

Improved Shape Determination for Autonomous State Estimation

By Benjamin BERCOVICI,¹⁾ Jay MCMAHON,¹⁾

¹⁾*Department of Aerospace Engineering Sciences, University of Colorado at Boulder*

(Received June 21st, 2017)

This paper provides insight into the performance of a new filtering approach for estimation of the shape of an unknown target by means of a single LIDAR instrument mounted on a known observing spacecraft. The algorithm presented here proceeds by sequentially processing point-cloud observations so as to solve for the deviation in the shape model's vertices coordinates. The shape is refined by adding degrees of freedom where measurement residuals are found to be bad. The performance of the algorithm is demonstrated on the shape model of KW4 Alpha and Itokawa, using a very coarse a-priori. The filter was able to reconstruct the shape with a satisfying degree of accuracy. Future work will focus on augmenting the estimated state with the relative position and attitude of the targeted shape, in addition to relying on more realistic dynamics and observation model.

Key Words: shape determination, filtering, Lidar, observability

Nomenclature

$a^T b$: Dot product of the \mathbb{R}^N vectors a and b
$[\tilde{x}]$: Matrix of the operator $v \mapsto x \times v = [\tilde{x}]$
\hat{n}_j	: j -th unit outward-pointing normal vector
P_k	: Coordinate of the k -th Lidar pixel
\hat{u}_k	: Unit vector directing the k -th lidar ray
$V_{i,j}$: i -th vertex coordinates of the j -th facet
ρ_k	: k -th range collected from the true shape
$\bar{\rho}_k$: k -th range collected from the a-priori shape
$\hat{\rho}_k$: k -th range collected from the corrected shape
I_3	: Identity matrix of $\mathbb{R}^{3,3}$

Subscripts

k	: Measurement index ($k \in \llbracket 0, N_m - 1 \rrbracket$)
j	: Facet index ($j \in \llbracket 0, N_f - 1 \rrbracket$)
i	: Local vertex index ($i \in \{0, 1, 2\}$)
o	: Observation time index ($o \in \llbracket 0, N_p - 1 \rrbracket$)

1. Introduction

The development of modern batch and sequential orbit determination methods kicked off during the Apollo program.¹⁾ The first implementation of an attitude filter followed shortly after, once the modeling of rotational dynamics improved.²⁾ Besides the development of more advanced filtering architectures, some recent improvements in attitude determination stemmed from the development of new attitude sets such as Modified Rodrigues Parameters. These alternative parametrizations provide an elegant solution to attitude singularities, along with a constraint-free structure.³⁾ It must be noted that these early efforts were concerned with the estimation of the position and attitude state of a known spacecraft. In particular, various observation types related to angles and angle-rate of change could be provided by star trackers or speed gyros. Those strategies are thus not as easily applicable when in-situ measurements of these quantities are not available.

The last 15 years have showed a new emphasis put on spacecraft autonomy. The success of advanced robotic missions such as satellite servicing or proximity operations about small bodies like asteroids or comets was found to be heavily dependent on

the capacity of spacecraft to operate autonomously as they carry out their mission.⁴⁾ Achieving science or engineering goals without reducing the mission envelope thus requires spacecraft to perform data processing and decision making without external input. This is a textbook example of where advanced state and parameter estimation techniques are needed. For instance, orbital debris mitigation can only be addressed by means of autonomous robotic servicing spacecraft if one was able to remotely determine the state, inertia, or any other relevant parameter of a non-cooperative target for which little if no a priori information is available.⁵⁾ Pioneering rendez-vous, remote inspection and stand-off of a servicer and a non-cooperative target was demonstrated in 2012 in the frame of the PRISMA experiment,⁶⁾ using angle-only measurements provided by an optical-camera and ground-in-the-loop processing. The final relative separation between the two spacecraft was close to 3 kilometers in average, which is too far to allow resolved observations of the target.

Decreasing the relative distance between the target and the observing spacecraft is thus necessary to obtain a more favorable observation geometry, enabling one to get insight into the target's rotational dynamics, inertia parameters or even shape.

The inertia estimation case is actually multifold. It covers at least two scenarios, whether external torques are present or absent. Beginning with cases where external torques are present, one can identify two subcases: either the torques are a consequence of the target's dynamics (maneuver, SRP,...) or the action of an external agent (such as an inspecting spacecraft) applying a known torque to the target to as to make its inertia tensor observable⁷⁾⁸⁾⁹⁾ Small bodies like asteroids or comets have also been a target of interest, because of the coupling of their rotational motion with their orbit about the Sun. The example of 4179 Toutatis demonstrated how the inertia tensor could be estimated by means of Doppler data collected over years of observation passes, while accounting for external torques.¹⁰⁾ Moving on to the case where external torques are absent, or neglected, it becomes clear from Euler's equations that the full inertia tensor becomes non-observable. However, it is still possible to extract insightful results from the one-dimensional space of estimated inertia tensors, such as the direction of the principal axes. Ele-

gant batch-like approaches such as the one developed in¹¹⁾ can be utilized to estimate the inertia tensor along with the position of the target's center of mass in a purely torque-free manner, but only if angular velocity measurements are available in addition to angles.

