(Preprint) AAS 23-126

SMALL BODY NAVIGATION AND GRAVITY ESTIMATION USING KALMAN FILTER AND LEAST-SQUARES FITTING

Julio C. Sanchez* **and Hanspeter Schaub†**

This paper considers the problem of simultaneous navigation and inhomogeneous gravity estimation around a small body. The available measurements are acquired by an on-board camera able to track small body landmarks. The proposed solution approach combines a dynamic model compensated unscented Kalman filter with a least-squares gravity fitting through gradient descent. The filter fills a training dataset with position and unmodeled acceleration estimates. Prior to the filter, the measurements are processed to determine an initial estimate of the spacecraft position and its uncertainty. Both spherical harmonics and mascon gravity models have been embedded within the proposed scheme. The numerical results demonstrate that the mascon model does not diverge within the Brillouin sphere, as is the case for spherical harmonics, while not loosing accuracy outside of it.

INTRODUCTION

Small bodies exploration enables a deeper understanding of the Solar System formation processes[1] and planetary protection techniques.[2] Spacecraft dynamics, in the vicinity of small bodies, is highly perturbed from Keplerian motion[3] due to the strong effects of inhomogeneous gravity and solar radiation pressure. These perturbations have to be taken into account in order to design safe flight operations. However, in the early stage small body missions, only coarse information of the body shape can be inferred through ground-based radar and telescope measurements before launch. While this is not a concern for solar radiation pressure (as it does not depend on the small body properties), it certainly is for the small body gravity field. Consequently, the inhomogeneous gravity field can only be sensed while on the fly through in-situ observations. The classic approach for small bodies gravity estimation largely relies on ground-based systems for radiometric measurements and data processing.[4] As communication with Earth-based mission control is required, the whole process is slow due to signal delay and Deep Space Network accessibility. Consequently, there is a need of on-board autonomy for the small body gravity field estimation.

Recently, several publications[5–10] are proposing solutions to the autonomous gravity estimation problem. References 5-6 consider satellites in close formation, namely swarm, for simultaneous navigation and gravity estimation. These works use landmark-based measurements and inter-satellite ranging within a centralized unscented Kalman filter (UKF). The filter estimates the spacecrafts state and spherical harmonics coefficients. Second-order degree gravity and solar sail reflectivity degradation are inferred in Ref. 7. These parameters are estimated via a sequential fitting of

*MSCA postdoctoral fellow, Ann and H.J. Smead Department of Aerospace Engineering Sciences, University of Colorado, Boulder, 431 UCB, Colorado Center for Astrodynamics Research, Boulder, CO, 80309.

†Professor and Department Chair, Schaden Leadership Chair, Ann and H.J. Smead Department of Aerospace Engineering Sciences, University of Colorado, Boulder, 431 UCB, Colorado Center for Astrodynamics Research, Boulder, CO, 80309. AAS Fellow, AIAA Fellow.

the observed perturbing acceleration. However, the simulations lack realism as perfect knowledge of the spacecraft position is assumed. A gravity model learning-based predictive control is developed in Ref. 8. The spherical harmonics coefficients are inferred by averaging individual estimates of a satellite constellation, thus mitigating outliers and speeding the convergence. Each satellite embarks its own UKF for simultaneous state and gravity estimation by acquiring landmarks-based measurements. In Ref. 9, Hopfield neural networks are demonstrated to provide similar performance as extended Kalman filtering (EKF) for the spherical harmonics coefficients estimation. A dynamical compensated filter generating a dataset to train a physics-informed neural network gravity representation is proposed by Ref. 10. The authors of this paper utilize a similar approach by combining a dynamical model compensated unscented Kalman filter (DMC-UKF) with a least-squares fit of spherical harmonics coefficients.[11]

The literature review highlights a clear preference for the spherical harmonics model[5–9,11] (except Ref 10). However, the spherical harmonics model is known to diverge within the Brillouin sphere, thus being unfit for descent and landing operations. In the works with the higher fidelity of measurements,[5,6,8] the spherical harmonics may be preferred in order to maintain the filter dimensionality as low as possible (state and gravity are jointly estimated). Alternatively, if the filter gravity coefficients are substituted by the unmodeled acceleration, the simultaneous navigation and gravity estimation process may be more flexible. For example, there are gravity models with the potential to avoid the Brillouin sphere divergence (such as mascon[12] or physics informed neural networks[13]) that may be of consideration. In particular, this manuscript uses mascon models by a priori fixing the point masses randomly within the small body volume. Consequently, the decision variables are the mascon standard gravity parameters.

The main contribution of this work is the development of a simultaneous navigation and gravity estimation scheme that is flexible to models other than spherical harmonics. In particular, the mascon model has been implemented and the results demonstrate its superiority, in terms on the global gravity accuracy, over spherical harmonics. The proposed strategy just relies on an on-board camera able to track landmarks on the small body surface (the image processing that provides the landmarks pixels is out of the scope of this work). By using the visible landmarks pixels, the satellite position and associated uncertainty can be determined. This estimation is used as the measurement input to the DMC-UKF which fills the position and unmodeled accelerations datasets. When the dataset is full after a certain period of time, the gravity parameters are fitted using a momentum-based gradient descent algorithm. This updates the filter gravity model and the process is sequentially repeated.

The structure of the manuscript is as follows. The dynamics around small bodies is firstly introduced while making emphasis on gravity models. Secondly, the pinhole camera model, that maps 3D coordinates to pixels, is presented. Subsequently, the landmark-based position determination, DMC-UKF and gravity estimation algorithms are described. Then, numerical simulations are shown and discussed. Finally, the paper is finished with some conclusions and future work remarks.

DYNAMICS AROUND SMALL BODIES

A spacecraft orbiting in the proximity of a small body is perturbed by the inhomogeneous gravity field, the Sun's third body gravity and the solar radiation pressure such that

$$\ddot{\mathbf{r}} = -\frac{\mu \mathbf{r}}{r^3} + \mathbf{R}_A^N \mathbf{a}_{\text{grav}} + \mathbf{a}_{\odot} + \mathbf{a}_{\text{SRP}}, \tag{1}$$

where $\mathbf{r}$ is the spacecraft position expressed in an inertial frame, μ is the small body standard gravity parameter and $\mathbf{a}_{\text{grav}}$ is the inhomogeneous gravity acceleration (expressed in the small body rotating

frame). The term $\mathbf{R}_A^N$ is the direction cosine matrix from the rotating small body centred frame to the inertial one. It is assumed that the small body rotates around its major inertia axis z_A (as is the usual case for small bodies), thus

$$\mathbf{R}_A^N = \begin{bmatrix} \cos\left(\text{LST}_0 + \omega_A t\right) & -\sin\left(\text{LST}_0 + \omega_A t\right) & 0 \\ \sin\left(\text{LST}_0 + \omega_A t\right) & \cos\left(\text{LST}_0 + \omega_A t\right) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{2}$$

where LST_0 is the small body initial local sidereal time and ω_A its rotational velocity. The solar perturbations, namely the third body gravity $\mathbf{a}_\odot$ and solar radiation pressure $\mathbf{a}_{\text{SRP}}$, are described as

$$\mathbf{a}_\odot = -\mu_\odot \left(\frac{\mathbf{r}_A + \mathbf{r}}{\|\mathbf{r}_A + \mathbf{r}\|_2^3} - \frac{\mathbf{r}_A}{r_A^3} \right), \qquad \mathbf{a}_{\text{SRP}} = \frac{C_R S W_\oplus r_\oplus^2}{mc\|\mathbf{r}_A + \mathbf{r}\|_2^3}(\mathbf{r}_A + \mathbf{r}), \tag{3}$$

where $\mathbf{r}_A$ is the small body relative position with respect to the Sun and $\mu_\oplus = 1.3271244 \cdot 10^{20}$ m^3/s^2 is the Sun's standard gravity parameter. The solar radiation pressure is described with the cannonball model. The term m is the spacecraft mass, C_R its reflection coefficient, S is the exposed surface to the photons, $W_\oplus = 1366$ W/m^2 is the mean energy flux received from the Sun at the mean orbital distance of $r_\oplus = 1$ AU and $c = 3 \cdot 10^8$ m/s^2 is the speed of light.

Inhomogeneous gravity field

This section describes the inhomogeneous gravity models, that are being used throughout this paper, characterizing the term $\mathbf{a}_{\text{grav}}$. These are the polyhedron, spherical harmonics and mascon models respectively.

Polyhedron: the classic polyhedron gravity model, as described by Ref. 14, assumes the small body has a constant density. The small body shape is approximated by a polyhedron which is defined in terms of faces composed by vertexes. The inhomogeneous gravity acceleration is as follows

$$\mathbf{a}_{\text{grav}} = \frac{\mu \mathbf{r}}{r^3} - \frac{\mu}{V} \left(\sum_{e \in \text{edges}} \mathbf{E}_e \cdot \mathbf{r}_e L_e - \sum_{f \in \text{faces}} \mathbf{F}_f \mathbf{r}_f w_f \right), \tag{4}$$

where V is the body volume, $\mathbf{r}_e$ is the relative position of the evaluation point with respect to the edge origin, $\mathbf{E}_e$ is the dyad product resulting from the edge and face normals, L_e is the potential of the edge as a 1D wire, $\mathbf{r}_f$ is the relative position of the evaluation point with respect to a vertex on a face, $\mathbf{F}_f$ is the outer product of the face normal vector and w_f is the solid angle of the face as viewed from the evaluation point. The explicit details of these terms can be consulted in Ref. 14. Note that the Keplerian term is added in order to be consistent with Eq. (1). As the polyhedron shape is converged to the real one by augmenting the number of faces and vertexes, the exterior gravity field also does (if the constant density assumption holds). However, the computational burden increases steeply due to the summations over all edges and faces of Eq. (4). This precludes the use of this model for real-time applications. In this work, the polyhedron gravity is used as ground truth model for the simulations.

