

HIGH-FIDELITY SMALL BODY LANDER SIMULATIONS

*Stefaan Van wal and Daniel J. Scheeres**

University of Colorado Boulder
Celestial and Spaceflight Mechanics Laboratory
431 UCB, CO 80309, USA

Simon Tardivel

California Institute of Technology
Jet Propulsion Laboratory
CA 91109, USA

ABSTRACT

This paper provides an overview of the elements required for high-fidelity simulations of the motion of landers deployed to the surfaces of small bodies. Previous work on the motion of spherical landers is expanded to the general case of arbitrary shapes, which is challenging due to the high restitution of the necessary gravity, shape, and surface rock models. We implement established techniques that reduce the numerical burden of handling shape models by distinguishing between models used for gravity field evaluations and surface interactions. Significant attention is given to the implementation of a robust collision model for the non-eccentric collisions of arbitrary lander shapes. A C++ implementation of this software enables quick verification of deployment and surface mobility strategies in the presence of uncertainties in the lander release conditions, lander properties, and surface properties.

Index Terms— Small body, lander, rover, simulation, deployment, collision

1. INTRODUCTION

The small bodies of our Solar System have matured into common targets for large space missions. Specifically, the missions to these asteroids, comets, and small moons aim to address three distinct goals: first, the pristine condition of most small bodies provides insight into the early conditions of the Solar System, shedding light on its formation. Secondly, missions to small bodies provide a way of validating planetary defense strategies to deflect hazardous near-Earth objects. Finally, the analysis of small body (sub-)surfaces provides insight into the feasibility of in-situ resource utilization techniques.

Over the past two decades, the *NEAR-Shoemaker*, *Hayabusa*, and *Rosetta* spacecraft have established a core understanding of the origins, characteristics, and dynamics of small bodies. These mission mainly performed remote sensing operations about their respective targets; *Rosetta's Philae* lander further demonstrated the feasibility of small body lan-

ders [1]. In the coming decade, the *Hayabusa-2*, *OSIRIS-REx*, and *AIDA* missions will continue to expand our knowledge of small bodies and carry out remote sensing, sample return, and orbit deflection operations at their respective targets. *Hayabusa-2* also carries the *MASCOT* and *MINERVA* lander/rover, which will carry out surface mobility operations using momentum exchange mechanisms. This allows them to return scientific measurements from multiple sites on their target, asteroid Ryugu [2, 3]. Similar cubesat landers are proposed to be included on the *AIDA* mission [4].

One important aspect in the design of lander/rover operations is the release and deployment strategy used to deliver the craft to the surface of its target. This directly affects the lander hardware (*e.g.* battery) and the inherent risk to the mothership. Similarly, simulations of surface mobility operations will provide insight into the dynamics and controllability of a rover following deployment to the surface. Previous work has showed successful, low-risk deployment to a target along its unstable manifolds [5]. This strategy has been extensively verified with numerical simulations for spherical landers, using a statistical model of the surface rock distribution [6, 7]. While this provides a tractable approximation to the motion of landers with arbitrary shapes and establishes important simulation techniques, we must expand these simulations to be able to handle such shapes and generate accurate rock distributions.

In this work, we present an overview of the elements required to carry out high-fidelity simulations of these small body landers with arbitrary shapes. In Sec. 2, we briefly review the equations that express the lander motion. This motion is governed by gravitational forces, of which the evaluation is detailed in Sec. 3, along with techniques to make this evaluation computationally feasible. Similarly, we present techniques to convert the high-resolution surface shape and rock models into tractable subsets that allow for collision detection in Sec. 4. Sec. 5 then reviews a robust strategy for computing the impulsive forces that act in these collisions. The applications of these high-fidelity simulations are discussed in Sec. 6, and Sec. 7 finally provides some conclusions and future work directions for this research.

*This research was supported by NASA's SSERVI program (Institute for the Science of Exploration Targets) through institute grant number NNA14AB03A.

2. FRAMEWORK

Simulations of the motion of a lander in the neighborhood of a target small body first require set-up of the proper reference frames and equations of motions. For simplicity, this is done here for unitary bodies, though the expressions given are easily expanded to binary systems.

2.1. Reference Frames

In order to set up the state variables and equations of motion of the lander-target system, we first define three reference frames:

1. The inertial frame \mathcal{N} , with origin at the target center of mass and axes fixed inertially.
2. The rotating target frame \mathcal{T} , with origin at the target center of mass and axes fixed to the target.
3. The rotating lander frame \mathcal{L} , with origin at the lander center of mass and axes aligned with the lander principal inertia axes.