The next step naturally consists in complementing the inertia tensor of the observed target with a proper description of its shape. This process pertains to the reconstruction of a parametrized shape model representative of the target. The parametrization can either be implicit or explicit. One of the state-of-the-art of the implicit techniques is known as the Poisson Surface Reconstruction (PSR). It aims at reconstructing the indicator function representative of the inside of a dense, fully-registered point cloud.¹²⁾ The quality of the reconstructed point cloud is excellent, as the variational methods PSR relies on are ensured to return a continuous isosurface from the source point cloud. Such point clouds can be comprised of hundred of millions of data points, often time collected over *observation campaigns*, making the global shape reconstruction a very computationally-intensive process. Yet, they remain the method of choice when the quality of the reconstructed shape model is paramount, and when storage capabilities or computational power are not a concern.

An interesting contribution relying on a more sustainable Bayesian framework was provided by Lichter and al., in which they performed shape, inertia and attitude parameter estimation. The shape was parametrized implicitly using voxels.¹³⁾ However, it must be noted that the observation model retained in this study was fairly optimistic, as it was assuming that a fully-registered point cloud was readily available from a formation of spacecraft carrying LIDAR instruments.

A recent unifying approach relying on factor graphs was proposed to carry out the determination of position, attitude and shape simultaneously.¹⁴⁾ This Bayesian framework inherited from ground-based SLAM techniques was tested on the SPHERES testbed onboard the ISS to validate this approach. If this method appears to perform really well from an estimate quality standpoint, it is recognized as too slow to operate real-time due to the increasing complexity of the graph model as more observations are accumulated. Optimal landmark tracking was also shown to provide a satisfying estimate of the relative state between Rosetta and 67P/Churyumov-Gerasimenko, but this approach was relying on ground-based processing and manually positioned landmarks.¹⁵⁾

It thus appears that a sequential framework able to autonomously perform position, attitude, inertia and shape determination real-time does not exist yet. If these problems have been solved independently, attempting to tackle them simultaneously has only been attempted by few. This paper thus aims at exploring the theory and implementation of a framework that we believe could fulfill this role. This framework leverages recent filtering results applied to LIDAR measurements collected by a unique instrument, to produce estimate of the aforementioned quantities on-line. Because it is one of the most daunting issues identified in our review of the literature, this paper will focus on the autonomous shape determination exclusively.

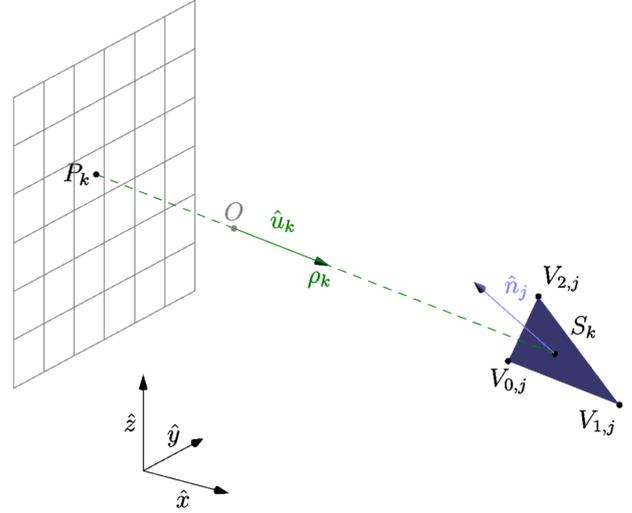


Fig. 1. Illustration of the u_k ray intersecting with the shape at S_k and the resulting range measurement ρ_k .

2. Method

2.1. Shape model parametrization

As pointed out in the introduction, a number of ways exist to represent a shape. An explicit shape parametrization by means of vertices and triangular facets was retained in this study. This was motivated by the relative ease with which one can formulate a mean-square estimation framework operating on the shape model parametrized like so. Moreover, this shape representation can easily be related to gravity field computation through a Polyhedron Gravity Model evaluation.¹⁶⁾

2.2. Shape model batch update

We are assuming that our shape of interest is being observed by a Lidar instrument collecting range measurements, following the same notations as in Figure 1. For the purpose of shape determination, a natural performance index one can seek to minimize is the \mathcal{L}_2 norm of the range residuals, namely

$$J^2 = \frac{1}{N_m} \sum_{k=0}^{N_m-1} (\rho_k - \hat{\rho}_k)^2 \quad (1)$$

Writing a first-order expansion of the estimated range measurement about some a-priori $\bar{\rho}_k$,

$$J^2 \simeq \frac{1}{N_m} \sum_{k=0}^{N_m-1} \left(\rho_k - \bar{\rho}_k - \frac{\partial \bar{\rho}_k}{\partial \mathbf{V}} \delta \mathbf{V} \right)^2 \quad (2)$$

