Planetary Defense Conference 2013

IAA-PDC13-04-23

GPU-Based Optical Navigation and Terminal Guidance Simulation of a Hypervelocity Asteroid Intercept Vehicle (HAIV)

Joshua Lyzhoft[a,1], Matt Hawkins[a,1,*], Brian Kaplinger[a,1], Bong Wie[a,2]

[a]*Iowa State University, 2271 Howe Hall, Ames, IA 50011-2271, USA*

Abstract

With increased awareness of the possibility of an asteroid impacting the Earth, there is a corresponding increased interest in developing techniques for mitigating the asteroid impact threat. One such concept is the hypervelocity asteroid intercept vehicle (HAIV). Hypervelocity target intercept using autonomous optical navigation strategies has been demonstrated by the Deep Impact mission to the 5-km comet Tempel 1. Potentially hazardous objects are as small as 50 meters in diameter, meaning that an advanced autonomous navigation and precision terminal guidance system is needed for the HAIV mission. This paper describes GPU-based optical navigation and guidance simulation of a HAIV during its terminal phase operation to intercept a small, 50-m class target asteroid.

Keywords: Optical navigation, terminal-phase guidance, GPU, hypervelocity asteroid intercept vehicle (HAIV)

1. Introduction

As potential asteroid threats to the Earth are being discovered, new and innovative missions must be designed to address the range of threats. New asteroid intercept spacecraft will need to incorporate autonomous navigation and guidance strategies for successful mitigation missions. The Deep Impact mission demonstrated the use of optical navigation and a terminal guidance scheme for a hypervelocity (10.3 km/s) intercept of a target comet [1, 2]. The successful collision with comet Tempel 1 (diameter of approximately 5 km) was achieved by pin-pointing a location on the sunlit side of the object, then guiding to this region. Design of a hypervelocity asteroid intercept vehicle (HAIV) that can potentially operate in a relative speed range of 10 to 30 km/s have been studied by Pitz et al. [3]. In the design of interest, the HAIV separates into two bodies, similar to the Deep Impact mission. The fore body creates a crater, and the aft body delivers a nuclear explosive device (NED) to the crater's interior. Delivering and detonating a NED creates sufficient energy, ranging from 300 kt to 2 Mt of TNT, to disrupt the target asteroid. Figure 1 illustrates a baseline notional design, related to that shown by Pitz et al. [3] as well as a terminal phase concept.

This paper describes the preliminary implementation of a GPU (Graphic Processing Unit) based optical navigation and guidance simulation of terminal phase hypervelocity intercept of a scaled 433 Eros polyhedron model. Using this model does not limit the utilization of other known asteroid shape archetypes. Data for this model has been obtained from the NEAR collected shape models [4]. Scaling this model by a factor of 333 generates an asteroid with a 50-m diameter, considered to be a lower limit for asteroids that may require hypervelocity intercept. A lighting and diffusion model creates a realistic simulated image, which can then be projected onto the camera plane, yielding realistic sensor inputs to drive the guidance algorithms. The camera model is based on flight-proven cameras used in previous missions.

With simulated images of the lit asteroid, the center of brightness (COB) of the figure can be determined for the guidance and navigation operations. The COB is calculated from light intensity of each camera pixel. The calculated COB is used to estimate the line-of-sight (LOS) vector from the spacecraft to the target. Knowledge

*Corresponding author

Email addresses: lyzhoft@iastate.edu (Joshua Lyzhoft), mhawkins@iastate.edu (Matt Hawkins), bdkaplin@iastate.edu (Brian Kaplinger), bongwie@iastate.edu (Bong Wie)

[1]Graduate Research Assistant, Asteroid Deflection Research Center, Department of Aerospace Engineering

[2]Vance Coffman Endowed Chair Professor, Asteroid Deflection Research Center, Department of Aerospace Engineering

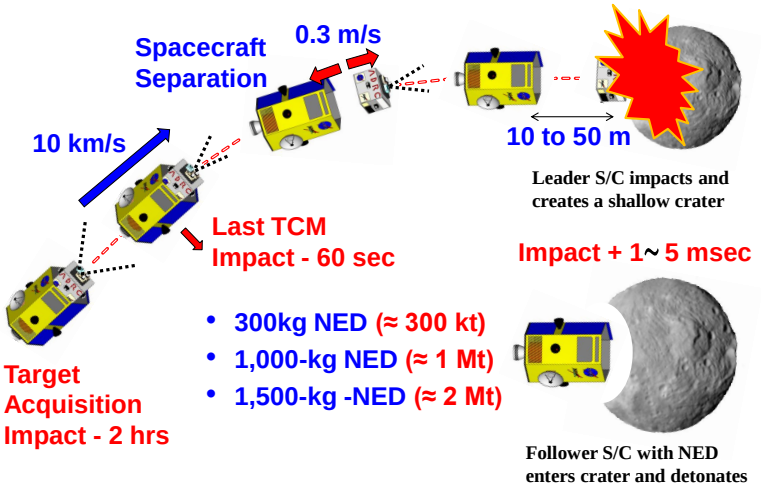


Figure 1: A baseline two-body HAIV mission concept.

of the LOS vector at different time steps yields a calculated LOS rate. Several different guidance laws are used, including classical proportional navigation (PN) and predictive guidance laws.

Along with simulation of camera images, this paper will demonstrate the plausibility of an asteroid impact for a two-body (comprised of fore and aft bodies) spacecraft separating at 60 seconds before the final intercept occurs. The separation mechanism concept is taken from the Deep Impact mission, where the impactor separated from the flyby spacecraft at a relative speed of approximately 36 cm/s. Imaging by a flyby craft has also been studied in multiple missions, such as Deep Space 1. Once the fore and aft body separation occurs, no additional control input is applied.

2. Rendering the Simulated Camera Data

2.1. 3D Target Polygon Model

This section addresses the rendering of the images in simulation memory that correspond to camera data to be processed by the spacecraft. We begin with a triangulation of surface points represented as a 3D wireframe polygon model. The target model is derived from data for the asteroid 433 Eros [1] collected by the NEAR mission, and has 200,700 faces [5]. This number of faces corresponds to the variable n_f of the computer model. Generally, the number of vertices needed to fill out the model, n_v, is strictly less than n_f, so it is more computationally efficient to do calculations on the vertices where possible. However, much of our information depends in some sense on an orientation of the body surface, for which the faces are necessary. A connectivity array stores the relationships between each set of vertex data and which face it applies to. If $\mathbf{R}_1$, $\mathbf{R}_2$, and $\mathbf{R}_3$ are the position vectors for the vertices of a face, as shown in Figure 2, then we have $\mathbf{X}_i$, $\mathbf{Y}_i$, and $\mathbf{Z}_i$ for $a \leq i \leq n_f$, defined as

$$\mathbf{X}_i = \mathbf{R}_2 - \mathbf{R}_1 \qquad \mathbf{Y}_i = \mathbf{R}_3 - \mathbf{R}_1 \qquad \mathbf{Z}_i = (\mathbf{R}_1 + \mathbf{R}_2 + \mathbf{R}_3)/3 \tag{1}$$

The unit normal vectors of each face are calculated and stored as

$$\mathbf{N}_i = \mathbf{X}_i \times \mathbf{Y}_i, \qquad \mathbf{N}_i \cdot \mathbf{Z}_i > 0 \tag{2}$$

The wireframe model uses a known rotation state of the target, though this information is not available to the spacecraft. A rotation matrix is applied to the initial conditions so that a simulated "real" state is known for the 3D model. To complete this system description, we require a unit sun vector, $\widehat{\mathbf{S}}$, in the direction of the sun from the target and a unit view vector, $\widehat{\mathbf{V}}$, in the direction of the spacecraft from the target. We assume that the true center of mass of the target is at the origin on this coordinate system.