Spherical harmonics: the spherical harmonics representation is the solution to Laplace's equation in a spherical domain.[15] Its normalized version, truncated to $\bar{i}$ degree, is as follows

$$\mathbf{a}_{\text{grav}} = \sum_{i=2}^{\bar{i}} \sum_{j=0}^{i} \frac{\mu}{r^2} \left(\frac{R_e}{r} \right)^i \begin{bmatrix} -(i+1) P_i^{(j)} (C_{ij} \cos(j\lambda) + S_{ij} \sin(j\lambda)) \\ \frac{j}{\cos\phi} P_i^{(j)} (-C_{ij} \sin(j\lambda) + S_{ij} \cos(j\lambda)) \\ \cos\phi P_i^{(j)'} (C_{ij} \cos(j\lambda) + S_{ij} \sin(j\lambda)) \end{bmatrix}, \tag{5}$$

where (λ, ϕ) are the evaluation longitude and latitude, R_e is a normalization radius, C_{ij} and S_{ij} are the spherical harmonics coefficients characterizing the gravity field, $P_i^{(j)}$ and $P_i^{(j)'}$ are the normalized Legendre function and its derivative. The spherical harmonics model is very popular within the astrodynamics community as it encodes fundamental gravity features with few parameters (e.g. a constant density ellipsoid is simply characterized by C_{20} and C_{22}). On the other hand, since it prescribes spherical functions, it diverges within the body shape circumscribing sphere (namely Brillouin sphere). For the sake of brevity, Eq. (5) describes the spherical harmonics expression with a singularity at $\phi = \pm\pi/2$. However, the non-singular Pines spherical harmonics model[16] is used in this paper.

Mascon: the mascon model was proposed in the 70s as an alternative to spherical harmonics in order to fit sparse data. It is based on the idea that the gravity field could be represented by a finite number of point masses (namely mascons) such that

$$\mathbf{a}_{\text{grav}} = -\sum_{k=1}^{n_M} \mu_k \frac{\mathbf{r} - \mathbf{r}_k}{\|\mathbf{r} - \mathbf{r}_k\|_2^3}, \tag{6}$$

where μ_k and $\mathbf{r}_k$ are, respectively, the standard gravity parameter and position of each mascon. The term n_M is the number of mascons. In order to avoid singularities in the external gravity field evaluation, the point masses shall be within the small body volume. This requires certain knowledge of the body shape (e.g. polyhedron geometry).

PINHOLE CAMERA MODEL

Optical navigation is one of the most reliable methods to autonomously acquire relative measurements with respect to a small body. Let assume a camera that is able to acquire small body images. From a high-level navigation perspective, the imaging can be described with a pinhole camera model that transforms 3D points to 2D features.[17] The projection of a 3D point, expressed in the camera reference frame, (x, y, z) to virtual image plane coordinates (u, v) is as follows

$$\begin{bmatrix} u \\ v \end{bmatrix} = \frac{f}{z} \begin{bmatrix} x \\ y \end{bmatrix}, \tag{7}$$

where f is the pinhole camera focal length. Since the image is digital, the 3D point will correspond to a pixel (p_x, p_y) in the image plane as follows

$$p_x = \begin{cases} \text{ceil}(u/w_p) & \text{if} \quad u > 0, \\ \text{floor}(u/w_p) & \text{if} \quad u < 0, \end{cases} \qquad p_y = \begin{cases} \text{ceil}(v/w_p) & \text{if} \quad v > 0, \\ \text{floor}(v/w_p) & \text{if} \quad v < 0, \end{cases} \tag{8}$$

where w_p is the camera pixel width. The projection process is illustrated in Fig. 1. Note that $\mathbf{0}_C$ is the camera aperture and $\mathbf{z}_C$ denotes the optical axis. The frame $C \equiv \{\mathbf{O}_C, \mathbf{x}_C\mathbf{y}_C\mathbf{z}_C\}$ defines the camera frame. The pixel width is determined by the camera sensor size and its resolution. An important parameter of the camera configuration is the field of view (FOV) which refers to what is visible through the lens. The field of view is characterized by horizontal and vertical angles that are related to the focal length and sensor size as

$$\text{FOV} \equiv 2\arctan(n_{p_x} w_p / 2f) \times 2\arctan(n_{p_y} w_p / 2f), \tag{9}$$

where n_{p_x} and n_{p_y} are, respectively, the horizontal and vertical number of pixels. Note that the camera focal length is the variable parameter that drives how much area of the small body is visible.

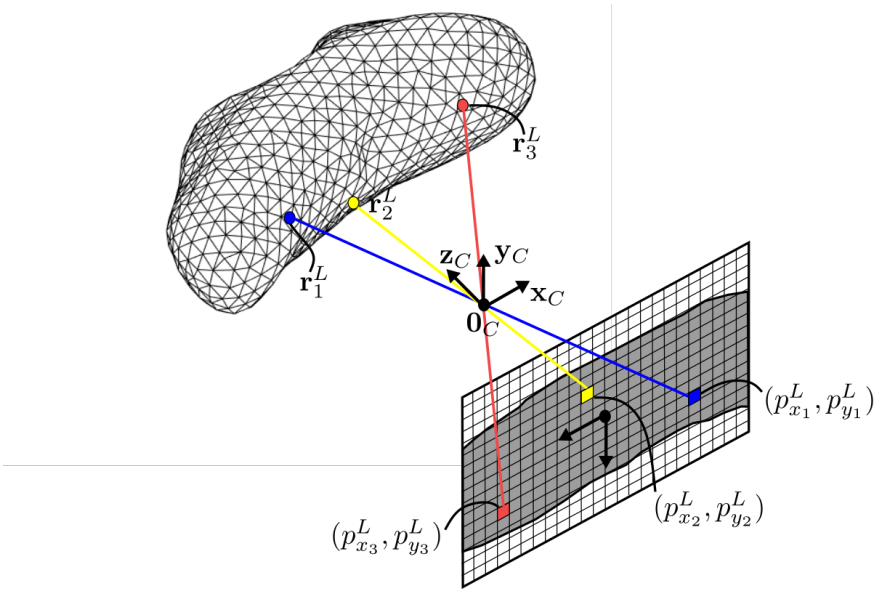

Figure 1. Illustration of the pinhole camera model.

SIMULTANEOUS NAVIGATION AND GRAVITY ESTIMATION

This section describes the simultaneous navigation and gravity estimation algorithms. These comprise the landmarks-based position determination, the dynamical model compensated unscented Kalman filter and the batch least-squares fitting of the gravity parameters.

Landmarks-based position determination

Landmarks are small body surface features (e.g. craters) that are mapped during high orbit phases to enable accurate relative navigation. Consequently, in this work, the landmark locations $\mathbf{r}_j^L$ are known to a certain extent. Moreover, by tracking them with the camera, the visible landmarks pixels $(p_{x_j}^L, p_{y_j}^L)$ are assumed to be the available measurements.

The visible landmark pixels and known locations could be directly used as inputs to the navigation filter.[5,8] Differently from that approach, in this work, the previous data is used for an initial determination of the spacecraft position and its associated uncertainty. The output of this initial determination is subsequently sent to the DMC-UKF. The landmarks-based position determination algorithm exploits the relative geometry between the spacecraft and landmarks as follows. The landmark-spacecraft lines should intersect in the camera aperture (from now on, it will be assumed to coincide with the spacecraft center of mass for the sake of simplicity). This geometrical property is illustrated in Fig. 1 for three landmarks. However, landmark locations will not be exactly known and pixelation error will be present due to the use of a digital camera. As a consequence, the intersection of all landmark-spacecraft lines is not unique due to the presence of errors. Due to this fact, they would be denoted as landmark-spacecraft pseudolines. Nonetheless, the spacecraft position (unique intersection point for the ideal case) could be approximately determined by finding the nearest point with respect to all the spacecraft-landmark pseudolines (in the least-squares sense). This algorithm is described below. Firstly, let detail how the landmark-spacecraft pseudoline can be constructed from the landmark pixel:

1. Center the landmark pixel:

$$p'_{x_j} = \begin{cases} p^L_{x_j} - 0.5 & \text{if} \quad p^L_{x_j} > 0, \\ p^L_{x_j} + 0.5 & \text{if} \quad p^L_{x_j} < 0, \end{cases} \qquad p'_{y_j} = \begin{cases} p^L_{y_j} - 0.5 & \text{if} \quad p^L_{y_j} > 0, \\ p^L_{y_j} + 0.5 & \text{if} \quad p^L_{y_j} < 0. \end{cases}$$

2. Transform it to 2D image coordinates: $(u'_j, v'_j) = w_p(p'_{x_j}, p'_{y_j})$.

3. Compute the line-of-sight vector $\mathbf{w}^C_j$ of the landmark from the spacecraft:

$$\mathbf{w}^C_j = \frac{1}{\sqrt{(u'_j)^2 + (v'_j)^2 + 1}} \begin{bmatrix} u'_j \\ v'_j \\ 1 \end{bmatrix}.$$

4. Project the line-of-sight to the small body centred fixed frame: $\mathbf{w}^A_j = \mathbf{R}^A_C \mathbf{w}^C$.

5. Construct the parametric equation of the landmark-spacecraft pseudoline: $s\mathbf{w}^A_j = \mathbf{r}' - \mathbf{r}^L_j$.