Introducing the prefit residuals $\delta \rho_k = \rho_k - \bar{\rho}_k$ and the notation $\tilde{H}_k = \frac{\partial \bar{\rho}_k}{\partial \mathbf{V}}$, the performance index gets rewritten into

$$J^2 = \frac{1}{N_m} \sum_{k=0}^{N_m-1} (\delta \rho_k - \tilde{H}_k \delta \mathbf{V})^2 \quad (3)$$

Setting the first partial derivative of J^2 relative to $\delta \mathbf{V}$ to zero, we obtain the so-called *normal form* of the filter:

$$(H^T H) \delta \mathbf{V} = H^T \delta \boldsymbol{\rho} \quad (4)$$

with

$$H = \begin{bmatrix} \tilde{H}_0 \\ \tilde{H}_1 \\ \dots \\ \tilde{H}_{N_m-1} \end{bmatrix} \quad (5)$$

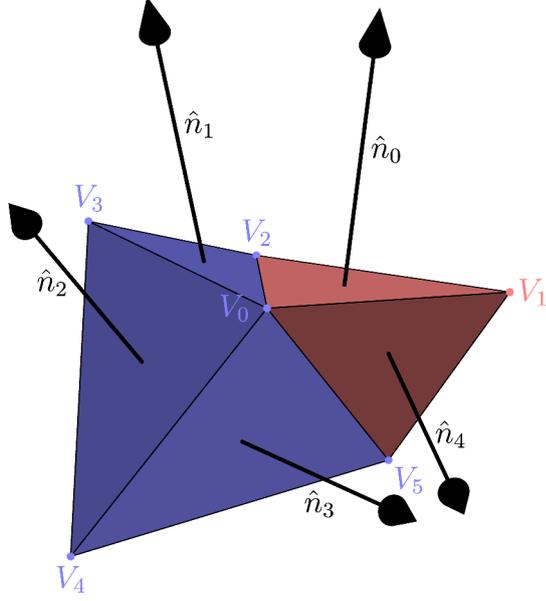


Fig. 2. Illustration of a patch of visible facets (light blue) bounded by two unobserved facets (light red). Observable and unobservable vertices are highlighted using the same color code.

$$\delta\rho = \begin{bmatrix} \delta\rho_0 \\ \delta\rho_1 \\ \dots \\ \delta\rho_{N_m-1} \end{bmatrix} \quad (6)$$

It must be noted that Equation (4) cannot actually be solved, as the matrix $H^T H$ is singular. This singularity stems from two reasons:

1. $\delta\mathbf{V}$ contains coordinates of vertices that belong to facets that were never seen in any of the N_m Lidar measurements. Some components of $\delta\mathbf{V}$ are thus completely unobservable.
2. Even though $\delta\mathbf{V}$ is reduced to deviations in the coordinates of observable vertices (e.g. belonging to observed facets), it still encompasses observable deviations directions (e.g. normal to each facet) and unobservable directions (e.g. tangential to each facet). The latter would cause a change in the facet's vertices coordinates that yield no variation in the measured range.

The first point can be taken care of by simply keeping track of which facets were seen by the Lidar, and by only including coordinates of the vertices belonging to those observable facets in $\delta\mathbf{V}$. Addressing the second point requires rewriting the solved-for deviation of the l -th seen vertex in the a-priori shape δV_l as a linear combination of the normal vectors of the facets the l -th observable vertex belongs to, thus explicitly excluding any non-observable displacement :

$$\delta V_l = \sum_{j \text{ such that } V_l \in j} \alpha_{l,j} \hat{n}_j \quad (7)$$

where the sum can include up to three terms, since an arbitrary displacement can be decomposed into any set of three linearly independent vector.

The corrected normal equation thus reads

$$\Lambda\alpha = \mathbf{N} \quad (8)$$

where α holds the normal components of all the solved-for displacements. The matrix N maps the now solved-for normal component of the displacements along each normal to the deviation in the coordinates of the observable vertices. Note that for computational efficiency, the information matrix and the normal matrix are computed in the following way:

$$\Lambda = (HN)^T (HN) = \sum_{k=0}^{N_m-1} (\tilde{H}_k N)^T (\tilde{H}_k N) \quad (9)$$

$$\mathbf{N} = (HN)^T \delta\rho = \sum_{k=0}^{N_m-1} N^T \tilde{H}_k^T \delta\rho_k \quad (10)$$

A damping term was finally added to the information matrix in the form of

$$\Lambda' = \Lambda + \beta I \quad (11)$$

where $\beta = 10$. This procedure effectively lowers the magnitude of the α and helps alleviating the linearization error inherent to this first-order filter.