2.2. Camera Pointing and Focus Plane

In order to decouple the attitude and translation mechanics of the present simulation system, a "perfect" pointing was adopted for the camera. Thus, the choice of estimated center of mass or previously computed center

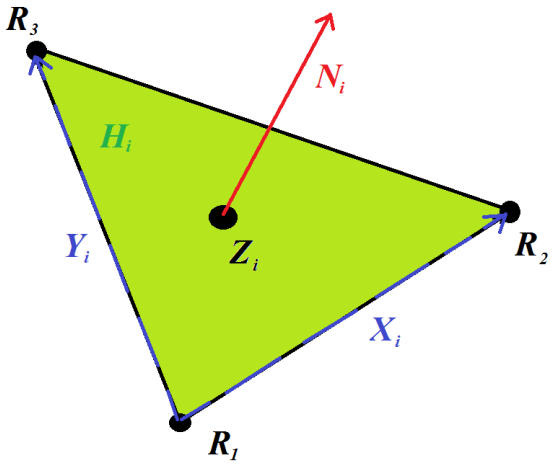


Figure 2: Facet geometry definitions.

of brightness is at the center of the camera focal plane, with the camera needing zero time to adjust to the new settings. Modifications to this scheme, including integration with spacecraft attitude are expected in the near future.

The camera focal plane distance was chosen arbitrarily as the distance from the spacecraft to target. Since this plane is perpendicular to $\widehat{\mathbf{V}}$ we can project the 3D model onto this plane by subtracting the component along this vector, thus the projected locations of the vertices are described by

$$\mathbf{P}_j = \mathbf{R}_j - \widehat{\mathbf{V}}, \qquad 1 \leq j \leq 3 \tag{3}$$

To save computational time and storage space, a condensed array of these values are computed only for the faces satisfying $\mathbf{N}_i \cdot \widehat{\mathbf{V}} > 0$. This results in only the faces visible to the spacecraft. Due to the decoupling of the camera dynamics, a direct orientation of this plane is not possible given the currently available information. Thus, a direction for camera "up" must be chosen. We choose unit vectors $\widehat{\mathbf{U}}$ and $\widehat{\mathbf{V}}$ such that

$$\widehat{\mathbf{W}} = \widehat{\mathbf{S}} \times \widehat{\mathbf{V}}, \qquad \widehat{\mathbf{U}} = -\widehat{\mathbf{V}} \times \widehat{\mathbf{W}} \tag{4}$$

Therefore, the sunlight will always come into frame from a horizontal direction, and "up" will always be orthogonal to the plane containing the target, the sun, and the spacecraft. The coordinates of each vertex in this plane can be computed as

$$\widetilde{\mathbf{P}}_j = \begin{bmatrix} u_j & w_j \end{bmatrix}^T = \begin{bmatrix} \mathbf{P}_j \cdot \widehat{\mathbf{U}} & \mathbf{P}_j \cdot \widehat{\mathbf{W}} \end{bmatrix}^T \tag{5}$$

At this point, what is visible to the spacecraft depends on camera parameters that interpret the "real" system. The resolution and field of view for the cameras simulated are listed in Table 1. At each time step, these are used to compute the half resolution of the image plane, R_h. If $\mathbf{R}_T$ is the position vector representing the spacecraft in the target frame, then this can be computed as

$$R_h = |\mathbf{R}_T| \tan\left(\frac{1}{2} f_v\right) \tag{6}$$

where f_v represents the field of view in radians (assumed isotropic). If x_r and y_r represent the x and y resolutions in the camera 2D pixel frame, then the information represented by each pixel corresponds to a size of $2R_h/x_r$ in the horizontal direction and $2R_h/y_r$ in the vertical direction.

Table 1: Simulated camera parameters

	High Resolution	Medium Resolution	Infrared
Resolution (x_r, y_r) (pixels)	1024 x 1024	1024 x 1024	512 x 512
Field of View (f_v) (radians)	2.05×10^{-3}	10.0×10^{-3}	10.0×10^{-3}
Pixel Size at 1,000 km (m)	2.0	9.8	19.5

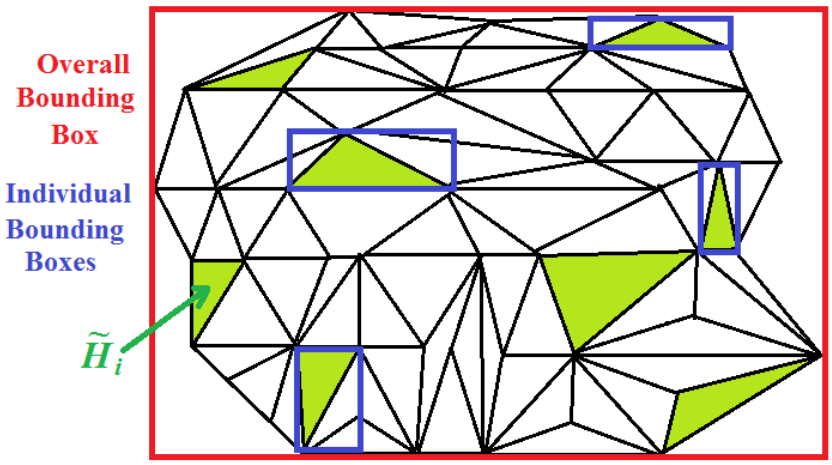


Figure 3: Parallel trim reduction technique.

2.3. Lighting Conditions

A simple flat shading model is used to calculate the brightness of the target surface. Given a diffuse lighting coefficient, k_d, and an ambient lighting coefficient, k_a, the corresponding brightness attributed to each face of the 3D model is computed as

$$C_i = \begin{cases} k_d\left(\mathbf{N}_i \cdot \widehat{\mathbf{S}}\right) + k_a & \mathbf{N}_i \cdot \widehat{\mathbf{S}} > 0 \\ 0 & \mathbf{N}_i \cdot \widehat{\mathbf{S}} \leq 0 \end{cases} \tag{7}$$

where a value C_i of 1 corresponds to perfect reflectivity of the incident sunlight at that distance. A greyscale colormap is used for human visualization. The diffuse coefficient works closely in line with the albedo, and is chosen to be 0.25 for the simulations. Since the cameras we are using have very little time to resolve detail of the body, we assume that their lower threshold for brightness is very low and therefore $k_a = 0$. This model is very fast, and can be computed completely in parallel for each face.