Note that s is an independent variable and $\mathbf{r}'$ is the set of points sweeping across the line. The distance l_j of an arbitrary point $\mathbf{r}$ to a spacecraft-landmark pseudoline can be computed as

$$l_j = \|(\mathbf{r} - \mathbf{r}^L_j) \times \mathbf{w}^A_j\|_2. \tag{10}$$

Let recall that $\mathbf{w}^A_j$ is a unit vector. Then, the goal is to find the point $\mathbf{r}$ that minimizes the sum of the squared distances L_{lmk} for n_L landmarks. The variable L can be expressed as

$$\begin{aligned} L_{\text{lmk}} = \sum_{j=1}^{n_L} l_j^2 &= \sum_{j=1}^{n_L} [(\mathbf{r} - \mathbf{r}^L_j) \times \mathbf{w}^A_j]^T [(\mathbf{r} - \mathbf{r}^L_j) \times \mathbf{w}^A_j] \\ &= \sum_{j=1}^{n_L} (\mathbf{r} - \mathbf{r}^L_j)^T (\mathbf{r} - \mathbf{r}^L_j) - [(\mathbf{r} - \mathbf{r}^L_j) \cdot \mathbf{w}^A_j]^T [(\mathbf{r} - \mathbf{r}^L_j) \cdot \mathbf{w}^A_j], \end{aligned} \tag{11}$$

where the identity $(\mathbf{a} \times \mathbf{b})^T (\mathbf{a} \times \mathbf{b}) = (\mathbf{a}^T \mathbf{a})(\mathbf{b}^T \mathbf{b}) - (\mathbf{a}^T \mathbf{b})(\mathbf{b}^T \mathbf{a})$ has been used. The closest point $\mathbf{r}^*$ can be computed by taking the first order derivative of L with respect to $\mathbf{r}$ and equaling to zero

$$\frac{dL_{\text{lmk}}}{d\mathbf{r}} = \sum_{j=1}^{n_L} \frac{dl_j^2}{d\mathbf{r}}\bigg|_{r=r^*} = \sum_{j=1}^{n_L} 2(\mathbf{r}^* - \mathbf{r}^L_j) - 2[(\mathbf{r}^* - \mathbf{r}^L_j) \cdot \mathbf{w}^A_j]\mathbf{w}^A_j = \mathbf{0}, \tag{12}$$

then, $\mathbf{r}^*$ is determined by

$$\left(n_L \mathbf{I} - \sum_{j=1}^{n_L} \mathbf{w}^A_j (\mathbf{w}^A_j)^T \right) \mathbf{r}^* = \sum_{j=1}^{n_L} \mathbf{r}^L_j - (\mathbf{r}^L_j \cdot \mathbf{w}^A_j)\mathbf{w}^A_j, \tag{13}$$

which is a simple 3x3 system of linear equations where $\mathbf{r}^*$ can be easily cleared by inverting the left-side matrix. There are two cases where Eq. (13) is unsolvable. The first one is $n_L = 1$ where it is evident that at least another pseudoline is needed to have an intersection. The other case is when all the pseudolines are parallel.

An estimation on the $\mathbf{r}^*$ uncertainty, in terms of its covariance matrix $\mathbf{\Sigma_{zz}}$, can be obtained through the reduced chi-squared statistic of the residuals χ^2

$$\mathbf{\Sigma_{zz}} = \chi^2 (\mathbf{J}_L \mathbf{J}_L^T)^{-1}, \quad \chi^2 = \frac{L_{\text{lmk}}(\mathbf{r}^*)}{3n_L - 3}, \quad \mathbf{J}_L = [(\mathbf{w}^A_1)^\times, \ldots, (\mathbf{w}^A_{n_L})^\times], \tag{14}$$

where $(\mathbf{w}^A_j)^\times$ is the cross-product matrix of each line-of-sight vector.

Dynamical model compensated unscented Kalman filter

The DMC-UKF has the double task of estimating the spacecraft state (position and velocity) and the umodeled perturbing acceleration. The DMC approach relies on augmenting the filter state with the unmodeled acceleration term[18] which allows to recursively estimate the dynamics missing part.

The UKF[19] has the distinctive capability to transform Gaussian distributions with non-linear functions. Although, by applying a nonlinear transformation, the resulting distribution will not be Gaussian, the UKF approximates the result with a higher degree of accuracy than a linearized EKF. To this end, the UKF uses a particle-based approach, namely unscented transform (UT), which chooses a reduced set of samples (namely sigma points $\boldsymbol{\chi}_{\mathbf{x}}$) around the initial distribution mean as

$$\boldsymbol{\chi}_{\mathbf{x}}^{[j]} = \hat{\mathbf{x}} + \text{sgn}(j) \cdot \left(\sqrt{(n+\lambda)\boldsymbol{\Sigma}_{\mathbf{xx}}} \right)_{|j|}, \quad j = -n \dots n, \tag{15}$$

where $\hat{\mathbf{x}}$ is the distribution mean, $\boldsymbol{\Sigma}_{\mathbf{xx}}$ its covariance and n its dimension. Let use the UT to transform the initial distribution, $\mathbf{x} \sim N_n(\mathbf{x}, \boldsymbol{\Sigma}_{\mathbf{xx}})$, to $\mathbf{y} = \mathbf{f}(\mathbf{x})$. The UT works by applying the function $\mathbf{f}$ to each sigma point and using the outcome samples to reconstruct the final distribution $\mathbf{y} \sim N_m(\mathbf{y}, \boldsymbol{\Sigma}_{\mathbf{yy}})$ as

$$\hat{\mathbf{y}} = \sum_{j=-n}^{n} w_m^{[j]} \mathbf{f}(\boldsymbol{\chi}_{\mathbf{x}}^{[j]}), \qquad \boldsymbol{\Sigma}_{\mathbf{yy}} = \sum_{j=-n}^{n} w_c^{[j]} (\hat{\mathbf{y}} - \mathbf{f}(\boldsymbol{\chi}_{\mathbf{x}}^{[j]})(\hat{\mathbf{y}} - \mathbf{f}(\boldsymbol{\chi}_{\mathbf{x}}^{[j]}))^T. \tag{16}$$

Process propagation: the DMC-UKF dynamics is modeled as

$$\frac{d}{dt} \begin{bmatrix} \mathbf{r} \\ \mathbf{v} \\ \mathbf{a} \end{bmatrix} = \begin{bmatrix} \mathbf{v} \\ -\mu \mathbf{r}/r^3 + \mathbf{a}_{\text{grav}}(\mathbf{r}) + \mathbf{a}_{\odot}(\mathbf{r}) + \mathbf{a}_{\text{SRP}}(\mathbf{r}) + \mathbf{a} \\ \mathbf{0} \end{bmatrix}, \tag{17}$$

where the small body standard gravity parameter μ, the Sun's third-body gravity and solar radiation pressure features are assumed to be known. Then, the unknown dynamics component corresponds to the inhomogeneous gravity $\mathbf{a}_{\text{grav}}$ which is compensated with the unmodeled acceleration $\mathbf{a}$. For the sake of simplicity (avoid tuning parameters), a zero order Gauss Markov process is assumed for the unmodeled acceleration dynamics. Equation (17) is propagated using a simple forward Euler integration as

$$\mathbf{x}_k = \mathbf{x}_{k-1} + \dot{\mathbf{x}}_{k-1} \Delta t_{\text{int}}, \tag{18}$$

where $\mathbf{x} = [\mathbf{r}^T, \mathbf{v}^T, \mathbf{a}^T]^T$ denotes the DMC-UKF state and Δt_{int} is the integration time step (which could be smaller than the measurement sampling time to augment the integration accuracy).

State to measurements transformation: the map from state to incoming measurements is

$$\mathbf{r} = \mathbf{R}_A^N \mathbf{r}^*, \tag{19}$$

where $\mathbf{r}^*$ is the position, expressed in the rotating small body centred frame, determined in Eq. (13).

DMC-UKF algorithm: the implementation of the DMC-UKF is described in Algorithm 1. Let start with the current DMC-UKF state $\hat{\mathbf{x}}_0$ and covariance $\boldsymbol{\Sigma}_{\mathbf{xx}_0}$. When measurements arrive after a certain period of time, the state is updated as follows. The steps 2-3 generate an a-priori estimation of the state $\mathbf{x}^-$ and covariance $\boldsymbol{\Sigma}_{\mathbf{xx}}^-$ through the process (referred as $\mathbf{f}$) UT. The process covariance $\boldsymbol{\Sigma}_{\mathbf{ff}}$ is added in order to account for the mismatch between truth and process dynamics. The steps

4-5 map (referred as $\mathbf{g}$) the a-priori state distribution to measurement space which yields a Gaussian with mean $\hat{\mathbf{z}}^-$ and covariance $\mathbf{\Sigma}^-_{\mathbf{zz}}$. The covariance $\mathbf{\Sigma}_{\mathbf{zz}}$ is added to account for the measurements dispersion (that is the landmarks-based position determination uncertainty). The step 6 computes the cross-correlation matrix $\mathbf{H}_{\mathbf{xz}}$ between state and measurements. This matrix is used in step 7 to compute the Kalman gain $\mathbf{K}$. Finally, by using the measurement $\mathbf{z}$ (landmarks-based position determination), the Kalman linear equation is applied to the a-priori state and uncertainty in order to yield the a-posteriori distribution mean $\hat{\mathbf{x}}$ and covariance $\mathbf{\Sigma}_{\mathbf{xx}}$. Following,[19] the weights are

Algorithm 1: DMC-UKF

1 **begin**
2 Apply the UT Eq. (15)-(16) to the process of Eq. (18): $(\hat{\mathbf{x}}^-, \mathbf{\Sigma}^-_{\mathbf{xx}}) \equiv \text{UT}(\hat{\mathbf{x}}_0, \mathbf{\Sigma}_{\mathbf{xx}_0}, \mathbf{f})$;
3 Add the process uncertainty: $\mathbf{\Sigma}^-_{\mathbf{xx}} \leftarrow \mathbf{\Sigma}^-_{\mathbf{xx}} + \mathbf{\Sigma}_{\mathbf{ff}}$;
4 Apply the UT Eq. (15)-(16) to transform from state to measurement space using Eq. (19): $(\hat{\mathbf{z}}^-, \mathbf{\Sigma}^-_{\mathbf{zz}}) \equiv \text{UT}(\hat{\mathbf{x}}^-, \mathbf{\Sigma}^-_{\mathbf{xx}}, \mathbf{g})$;
5 Add the uncertainty of measurements from Eq. (14): $\mathbf{\Sigma}^-_{\mathbf{zz}} \leftarrow \mathbf{\Sigma}^-_{\mathbf{zz}} + \mathbf{\Sigma}_{\mathbf{zz}}$;
6 Compute the cross-correlation matrix between state and measurements: $\mathbf{H}_{\mathbf{xz}} = \sum_{j=-n}^{n} w_c^{[j]} \left(\boldsymbol{\chi}_{\mathbf{x}}^{[j]} - \hat{\mathbf{x}}^-\right) \left(\boldsymbol{\chi}_{\mathbf{z}}^{[j]} - \hat{\mathbf{z}}^-\right)^T$;
7 Compute the Kalman gain: $\mathbf{K} = \mathbf{H}_{\mathbf{xz}'} \mathbf{\Sigma}^{-1}_{\mathbf{zz}'}$;
8 Update the state with incoming measurements from Eq. (13): $\hat{\mathbf{x}} = \hat{\mathbf{x}}^- + \mathbf{K}(\mathbf{z} - \hat{\mathbf{z}}^-), \quad \mathbf{\Sigma}_{\mathbf{xx}} = \mathbf{\Sigma}^-_{\mathbf{xx}}(\mathbf{I} - \mathbf{H}_{\mathbf{xz}}\mathbf{K}^T)$;
9 **end**

defined as $w_m^{[0]} = \lambda/(\lambda + n)$, $w_c^{[0]} = w_m^{[0]} + 1 - \alpha^2 + \beta$ and $w_m^{[j]} = w_c^{[j]} = 1/(2n + 2\lambda)$. Therefore, the DMC-UKF tuning parameters are the process noise covariance $\mathbf{\Sigma}_{\mathbf{ff}}$ and the variables $\{\alpha, \beta, \lambda\}$ controlling the sigma point spread (see Eq. (15)) and UT weights.