We may now define the state variables that describe the dynamical state of the spacecraft-target system. As surface interactions between the two bodies are computed relative to the target surface, the spacecraft motion is expressed primarily in the \mathcal{T} -frame. The system state consists of:

1. The spacecraft position \mathbf{r} , expressed in the \mathcal{T} -frame.
2. The spacecraft linear velocity $\dot{\mathbf{r}}$, expressed in the \mathcal{T} -frame.
3. The spacecraft attitude quaternion \mathbf{q} , expressed in the \mathcal{T} -frame.
4. The spacecraft angular velocity $\boldsymbol{\omega}$, expressed in the \mathcal{L} -frame.
5. The target attitude quaternion \mathbf{Q} , expressed in the \mathcal{N} -frame.
6. The target angular velocity $\boldsymbol{\Omega}$, expressed in the \mathcal{T} -frame.

2.2. Equations of Motion

The equations of motion of the state elements, in their respective reference frames, are equal to:

$$\frac{d\mathbf{r}}{dt} = \dot{\mathbf{r}} \quad (1)$$

$$\frac{d\dot{\mathbf{r}}}{dt} = \mathbf{g} - \boldsymbol{\Omega} \times \boldsymbol{\Omega} \times \mathbf{r} - 2\boldsymbol{\Omega} \times \dot{\mathbf{r}} + \mathbf{a}_E \quad (2)$$

$$\frac{d\mathbf{q}}{dt} = \frac{1}{2} \begin{bmatrix} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & -q_3 & q_2 \\ q_2 & q_3 & q_0 & -q_1 \\ q_3 & -q_2 & q_1 & q_0 \end{bmatrix} \begin{bmatrix} 0 \\ \boldsymbol{\omega} \end{bmatrix} \quad (3)$$

$$\frac{d\boldsymbol{\omega}}{dt} = -[\mathbf{I}]^{-1} (\boldsymbol{\omega} \times [\mathbf{I}]\boldsymbol{\omega}) \quad (4)$$

$$\frac{d\mathbf{Q}}{dt} = \frac{1}{2} \begin{bmatrix} Q_0 & -Q_1 & -Q_2 & -Q_3 \\ Q_1 & Q_0 & -Q_3 & Q_2 \\ Q_2 & Q_3 & Q_0 & -Q_1 \\ Q_3 & -Q_2 & Q_1 & Q_0 \end{bmatrix} \begin{bmatrix} 0 \\ \boldsymbol{\Omega} \end{bmatrix} \quad (5)$$

$$\frac{d\boldsymbol{\Omega}}{dt} = \mathbf{0} \quad (6)$$

where $[\mathbf{I}]$ is the inertia matrix of the lander, and where the target body is assumed to be in uniform rotation, as implied by Eq. 5. In Eq. 2, \mathbf{g} is the gravitational attraction of the target body, and \mathbf{a}_E is the external acceleration, which may include sources such as solar radiation pressure and accelerations during continued contact with the surface. Here, we set $\mathbf{a}_E = \mathbf{0}$, *i.e.* we ignore contact motion of a lander on the surface of its target, and only consider impulsive forces that occur during collisions.

These equations of motion are propagated numerically using a fifth-order Runge-Kutta method, with a fourth-order local truncation error estimate, *i.e.* the RK5(4) method. The integrator is further equipped with event detection capability in order to detect collisions between the lander and its target body. We implement this integrator in C++ to enable fast simulations of a large number of trajectories.

3. GRAVITY

As the time required for a lander to come to rest on the small body surface following release from a mothership is on the order of hours, the gravitational attraction of the target body is the dominant force on the lander. We therefore ignoring perturbing forces such as solar radiation pressure. In this section, we review our modeling of the small body gravitational field, and present techniques to mitigate numerical challenges inherent in these models. These techniques were developed with the specific purpose of lander/rover simulations by [6].

3.1. Polyhedron Model

The complex and irregular gravitational field of the target small body is implemented using the constant-density polyhedron model [8]. This model represents the small body shape through a collection of vertices \mathbf{P}_i , edges \mathcal{E}_{ij} , and facets \mathcal{F}_{ijk} . In order to evaluate the gravitational field of such a polyhedron, we first compute the outward-pointing normals $\hat{\mathbf{N}}_{\mathcal{F}_{ijk}}$ of each facet and $\hat{\mathbf{N}}_{\mathcal{E}_{ij}}$ of each edge with a ray-tracing technique. Using these normals, we compute the dyad of every edge and facet, as:

$$\mathbf{E}_{\mathcal{E}_{ij}} = \hat{\mathbf{N}}_{\mathcal{F}_{ijk}} \hat{\mathbf{N}}_{\mathcal{E}_{ij}}^T + \hat{\mathbf{N}}_{\mathcal{F}_{ijl}} \hat{\mathbf{N}}_{\mathcal{E}_{ji}}^T \quad (7)$$

$$\mathbf{F}_{\mathcal{F}_{ijk}} = \hat{\mathbf{N}}_{\mathcal{F}_{ijk}} \hat{\mathbf{N}}_{\mathcal{F}_{ijk}}^T \quad (8)$$

As the normals and dyads are invariant, they are pre-computed prior to simulations in order to reduce the computational overhead. When evaluating the gravitational field at