When additional lighting detail is desired, the following interpolation model due to Gouraud [6] is used. Let each vertex have a normal $\widetilde{\mathbf{N}}_1$, $\widetilde{\mathbf{N}}_2$, and $\widetilde{\mathbf{N}}_3$ computed as an average of the adjacent faces, found from inverting the connectivity array. Then, a corresponding value for brightness at each vertex $\widetilde{C}_1$, $\widetilde{C}_2$, and $\widetilde{C}_3$ is computed using the flat shading model equation. If a pixel representing the image plane at $[u\ w]^T$ is calculated to intercept this triangle, then an inverse squared weighted average is assigned to the pixel, as follows:

$$C\left(u, w\right) = \frac{\alpha_1}{\alpha_T}\widetilde{C_1} + \frac{\alpha_2}{\alpha_T}\widetilde{C_2} + \frac{\alpha_3}{\alpha_T}\widetilde{C_3} \tag{8}$$

where $\alpha_T = \alpha_1 + \alpha_2 + \alpha_3$ and

$$\alpha_j = \frac{1}{\left(u - u_j\right)^2 + \left(w - w_j\right)^2} \quad 1 \leq j \leq 3 \tag{9}$$

2.4. Pixel Value Assignment

For either the flat or interpolated shading models, the facet at which a pixel intercepts the target projection must be determined. This is done on the GPU in parallel to reduce computational time, especially using a higher fidelity target model. Each pixel is assigned coordinates in the u, w plane representing the center of the box over which the pixel is integrated. This works well when the size of a pixel is on the scale of the surface facets, but an averaged value needs to be used at greater distances. To determine the intercepted triangle, we first compute the upper and lower bounds of each visible triangle in parallel. Then, a reduction is performed to get the outer limits of the target in the UW system. This process is represented in Figure 3. Pixels outside of this range can automatically be assigned a value of 0. In fact, only the location and value of nonzero image components are stored in the present computational model.

The arrays containing the limits of each facet are sorted, and then a scan of these arrays is completed. This lets us assign a vector of possible intersections for each pixel. This vector contains the IDs of all facets through which the pixel passes through the bounding box. This is generally a low number. For each candidate intersection, the convex hull of the triangle, Hi, is identified, as shown in Figure 1. This is projected onto the UW plane, forming

the set $\widetilde{H}_i$ shown in Figure 2. If $\widetilde{\mathbf{X}}_i$ and $\widetilde{\mathbf{Y}}_i$ are the projections of $\mathbf{X}_i$ and $\mathbf{Y}_i$ in the UW plane, then the coordinates of the pixel can be expressed as

$$\begin{bmatrix} u \\ w \end{bmatrix} = \widetilde{\mathbf{P}}_1 + d_1\widetilde{\mathbf{X}}_i + d_2\widetilde{\mathbf{Y}}_i \tag{10}$$

which can be viewed as a transformation into the affine system defined by these two vectors. It is a well-known result from geometry that the point $[u\ w]^T$ lies within the triangle if $d_1 > 0$, $d_2 > 0$, and $d_1 + d_2 < 1$. This generally assigns a single facet as the possibility for intersection. However, at pixels near the boundary of the target, several potential intersection may occur. In these cases, the closest triangle (largest $\mathbf{Z}_i \cdot \widehat{\mathbf{V}}$) is chosen.

2.5. Line of Sight Vector

For the present implementation, the estimated line of sight vector is chosen to be the center of brightness. If $I(u, v)$ is a matrix storing the values of brightness for the image, then the centroid of the image array is computed using an weighted average, as follows:

$$\bar{u} = \frac{\sum I(u,v)\, u}{\sum I(u,v)}; \qquad \bar{w} = \frac{\sum I(u,v)\, w}{\sum I(u,v)} \tag{11}$$

The line of sight vector in the spacecraft frame is therefore

$$\mathbf{\Lambda} = \begin{bmatrix} \bar{u} \\ \bar{w} \end{bmatrix} - \mathbf{R}_T \tag{12}$$

3. Orbital Dynamics of an Interceptor Spacecraft

The target asteroid can be modeled as a point mass in a standard heliocentric Keplerian orbit, as follows:

$$\begin{aligned} \dot{\mathbf{r}}_T &= \mathbf{v}_T \\ \dot{\mathbf{v}}_T &= \mathbf{g} \end{aligned} \tag{13}$$

where $\mathbf{r}_T$ and $\mathbf{v}_T$ are the position of the velocity vectors of the target and $\mathbf{g}$ is the gravitational acceleration due to the sun, expressed as

$$\mathbf{g} = -\frac{\mu_\odot \mathbf{r}_T}{r^3} \tag{14}$$

where $\mu_\odot$ is the solar gravitational parameter. Similarly, the motion of the spacecraft is described by

$$\begin{aligned} \dot{\mathbf{r}}_S &= \mathbf{v}_S \\ \dot{\mathbf{v}}_S &= \mathbf{g} + \mathbf{a} \end{aligned} \tag{15}$$

where $\mathbf{r}_S$ and $\mathbf{v}_S$ are the position of the velocity vectors of the spacecraft and $\mathbf{g}$ is again the gravitational acceleration due to the sun, and $\mathbf{a}$ is the control acceleration provided by the control thrusters. In this paper, a boldfaced symbol indicates a column matrix of a physical vector expressed in a chosen inertial reference frame.

In general, we have $\mathbf{g} = \mathbf{g}(\mathbf{r}, t)$. For some guidance problems the gravitational acceleration can be considered constant or negligible, but for asteroid terminal guidance missions, the gravitational acceleration must be considered a nonlinear function of position. There are some other disturbing accelerations that act on the spacecraft, such as radiation pressure and the gravitational acceleration due to the asteroid. However, intercept and rendezvous missions to small asteroids can neglect these.

The relative position of the spacecraft with respect to the target is then described by

$$\mathbf{r} = \mathbf{r}_S - \mathbf{r}_T \tag{16}$$

The equation of motion of the spacecraft with respect to the target becomes

$$\ddot{\mathbf{r}} = \mathbf{g} + \mathbf{a} \tag{17}$$

where $\mathbf{g}$ represents the sum of apparent gravitational accelerations on the target.

4. Guidance Laws

The guidance laws that will be used in the simulations will be presented here. A detailed derivation is not included in this paper, but can be found in [8].

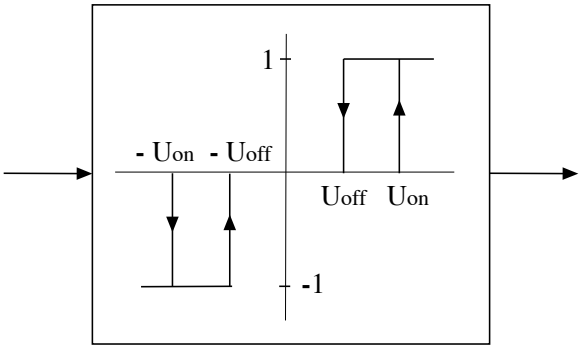


Figure 4: Schmitt trigger logic.

4.1. Classical Proportional Navigation (PN) Guidance

The classical proportional navigation (PN) guidance law commands accelerations perpendicular to the Line-of-Sight (LOS) direction. The PN guidance law is expressed as

$$a = nV_c\dot{\lambda} \tag{18}$$

where a is the acceleration command, n is the effective navigation ratio, a designer-tunable parameter, V_c is the closing velocity, and $\dot{\lambda}$ is the line-of-sight rate.