Batch least-squares gravity fitting

The DMC-UKF provides estimates of the spacecraft position and the unmodeled acceleration. By stacking these during a certain period of time, a training dataset ($\mathbf{r}_{\text{data}}$,$\mathbf{a}_{\text{data}}$) can be formed as

$$\mathbf{r}_{\text{data}} = \begin{bmatrix} \hat{\mathbf{r}}_0 \\ \vdots \\ \hat{\mathbf{r}}_N \end{bmatrix}, \qquad \mathbf{a}_{\text{data}} = \begin{bmatrix} \mathbf{a}_{\text{grav}}(\hat{\mathbf{r}}_0) + \hat{\mathbf{a}}_0 \\ \vdots \\ \mathbf{a}_{\text{grav}}(\hat{\mathbf{r}}_N) + \hat{\mathbf{a}}_N \end{bmatrix}. \tag{20}$$

Using the previous dataset, a least-squares fitting of the unmodeled acceleration can be posed through the following functional

$$L_{\text{grav}} = \frac{[\mathbf{a}_{\text{fit}}(\mathbf{r}_{\text{data}}) - \mathbf{a}_{\text{data}}]^T [\mathbf{a}_{\text{fit}}(\mathbf{r}_{\text{data}}) - \mathbf{a}_{\text{data}}]}{N}, \tag{21}$$

where $\mathbf{a}_{\text{fit}}$ is the predicted acceleration by the fitted gravity parameters. Both spherical harmonics and mascon models are used. The specific details for each model is described below.

Spherical harmonics: as it can be derived from Eq. (5), the spherical harmonics coefficients are linear with respect to the inhomogeneous gravity. These coefficients can be stacked as $\mathbf{y}_{\text{SH}}$ and the

gravity Jacobian matrix $\mathbf{J}_{\text{SH}}$ with respect to them, evaluated at the dataset, can be derived as

$$\mathbf{y}_{\text{SH}} = \begin{bmatrix} C_{20} \\ \vdots \\ C_{\bar{i}\bar{i}} \\ S_{22} \\ \vdots \\ S_{\bar{i}\bar{i}} \end{bmatrix}, \qquad \mathbf{J}_{\text{SH}} = \begin{bmatrix} \left.\dfrac{\partial \mathbf{a}_{\text{grav}}}{\partial C_{20}}\right|_{\mathbf{r}=\hat{\mathbf{r}}_0} & \cdots & \left.\dfrac{\partial \mathbf{a}_{\text{grav}}}{\partial S_{\bar{i}\bar{i}}}\right|_{\mathbf{r}=\hat{\mathbf{r}}_0} \\ \vdots & \ddots & \vdots \\ \left.\dfrac{\partial \mathbf{a}_{\text{grav}}}{\partial C_{20}}\right|_{\mathbf{r}=\hat{\mathbf{r}}_N} & \cdots & \left.\dfrac{\partial \mathbf{a}_{\text{grav}}}{\partial S_{\bar{i}\bar{i}}}\right|_{\mathbf{r}=\hat{\mathbf{r}}_N} \end{bmatrix}, \tag{22}$$

where the second-order tesseral coefficients S_{21} and C_{21} vanish because it has been assumed that the small body rotation axis is aligned with its principal inertia axis (thus, they are null). By using the terms of Eq. (22), the following linear relation between data and fitting parameters holds $\mathbf{a}_{\text{fit}} = \mathbf{A}_{\text{SH}}(\mathbf{r}_{\text{data}})\mathbf{y}_{\text{SH}}$.

Mascon: the mascon model (see Eq. (6)) has two parameters for each point mass that are its gravity parameter μ_k and its position $\mathbf{r}_k$. The mascon acceleration is linear with respect to the gravity parameter but nonlinear with the position. In this work, the mascon locations are fixed to avoid a costly non-linear optimization. Accordingly, the single decision variable is the standard gravity of each point mass. Similarly to the spherical harmonics case, the point masses standard gravity are stacked in $\mathbf{y}_{\text{M}}$ and the corresponding Jacobian matrix $\mathbf{J}_{\text{M}}$ is formed as

$$\mathbf{y}_{\text{M}} = \begin{bmatrix} \mu_1 \\ \vdots \\ \mu_{n_M} \end{bmatrix}, \qquad \mathbf{J}_{\text{M}} = -\begin{bmatrix} \dfrac{\hat{\mathbf{r}}_0 - \mathbf{r}_1}{\|\hat{\mathbf{r}}_0 - \mathbf{r}_1\|_2^3} & \cdots & \dfrac{\hat{\mathbf{r}}_0 - \mathbf{r}_{n_M}}{\|\hat{\mathbf{r}}_0 - \mathbf{r}_{n_M}\|_2^3} \\ \vdots & \ddots & \vdots \\ \dfrac{\hat{\mathbf{r}}_N - \mathbf{r}_1}{\|\hat{\mathbf{r}}_N - \mathbf{r}_1\|_2^3} & \cdots & \dfrac{\hat{\mathbf{r}}_N - \mathbf{r}_{n_M}}{\|\hat{\mathbf{r}}_N - \mathbf{r}_{n_M}\|_2^3} \end{bmatrix}, \tag{23}$$

thus, the mascons predicted acceleration is $\mathbf{a}_{\text{fit}} = \mathbf{J}_{\text{M}}(\mathbf{r}_{\text{data}})\mathbf{y}_{\text{M}}$. In order to generate a mascon distribution consistent with the previous center of mass, the following linear constraint is considered

$$\sum_{k=1}^{n_M} \mu_k \mathbf{r}_k = \mathbf{A}_{\text{M}}\mathbf{y}_{\text{M}} = \mathbf{0}, \quad \mathbf{A}_{\text{M}} = \begin{bmatrix} \mathbf{r}_1 & \dots & \mathbf{r}_{n_M} \end{bmatrix}. \tag{24}$$

Let recall that the mascon locations are fixed a-priori by assuming a coarse polyhedron shape is available. In particular, the point masses are randomly distributed within the small body volume. In order to distribute the mascons uniformly around the small body, a cuboid embedding the small body is divided into eight octants with an equal number of point masses. For each octant, a random location with equal probability produces a potential mascon. Then, by using the polyhedron shape, this random location is assigned to a mascon or discarded by evaluating the normalized Laplacian. If the point is discarded, another random sample is generated until the specified number of mascons is reached. For a polyhedron, the normalized Laplacian is 0 for an exterior point and 4π for an interior one.[14] An example of a generated 100 mascons distribution is shown in Fig. 2. Another aspect deserving a comment is that the mascon gravity parameters μ_k can take negative values. This is allowed because the objective of this paper is just to generate an accurate representation of the exterior inhomogeneous gravity field.

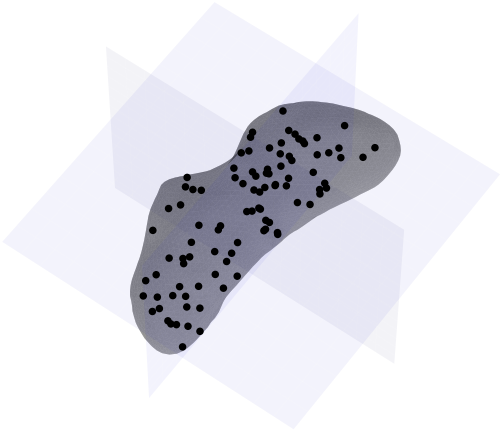

Figure 2. Illustration of a mascon random distribution for $n_M = 100$.

Momentum-based gradient descent: for both the spherical harmonics and mascon models (from now on, their sub-indexes are omitted), the loss function of Eq. (21) can be expanded as

$$L_{\text{grav}} = \frac{\mathbf{y}^T\mathbf{J}^T\mathbf{J}\mathbf{y} + 2\mathbf{a}_{\text{data}}^T\mathbf{J}\mathbf{y} + \mathbf{a}_{\text{data}}^T\mathbf{a}_{\text{data}}}{N}, \tag{25}$$

which is the least-squares fitting of a linear model in parameters. The optimal solution of Eq. (25) is known to be $\mathbf{y}^* = (\mathbf{J}^T\mathbf{J})^{-1}\mathbf{J}^T\mathbf{a}_{\text{data}}$. However, when the dataset is generated online it will be possibly noisy and biased. This may cause the Jacobian matrix to be ill conditioned, specially if models with a high number of parameters are fitted. Moreover, as the number of parameters increase, the matrix inversion may be extremely costly from the computational perspective. For these two reasons, a momentum-based gradient descent method is used for both models.

The momentum-based gradient descent implementation is detailed in Algorithm 2. The acceleration dataset is adimensionalized with a standard scaler by substracting the mean and dividing by the standard deviation of each component (see step 2). Since both the spherical harmonics and mascon models are linear in parameters, the scaled Jacobian is computed outside the training loop (see step 3). Then, the fitting parameters and momentum are initialized in step 4. The fitting parameters are initialized with their current knowledge (e.g. if a data batch has been fitted previously, the training of the new data batch continues from that estimation). The gradient descent executes the loop of steps 5-13 until a maximum number of iterations is reached. In step 6, the scaled acceleration fit with the current parameters is computed. This serves to compute the loss function gradient with respect to the fitting parameters (see step 7). The momentum term is updated in step 8 and it is added to the parameters update, which pushes faster towards the minimum, in step 9. The step 11 only applies to the mascon model and guarantees the satisfaction of Eq. (24). It projects the unconstrained new solution, as per step 9, to the nearest point within the feasible solution space. The tuning parameters of the gradient descent are the learning rate κ and the momentum coefficient $\eta \in [0, 1]$. Note that the decision variables have not been normalized, thus learning rate would differ between spherical harmonics and mascon models.