Fig. 1. Illustration of (left) a high-resolution shape model and (right) a low-resolution gravity model of asteroid Itokawa.

some position \mathbf{r} , we first compute the per-edge factor $L_{\mathcal{E}_{ij}}$ and per-facet factor $\omega_{\mathcal{F}_{ijk}}$ of all edges and facets of the target polyhedron:

$$L_{\mathcal{E}_{ij}} = \ln \left(\frac{r_i + r_j + e_{ij}}{r_i + r_j - e_{ij}} \right) \quad (9)$$

$$\omega_{\mathcal{F}_{ijk}} = 2 \arctan(\varpi) \quad (10)$$

with:

$$\varpi = \frac{\mathbf{r}_i \cdot (\mathbf{r}_j \times \mathbf{r}_k)}{r_i r_j r_k + r_i (\mathbf{r}_j \cdot \mathbf{r}_k) + r_j (\mathbf{r}_k \cdot \mathbf{r}_i) + r_k (\mathbf{r}_i \cdot \mathbf{r}_j)} \quad (11)$$

where \mathbf{r}_i is the vector from vertex \mathbf{P}_i to the position \mathbf{r} , *i.e.* $\mathbf{r}_i = \mathbf{r} - \mathbf{P}_i$, and $r_i = \|\mathbf{r}_i\|$. Using this, we may finally evaluate the gravitational potential U , gravitational attraction \mathbf{g} , and gravity gradient matrix $\mathbf{\Gamma}$ at the position \mathbf{r} as [8]:

$$U(\mathbf{r}) = \frac{1}{2} G \rho \sum_{\mathcal{E}_{ij}} \mathbf{r}_{\mathcal{E}_{ij}} \cdot \mathbf{E}_{\mathcal{E}_{ij}} \cdot \mathbf{r}_{\mathcal{E}_{ij}} \cdot L_{\mathcal{E}_{ij}} \quad (12)$$

$$- \frac{1}{2} G \rho \sum_{\mathcal{F}_{ijk}} \mathbf{r}_{\mathcal{F}_{ijk}} \cdot \mathbf{F}_{\mathcal{F}_{ijk}} \cdot \mathbf{r}_{\mathcal{F}_{ijk}} \cdot \omega_{\mathcal{F}_{ijk}}$$

$$\mathbf{g}(\mathbf{r}) = -G \rho \sum_{\mathcal{E}_{ij}} \mathbf{E}_{\mathcal{E}_{ij}} \cdot \mathbf{r}_{\mathcal{E}_{ij}} \cdot L_{\mathcal{E}_{ij}} \quad (13)$$

$$+ G \rho \sum_{\mathcal{F}_{ijk}} \mathbf{F}_{\mathcal{F}_{ijk}} \cdot \mathbf{r}_{\mathcal{F}_{ijk}} \cdot \omega_{\mathcal{F}_{ijk}}$$

$$\mathbf{\Gamma}(\mathbf{r}) = G \rho \sum_{\mathcal{E}_{ij}} \mathbf{E}_{\mathcal{E}_{ij}} \cdot L_{\mathcal{E}_{ij}} - G \rho \sum_{\mathcal{F}_{ijk}} \mathbf{F}_{\mathcal{F}_{ijk}} \cdot \omega_{\mathcal{F}_{ijk}} \quad (14)$$

where $\mathbf{r}_{\mathcal{E}_{ij}}$ and $\mathbf{r}_{\mathcal{F}_{ijk}}$ are the vectors from the position \mathbf{r} to respectively any vertex of edge \mathcal{E}_{ij} and facet \mathcal{F}_{ijk} . High-resolution small body shape models consist of several hundred thousand facets; evaluations of Eqs. 12 through 14 for a these models are therefore computationally expensive and significantly increase the runtime of simulations. Therefore, we implement two techniques that were developed by [6] and shown to be highly effective at reducing this computational load. For a more detailed discussion, the reader is referred to this work.

3.2. Linearization

The first technique applies a linearization of the gravity field. When the lander velocity or integration step size are small, the lander moves only a small distance between successive integration steps. As a result, we may approximate the gravitational field at some \mathbf{r} using a previous evaluation at some \mathbf{r}_0 , provided that the distance $\Delta r = \|\mathbf{r} - \mathbf{r}_0\|$ is small. When this is true, the gravitational potential and attraction at \mathbf{r} are approximately equal to:

$$U(\mathbf{r}) \simeq U(\mathbf{r}_0) + (\mathbf{r} - \mathbf{r}_0) \cdot \mathbf{g}(\mathbf{r}_0) \quad (15)$$

$$+ (\mathbf{r} - \mathbf{r}_0) \cdot \mathbf{\Gamma}(\mathbf{r}_0) \cdot (\mathbf{r} - \mathbf{r}_0)$$

$$\mathbf{g}(\mathbf{r}) \simeq \mathbf{g}(\mathbf{r}_0) + (\mathbf{r} - \mathbf{r}_0) \cdot \mathbf{\Gamma}(\mathbf{r}_0) \quad (16)$$

Through the selection of some Δr_{max} , we control the allowable error of this gravity field linearization. In verification tests, we find that setting Δr_{max} to 1/100th of the mean volumetric radius of a target body yields a relative error in the gravitational acceleration on the order of 10^{-5} .

