploitation of the optimal beacons selection method, and by introducing light effects corrections inside the model equations. In detail, the light-time effect has been corrected by including the vector $\boldsymbol{\tau}$ inside the state representation. Whereas, the correction of the light aberration effect has been performed at the acquisition of a new observation and before the estimation measurement-update. Finally, alternative filtering approaches have been developed and their performances analysed to prevent the rise of numerical issues after the code deployment on a real platform. Among the five schemes described in this work, Method D has been chosen as the most appropriate alternative to be adopted onboard a CubeSat. To study its actual behaviour and performances with a PIL, Method D has been deployed on a Raspberry-Pi. The filter estimates the probe position and velocity with 3σ bounds of the order of 700 km and 0.09 m/s, respectively. The suitability of Method D for CubeSats applications is so verified.

Acknowledgments

This research is part of EXTREMA, a project that has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant Agreement No. 864697).

Bibliography

[1] T. Villela, C. Costa, A. Brandão, F. Bueno, and R. Leonardi. Towards the thousandth cubesat: A statistical overview. *International Journal of Aerospace Engineering*, 2019:1–13, 01 2019. DOI: 10.1155/2019/5063145.

[2] R. Walker, D. Binns, C. Bramanti, M. Casasco, P. Concari, D. Izzo, D. Feili, P. Fernandez, J. Fernandez, P. Hager, D. Koschny, V. Pesquita, N. Wallace, I. Carnelli, M. Khan, M. Scoubeau, and D. Taubert. Deep-space cubesats: Thinking inside the box. *Astronomy & Geophysics*, 59:5.24–5.30, 10 2018. DOI: 10.1093/astrogeo/aty232.

[3] S. Speretta, F. Topputo, J. Biggs, P. Di Lizia, M. Massari, K. Mani, D. Dei Tos, S. Ceccherini, V. Franzese, A. Cervone, P. Sundaramoorthy, R. Noomen, S. Mestry, A. do Carmo Cipriano, A. Ivanov, D. Labate, L. Tommasi, A. Jochemsen, J Gailis, R. Furfaro, V. Reddy, J. Vennekens, and R. Walker. Lumio: achieving autonomous operations for lunar exploration with a cubesat. In *2018 SpaceOps Conference*, page 11, 2018. DOI: 10.2514/6.2018-2599.

[4] F. Topputo, Y. Wang, C. Giordano, V. Franzese, H. Goldberg, F. Perez-Lissi, and R. Walker. Envelop of reachable asteroids by m-argo cubesat. *Advances in Space Research*, 67(12):4193–4221, 2021. DOI: 10.1016/j.asr.2021.02.031.

[5] H. R. Goldberg, O. Karatekin, B. Ritter, A. Herique, P. Tortora, C. Prioroc, B. G. Gutierrez, P. Martino, and I. Carnelli. The juventas cubesat in support of esa's hera mission to the asteroid didymos. In *33rd Annual AIAA/USU Conference on Small Satellites*, 8 2019. Logan, Utah, USA.

[6] F. Ferrari, V. Franzese, M. Pugliatti, C. Giordano, and F. Topputo. Preliminary mission profile of hera's milani cubesat. *Advances in Space Research*, 67(6):2010–2029, 2021. DOI: 10.1016/j.asr.2020.12.034.

[7] I. Carnelli, P. Martino, D. Escorial, A. Rugina, J. Gil, H. Greus, A. Valverde, A. Zuccaro, B. Bonnafous, C. M. Moya, C. Honvault, F. P. Lissi, G. Tzeremes, M. Khan, M. Küppers, M. Tourloukis, P. Muñoz, P. Concari, R. Moissl, and A. Accomazzo. Esa's hera mission to asteroid dimorphos. In *Europlanet Science Congress 2020*, 09 2020. Virtual. DOI: 10.5194/epsc2020-1119.

[8] S. M. Lichten, D. S. Abraham, B. Arroyo, S. W. Asmar, J. Bell, and C. D. Edwards. Allocation of deep space network ground system tracking and communications assets during the 2020-2021 timeframe of the "mars armada". In *2018 SpaceOps Conference*, 2018. Marseille, France. DOI: 10.2514/6.2018-2502.

[9] M. Quadrelli, Lincoln J. Wood, J. E. Riedel, M. McHenry, M. Aung, L. Cangahuala, R. Volpe, P. Beauchamp, and J. A. Cutts. Guidance, navigation, and control technology assessment for future planetary science missions. *Journal of Guidance Control and Dynamics*, 38:1165–1186, 2015. DOI: doi.org/10.2514/1.G000525.

[10] R. Karimi and D. Mortari. Interplanetary autonomous navigation using visible planets. *Journal of Guidance, Control, and Dynamics*, 38:1–6, 04 2015. DOI: 10.2514/1.G000575.

[11] V. Franzese and F. Topputo. Optimal beacons selection for deep-space optical navigation. *The Journal of the Astronautical Sciences*, 67:1775–1792, 11 2020. DOI: 10.1007/s40295-020-00242-z.

[12] S. B. Broschart, N. Bradley, and S. Bhaskaran. Kinematic approximation of position accuracy achieved using optical observations of distant asteroids. *Journal of Spacecraft and Rockets*, 56(5):1383–1392, 2019. DOI: 10.2514/1.A34354.

[13] J. A. Christian. Starnav: Autonomous optical navigation of a spacecraft by the relativistic perturbation of starlight. *Sensors*, 19(19), 2019. DOI: 10.3390/s19194064.

[14] D. Mortari and D. Conway. Single-point position estimation in interplanetary trajectories using star trackers. *Celestial Mechanics and Dynamical Astronomy*, 128, 11 2016. DOI: 10.1007/s10569-016-9738-4,.

[15] C. N. D'Souza J. R. Carpenter. Navigation Filter Best Practices. Technical Report 20180003657, NASA, 04 2018.

[16] D. Simon. *Alternate Kalman filter formulations*, chapter 6, pages 149–181. John Wiley & Sons, Ltd, 2006.

[17] E. Buckingham. On Physically Similar Systems; Illustrations of the Use of Dimensional Equations. *Physical Review*, 4(4):345–376, October 1914.