Basilisk implementation

The simultaneous navigation and gravity estimation algorithms have been embedded within the Basilisk (BSK) astrodynamics simulation framework.[20] The BSK software is composed of Python

Algorithm 2: Momentum-based gradient descent for gravity estimation

begin
  Scale acceleration data: $\bar{\mathbf{a}}_{\text{data}} \leftarrow (\mathbf{a}_{i,\text{data}} - \hat{\mathbf{a}}_{\text{data}})/\boldsymbol{\sigma}_{\text{data}}$;
  Compute scaled Jacobian: $\bar{\mathbf{J}} \leftarrow \mathbf{J}(\mathbf{r}_{\text{data}})/\boldsymbol{\sigma}_{\text{data}}$;
  Initialize the fitting parameters and momentum: $\mathbf{y} = \mathbf{y}_0$, $\Delta\mathbf{y} = \mathbf{0}$;
  for $k \leftarrow 0$ **to** maxiter **do**
    Compute scaled fitting: $\bar{\mathbf{a}}_{\text{fit}} = \bar{\mathbf{J}}\mathbf{y} - (\bar{\mathbf{a}}_{\text{data}}/\boldsymbol{\sigma}_{\text{data}})$;
    Compute gradient $\boldsymbol{\nabla}\bar{L}_{\text{grav}} = 2\bar{\mathbf{J}}^T(\bar{\mathbf{a}}_{\text{fit}} - \bar{\mathbf{a}}_{\text{data}})/N$;
    Compute momentum: $\Delta\mathbf{y} \leftarrow \eta\Delta\mathbf{y} + \kappa\boldsymbol{\nabla}\bar{L}_{\text{grav}}$;
    Do update step: $\mathbf{y} \leftarrow \mathbf{y} - \Delta\mathbf{y}$;
    if model $\equiv$ mascon **then**
      Project to the feasible solution space of Eq. (24): $\mathbf{y} \leftarrow (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{y}$
    end
  end
end

modules written in C/C++. This offers the user to setup the simulation with Python scripts while having the execution speed of C/C++ compiled coding.

The BSK implementation is shown in Fig. 3. It is composed of the simulation and flight software processes that are executed with different sampling rates (dynamics integration step and DMC-UKF rate). The simulation process manages the system ground truth by computing the small body gravity, Sun 3rd body gravity and solar radiation pressure to propagate the spacecraft state. The flight software process comprises the aforementioned algorithms (landmarks-based position determination, DMC-UKF and gravity fitting). The first block, related to the pinhole camera, loads a landmarks database and generates the visible pixels in the image. Then, the landmarks based position determination algorithm computes the spacecraft position and covariance. These estimates are used as input measurements to the DMC-UKF block that fills the spacecraft state and unmodeled acceleration datasets. When these are full, the momentum-based gradient descent finds the gravity parameters fitting, in the least-squares sense, the data. Then, the dataset is emptied and the filter gravity parameters are updated with the fitted ones (thus, reducing system uncertainty). This process is depicted in Fig. 4.

NUMERICAL RESULTS

This section aims to validate the simultaneous navigation and gravity estimation algorithms with numerical simulations. The target small body is 433 Eros which has a standard gravity parameter $\mu = 4.4627547 \cdot 10^5$ m^3/s^2 and a rotational period $T_A = 5.27$ h. Its ground truth gravity field is modelled using a Eros polyhedron shape with 7790 triangular faces[1]. Eros heliocentric orbital parameters are $\{a = 1.4583 \text{ AU}, e = 0.2227, i = 10.829^\circ, \Omega = 304.4^\circ, \omega = 178.9^\circ, \nu_0 = 246.9^\circ\}$ and its orientation is defined by $\{\text{RA} = 11.369^\circ, \text{dec} = 17.227^\circ, \text{LST}_0 = 0^\circ\}$.

The spacecraft is placed in a stable orbit for the simulation duration which is of 10 initial orbital periods (6.825 days in total). The initial orbital elements are $\{a_0 = 34 \text{ km}, e_0 = 0.001, i_0 = 45^\circ, \Omega_0 = 48.2^\circ, \omega_0 = 347.8^\circ, \nu_0 = 85.3^\circ\}$. The spacecraft mass is taken as $m = 750$ kg, its

[1] https://sbnarchive.psi.edu/pds3/near/NEAR_A_5_COLLECTED_MODELS_V1_0/data/msi/

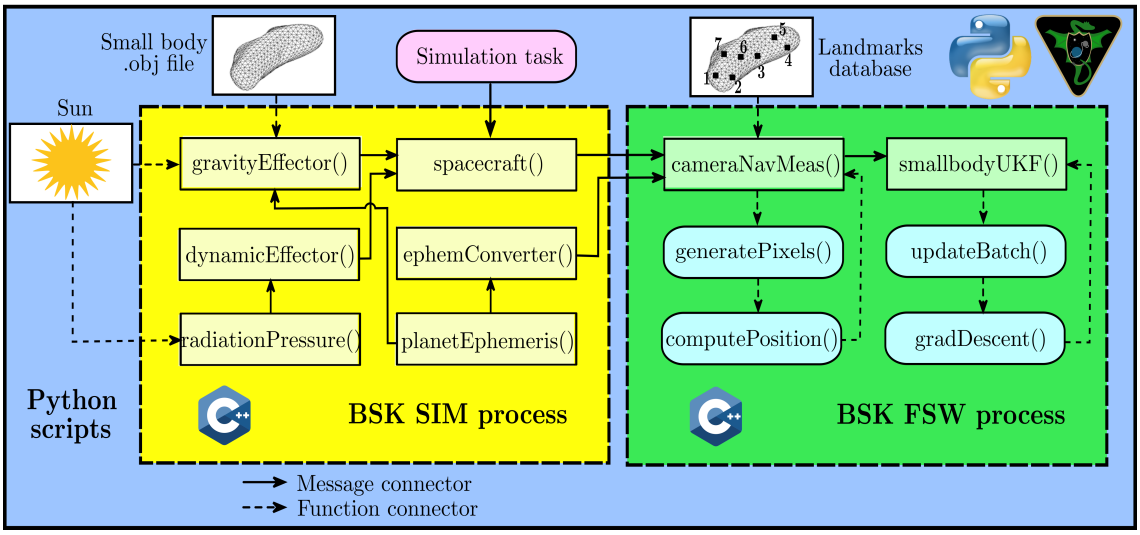

Figure 3. Diagram of simultaneous navigation and gravity estimation in the Basilisk simulation framework.

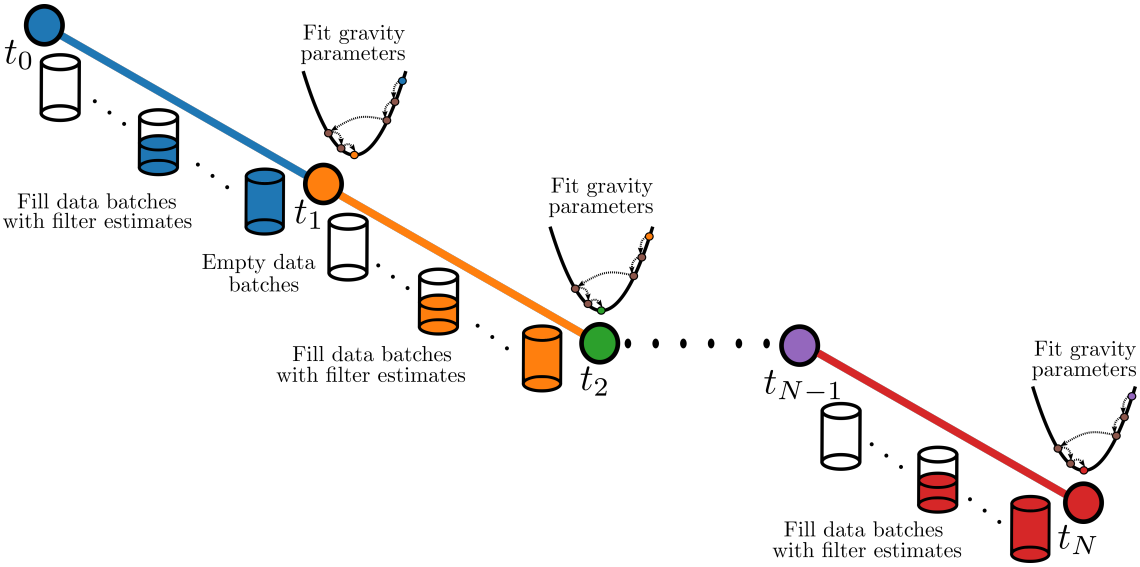

Figure 4. Illustration of the gravity estimation process.

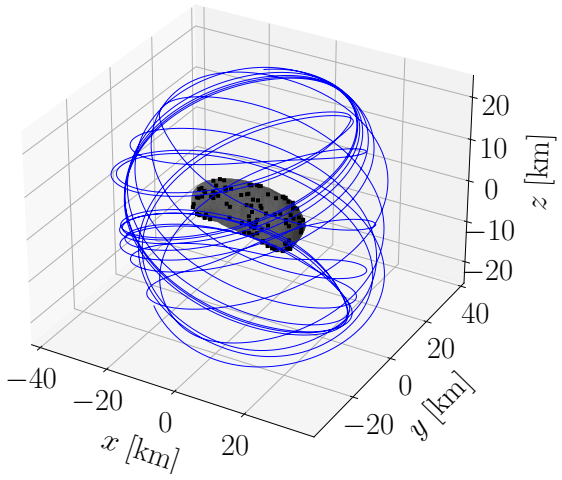

Figure 5. Spacecraft trajectory for the simulations. Black squares ≡ landmarks.

reflection coefficient as $C_R = 1.2$ and the solar radiation pressure exposed area as $S = 1.1$ m^2. The simulated navigation camera is characterized by a 4:3 aspect ratio with a 17.3×13 mm sensor size and a resolution of 2048×1536 px. Then, the pixel width is $w_p = 8.447$ μm which is on the order of magnitude of cameras used in small bodies navigation. During the entire simulation, the camera is assumed to be always pointing towards Eros center of mass and generates measurements with a sampling rate of 1 min. The available landmarks dataset is composed of 100 random points (centers of the polyhedron facets) of Eros surface. The trajectory, expressed in the rotating small body centred frame, and landmarks are shown in Fig. 5.