3.3. Gravity Model Resolution

The second technique involves the reduction of the shape model used for gravity field evaluations. Indeed, the gravity field of a high-resolution model can be approximated by a reduced-resolution model of the same shape. In order to construct these models, we first triangulate the largest ellipsoid that fits entirely within the original shape model. The vertices of this ellipsoid are then projected outwards until they intersect the original shape model; these intersection points define the vertices of the reduced model. This model is then translated such that its center of mass matches that of the original model. Finally, as the volume of the new model differs from that of the original model, we adjust its density such that its total mass matches that of the original model. In Fig. 1, we illustrate a high-resolution shape model and its corresponding reduced gravity model. As we can see, the reduced model

maintains the general shape of the body, but has lost small local variations.

By controlling the number of vertices in the initial ellipsoid triangulation, we control the resolution of the final shape model and the corresponding gravity field error. In verification tests, we find that 162-vertex models yield a surface gravity error of approximately 1% for relatively smooth small body shapes. For more irregular models, a 642-vertex model yields the same error. Using these two techniques, the computational cost of gravity field evaluations is significantly reduced, while maintaining control over the associated error.

4. SURFACE

Lander/rover spacecraft operating on the surface of a small body dissipate energy due to interactions with the surface, primarily through collisions. As the shape and orientation of the local surface strongly affects these interactions, it is necessary to consider high-resolution shape models in computations of the surface interaction; simplified models may not accurately capture local topography that may serve as basins of attraction or rejection for objects moving on/over the surface. Additionally, it has been shown that rocks and boulders on the small body surface play an important role in the dissipation process, both from an energetic and a topographic point of view [6, 7]. A model of the surface rock distribution must therefore be included in addition to a high-resolution shape model.

Once again, this introduces issues with regard to computational burden, which are most significant when performing collision detection between a lander and the small body surface, as this detection requires distance computations between all features (vertices, edge, and facets) of both the lander and the small body. Fortunately, the burden can be significantly reduced using three techniques: the division of the ‘global’ small body surface into a number of smaller, ‘local worlds’, the procedural generation of rocks on these local worlds, and the use of bounding spheres.

4.1. Atlas

Prior to performing simulations, we follow the technique of [6] and construct the *atlas* of the target small body. This is done by creating a latitude-longitude grid, where individual cells contain those target surface features that fall within the cells’ latitude-longitude span. In order to establish a consistent grid size across the surface, we create one atlas relative to the rotation axis of the target (‘Z-atlas’), and a second atlas relative to an axis in its equatorial plane (‘X-atlas’), as illustrated in Fig. 2. Each of the two atlases spans $\pm 45^\circ$ of latitude, thus covering the entire small body surface with similar cell sizes.

When propagating the lander motion, the simulation continuously updates the ‘active’ local world based on the current lander latitude and longitude. This technique ensures the

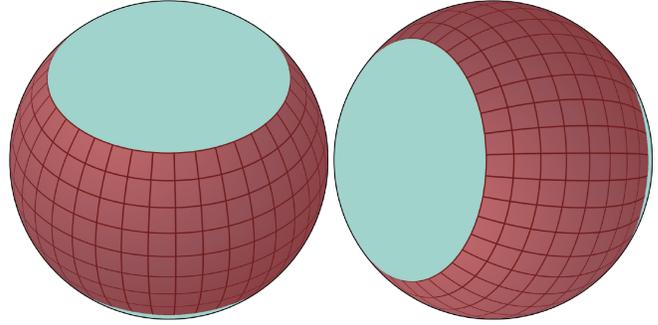


Fig. 2. Illustration of the grids that comprise (left) the Z-atlas and (right) the X-atlas [7].

correct surface features are active when the lander closely approaches the target surface, and enables collision detection with a high-resolution surface at low computational costs, as we only ever perform distance computations to a small subset of the total number of surface features.

4.2. Rock Distribution

This division of the small body surface into local worlds can be further exploited when generating rocks on the small body surface. The inclusion of such rock distributions is not a novel technique and has been shown in previous work to be an important factor in lander dissipation. In this work, the presence of rocks is accounted for through a stochastic model that imposes a degree of randomness of collisions between a lander and the small body surface [6]. While this strategy is numerically very effective and performs well at modeling the first few impacts of a lander trajectory, it has a number of drawbacks. Specifically, a stochastic model limits the surface slopes at which a lander can come to rest, and cannot produce repeatable collisions of a lander at the same surface location. These limitations are most prominent when the range of motion of the lander, *i.e.* its velocity, is small. In order to remove these limitations, we must include a ‘full’ rock model in which rocks are shaped and placed on the surface, and considered as surface features in collision detection. Depending on the minimum considered radius of a rock, the total number of rocks on a small body may well exceed the millions, making it infeasible to both generate and store these rocks in memory.