For implementation in 3D coordinates, it is easier to work with the LOS vector, defined as

$$\mathbf{\Lambda}(t) = \frac{\mathbf{r}(t)}{r(t)} \tag{19}$$

The line-of-sight rate is then denoted as $\dot{\mathbf{\Lambda}}(t)$. The PN guidance law is expressed in 3 dimensions as

$$\mathbf{a} = nV_c\dot{\mathbf{\Lambda}} = nV_c\begin{bmatrix}\dot{\Lambda}_x\\ \dot{\Lambda}_y\\ \dot{\Lambda}_z\end{bmatrix} \tag{20}$$

4.1.1. Pulsed Guidance

For simple asteroid intercept, the terminal velocity is not specified, and is assumed to be the closing velocity for PNG. The PNG law assumes that continuously variable thrust is available. For thrusters with no throttling ability, a different approach to guidance laws is needed. Two approaches to formulating guidance laws for fixed-thrust-level (on-off) guidance laws are PN-based guidance laws and predictive guidance laws.

The PNG law continuously generates acceleration commands to achieve intercept. Due to its feedback nature, PNG will continue to generate guidance commands until intercept is achieved. A special case of PNG occurs when the interceptor is on a direct collision course. When this is true, the guidance commands will be zero. When using PNG logic, then, an acceleration command of zero means that the interceptor is instantaneously on a direct collision course. This fact can be exploited to use PNG logic for constant-thrust engines.

Pulsed PNG (PPNG) logic computes the required acceleration commands from PNG, but applies them in continuous-thrust pulses. PPNG will "overshoot" the amount of correction specified by PNG, until the PNG command is zero. At that point, the interceptor is instantaneously on a collision course, and the engines are turned off. If there were no external accelerations or disturbances, the interceptor would continue on an interception course. Because of the acceleration due to the sun, this will not be the case, and a further engine firing will be required later as the interceptor "drifts" further and further from the straight-line collision path.

Pulsed guidance can be employed using the so-called Schmitt trigger, or other pulse-modulation schemes. Using a Schmitt trigger, acceleration commands are calculated by the PN guidance law as before. The trigger commands the divert thrusters to turn on once the commanded acceleration exceeds a certain magnitude, chosen by the designer, and off when the commanded acceleration reaches a designer-chosen cutoff. With traditional PN guidance the LOS rate must reach zero for a successful intercept. Therefore the second cutoff is typically selected as zero. The trigger control logic for pulsed proportional navigation guidance (PPNG) is shown in Figure 4.

4.2. Predictive Feedback Guidance Laws

A different class of guidance laws, which also uses on-off pulses, are the predictive guidance schemes. In this paper, two versions of time-varying state transition matrix (STM) guidance are considered. Predictive guidance

laws command a required velocity, $\mathbf{v}_{req}$. Subtracting the current velocity $\mathbf{v}$ from this gives the velocity to be gained, or $\Delta\mathbf{v}$. The simplest way to generate acceleration commands is to align the thrust vector with the $\Delta\mathbf{v}$ vector. When the desired velocity is achieved, $\Delta\mathbf{v}$ is zero and the engine is cut off. The velocity to be gained is

$$\Delta\mathbf{v} = \mathbf{v}_{req} - \mathbf{v} \tag{21}$$

For a given acceleration magnitude a, the direction of the thrust acceleration should be aligned with the velocity-to-be-gained vector, that is

$$\mathbf{a} = \frac{\Delta\mathbf{v}}{\|\Delta\mathbf{v}\|} \tag{22}$$

For asteroid intercept, with small changes in closing velocity, the relative position of the spacecraft at the end of the mission can be estimated as

$$\mathbf{r}\left(t_f\right) \cong \tilde{\mathbf{r}}_{t_f} = \Phi_1\left(t\right)\mathbf{r}\left(t\right) \tag{23}$$

where

$$\Phi_1 = \mathbf{I} + \frac{1}{2}\mathbf{G}\Delta t^2 = \begin{bmatrix} 1 + \frac{3\mu\Delta t^2 r_1^2}{2r_0^5} - \frac{\mu\Delta t^2}{2r_0^5} & \frac{3\mu\Delta t^2 r_1 r_2}{2r_0^5} \\ \frac{3\mu\Delta t^2 r_1 r_2}{2r_0^5} & 1 + \frac{3\mu\Delta t^2 r_2^2}{2r_0^5} - \frac{\mu\Delta t^2}{2r_0^3} \end{bmatrix} \tag{24}$$

where $r_0 = \sqrt{r_1^2 + r_2^2}$ and the subscript $\odot$ is dropped for simplicity.

4.2.1. Predictive Impulsive Guidance

For practical implementation, we adopt the following definitions:

$$\begin{aligned} V_c\left(t\right) &= \dot{r}\left(t\right) \\ \mathbf{\Lambda}\left(t\right) &= \frac{\mathbf{r}\left(t\right)}{r\left(t\right)} \\ \mathbf{\Lambda}_c\left(t\right) &= \frac{\tilde{\mathbf{r}}\left(t_f\right)}{\tilde{r}\left(t_f\right)} \\ t_{go} &= \Delta t \end{aligned} \tag{25}$$

Then, we have the predictive impulsive guidance law of the form

$$\Delta\mathbf{v} = V_c\mathbf{\Lambda}_c - \mathbf{v} \tag{26}$$

4.2.2. Kinematic Impulsive Guidance

By making some additional kinematic simplifications, we can arrive at the kinematic impulsive law of the form

$$\Delta\mathbf{v} = V_c\left(\mathbf{\Lambda}_c - t_{go}\dot{\mathbf{\Lambda}} - \mathbf{\Lambda}\right) \tag{27}$$

5. Leader-Follower Separation

The separation mechanism for the HAIV spacecraft is assumed to be like the mechanism for the Deep Impact mission. In that mission, separation pyros fired allowing a spring to uncoil and separate the two spacecraft at a speed of about 35 cm/s. In order to test the feasibility of employing such a separation mechanism, Monte Carlo tests are run with uncertainties in the direction of the separation. First, a coordinate system is established via the following equation:

$$\widehat{\mathbf{Q}} = \widehat{\mathbf{S}} \times \mathbf{\Lambda} \tag{28}$$

where $\mathbf{\Lambda}$ is the LOS direction, $\widehat{\mathbf{S}}$ is the unit sun vector, and $\widehat{\mathbf{Q}}$ completes the orthogonal system. Figure 5 shows the two angles that will be randomly drawn for Monte Carlo simulation. The angle α_1 gives deviation from the line-of-sight vector, while α_2 rotates around the LOS vector to give three-dimensional displacement.