For the sake of illustration, Figure 2 shows a set of observed and unobserved facets. Only the vertices belonging to at least one observed facet are themselves observable. Assuming that $\delta\mathbf{V}$ holds the deviations in the coordinates of V_0, V_2, V_3, V_4 and V_5 in this order, the N matrix corresponding to Figure 2 would read

$$N = \begin{bmatrix} \hat{n}_1 & \hat{n}_2 & \hat{n}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \hat{n}_1 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \hat{n}_1 & \hat{n}_2 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \hat{n}_2 & \hat{n}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \hat{n}_3 \end{bmatrix} \quad (12)$$

2.3. Observation Model

We are using the notations introduced in Figure 1. Given a lidar ray direction \hat{u}_k emitted from the pixel P_k and intersecting with the j -th facet at S_k , there is a positive real number ρ_k such that

$$P_k S_k = \rho_k \hat{u}_k \quad (13)$$

Taking the dot product of this expression with the normal to the facet \hat{n}_j , we get

$$P_k S_k^T \hat{n}_j = \rho_k \hat{u}_k^T \hat{n}_j \quad (14)$$

But since

$$P_k S_k^T \hat{n}_j = (P_k V_{0,j} + S_k V_{0,j})^T \hat{n}_j = P_k V_{0,j}^T \hat{n}_j + 0 \quad (15)$$

We have

$$\rho_k = \frac{\hat{n}_j^T P_k V_{0,j}}{\hat{n}_j^T \hat{u}_k} \quad (16)$$

The expression of the range measurement and the associated partials can be obtained from

$$\rho_k = \frac{P_k V_{0,j}^T \hat{n}_j}{\hat{n}_j^T \hat{u}_k} \quad (17)$$

$$= \frac{(V_{0,j} - P_k)^T \hat{n}_j}{\hat{n}_j^T \hat{u}_k} \quad (18)$$

$$= \frac{(V_{0,j} - P_k)^T n_j}{n_j^T \hat{u}_k} \quad (19)$$

where n_j is the outward surface normal of facet j ,

$$n_j = (V_{1,j} - V_{0,j}) \times (V_{2,j} - V_{0,j}) \quad (20)$$

Using n_j instead of its dimensionless counterpart \hat{n}_j drastically simplifies the expression of the partial derivatives required to set up the filter. We thus have

$$\frac{\partial \rho_k}{\partial V_{0,j}} = \frac{n_j^T}{\hat{u}_k^T n_j} + \frac{(V_{0,j} - P_k)^T}{\hat{u}_k^T n_j} \left[I_3 - \frac{n_j \hat{u}_k^T}{\hat{u}_k^T n_j} \right] \frac{\partial n_j}{\partial V_{0,j}} \quad (21)$$

$$\frac{\partial \rho_k}{\partial V_{(1,2),j}} = \frac{(V_{0,j} - P_k)^T}{\hat{u}_k^T n_j} \left[I_3 - \frac{n_j \hat{u}_k^T}{\hat{u}_k^T n_j} \right] \frac{\partial n_j}{\partial V_{(1,2),j}} \quad (22)$$

$$\frac{\partial n_j}{\partial V_{0,j}} = [\tilde{V}_{2,j}] - [\tilde{V}_{1,j}] \quad (23)$$

$$\frac{\partial n_j}{\partial V_{1,j}} = [\tilde{V}_{0,j}] - [\tilde{V}_{2,j}] \quad (24)$$

$$\frac{\partial n_j}{\partial V_{2,j}} = [\tilde{V}_{1,j}] - [\tilde{V}_{0,j}] \quad (25)$$

Equation (10) can now be solved for the displacement coordinates α . The corresponding deviation in the coordinates of the observable vertices is thus given by

$$\delta \mathbf{V} = N \alpha \quad (26)$$

Applying this deviation to the corresponding vertices effectively updates the shape. This batch procedure is iterated five times every time the Lidar sends a "flash" towards the true target.

2.4. Facet splitting

One of the drivers of our approach is to let the filter determine by itself where and if new degrees of freedom should be added to the estimated shape model. Namely, additional state parameters come in the form of new vertices that are inserted at appropriately chosen locations. Ideally, one could begin the shape determination procedure with a very coarse, isotropic shape model, that would get further refined based on where residuals indicate underfitting of the measurement data. Such underfitting is characteristic of a lack of local control over the shape.

To this end, the facet found to have the worst observation residuals among all facets at a given measurement time is split, along with its neighbors, according to the splitting scheme shown on Figure 3. This procedure adds three new vertices to the estimated shape model, along with ten facets. If four old facets are effectively removed from the shape model, their vertices are conserved and properly reassigned to the new facets so as to retain a consistent surface normal orientation. Note that a facet and its children were not allowed to be subdivided more than 5 times.

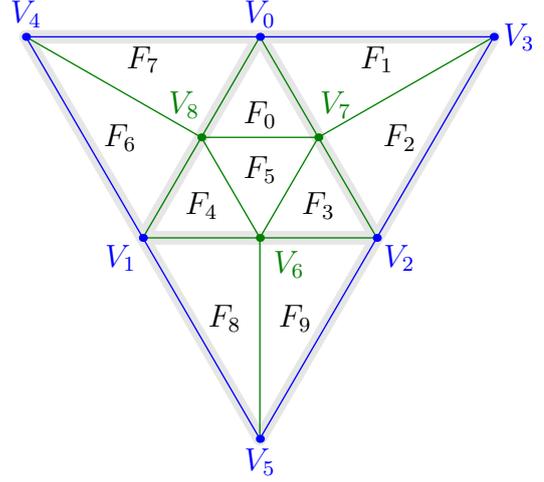


Fig. 3. Splitting scheme used to decompose 4 facets (in grey) into 10 facets after introducing 3 new vertices (in green), keeping the 6 vertices that were forming the original facets (in blue).