Instead, a single local world will contain only a manageable few hundred rocks; the exact number follows directly from the surface area of the local world and the statistical distribution of the rocks. Similarly, we can invert the rock size distribution and apply a random number generator to shape and place the rocks on the local world. By controlling the seed of the random number generator, we ensure that this procedural generation is consistent, *i.e.* that the same rocks are always created on the same facet. This holds for any arbitrary

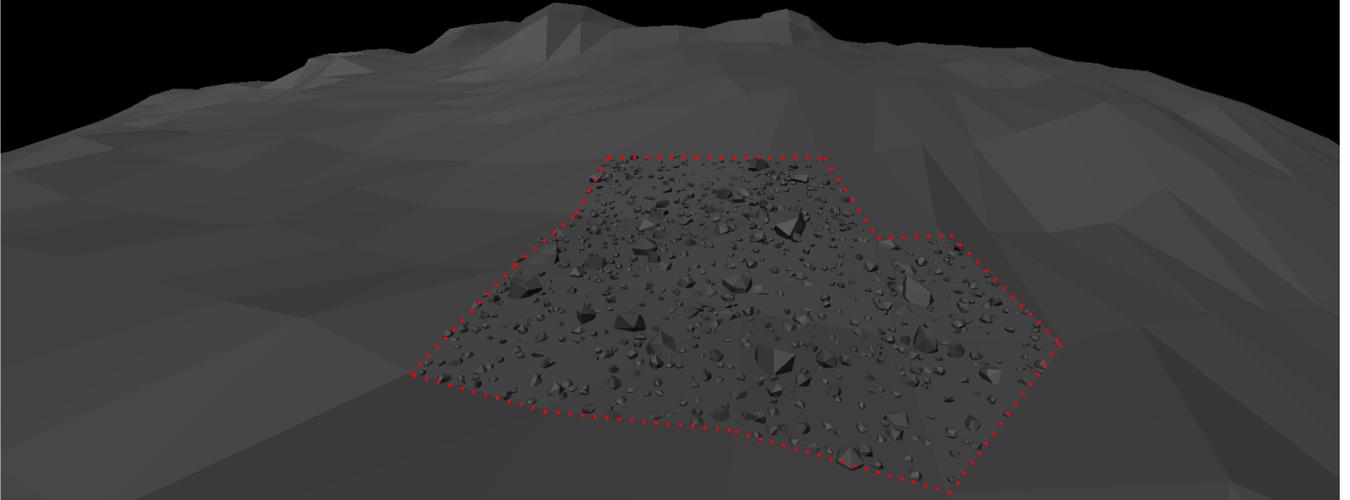


Fig. 3. Illustration of a small body surface with an active local world marked in red, with procedurally generated rocks.

rock model, as long as the rocks on a given facet are generated independently of the geometry surrounding that facet. Fig. 3 illustrates an arbitrary active world with generated rocks. The visualization software used to generate this and subsequent figures was developed in [6] and expanded in [7]. In this particular model shown, rocks are created as regular icosahedra with a superimposed random variation on their vertices, that are then partially sunk into the surface. This method of creating and shaping rocks is consistent with those used to generate the *a priori* stochastic model in [6], but has not been applied for procedural generation before.

4.3. Bounding Spheres

The efficiency of collision detection can be increased even further through the use of *bounding spheres*. Specifically, we define bounding spheres around each local world, as well as around the lander, each sphere encompassing its entire contents. Following this definition, collisions between the lander and the active local world are possible only when their bounding spheres intersect. Therefore, it is sufficient to compute the distance between the centers of the bounding spheres in most cases; the ‘full’ distance computation between all features of the lander and the local world is only carried out when the bounding spheres are found to intersect or contain one another. This process is illustrated in Fig. 4. In a similar process, we create bounding spheres for the surface rocks to enable efficient collision detection between an arbitrarily-shaped lander and several hundred rocks.

5. COLLISIONS

As discussed before, we propagate the motion of a lander in the small body environment using an integrator with event capability. Indeed, at every time step, the integrator computes

the minimum distance between the lander and the active local world, and examines whether the surface penetration constraint has been violated. If it has, the integrator converges on the instant just before the collision, and then handles the collision event before continuing with a successive integration arc.

Here, we review the methodology applied to handle these collisions. Although simple algebraic collision laws can robustly capture the impulsive forces present in collisions of a spherical body, these laws cannot be applied in the *non-eccentric* collisions of a body with an arbitrary shape where the contact point, body center of mass, and normal force, are not aligned. When this is the case, algebraic laws often result in energy increases or spin reversals in such collisions [9]. Instead, we follow the approach of [10], where a numerical integration is performed over the span of a collision to correctly compute the impulsive forces.