For the subsequent results, the DMC-UKF parameters are $\{\alpha = 0, \beta = 2, \lambda = 10^{-3}\}$. The filter is initialized with the spacecraft state truth values and null unmodeled acceleration. The initial DMC-UKF state covariance matrix and the process noise are

$$\begin{aligned} \mathbf{\Sigma_{xx}}(t_0) &= \begin{bmatrix} (10)^2\mathbf{I}\ (\text{m})^2 & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & (10)^2\mathbf{I}\ (\text{mm/s})^2 & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & (1)^2\mathbf{I}\ (\mu\text{m/s}^2)^2 \end{bmatrix}, \\ \mathbf{\Sigma_{ff}} &= \begin{bmatrix} (0.1)^2\mathbf{I}\ (\text{m})^2 & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & (1)^2\mathbf{I}\ (\text{mm/s})^2 & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & (2)^2\mathbf{I}\ (\mu\text{m/s}^2)^2 \end{bmatrix}. \end{aligned} \tag{26}$$

The gradient-descent maximum number of iterations is 10000, its momemtum coefficient is tuned as $\eta = 0.9$ and the learning rate is $\kappa_{\text{SH}} = 10^5$ for spherical harmonics and $\kappa_M = 10^{-6}$ for the mascon model (let recall that the decision variables have not been adimensionalized).

Landmarks-based position determination

This paragraph aims to assess the landmarks-based position determination algorithm and choose an optimal camera focal length. Since the goal is just to test the algorithm, no lighting constraints are introduced and all landmarks that fall within the camera FOV are visible. Firstly, let vary the diagonal field of view evenly between 94° and 23° which correspond to focal lengths of $f \equiv \{10, 12, 14, 17, 25, 40, 52\}$ mm. The distribution of the position determination errors, by only taking

into account pixelation error, is shown in Fig. 6 (*left*). It can be deduced that the optimal focal length configuration lies between 17 and 40 mm (63°-30° of diagonal FOV). There is a tradeoff between pixel accuracy (which augments as the FOV is narrower) and a nonambiguous distribution of visible landmarks (which is favored by a wider FOV). Since $f = 25$ mm seems the optimal value, the impact of uncertainty in the landmarks locations is analyzed for this focal length. To this end, the landmarks locations are perturbed with 1-σ Gaussian errors of $\{0, 5, 15, 30, 50\}$ m. The distribution of the position determination errors is shown in Fig. 6 (*right*). As expected, the errors augment accordingly to the landmarks uncertainty. In particular, the average position errors are $\{2.01, 4.83, 13, 26, 44\}$ m respectively.

In the online gravity estimation paragraph (end of this section), the optimal focal length of $f = 25$ mm is used. In that case, the small body rear and top are seen as in Fig. 7 from a distance $r = 34$ km (in the order of magnitude of the simulation).

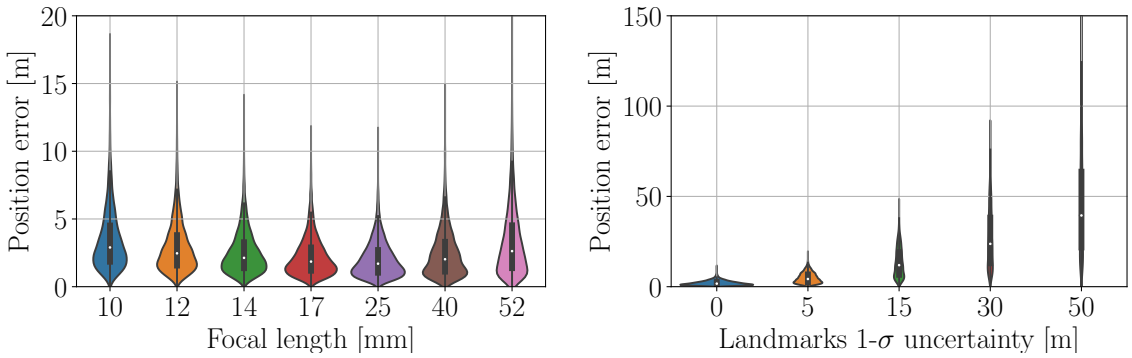

Figure 6. Distribution of position errors with respect to focal length (*left*) and landmarks uncertainty (*right*).

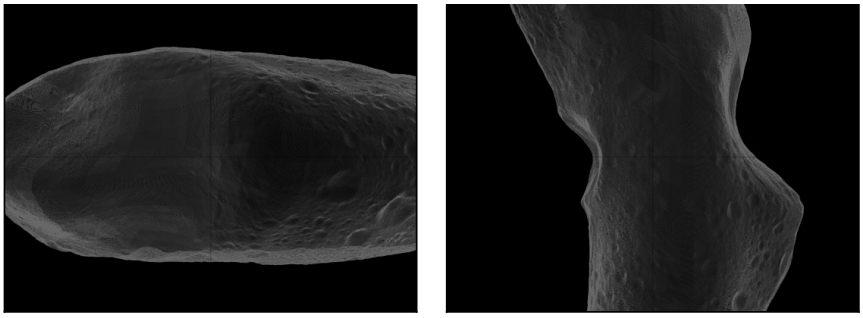

Figure 7. Rear (*left*) and top (*right*) small body images from $r = 34$ km and $f = 25$ mm.

Offline fitting of gravity models

Let now assess both the gradient descent algorithm and the accuracy of the gravity determination under a true position-acceleration dataset (values along the trajectory). The data sampling rate is the same as the camera measurements (1 min). The data batch is composed of 982 points that correspond to a duration of one initial orbital period. Then, 10 sequential fitting processess are carried out. No initial knowledge is assumed on the gravity parameters, thus they are initialized with

null values. The spherical harmonics maximum degree is ranged between 2 and 8 (3 to 75 variables). The mascon models number of point masses is varied through $n_M = \{40, 100, 200, 400\}$. Since the mascons locations are generated randomly, 500 different samples are tested for each model in order to provide accuracy bounds. The gravity acceleration error δa is quantified as follows

$$\delta a(\mathbf{r}) = \frac{\|\mathbf{a}_{\text{grav}}(\mathbf{r}) - \mathbf{a}^{\text{poly}}_{\text{grav}}(\mathbf{r})\|_2}{\|-\mu\mathbf{r}/r^3 + \mathbf{a}^{\text{poly}}_{\text{grav}}(\mathbf{r})\|_2}, \tag{27}$$

where $\mathbf{a}_{\text{grav}}$ is the inhomogeneous acceleration for the fitted model and $\mathbf{a}^{\text{poly}}_{\text{grav}}$ is the one from the ground truth polyhedron. The Keplerian term (though being assumed as known) is added in the denominator for the sake of relevance (it does not make sense to evaluate the error with respect to the inhomogeneous component if its value is too small). The average global gravity errors, per orbital range, of the fitted models are shown in Fig. 8. This is computed by evaluating the gravity acceleration error on a 3D grid (64000 exterior points) around the small body. The error with a simple Keplerian model (which is the initial guess) is also shown. Several trends can be derived. By increasing the spherical harmonics degree, the accuracy improves outside of the Brillouin sphere (with a radius of $r = 17.68$ km) but highly degrades within its interior (being more inaccurate than the basic Keplerian model). Moreover, the accuracy outside Brillouin sphere improves little once 4th degree is reached. On the other hand, the mascons models (being the dots the mean of the 500 random realizations and the filled areas their variability), improves the basic Keplerian model for all orbital ranges. The results variability is narrowed down as the number of point masses is increased because since the distributions shall resemble more to each other.

Let compare the fitted models by taking the best spherical harmonics errors for each orbital range ($\{\text{Surf.} - 17.68, 17.68 - 30, 30 - 40, 40 - 50\}$ km), which are $\{50.60, 1.446, 0.268, 0.167\}$ %, and the worst realization of the $n_M = 400$ mascon model that is $\{11.08, 0.573, 0.227, 0.165\}$ %. It can be easily derived that the mascon models have the superiority in the low altitude range around the Brillouin sphere while not loosing accuracy medium to high altitudes (if there is an enough number of point masses). This argument is further highlighted in Fig. 9 where 2D gravity error maps are plotted on the xy, xz and yz planes for the spherical harmonics 4th degree model and a mascon random realization with $n_M = 400$. The spacecraft trajectory is also shown to provide visual insight on the dataset spatial variations.

Simultaneous navigation and gravity estimation

This last paragraph is devoted to test the whole simultaneous navigation and gravity estimation process. The measurements are provided with the camera focal length of $f = 25$ mm. The landmarks locations are known uncertainly with a 1-σ Gaussian error of 5 m. In these online simulations, the Sun's lighting constraint is added for the sake of fidelity. Consequently, there are moments where the spacecraft is in the small body dark side and there are no visible landmarks. In practice, this creates measurements gaps that reduce the dataset by a 22% for the simulated orbit.

For a mascon random distribution of $n_M = 400$, the filter position errors and inhomogeneous gravity knowledge are shown in Fig. 10-11 respectively. In Fig. 10, the estimated position error increase sharply when landmarks start vanishing or are being reacquired. Nonetheless, when the interval with measurements is long enough, the error is usually kept below 5 m during intermediate periods. Figure 11 shows the filter is able to track the unmodeled acceleration (first orbital period) and that the subsequent gravity training is efficient. It can also be observed that the transition lighting periods also produces sharp errors on $\mathbf{a}$. However, the filter is always able to track

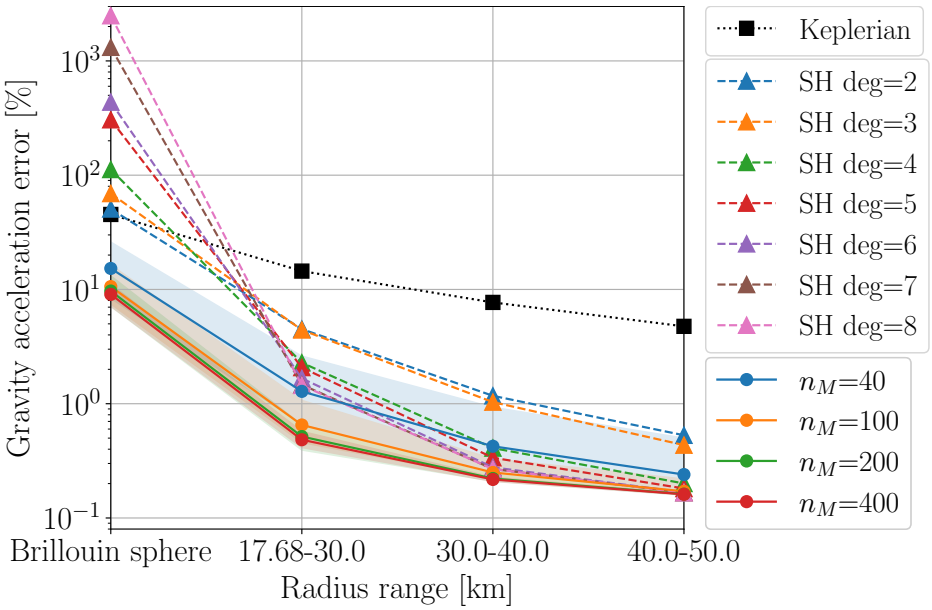

Figure 8. Average gravity error per orbital range with offline estimation.