The components of the velocity change to be imparted by the separation are given by

$$V_\Lambda = \cos\left(\alpha_1\right) \tag{29}$$

$$V_W = \sin\left(\alpha_1\right)\cos\left(\alpha_2\right) \tag{30}$$

$$V_S = \sin\left(\alpha_1\right)\sin\left(\alpha_2\right) \tag{31}$$

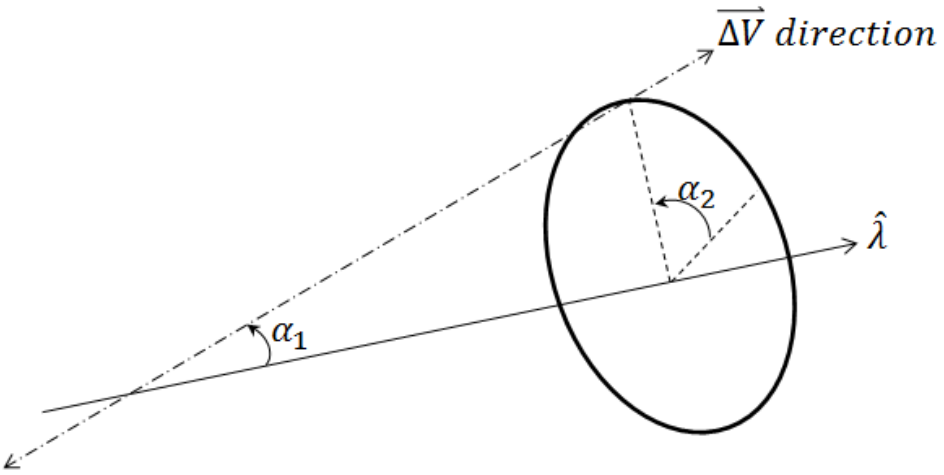


Figure 5: Separation angle definitions.

The total velocity change direction becomes

$$\widehat{\Delta \mathbf{v}} = V_\Lambda \mathbf{\Lambda} + V_W \widehat{\mathbf{Q}} + V_S \widehat{\mathbf{S}} \tag{32}$$

The velocity change imparted to the fore and aft bodies is found as

$$\mathbf{v}_1 = -|\Delta \mathbf{v}| \frac{m_2}{m_2 + m_1} \widehat{\Delta \mathbf{v}} \tag{33}$$

$$\mathbf{v}_2 = |\Delta \mathbf{v}| \frac{m_1}{m_2 + m_1} \widehat{\Delta \mathbf{v}} \tag{34}$$

The new velocities of the fore and aft bodies are finally expressed in the inertial coordinates as

$$\mathbf{v}_{ForeBody} = \mathbf{v}_{sc^-} + \mathbf{v}_2 \tag{35}$$

$$\mathbf{v}_{AftBody} = \mathbf{v}_{sc^-} + \mathbf{v}_1 \tag{36}$$

where $\mathbf{v}_{sc^-}$ is the velocity vector of the complete spacecraft before separation.

6. Imaging and Guidance Results

Two different types of missions, hypervelocity intercept and leader-follower separation, were simulated using the GPU-based optical navigation simulator described in this paper. The target is a model of 433 Eros, scaled down so that the longest dimension is 150 m. The target rotates with a period of 5 hours. The initial conditions are perturbed such that the spacecraft will miss the target by approximately 15 km without control input. The LOS rate, used for spacecraft acceleration commands, is found by numerically differentiating the LOS unit vector. Images are obtained by the camera at a maximum rate of 1 Hz. In this study, the LOS rate is found with a simple first-order finite differencing scheme. More refined algorithms and filters will be employed as the simulator is further developed. The simulations start at I-2 hours, which for most targets is approximately the threshold time for the sensors to detect any pixels.

6.1. Centroiding Algorithm

The simulated camera data rendering process described in Section 2 results in a simulated 1024 x 1024 pixel camera image. From this image, the center of brightness is found according to Equation (11). The center of brightness so found is based on the visible surface of the asteroid, with varying brightness according to surface geometry and orientation with respect to the sun. For most asteroid missions, the asteroid's shape and attitude at impact time will not be well-known in advance, so the center of brightness must serve for targeting purposes. However, this can vary greatly from the asteroid's center of mass, which is the ideal target for hypervelocity impact. Figure 6 shows an image of the Eros model during a simulation run. The magenta crosshair shows the calculated center of brightness, while the known center of mass is shown by the blue circle.

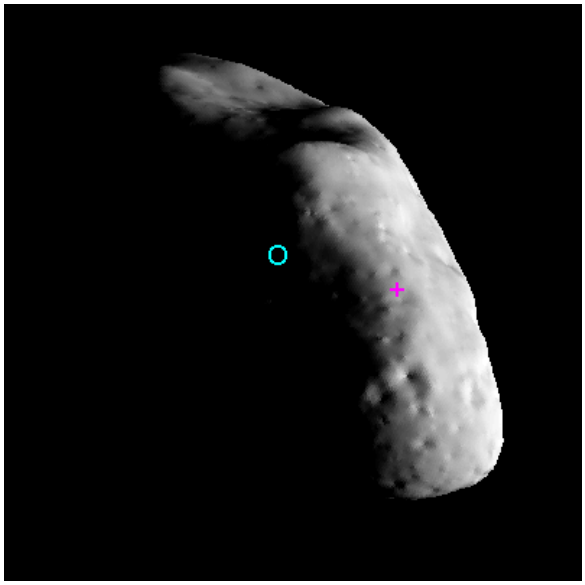


Figure 6: Known center of mass and calculated center of brightness.

6.2. Hypervelocity Intercept

Hypervelocity impact was simulated using both PN guidance and predictive guidance laws. To demonstrate the challenges of hitting such a small target at hypervelocity, Figure 7 shows the simulated camera images at four different times for a 150-m asteroid. The first is taken at I-30 minutes, when there are pixels illuminated, but not enough to see in the total field of view. At I-5 minutes the entire shape of the asteroid can be seen, but the longest axis is only on the order of 10 pixels. At I-2 seconds the target is easily seen, but still fits within the camera's field of view. A final image at I-$\frac{1}{8}$ seconds shows the asteroid taking up almost the entire field of view, verifying that impact in fact occurs.

The amount of information available to the sensors depends on the size of the asteroid. The HAIV is designed to hit targets as small as 50 meters. Figure 8 shows a comparison of the camera image at I-2 seconds for a scaled-down 50-m target and a scaled-up 450-m target. This figure underscores the small amount of sensor information available for smaller targets.

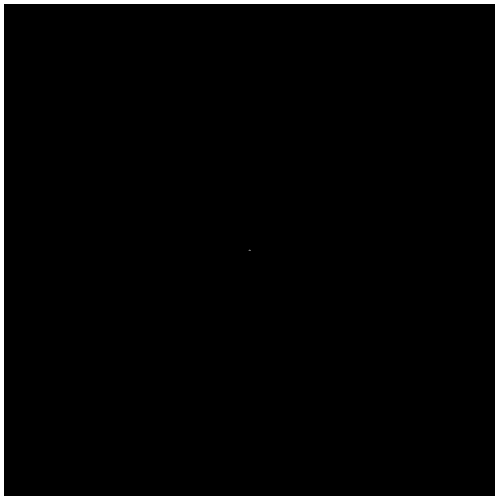
Different target sizes also have an effect on the centroiding algorithm. The location of the centroid in pixel coordinates can "jump" between frames for several reasons. One source of centroid jump is the growing size of the asteroid in the camera frame. As the spacecraft approaches the target, the apparent size of the target increases, with new pixels becoming visible on the edge of the figure. These pixels have an outsized effect on the centroid location, being the farthest from the current centroid. Target rotation also causes centroid jump, as shadowing effects and newly visible faces change the centroid location independent of camera resolution.