5.1. Geometry

Using the converged, pre-collision lander state, we define the collision frame \mathcal{C} , with the origin at the contact point \mathbf{H} between the lander and the small body surface, and orthonormal axes $\{\hat{\mathbf{t}}_1, \hat{\mathbf{t}}_2, \hat{\mathbf{n}}\}$. The $\hat{\mathbf{n}}$ axis is normal to the tangent plane of collision; the two tangential axes $\hat{\mathbf{t}}_1$ and $\hat{\mathbf{t}}_2$ (arbitrarily) span this plane. We further define the collision vector $\mathbf{c} = \mathbf{H} - \mathbf{r}$. In order to handle the collision, the quantities $\dot{\mathbf{r}}$, $\boldsymbol{\omega}$, and \mathbf{c} are transformed into the \mathcal{C} -frame. Assuming that the rigid body and surface are mutually impenetrable at the point of contact, the lander will be subject to a contact impulse \mathbf{P} , which affects its linear and angular velocity as:

$$M d\dot{\mathbf{r}} = d\mathbf{P} \quad (17)$$

$$[\mathbf{I}] d\boldsymbol{\omega} = \mathbf{c} \times d\mathbf{P} \quad (18)$$

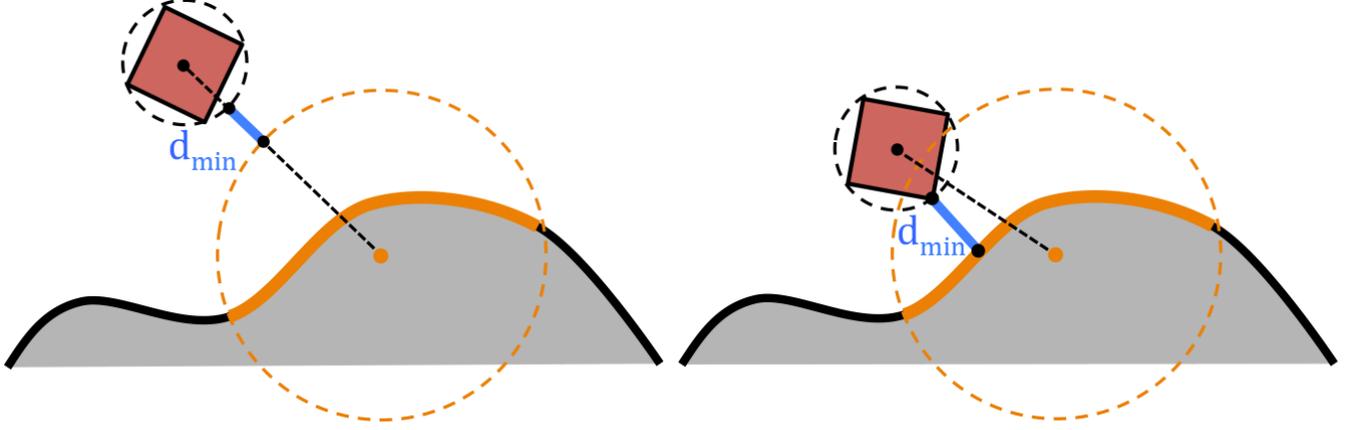


Fig. 4. Illustration of the lander-target minimum-distance computation when the bounding spheres (*left*) do not intersect and (*right*) do intersect.

where M is the mass of the lander and $[\mathbf{I}]$ is its inertia matrix, expressed in the \mathcal{C} -frame. The contact point velocity \mathbf{v} is determined by both the linear and angular velocity of the lander:

$$\mathbf{v} = \dot{\mathbf{r}} + \boldsymbol{\omega} \times \mathbf{c} \quad (19)$$

which is again expressed in the \mathcal{C} -frame, such that the third component of \mathbf{v} represents the normal contact point velocity, v_3 which must always be negative at the start of a collision. We will make use of the lander inverse inertia matrix, m^* , a symmetric matrix whose elements are defined as:

$$m_{ij}^* = M^{-1} \delta_{ij} + \sum_{k=1}^3 \sum_{l=1}^3 \sum_{m=1}^3 \sum_{n=1}^3 \varepsilon_{ikm} \varepsilon_{jln} I_{kl}^{-1} c_m c_n \quad (20)$$

where δ_{ij} is the Kronecker delta, ε_{ijk} is the permutation tensor, and I_{kl}^{-1} are the elements of the inverse of the inertia matrix¹ $[\mathbf{I}]$. Using this definition, we find for the elements of m^* :

