

Optimal low-thrust navigation around bilobate asteroids with sequential convex programming

Rory Lipkis, *Stanford University, Department of Aeronautics and Astronautics*

Introduction

As interest in asteroid exploration missions increases, there is potential use for low-thrust control around irregularly-shaped gravitating bodies. Recent missions to asteroids 67P/Churyumov–Gerasimenko (Rosetta) and 486958 Arrokoth (New Horizons) revealed bilobate shapes, which appear to be a relatively common feature of asteroids (see Figure 1). These shapes, also known as *contact binaries*, produce highly nonspherical gravitational potentials that cannot be easily navigated.

Stacey and D’Amico 2018¹ describe a method by which satellite formations could be used to perform initial estimation of asteroid gravitational fields. Although such gravitational fields must necessarily be very weak (or else the bodies would be pulled into spheroid), future missions that entail a large number of asteroid visits or independent spacecrafts must contend with potentially stringent thrust constraints, which limit the spacecraft control ability to within an order of magnitude of the surface gravity. In this domain, the irregularity of the gravitational becomes a nontrivial consideration.

Irregular gravitational fields are typically represented by their decomposition into spherical harmonics, the basis of Legendre polynomials. The standard orbital element representation of orbital motion can be modified to account for low-order harmonics, such as the J_2 oblateness correction, with the introduction of mean and osculating elements, which specify an “instantaneous” unperturbed orbit. This extension allows control schemes that rely on orbital elements to be easily extended to find optimal thrust solutions around slightly oblate bodies. Though useful for orbits around planets, this method does not generalize to trajectories in highly irregular gravitational fields, whose specifications require extremely high order spherical harmonics, and where the no-

tion of orbital elements ceases to be meaningful altogether. Since such trajectories are rarely stable orbits, and in fact are nearly always chaotic, they cannot be meaningfully specified by any set of periodic elements, and the Cartesian position and velocity state must be used instead.



Figure 1: Comet 67P, *via Wikipedia*.

Asteroid model

This paper examines a simplified model of a bilobate asteroid, consisting of two spherical masses of constant density, each with radius R_i , oriented along an axis $\hat{\mathbf{a}}$. By modeling the asteroid in this way, the costly spherical harmonic representation is replaced by a direct Newtonian calculation, which allows for much higher “orders” to be effectively considered, despite the reduced detail. The gravitational field of the asteroid is then given by

$$\mathbf{f}_g = -\frac{GM_1(\mathbf{r} - \mathbf{r}_1)}{|\mathbf{r} - \mathbf{r}_1|^3} - \frac{GM_2(\mathbf{r} - \mathbf{r}_2)}{|\mathbf{r} - \mathbf{r}_2|^3}$$

¹Stacey, Nathan, and Simone D’Amico. “Autonomous Swarming for Simultaneous Navigation and Asteroid Characterization.” In *AAS/AIAA Astrodynamics Specialist Conference*. 2018.

where

$$\begin{aligned}\mathbf{r}_1 &= \frac{m_2}{m_1 + m_2}(R_1 + R_2)\hat{\mathbf{a}} \\ \mathbf{r}_2 &= -\frac{m_1}{m_1 + m_2}(R_1 + R_2)\hat{\mathbf{a}}\end{aligned}$$

are the positions of the lobe centers in the center-of-mass frame.

Sequential convex programming

The system is formalized by the state \mathbf{s} , formed by the concatenation of the spacecraft position and velocity. Along with the control input \mathbf{u} , the dynamics are written as

$$\dot{\mathbf{s}} = \mathbf{f}(\mathbf{s}, \mathbf{u}) = \begin{bmatrix} \mathbf{v} \\ \mathbf{f}_g + \mathbf{u} \end{bmatrix}$$

This is linearized around a reference state \mathbf{s}_0 and control input \mathbf{u}_0 by the first-order expansion

$$\dot{\mathbf{s}} = \mathbf{f}(\mathbf{s}_0, \mathbf{u}_0) + A(\mathbf{s} - \mathbf{s}_0) + B(\mathbf{u} - \mathbf{u}_0)$$

The system matrix A is given by

$$A = \frac{\partial \mathbf{f}}{\partial \mathbf{s}}(\mathbf{s}_0, \mathbf{u}_0) = \begin{bmatrix} \mathbf{0}_{3 \times 3} & \mathbf{I}_{3 \times 3} \\ A_1 + A_2 & \mathbf{0}_{3 \times 3} \end{bmatrix}$$

where

$$A_i = \frac{\mu(3\mathbf{d}_i\mathbf{d}_i^T - |\mathbf{d}_i|^2\mathbf{I}_{3 \times 3})}{|\mathbf{d}_i|^5}$$

and $\mathbf{d}_i = \mathbf{r} - \mathbf{r}_i$ is the separation from a lobe center. The control matrix B is given by

$$B = \frac{\partial \mathbf{f}}{\partial \mathbf{u}}(\mathbf{s}_0, \mathbf{u}_0) = \begin{bmatrix} \mathbf{0}_{3 \times 3} \\ \mathbf{I}_{3 \times 3} \end{bmatrix}$$

The sequential convex programming method solves at each iteration the problem

$$\begin{aligned}\min \quad & J(\mathbf{s}, \mathbf{u}) \\ \text{s.t.} \quad & \dot{\mathbf{s}}(t) = \mathbf{f}_k(\mathbf{s}(t), \mathbf{u}(t)) \\ & \mathbf{s}(0) = \mathbf{s}_0 \\ & \mathbf{u}(t) \in \mathcal{U} \\ & |\mathbf{s}(t) - \mathbf{s}_k(t)| \leq \rho, \quad t \in [0, t_f]\end{aligned}$$

where \mathbf{f}_k is the linearization of the dynamics with the previous solution trajectory \mathbf{s}_k and control history \mathbf{u}_k used as references. The dynamics constraint is enforced by a naive Euler integration that equates

$$\mathbf{s}_{t+1} = \mathbf{s}_t + \mathbf{f}_k(\mathbf{s}(t), \mathbf{u}(t))\Delta t$$

Implementation

Several things must be noted:

- (a) The equations of gravitational motion linearize notoriously poorly, and thus the trust region ρ must be monitored closely.
- (b) Neither the exact nor linearized dynamics admit a useful non-dimensional form. As a result, actual values for distance and mass must be used, which can result in extremely large costs that destabilize the convex solver. The cost matrices must be carefully scaled to produce a cost of a reasonable order of magnitude.
- (c) Additionally, the cost is a function of the control history and the final state, but not the entire trajectory:

$$J(\mathbf{s}, \mathbf{u}) = \mathbf{s}(t_f)^T Q_f \mathbf{s}(t_f) + \int_0^{t_f} \mathbf{u}(t)^T R \mathbf{u}(t) dt$$

The large distances involved at the beginning of a regularization problem would otherwise result in large terms in the cost that completely overwhelmed the control terms.

Example mission