2.5. Facet recycling

Due to the updates in the vertices coordinates along with the introduction of new facets, it is necessary to ensure the mesh does not degenerate. Degeneracy can occur when a facet shrinks to the point that it almost reduces to an edge. Such degenerated facets are tracked and recycled, according to the recycling scheme presented on Figure 4. Tracking facet degeneracy helps preventing issues such as self-intersecting facets, as degenerate facets are often time a precursor of the former. The criterion used to flag a facet as degenerate was the minimum angle between two edges of the same facet, that was becoming less than $\eta_{min} = 15^\circ$. This scheme was also used to recycle any pair of facet (f_0, f_1) whose normal vectors (\hat{n}_0, \hat{n}_1) were found to be spuriously oriented. That is, such a pair was discarded when it was satisfying

$$\hat{n}_1^T \hat{n}_2 < -\cos(\gamma_{min}) \quad (27)$$

where γ_{min} was set to 20° . This check helped ensuring that no facet was folding over one of its neighbors.

2.6. Summary

A flowchart illustrating the general functioning of our method is provided on Figure 5. In the presented scenario, N_f Lidar flashes are cast at times t_1, \dots, t_f towards the true shape model. At every observation time t_o , a set of true measurements ρ_o is collected from this reference shape. The lidar generates a set of computed observations from the estimated shape model and only retains N_m residuals value by excluding outliers. N_m will always be less or equal than the lidar's resolution. These residuals are then processed by the batch, which returns deviations in the coordinates of the vertices that were observed on the estimated shape. The shape is then updated and the process repeats itself for N_{iter} iterations. Facet insertion/recycling then takes place, and time moves forward to the next observation. The body-fixed frames of the reference, estimate and lidar are updated to reflect their new state and the process goes on.

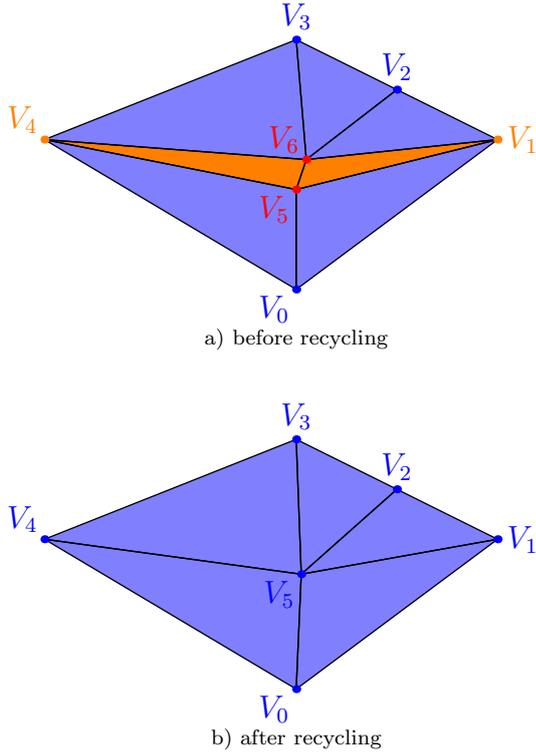


Fig. 4. Illustration of the facet recycling scheme. The highlighted facets formed by V_4, V_5, V_6 and V_5, V_1, V_6 should be recycled as their smallest vertex angle is less than a threshold value. The recycling consists in merging the vertices along the smallest edge. The facet degeneracy vanishes after facet recycling is complete. In this example, one vertex and two facets were thus removed from the shape model.

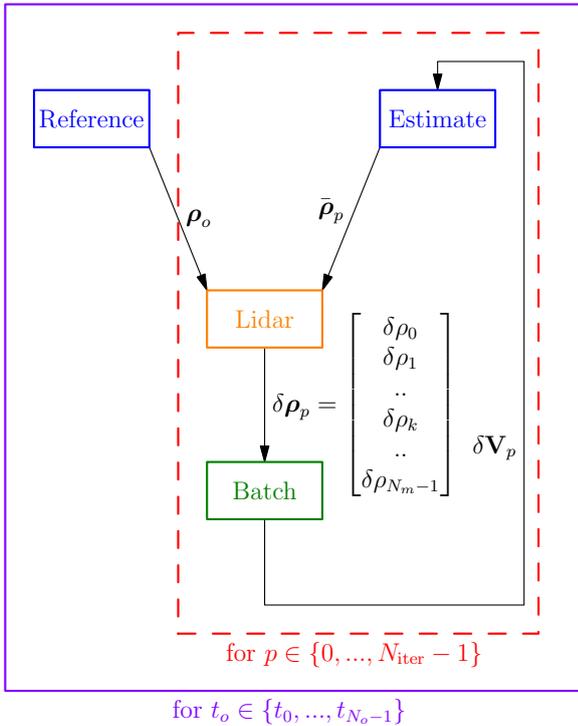


Fig. 5. Functioning of the autonomous shape determination algorithm

3. Results

3.1. Setup

The performance of the filter was illustrated on asteroids KW4 Alpha and Itokawa, for which shape models are available. The first of these two shape models presents a spin-top like shape that was obtained by means of high-resolution radar data from the primary body of the binary system 1999 KW4.¹⁷⁾ This KW4 shape model was comprised of 9168 facets and 4586 vertices. The Itokawa shape model derives from the one established from the observations collected by the Hayabusa spacecraft,¹⁸⁾ downsampled to a resolution of 768 facets and 386 vertices. The two shape models are shown on Figure 6 and Figure 7.

