Landmark-Based Orbit Determination about an Asteroid

Ken Kuppa and Dahlia Baker

4/28/2021

Advanced State Estimation

Problem Overview: Motivation

- Landmark based navigation enables autonomy for planetary and small body missions [1]
- This solution must be robust to extended measurement gaps when passing over the unlit portion of a surface
- This project aims to investigate the robustness of the following filters:
    - Unscented Kalman Filter (UKF)
    - Rao-Blackwellized Particle Filter (RBPF)
    - Gauss Sum Unscented Kalman Filter (GSUKF)

- ***Problem Statement:*** Using precise landmark position information identified from the surface of an asteroid and IMU data taken onboard, estimate the spacecraft position, velocity, and attitude in the body-fixed frame, as well as the gravitational mass parameter of the planetary body despite error in measurement data, IMU uncertainty, and process noise.

- ***Key Variables:*** the filter state will consist of the spacecraft position, velocity, and attitude (represented as a quaternion), estimated IMU errors, target body gravity parameters (mu, J2, etc.), and measurement and process noise.

$$\hat{\mathbf{x}}_{S/C} = [\hat{x}, \hat{y}, \hat{z}, \dot{\hat{x}}, \dot{\hat{y}}, \dot{\hat{z}}, a_b, a_{sf}, g_b, g_{sf}, \hat{q}_i^b, \mu, J_2]^T$$

Problem Overview: Models

- ***IMU Model -*** accelerator and gyroscope

- ***Camera Model -*** Line-of-Sight measurements to each and any identifiable landmark. Will be expanded to include a field-of view discrimination scheme based on real mission instrumentation standards and simulated orbit distance

$$y(t_k) = LOS(l_1, l_2, \cdots, l_i) = \begin{bmatrix} x_{S/C}(1:3) - \vec{l}_1 \\ x_{S/C}(1:3) - \vec{l}_2 \\ \vdots \\ x_{S/C}(1:3) - \vec{l}_i \end{bmatrix}_{BF}$$

$$\vec{l}_i = [x, y, z]_{BF}$$

- ***Dynamics Model -*** two-body orbit problem with higher order gravity terms as perturbing accelerations

$$\ddot{\mathbf{r}} = \frac{\mu}{r^3}\mathbf{r} + \frac{\partial U}{\partial \mathbf{r}}$$

CCAR

Objectives Overview

ORCCA

✓ - completed
✓ - in progress

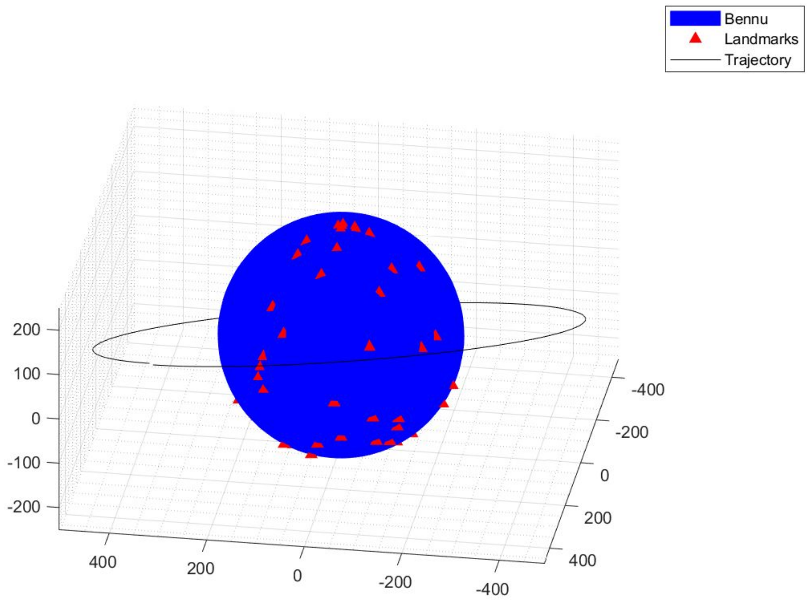
- ***Technical Objectives:***
  - **Level 1**
    - Characterize the orbit determination problem with the appropriate dynamics, measurement, and noise models ✓
    - Implement an Unscented Kalman Filter over multiple orbits of simulated data which includes lit/unlit transition gaps ✓
    - Implement a Rao-Blackwellized Particle Filter with covariances below that of unrestricted IMU drift ✓
  - **Level 2**
    - Improve the RBPF solution with IMU drift as an error comparison
    - Investigated sensor fusion with the integration of StarTracker information and compare results using the particle filter
    - Implement a Gaussian Mixture Model approach to compare to the UKF and RBPF, using one metric of split and one metric for compression ✓
  - **Level 3 -** Investigate and compare Gaussian Mixture Model approaches with varying levels of split and compression
- ***Levels of Success:***
  - The work will be considered successful if filters at the Level 2 priority are able to perform orbit determination and gravity estimation after gaps in the measurement data
  - UKF can be considered successful despite a divergence post-measurement gap
  - RBPF will be robust to time gap and GMM will be robust to time gap
- ***Milestones/Metrics:***
  - Have dynamics and measurement models completed ✓
  - Working implementation of UKF ✓
  - RBPF implementation producing better estimates than IMU drift ✓
  - All navigation solutions will constrain s/c position and velocity, attitude, and gravity uncertainty below a certain threshold