Hypervelocity impact missions using PN and pulsed PN guidance were simulated for a 150-m asteroid. Figures 9 and 10 show results for these simulations. The figures show the trajectories of the spacecraft and the target, the evolution of the centroid location relative to the center of the image, the control history, total Δv used, LOS angle, and LOS angle rate. In these simulations, a simple filtering scheme was included to ignore one-time jumps in the LOS rate, as these are more likely an artifact of the limited resolution of the camera than an actual physical effect. As the optical navigation simulator continues to be developed, more refined filtering will be included.

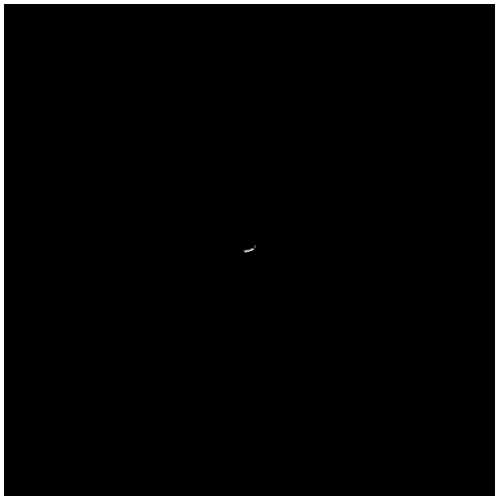
Hypervelocity impact missions using the kinematic impulsive guidance law were simulated for both a 50-m and a 450-m target asteroid. Figures 11 and 12 show results for these simulations. The figures show the trajectories of the spacecraft and the target, the evolution of the centroid location relative to the center of the image, the control history, total Δv used, LOS angle, and LOS angle rate. From the figures, it can be seen that centroid jump is more prominent for the 450-m target. Because the kinematic impulsive law generates guidance commands with one instantaneous value of LOS rate, the controls commanded and resulting Δv are similar for both cases. The predictive impulsive law produces similar results, and detailed results are omitted for the interest of saving space.

6.3. Leader-Follower Separation

The HAIV concept calls for two bodies, fore and aft, to separate shortly before impact and continue independently before hitting the target asteroid. It is assumed that the spacecraft consists of a 1200-kg fore body

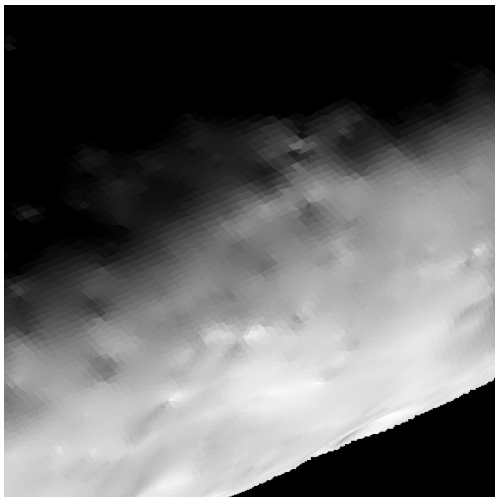


(a) I-30 minutes



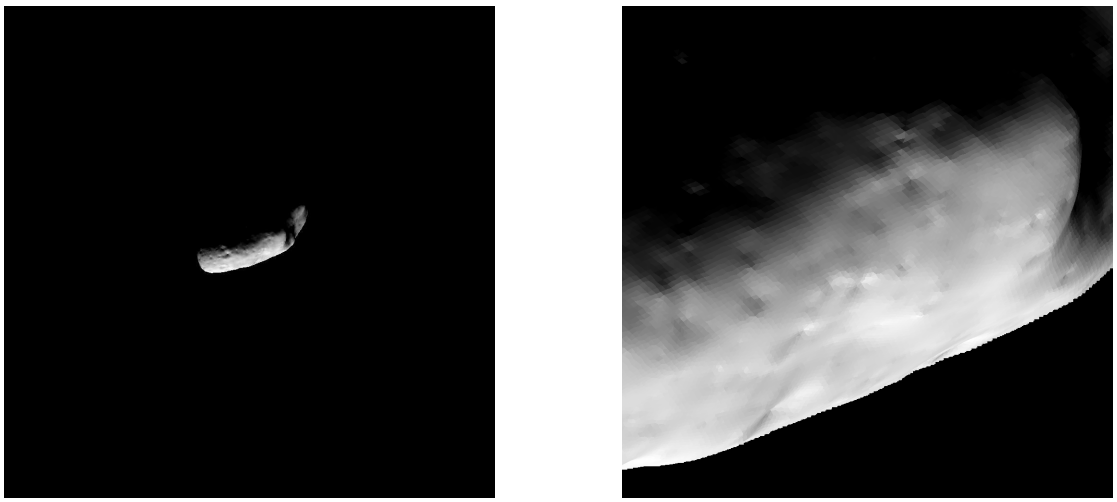
(b) I-5 minutes

(c) I-2 seconds



(d) I-$\frac{1}{8}$ seconds

Figure 7: Simulated camera images at different times-to-go in terminal guidance for a 150-m asteroid.



(a) 50-m target asteroid (b) 450-m target asteroid

Figure 8: Comparison of images at I-2 seconds for 50- and 450-m target asteroids.

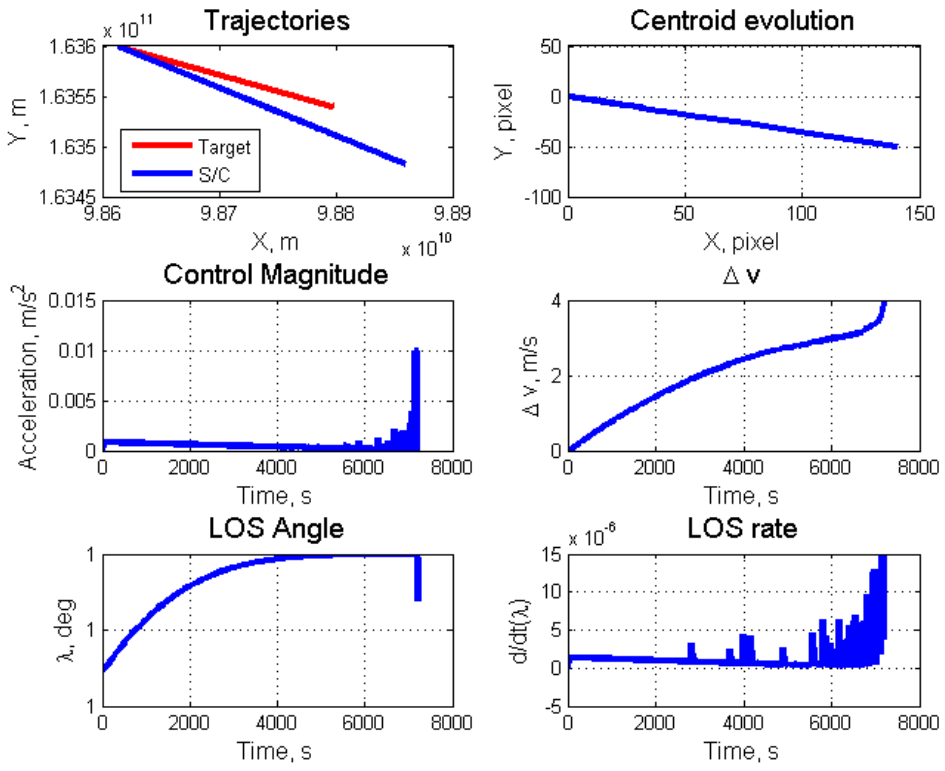


Figure 9: Simulation results for PN guidance of 150-m asteroid.