For both cases, the a-priori target used to initialize the filter was set to a faceted sphere of very coarse resolution, comprised of 80 facets and 42 vertices.. This a-priori was then corrected using observations collected over either of the two realistic shape models.

The Lidar instrument itself was on a circular orbit about the shape model's center of mass. The inclination of the orbit was set to $i = 80^\circ$, and the orbit rate was set to a constant value of $\omega_l = 2\pi \cdot 10^{-2}$ rad/s. Both reference and estimated shape models were spinning at a rate of $\omega_s = 2 \cdot 10^{-1}$ rad/s about the \hat{z} axis of the inertial frame of reference. ω_l and ω_s were chosen so as to be non-commensurable for good surface coverage. The Lidar line-of-sight axis was always pointing at the center of mass of the reference shape model. The observation time interval was set to $[t_1, t_f] = [0, 500](s)$ with a 1-second interval in between measurements. Finally, the range measurements were not corrupted by any noise. This is a current limitation of our approach, which will be addressed in our future work. Note that the typical range error provided by a Lidar instrument is in the order of a few centimeters, which is way less than the resolution of the shape models we are using in this paper.

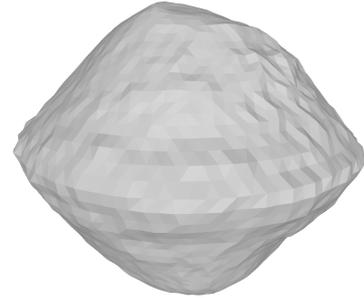


Fig. 6. Reference KW4 Alpha shape model

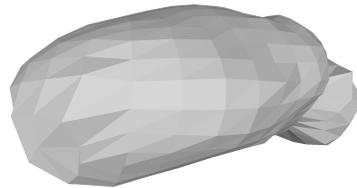


Fig. 7. Reference Itokawa shape model

3.2. Performance

The shape models estimate of KW4 and Itokawa after 500 seconds were respectively comprised of 1720 facets, 862 vertices and 720 facets, 362 vertices.

The resulting shape models are shown on Figure 9 and Figure 11. Visual inspection of the meshes revealed no folded facets or self-intersecting regions. A more quantitative measure of the shape quality is provided by the relative differences in volume and surface area between the estimated shape models and their corresponding reference. Both KW4 and Itokawa saw their volume and surface area captured with less than 0.5% of relative error, as shown on Figure 12 and Figure 13. It is noticeable that the KW4 surface area error does not continue decreasing after $t = 300$ seconds. This can be related to the difference between the allowable number of facets in the estimate and the number of facets in the true shape model. Letting each facet to be subdivided more would add the missing degrees of freedom enabling the shape to be captured more accurately.

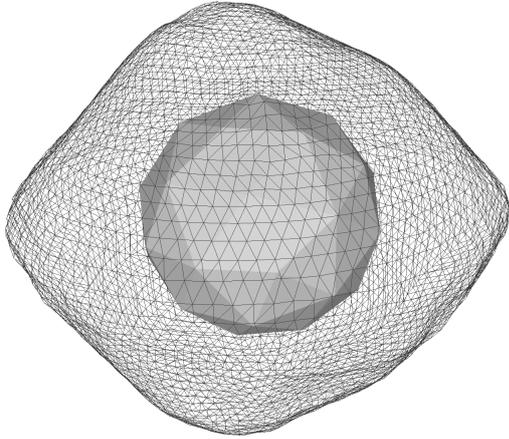


Fig. 8. A-priori of KW4 shape model overlaid with the reference mesh wireframe

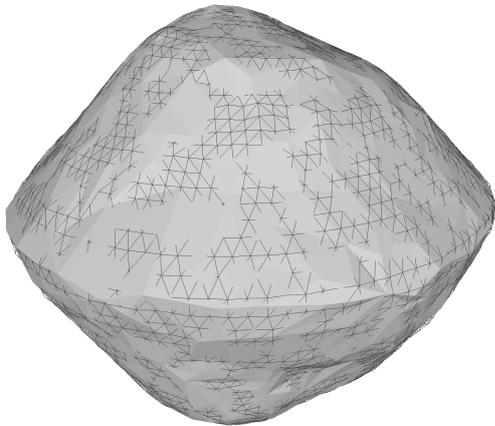


Fig. 9. Final estimate of the KW4 shape model overlaid with the reference mesh wireframe

This is exemplified by the better accuracy of the Itokawa shape model, obtained for an estimate comprised of nearly as many facets at the reference shape.