Problem Formulation

- Given a spherical body, each landmark location is generated like so:

$$\theta_{lmk} = \mathcal{U}[0, \pi]$$
$$\phi_{lmk} = \mathcal{U}[-\pi/2, \pi/2]$$
$$x_{lmk} = R_{body}\, cos(\phi_{lmk}) sin(\theta_{lmk})$$
$$y_{lmk} = R_{body}\, sin(\phi_{lmk}) sin(\theta_{lmk})$$
$$z_{lmk} = R_{body}\, cos(\theta_{lmk})$$

- 50 Landmarks were generated. Figure to the right.
- Trajectory shown in black



Level 1: UKF

Overview

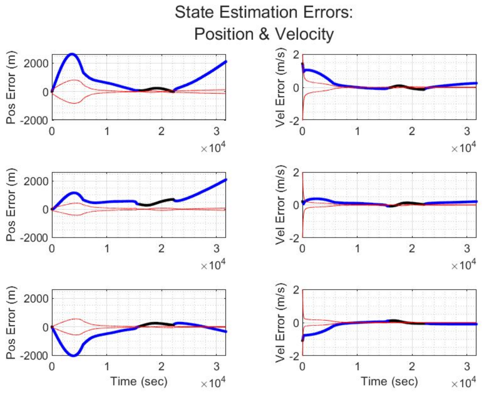
- **Objective:** Implement a UKF filter to estimate spacecraft pose and gravity (μ) from line-of-sight measurements of surface landmarks on Bennu.
- **Dynamics:** Two-body gravitational dynamics with process noise.
- **Measurements:** Line-of-sight measurements to surface landmarks

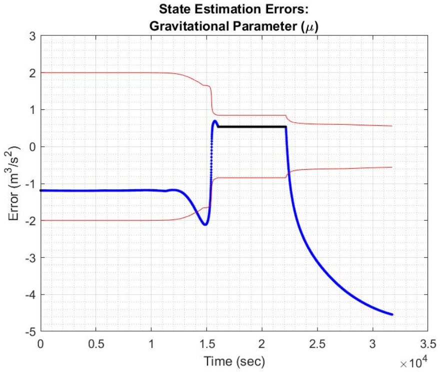
Results

- UKF Results are shown to the right.
- Initial conditions are generated as follows:

$$\mathbf{x}_0 = \mathcal{N}(\mathbf{x}_{0,true}, P_0)$$
$$\mathbf{x}_{0,true} = [r_0, v_0, \mu_0]^T$$
$$P_0 = \text{diag}([1000, 1000, 1000, 1, 1, 1, 1]]))$$

- Data blackouts are indicated by black parts of the estimates
- Covariance is shown in red.





Level 1: Rao-Blackwellized Particle Filter

Overview: RBPF

- ***Justification:***
  - Reduces dimensionality of the problem from the full hypothesis space [2] and requires fewer particles [3]
  - Computational efficiency is priority for any space-based onboard OD solution
  - Typically asteroid exploration includes an approach mapping, so a SLAM RBPF implementation is proposed to build a map while enhancing the navigation solution [4]

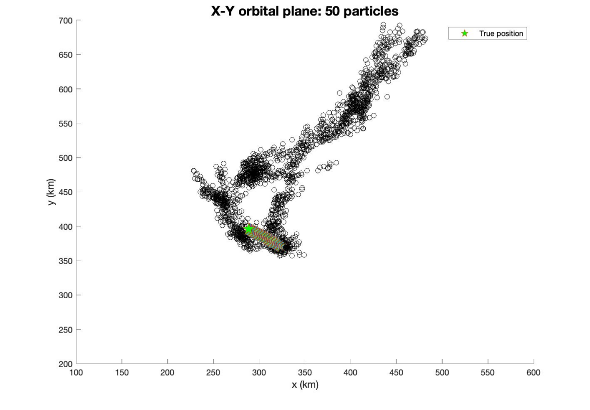
Same objective, dynamics, and measurements as each filter explored in this project:

- ***Objective:*** Implement a Rao-Blackwellized Particle filter to estimate spacecraft pose and gravity (μ) from line-of-sight measurements of surface landmarks on Bennu.
- ***Dynamics:*** Two-body gravitational dynamics with process noise
- ***Measurements:*** Line-of-sight measurements to surface landmarks

- ***Expected Algorithmic Approach:***
  - Repartition the system state vector into sampled state and analytical state
  - Apply Rao-Blackwell theorem sufficient statistics to determine particle number N_s
  - Initialize particles at time k to include the analytical filter variables
  - Complete analytical measurement update
  - Find filter normalizing constant from the Gaussian innovation likelihood
  - IS weight update
  - Resampling