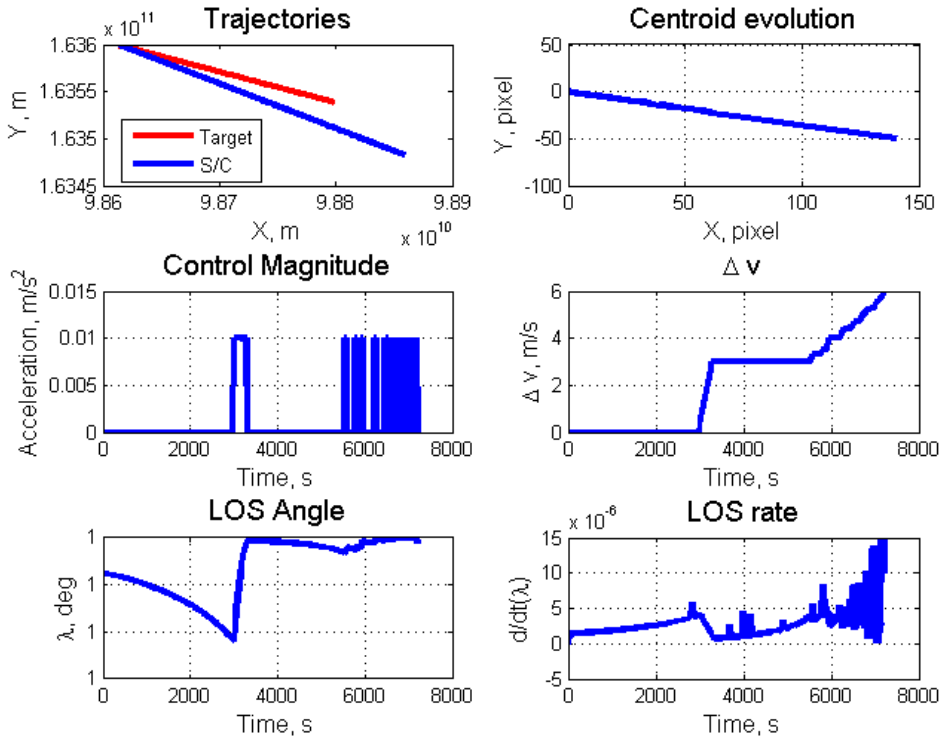


Figure 10: Simulation results for Pulsed PN guidance of 150-m asteroid.

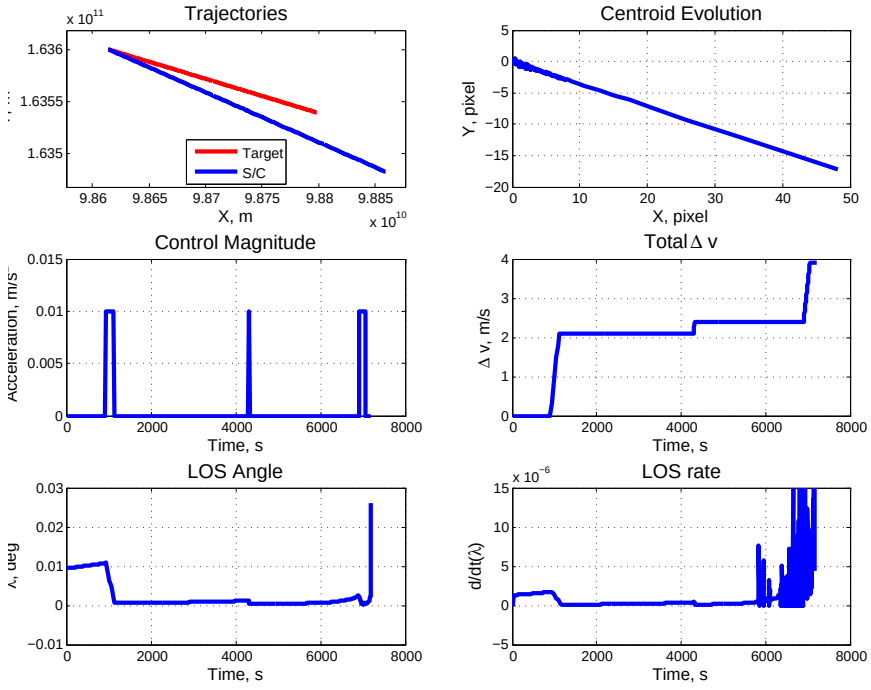


Figure 11: Simulation results for Kinematic Impulsive guidance of 50-m asteroid.

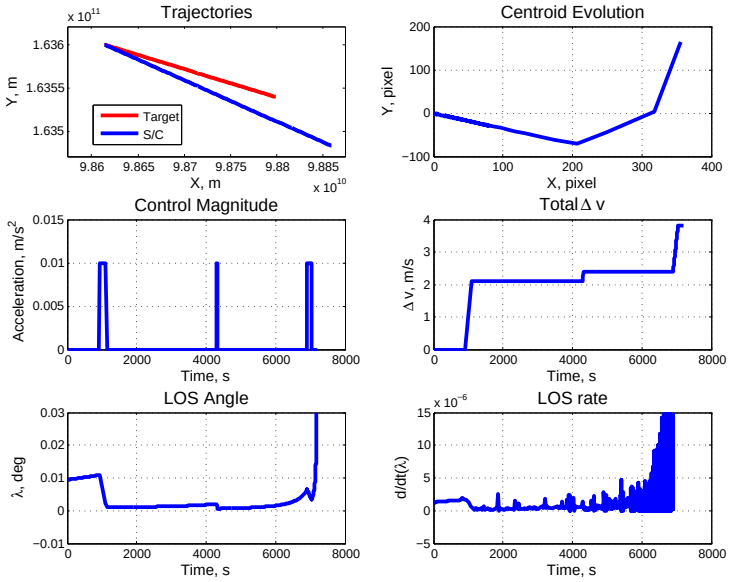


Figure 12: Simulation results for Kinematic Impulsive guidance of 450-m asteroid.