The current performance bottleneck resides in the ray-casting procedure, that is repeated N_f times for the reference shape model and $N_f * N_{iter}$ times for the estimated shape model. Most of the computational cost arises from finding the intersection of the cast rays with the targeted shape models. A brute-force approach relying on exhaustive facet search was used, but will be replaced by a voxel-grid based search shortly. The code was implemented in C++ using the Armadillo library.¹⁹⁾

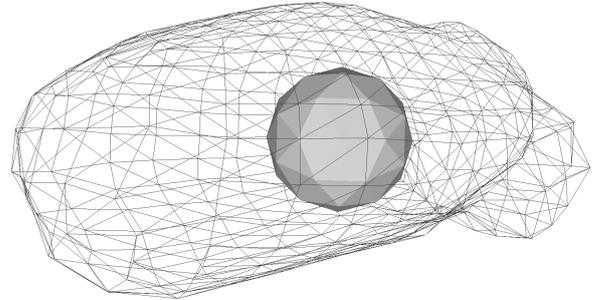


Fig. 10. A-priori of Itokawa shape model overlaid with the reference mesh wireframe

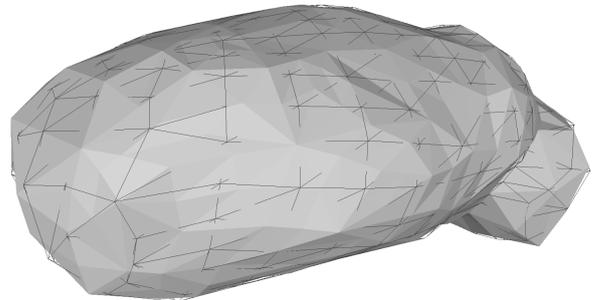


Fig. 11. Final estimate of the Itokawa shape model overlaid with the reference mesh wireframe

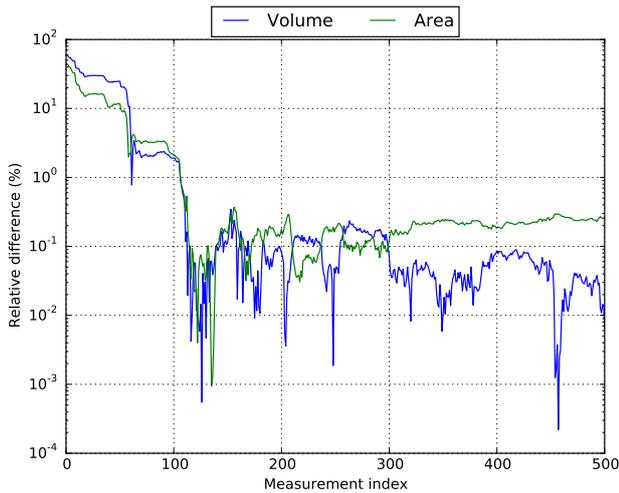


Fig. 12. Volume and surface area relative difference for the KW4 shape determination

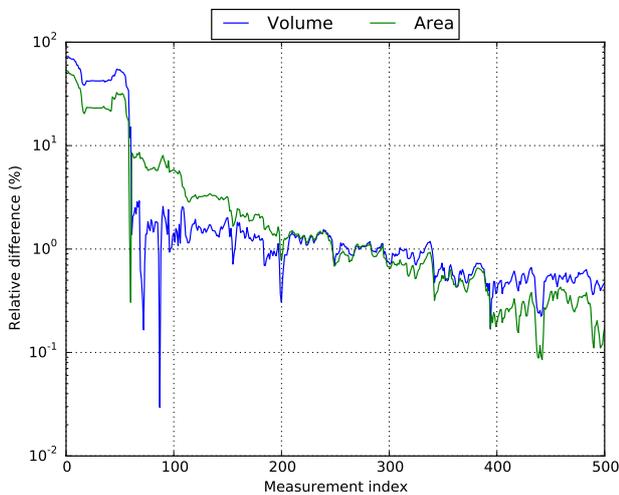


Fig. 13. Volume and surface area relative difference for the Itokawa shape determination

4. Conclusion

This paper validated the functioning of a new shape determination approach relying on sequential data processing for asteroids KW4 and Itokawa. This approach was enabled by the careful rewrite of the solved-for state deviation in a more observable form. Contrary to the state-of-the-art techniques of shape reconstruction, the shape determination problem is solved sequentially as data comes in, making the storage of raw point cloud measurements unnecessary. Performing mesh subdivision and facet recycling authorized the use of a very coarse a-priori that appears to converge to a satisfying estimate of the observed shape models. Future work will focus on solving the relative position, attitude determination problem. Moreover, incorporating realistic relative orbital dynamics will be considered, in addition to using noisy range measurements. This will enable us to complement our filter with a measure of the confidence in the estimated shape. The facet recycling scheme will also be reworked so as to improve its robustness. Finally, including the knowledge of the observing spacecraft's orbit into the filtering problem could help constraining the estimated shape model fur-

ther, thanks to the evaluation of the Polyhedron Gravity Model over the estimated shape model. This sequential filtering-based approach could thus enable spacecraft with limited memory or computational capabilities to perform on-board shape model reconstruction.