$$\begin{cases} m_{11}^* = M^{-1} + c_3^2 I_{22}^{-1} + c_2^2 I_{33}^{-1} - 2c_2 c_3 I_{23}^{-1} \\ m_{12}^* = c_2 c_3 I_{13}^{-1} - c_3^2 I_{12}^{-1} + c_1 c_3 I_{23}^{-1} - c_1 c_2 I_{33}^{-1} \\ m_{13}^* = c_2 c_3 I_{12}^{-1} - c_2^2 I_{13}^{-1} - c_1 c_3 I_{22}^{-1} + c_1 c_2 I_{23}^{-1} \\ m_{22}^* = M^{-1} + c_3^2 I_{11}^{-1} + c_1^2 I_{33}^{-1} - 2c_1 c_3 I_{13}^{-1} \\ m_{23}^* = c_1 c_3 I_{12}^{-1} - c_2 c_3 I_{11}^{-1} - c_1^2 I_{23}^{-1} + c_1 c_2 I_{13}^{-1} \\ m_{33}^* = M^{-1} + c_2^2 I_{11}^{-1} + c_1^2 I_{22}^{-1} - 2c_1 c_2 I_{12}^{-1} \end{cases} \quad (21)$$

The remaining elements are found by symmetry, *i.e.* $m_{21}^* = m_{12}^*$, $m_{31}^* = m_{13}^*$, and $m_{32}^* = m_{23}^*$.

¹Not to be confused with the inverse of the elements of the inertia matrix.

5.2. Normal Force

During the collision, the lander is subject to a normal impulse that enforces the non-penetration constraint. Assuming that the contact region is infinitesimally small and nondeformable, it can be shown that the normal impulse must always be compressive and monotonously increases during the (infinitesimally short) contact period. As such, we may resolve the collision in terms of the normal impulse p , by using it as a time-like variable for integration. It is noted that, due to the definition of the collision frame, the normal impulse $p = P_3$.

A collision consists of two phases: compression and restitution. During compression, the normal impulse reduces the normal contact point velocity v_3 to zero. In this phase, some of the lander's energy is removed by the work W_c done by the normal force, and stored in the lander. The restitution phase begins when the contact point reaches $v_3 = 0$, during which the normal force continues to act and returns part of the stored energy through the work W_r . The 'duration' of a collision in terms of the total normal impulse provided is determined by the *energetic* coefficient of restitution, $0 \leq e \leq 1$, which is defined as:

$$e^2 = \frac{W_r}{W_c} \quad (22)$$

We note that this definition is different from the more commonly used *kinetic* coefficient of restitution, which is defined in terms of the contact point normal velocity.

When resolving a collision, we thus first propagate the respective equations of motion until $v_3 = 0$ and compute the work done by the normal force during compression, as the area under the $v_3(p)$ curve. Using the energetic coefficient of restitution, we can then determine the work done by the normal force during restitution, which determines when the collision will terminate.

5.3. Presence of Sliding

In addition to the normal force, a lander will also be subject to a Coulomb friction force that attempts to drive the tangential contact point velocity to zero. In order to determine the magnitude of this friction force and related impulse, we make use of the *slip velocity*, s :

$$s = \sqrt{v_1^2 + v_2^2} \quad (23)$$

where v_1 and v_2 are the tangential contact point velocity components in respectively the directions of $\hat{\mathbf{t}}_1$ and $\hat{\mathbf{t}}_2$. When $s \neq 0$, we say that the contact point is in *slip*; when $s = 0$, we say that it is in *stick*. When slip occurs, the direction of slip can be defined by the angle ϕ measured in the tangent plane, from $\hat{\mathbf{t}}_1$, as:

$$\phi = \arctan\left(\frac{v_2}{v_1}\right) \quad (24)$$

where, numerically, we make use of the `atan2` function to determine the proper quadrant of ϕ . Respectively, the two tangential velocities can be expressed in terms of the slip velocity and angle as:

$$\begin{aligned} v_1 &= s \cos \phi \\ v_2 &= s \sin \phi \end{aligned} \quad (25)$$

5.4. Collision Equations of Motion

The state of the lander during collision is described by the contact impulse \mathbf{P} , the contact point velocity \mathbf{v} , and the lander velocities $\dot{\mathbf{r}}$ and $\boldsymbol{\omega}$. The latter two can be integrated following Eqs. 17 and 18; the equations of motion that govern \mathbf{P} and \mathbf{v} must be selected depending on the slip/stick state of the contact point, and are described here.

5.4.1. Slip

When the contact point is in slip, the contact impulse can be propagated with the following equations of motion:

$$\mathbf{dP} : \begin{cases} \frac{dP_1}{dp} = -\mu \cos \phi = -\frac{\mu v_1}{\sqrt{v_1^2 + v_2^2}} \\ \frac{dP_2}{dp} = -\mu \sin \phi = -\frac{\mu v_2}{\sqrt{v_1^2 + v_2^2}} \\ \frac{dP_3}{dp} = 1 \end{cases} \quad (26)$$

where $0 \leq \mu \leq 1$ is the coefficient of friction. Correspondingly, the contact point velocity varies as:

$$\mathbf{dv} : \begin{cases} \frac{dv_1}{dp} = -\mu m_{11}^* \cos \phi - \mu m_{12}^* \sin \phi + m_{13}^* \\ \frac{dv_2}{dp} = -\mu m_{21}^* \cos \phi - \mu m_{22}^* \sin \phi + m_{23}^* \\ \frac{dv_3}{dp} = -\mu m_{31}^* \cos \phi - \mu m_{32}^* \sin \phi + m_{33}^* \end{cases} \quad (27)$$

Eqs. 26 and 27 must be integrated simultaneously to correctly resolve the collision. When the coefficient of friction is

sufficiently large, slip may halt during either compression or restitution, *i.e.* $s \rightarrow 0$.