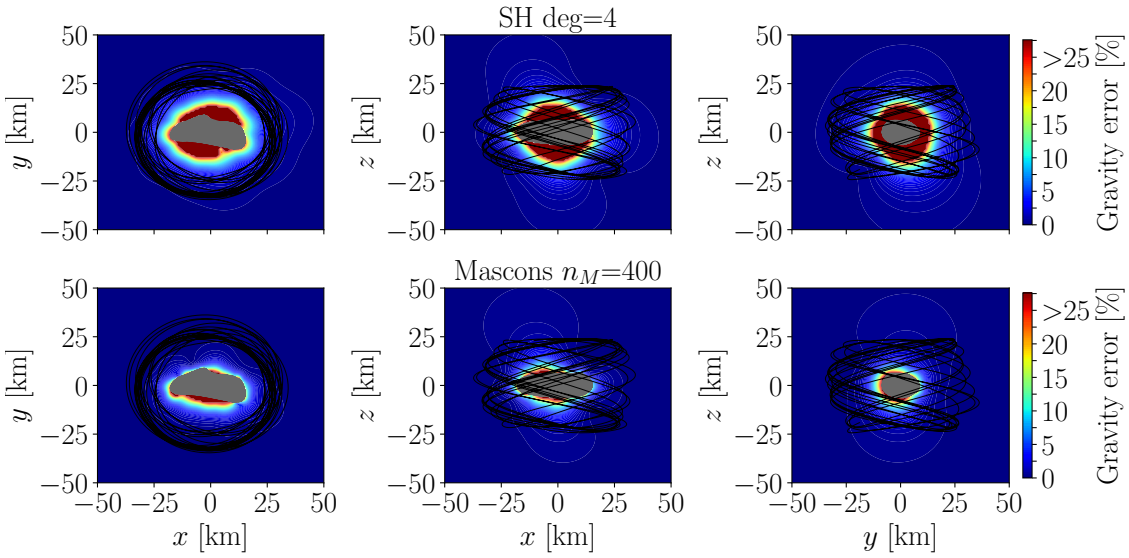

Figure 9. Gravity error 2D maps with offline estimation. Black: projected trajectory.

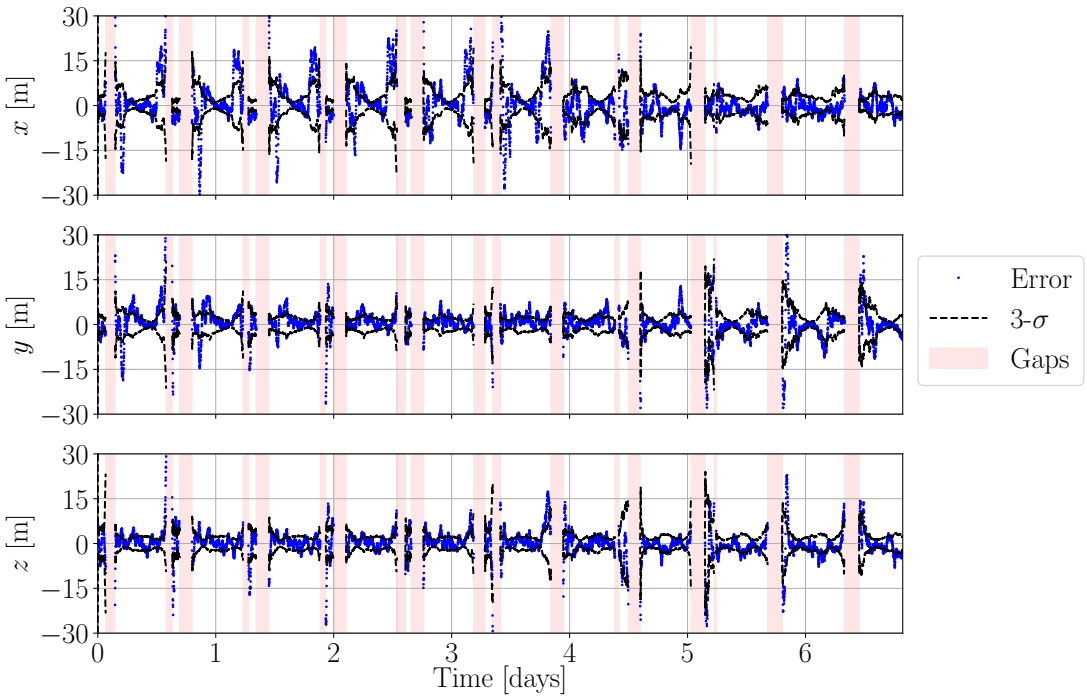

Figure 10. Filter position estimation error and uncertainty for a mascon random distribution ($n_M = 400$)**.**

the remaining unmodeled acceleration after the measurements are available again. Note that Fig. 11 shows $\mathbf{a}_{\text{grav}} + \mathbf{a}$, thus Fig. 10-11 is the dataset used for the gravity fitting. The final mascon distribution can be seen in Fig. 12. The heavier masses tend to be placed at the lateral lobes while the negative ones accumulate in the central region. It is not possible to deduce any explanation from that other than it appears symmetrical to keep the center of mass coincident with the origin.

The same simulations that has been carried out for the offline estimation are reproduced with the online configuration. The gravity error per orbital range is shown in Fig. 13 which yields similar conclusions to the offline estimation (superiority of the mascon model over the spherical harmonics one). Nonetheless, for this online case, augmenting the number of parameters does not necessarily seem to increase the accuracy. It can be observed that the most accurate mascon model corresponds to $n_M = 200$ and the spherical harmonics one, ignoring the Brillouin sphere, seems to be the 4th degree. For some of the models, the computational performance is reported in Table 1. The execution times are measured on a M1 Max processor running in single core. The mascon model is more costly to execute due to its higher number of parameters compared to spherical harmonics. Although being run in a high performance processor, these execution speeds (in the order of milliseconds for the DMC-UKF) may highlight the potential applicability of the simultaneous navigation and gravity estimation for on-board autonomy.

CONCLUSIONS

This paper has developed a simultaneous navigation and gravity estimation scheme around a small body. The main components are a landmarks-based position determination algorithm, a dynamical model compensated unscented Kalman filter and a momentum-based gradient descent to estimate the gravity parameters. The previous approach mainly relies on generating a dataset, with

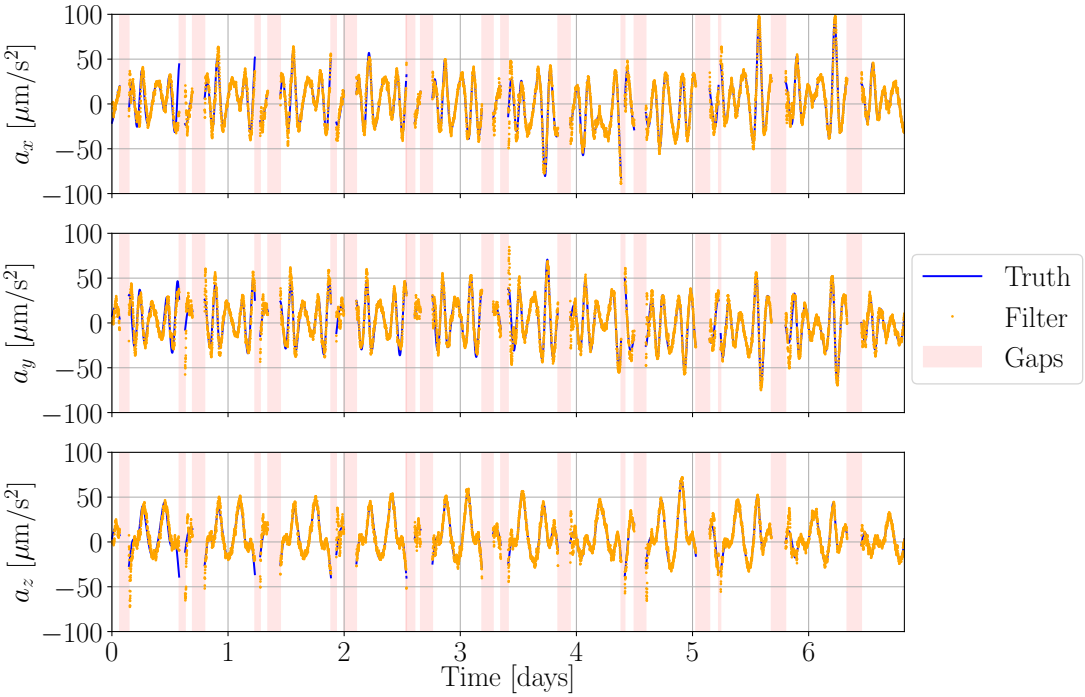

Figure 11. Filter inhomogeneous gravity acceleration estimation for a mascon random distribution ($n_M = 400$)**.**