Acknowledgments

The authors would like to express their thanks to NASA for funding this work under the grant PDART NNX16AG50G.

References

- 1) Byron D. Tapley, Bob E. Schutz, and George H. Born. *Statistical Orbit Determination*. Academic Press, 2004.
- 2) E. J. Lefferts, F. L. Markley, and M. D. Shuster. Kalman Filtering for Spacecraft Attitude Estimation. *Journal of Guidance, Control, and Dynamics*, 5(5):417–429, 1982.
- 3) John Crassidis and F Markley. Attitude Estimation Using Modified Rodrigues Parameters. *Nasa Conference Publication*, pages 71–84, 1996.
- 4) Joseph Starek, Marco Pavone, Issa A. Nesnas, and Açikmese, Behçet. Spacecraft autonomy challenges for next generation space missions. *Lecture Notes in Control and Information Sciences*, 460:1–34, 2016.
- 5) Marshall H Kaplan, Bradley Boone, Robert Brown, Thomas B Criss, and Edward W Tunstel. Engineering Issues for All Major Modes of In Situ Space Debris Debris Capture. *AIAA Space*, (September):1–20, 2010.
- 6) Simone D'Amico, J.-S. Ardaens, G. Gaias, H. Benninghoff, B. Schlepp, and J. L. Jørgensen. Noncooperative Rendezvous Using Angles-Only Optical Navigation: System Design and Flight Results. *Journal of Guidance, Control, and Dynamics*, 36(6):1576–1595, 2013.
- 7) Allan Y. Lee and Julie a. Wertz. In-Flight Estimation of the Cassini Spacecraft's Inertia Tensor. *Journal of Spacecraft and Rockets*, 39(1):153–155, 2002.
- 8) E. Bergmann and J. Dzielski. Spacecraft mass property identification with torque-generating control. *Journal of Guidance, Control, and Dynamics*, 13(1):99–103, 1990.
- 9) J.A. Keim, A.B. Açikmese, and J.F. Shields. Spacecraft Inertia Estimation Via Constrained Least Squares. *2006 IEEE Aerospace Conference*, pages 1–6, 2006.
- 10) Yu Takahashi, Michael W. Busch, and Daniel J. Scheeres. Spin State and Moment of Inertia Characterization of 4179 Toutatis. *The Astronomical Journal*, 146(4):95, 2013.
- 11) Heike Benninghoff and Toralf Boge. Rendezvous Involving a Non-Cooperative, Tumbling Target - Estimation of Moments of Inertia and Center of Mass of an Unknown Target. In *Proceedings of the International Symposium on Space Flight Dynamics*, number 1, pages 1–16, 2015.
- 12) Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. Poisson Surface Reconstruction. *Proceedings of the Symposium on Geometry Processing*, pages 61–70, 2006.
- 13) Matthew D. Lichter and Steven Dubowsky. Estimation of state, shape, and inertial parameters of space objects from sequences of range images. In *Intelligent Robots and Computer Vision XXI: Algorithms, Techniques, and Active Vision Proceedings*, volume 5267. SPIE, 2003.
- 14) Brent E. Tweddle, Alvar Saenz-Otero, John J. Leonard, and David W. Miller. Factor Graph Modeling of Rigid-body Dynamics for Localization, Mapping, and Parameter Estimation of a Spinning Object in Space. *Journal of Field Robotics*, 32(6):897–933, 2015.
- 15) Corwin Grant Olson. *Sequential Estimation Methods for Small Body Optical Navigation*. PhD thesis, University of Texas at Austin, 2016.
- 16) Robert A. Werner and Daniel 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*, 65(3):313–344, 1997.

- 17) Daniel J. Scheeres, E. G. Fahnestock, Steven J Ostro, Jean-Luc Margot, L. A. M. Benner, Stephen B Broschart, Julie Bellerose, Jon D Giorgini, Michael C Nolan, Christopher Magri, Petr Pravec, P. Scheirich, R. Rose, R. F. Jurgens, E M De Jong, S. Suzuki, and E. M. De Jong. Dynamical Configuration of Binary Near-Earth Asteroid (66391) 1999 KW4. *Science*, 314(November):1280–1283, 2006.
- 18) A Fujiwara, J Kawaguchi, D K Yeomans, M Abe, T Mukai, T Okada, J Saito, H Yano, M Yoshikawa, D J Scheeres, O Barnouin-Jha, a F Cheng, H Demura, R W Gaskell, N Hirata, H Ikeda, T Kominato, H Miyamoto, a M Nakamura, R Nakamura, S Sasaki, and K Uesugi. The rubble-pile asteroid Itokawa as observed by Hayabusa. *Science*, 312(5778):1330–4, 2006.
- 19) Conrad Sanderson and Ryan Curtin. Armadillo: a template-based c++ library for linear algebra. *Journal of Open Source Software*, 1:26–32, 2016.