Progress: RBPF

- ***Current Implementation: Regular Particle Filter with UKF, N_s = 50***
  - Began framework for developing the RBPF
  - Current simulation over 100 time steps
  - 50 particles
  - Resampling included - IS weight update
- ***Pending Updates***
  - Split state into analytical and sampled parts by RB theorem
  - Include analytical measurement update
  - Parallelization of dynamic update



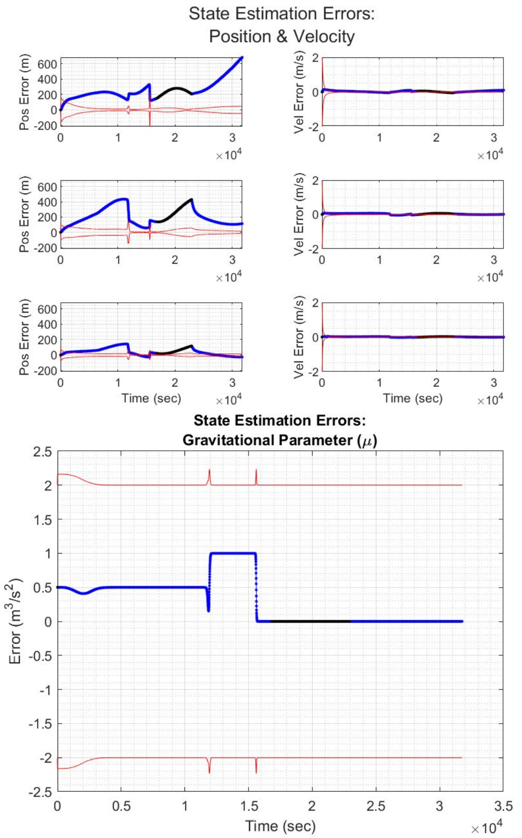
Level 2: GSF

Overview

- **Objective:** Implement a GSUKF filter to estimate spacecraft pose and gravity (μ) from line-of-sight measurements of surface landmarks on Bennu.
    - GM prior only with 3 mixands with equal weights
- **Dynamics:** Two-body gravitational dynamics with process noise.
- **Measurements:** Line-of-sight measurements to surface landmarks

Results

- GSUKF Results are shown to the right.
- Initial conditions are generated as a gaussian mixture
- Data blackouts are indicated by black parts of the estimates
- Covariance is shown in red.



CCAR

Pending Tasks

ORCCA

- ***Level 1:***
  - IMU model.
  - Augment J2 gravity perturbation to truth model dynamics. Leave filter dynamics unchanged.
  - Include camera parameters for landmark field of view discrimination
- ***Level 2:***
  - Improve on RBPF
  - Sensor fusion with Star Tracker data
  - GSUKF with mixture process and measurement models
- ***Level 3:***
  - Compare GMM tuning and compression

Technical Questions

- Implementing parallelization in particle filter:
  is this possible when the current particle state is dependent on it's previous time step quantity?
  - For context - Matlab complains about indexing and sliced variables
- Calculating estimates from GSUKF:
  - At each time, I loop through the measurements and update the weights sequentially. Is this the right way to do it?

```
for i = 2:length(t)
  %for each particle
   parfor j = 1:Ns
      %prediction step
      %UKF
      %xfilter is the current particle, ran through a UKF
      Xfilter = xsamplehist(:,j,i-1)+mvnrnd(zeros(6,1),sqrt(Qpf))';
      Pfilter = Pksamplehist(:,:,j,i-1);
```

```
% update weights
wts(jj) = wts(jj)*mvnpdf(losLand(:,kk),yhatm{jj},Pkyy);
```

References

1. Leonard, J. M., Geeraert, J. L., Page, B. R., French, A. S., Antreasian, P. G., Adam, C. D., Wibben, D. R., Moreau, M. C., & Lauretta, D. S. (2020). Osiris-rex orbit determination performance during the navigation campaign. *Advances in the Astronautical Sciences*, *171*, 3031–3050.
2. Ahmed, N., Casbeer, D., Cao, Y., & Kingston, D. (2017). Multitarget localization on road networks with hidden Markov Rao-Blackwellized particle filters. *Journal of Aerospace Information Systems*, *14*(11), 573–596. https://doi.org/10.2514/1.I010539
3. Ristic, Branko. & Arulampalm, Sanjeev. & Gordon, Neil. (2004). *Beyond the Kalman filter : particle filters for tracking applications*. Boston, Ma. ; London : Artech House
4. Grisetti, G., Stachniss, C., & Burgard, W. (2007). Improved techniques for grid mapping with Rao-Blackwellized particle filters. *IEEE Transactions on Robotics*, *23*(1), 34–46. https://doi.org/10.1109/TRO.2006.889486