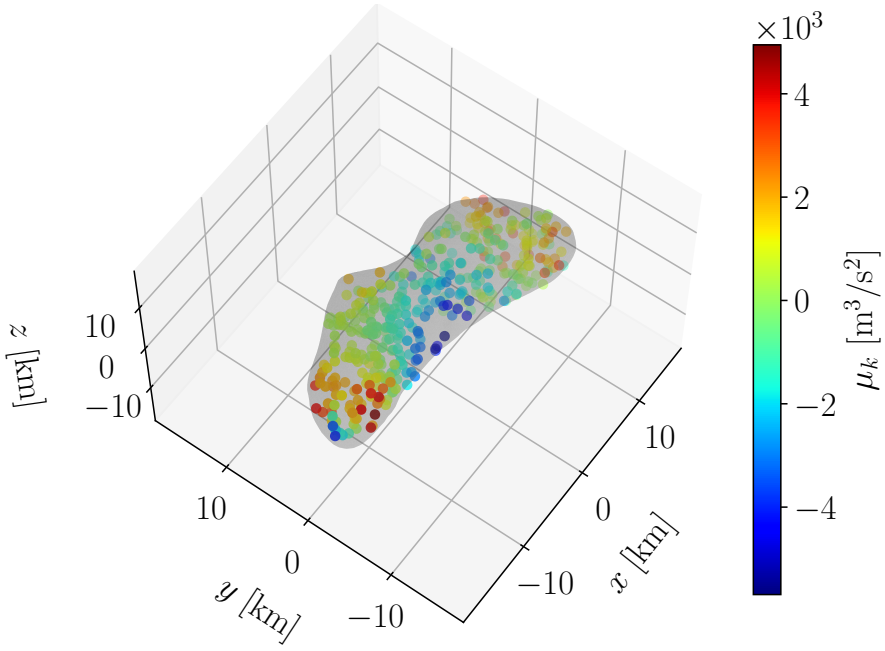

Figure 12. Fitted mascon random distribution ($n_M = 400$) **with simultaneous navigation and gravity estimation .**

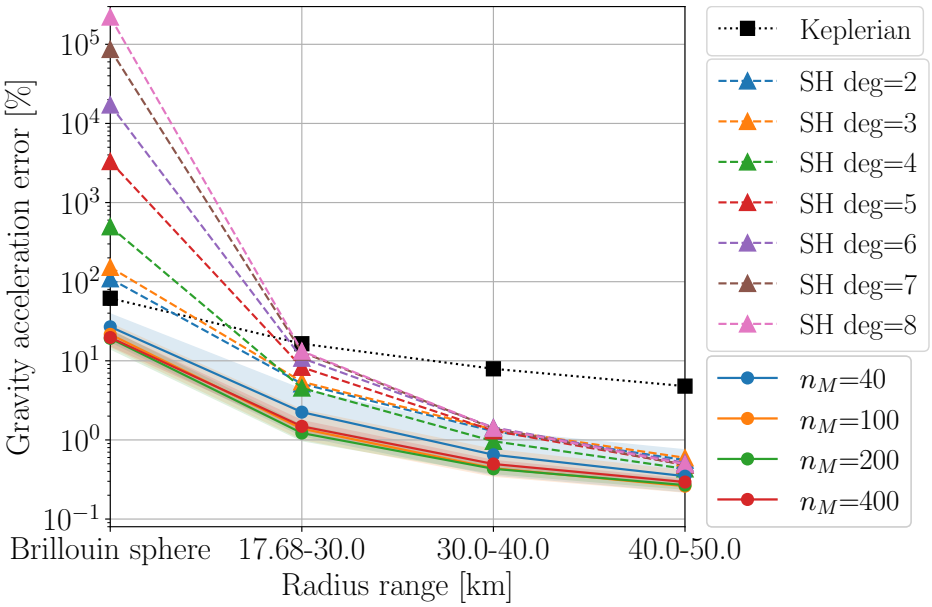

Figure 13. Average gravity error per orbital range with simultaneous navigation and gravity estimation.

	DMC-UKF step		Gravity fitting	
Model	Mean [ms]	Max. [ms]	Mean [s]	Max. [s]
Spher. harm. deg $= 2$	0.077	0.111	0.140	0.152
Spher. harm. deg $= 4$	0.096	0.174	0.359	0.388
Spher. harm. deg $= 6$	0.106	0.199	0.727	0.793
Spher. harm. deg $= 8$	0.133	0.217	1.265	1.383
Mascons $n_M = 100$	0.283	0.391	1.016	1.095
Mascons $n_M = 400$	1.035	1.116	4.084	4.430

Table 1. Computational times of simultaneous navigation and gravity estimation with each model.

the navigation filter, which serves to fit a gravity model. Then, the algorithm is flexible to models other than spherical harmonics. In particular, a mascon model has been implemented. The numerical results demonstrate the mascon model is able to overcome the spherical harmonics divergence within the Brillouin sphere while not loosing accuracy outside of it. This has been validated for both a true dataset and a noisy one, generated online, with data gaps due to lighting constraints. In conclusion, the use of the mascon model, within the developed simultaneous navigation and gravity estimation, could be of interest to plan subsequent autonomous descent and landing operations.

Future work may focus on improving certain characteristics of the mascon model. Specifically, it is envisioned that the point masses locations will be included as decision variables. This will make the mascon model to learn its optimal spatial distribution. Another line of work emerges from the consideration of an interconnected multi-satellite configuration that can provide ranging measurements. This may avoid the actual measurement gaps while producing a higher amount of relevant data which could positively impact the gravity estimation accuracy.

ACKNOWLEDGMENT

This work is part of a project that has received funding from the European Union's H2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 101025257.

REFERENCES

[1] J. C. Castillo-Rogez, M. Pavone, J. A. Hoffman, and I. A. D. Nesnas, "Expected science return of spatially-extended in-situ exploration at small Solar system bodies," *IEEE Aerospace Conference*, March 2012, pp. 1–15, 10.1109/AERO.2012.6187034.

[2] A. Cheng, A. Rivkin, P. Michel, J. Atchison, O. Barnouin, L. Benner, N. Chabot, C. Ernst, E. Fahnestock, M. Kueppers, P. Pravec, E. Rainey, D. Richardson, A. Stickle, and C. Thomas, "AIDA DART asteroid deflection test: Planetary defense and science objectives," *Planetary and Space Science*, Vol. 157, 2018, pp. 104–115, 10.1016/j.pss.2018.02.015.

[3] D. J. Scheeres, "Orbit Mechanics About Asteroids and Comets," *Journal of Guidance, Control, and Dynamics*, Vol. 35, No. 3, 2012, pp. 987–997, 10.2514/1.57247.

[4] J. K. Miller, A. S. Konopliv, P. Antreasian, J. J. Bordi, S. Chesley, C. E. Helfrich, W. M. Owen, T. C. Wang, B. G. Williams, D. K. Yeomans, and D. J. Scheeres, "Determination of Shape, Gravity, and Rotational State of Asteroid 433 Eros," *Icarus*, Vol. 155, January 2002, pp. 3–17, 10.1006/icar.2001.6753.

[5] N. Stacey and S. D'Amico, "Autonomous Swarming for Simultaneous Navigation and Asteroid Characterization," *AAS/AIAA Astrodynamics Specialist Conference*, Snowbird, United States of America, August 2018.

[6] N. Stacey, K. Dennison, and S. D'Amico, "Autonomous Asteroid Characterization through Nanosatellite Swarming," *IEEE Aerospace Conference*, Big Sky, United States of America, March 2022.

[7] J. D. Biggs and E. Ciccarelli, "In-situ tracking of a solar sail's characteristic acceleration using a robust active disturbance estimator," *5th International Symposium on Solar Sailing*, Aachen, Germany, 2019.

[8] J. C. Sanchez, R. Vazquez, J. D. Biggs, and F. Bernelli-Zazzera, "Orbit-attitude Predictive Control in the Vicinity of Asteroids with In Situ Gravity Estimation," *Journal of Guidance, Control, and Dynamics*, Vol. 45, No. 3, 2022, pp. 262–279, 10.2514/1.G005572.

[9] A. Pasquale, S. Silvestrini, A. Capannolo, P. Lunghi, and M. Lavagna, "Small bodies non-uniform gravity field on-board learning through Hopfield Neural Networks," *Planetary and Space Science*, Vol. 212, No. 105425, 2022, pp. 1–14, 10.1016/j.pss.2022.105425.

[10] J. R. Martin and H. Schaub, "Preliminary Analysis of Small-Body Gravity Field Estimation using Physics-Informed Neural Networks and Kalman Filters," *International Astronautical Congress*, Paris, France, September 2022.

[11] J. C. Sanchez and H. Schaub, "Semi-autonomous navigation and gravity estimation around small bodies," *2nd International Stardust Conference*, Noordwijk, The Netherlands, November 2022.

[12] R. P. Russell and P. T. Wittick, "Mascon Models for Small Body Gravity Fields," *AAS/AIAA Astrodynamics Specialist Conference*, Stevenson, United States of America, 2017.

[13] J. R. Martin and H. Schaub, "Physics-informed neural networks for gravity field modeling of small bodies," *Celestial Mechanics and Dynamical Astronomy*, Vol. 134, No. 46, 2022, 10.1007/s10569-022-10101-8.

[14] R. A. Werner and D. J. Scheeres, "Exterior gravitation of a polyhedron derived and compared with harmonic and mascon gravitation representations of asteroid 4769 Castalia," *Celestial Mechanics and Dynamical Astronomy*, Vol. 65, 1996, pp. 313–344, 10.1007/BF00053511.

[15] G. Balmino, "Gravitational potential harmonics from the shape of an homogeneous body," *Celestial Mechanics and Dynamical Astronomy*, Vol. 60, 1994, pp. 331–364, 10.1007/BF00691901.

[16] S. Pines, "Uniform Representation of the Gravitational Potential and its Derivatives," *AIAA Journal*, Vol. 11, No. 11, 1973, pp. 1508–1511, 10.2514/3.50619.

[17] M. Vetrisano and M. Vasile, "Autonomous navigation of a spacecraft formation in the proximity of an asteroid," *Advances in Space Research*, Vol. 57, No. 8, 2016, pp. 1783–1804, 10.1016/j.asr.2015.07.024.

[18] J. M. Leonard, F. G. Nievinski, and G. H. Born, "Gravity Error Compensation Using Second-Order Gauss-Markov Processes," *Journal of Spacecraft and Rockets*, Vol. 50, No. 1, 2013, pp. 217–229, 10.2514/1.A32262.

[19] E. Wan and R. Merwe, "The Unscented Kalman Filter for Nonlinear Estimation," *Adaptive Systems for Signal Processing, Communications, and Control Symposium*, Alberta, Canada, 2000.

[20] P. W. Kenneally, S. Piggott, and H. Schaub, "Basilisk: A Flexible, Scalar and Modular Astrodynamics Simulation Framework," *Journal of Aerospace Information Systems*, Vol. 17, No. 9, 2020, pp. 496–507, 10.2514/1.I010762.