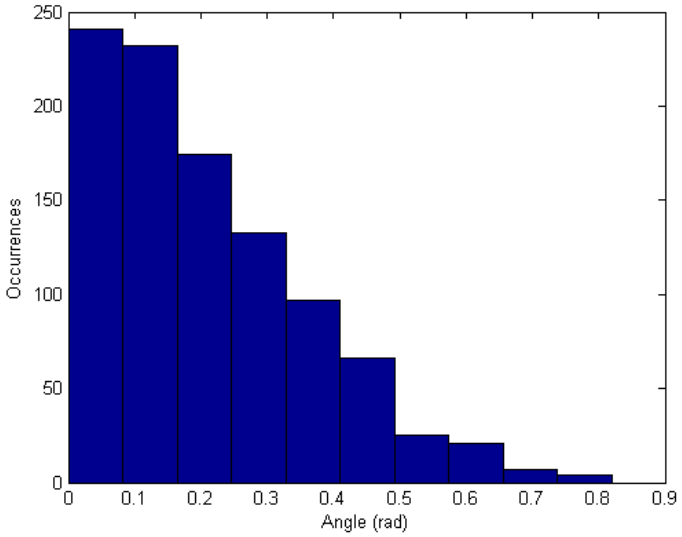


Figure 13: Histogram of α_1 for 1000 Monte Carlo simulations.

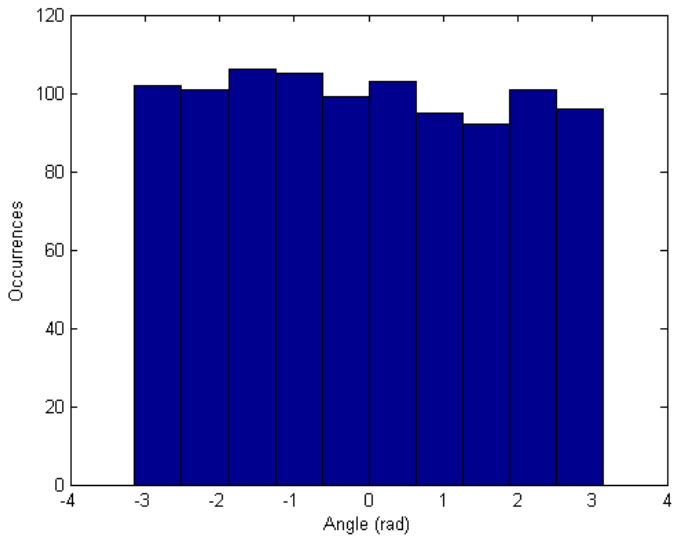


Figure 14: Histogram of α_2 for 1000 Monte Carlo simulations.

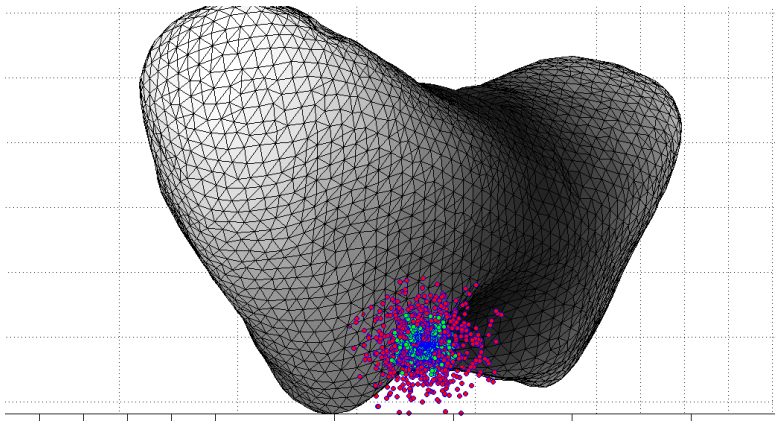


Figure 15: Impact distribution for leader (light blue) and follower (magenta).

and a 500-kg aft body. At I-60 seconds, the two bodies separate with a relative separation velocity of 35 cm/s. Once separation occurs, no further control accelerations are applied. In this simulation, 1000 Monte Carlo shots were run for a 150-m asteroid. The actual separation conditions were found in accordance with Section 5. The off-LOS angle, α_1, was selected from a half normal distribution, while the rotation angle, α_2, was selected from an even distribution from $-\pi$ to π. Figures 13 and 14 show histograms of the angles chosen for the Monte Carlo simulations.

With a separation instantaneously imparted, the fore and aft bodies continue until impacting (or missing) the asteroid. Figure 15 shows a scatter plot of leader hits in light blue and follower hits in magenta. For the extreme conditions tested, the follower misses the asteroid entirely in some cases.

7. Conclusions

In this paper, a new GPU-based optical navigation and terminal guidance simulation was described. A GPU-based simulator was used to simulate terminal phase hypervelocity intercept for an asteroid interceptor. The new GPU-based simulation tool provides realistic camera data for an optical navigation mission, given a model of a target asteroid. It is seen that various guidance laws can be used to achieve hypervelocity impact. Second, a late separation mechanism from the proposed HAIV mission was simulated to examine the feasibility of a two-body spacecraft performing a "double-hit" at a small target asteroid. Further development of this simulation tool will allow design and testing of navigation filters for optical navigation information, as well as higher-fidelity simulation of leader-follower separation.

Acknowledgments

This research work has been supported by a NASA Innovative Advanced Concepts (NIAC) Phase 2 Project entitled "An Innovative Solution to NASA's NEO Impact Threat Mitigation Grand Challenge and Its Flight Validation Mission Design." The authors would like to thank Dr. John (Jay) Falker, the NIAC Program Executive, for his support.

References

[1] Bhaskaran, S., Riedel, J. E., Synnott, S. P., "Autonomous Target Tracking of Small Bodies During Flybys," AAS/AIAA Spaceflight Mechanics Meeting, Maui, HI, 2004.

[2] Kubitschek, D.-G., "Impactor Spacecraft Encounter Sequence Design for the Deep Impact Mission," Space Systems Engineering Conference, Atlanta, GA, 2005.

[3] Pitz A., Kaplinger B., Vardaxis G., Winkler T., and Wie B., "Conceptual Design of a Hypervelocity Asteroid Intercept Vehicle (HAIV) and its Flight Validation Mission," AIAA 2012-4873, AIAA Guidance, Navigation, and Control Conference, Minneapolis, Minnesota, Aug 13-16, 2012.

[4] Raugh, A. C. "NEAR Collected Shape and Gravity Models." NEAR-A-5-COLLECTED MODELS-V1.0. N.p., 19 Apr. 2002. Web. http://sbn.psi.edu/pds/resource/nearbrowse.html.

[5] Thomas, P.C., et al., "Eros: Shape, Topography and Slope Processes," Icarus, Volume 155, pp. 18-37, 2002.

[6] Gouraud, H. "Continuous Shading of Curved Surfaces," IEEE Transactions on Computers, Volume C-20 (6): pp. 623629, 1971.

[7] Frauenholz, R. S., Bhat, R. S., Chesley, N., Mastrodemos, N., Owen Jr., W. M., and Ryne, M. S., "Deep Impact Navigation System Performance," *Journal of Spacecraft and Rockets*, Vol. 45, No. 1, 2008, pp. 39-56.

[8] Hawkins, M., Guo, Y., and Wie, B., "Spacecraft Guidance Algorithms for Asteroid Intercept and Rendezvous Missions," *International Journal of Aeronautical and Space Sciences*, Vol. 13, 2012, pp. 345-360.