When this happens, the contact point will either stick for the remainder of the collision, or immediately resume slip in some direction $\hat{\phi}$. To determine which of the two cases occurs, we compute the critical coefficient of friction, $\bar{\mu}$, which provides the minimum value of μ required for permanent stick:

$$\bar{\mu} = \frac{\sqrt{\alpha^2 + \beta^2}}{\gamma} \quad (28)$$

where:

$$\begin{cases} \alpha = m_{11}^* m_{23}^* - m_{12}^* m_{13}^* \\ \beta = m_{22}^* m_{13}^* - m_{12}^* m_{23}^* \\ \gamma = m_{11}^* m_{22}^* - m_{12}^* m_{12}^* \end{cases} \quad (29)$$

5.4.2. Continued stick

If $\mu \geq \bar{\mu}$, the contact point will remain in stick under the influence of a friction impulse $\bar{\mu} \cdot dp$ in a direction $\bar{\phi} - \pi$, such that the contact point velocity equations of motion become:

$$\mathbf{dv} : \begin{cases} \frac{dv_1}{dp} = 0 \\ \frac{dv_2}{dp} = 0 \\ \frac{dv_3}{dp} = -\mu m_{31}^* \cos \bar{\phi} - \mu m_{32}^* \sin \bar{\phi} + m_{33}^* \end{cases} \quad (30)$$

where the angle $\bar{\phi}$ can be computed as:

$$\bar{\phi} = \arctan\left(\frac{\alpha}{\beta}\right) \quad (31)$$

which is only a function of the collision geometry and the mass properties of the rigid body.

Correspondingly, the impulse equations become:

$$\mathbf{dP} : \begin{cases} \frac{dP_1}{dp} = \bar{\mu} \cos(\bar{\phi} - \pi) \\ \frac{dP_2}{dp} = \bar{\mu} \sin(\bar{\phi} - \pi) \\ \frac{dP_3}{dp} = 1 \end{cases} \quad (32)$$

5.4.3. Slip reversal

If instead $\mu > \bar{\mu}$, slip will resume immediately after reaching the $s = 0$ point. In this second phase of slip, the direction of slip $\hat{\phi}$ will be constant, and can be found as the root of $h(\mu, \phi) = 0$ for which $g(\mu, \phi) > 0$. The two functions are defined as:

$$\begin{aligned} g(\mu, \phi) &= m_{13}^* c \phi + m_{23}^* s \phi - \mu m_{11}^* c^2 \phi \\ &\quad - 2\mu m_{12}^* s \phi c \phi - \mu m_{22}^* s^2 \phi \end{aligned} \quad (33)$$

$$\begin{aligned} h(\mu, \phi) &= -m_{13}^* s \phi + m_{23}^* c \phi \\ &\quad + \mu (m_{11}^* - m_{22}^*) s \phi c \phi + \mu m_{12}^* (s^2 \phi - c^2 \phi) \end{aligned} \quad (34)$$

where $s\phi = \sin\phi$ and $c\phi = \cos\phi$. If it has been determined that slip reversal occurs, the collision can be propagated using the equations of motion for slip, as given by Eqs. 26 and 27, by simply substituting $\phi = \hat{\phi}$.

5.5. Numerical Issues

Inspection of Eq. 26 reveals a singularity that occurs when $s \rightarrow 0$, that is, when the contact point approaches stick. Indeed, when $s \rightarrow 0$, also $v_1 \rightarrow 0$ and $v_2 \rightarrow 0$, such that the quantities ϕ , dP_1/dp , and dP_2/dp become ill-defined. A numerical integrator will therefore be unable to properly reach the point of stick where $s = 0$.

To resolve this issue, we use the method by [11] and introduce the 'stretching' variable τ , which is related to the normal impulse through:

$$d\tau = \frac{\mu}{\sqrt{v_1^2 + v_2^2}} dp = \frac{\mu}{s} dp \Rightarrow \frac{dp}{d\tau} = \frac{s}{\mu} \quad (35)$$

This *blow-up transformation* effectively rescales the normal impulse such that the singularity at $s \rightarrow 0$ is shifted to $\tau \rightarrow \infty$, allowing for a stable integration. To avoid further numerical issues that occur with the computation of s from v_1 and v_2 , we also perform a change of variables from the Cartesian (v_1, v_2) to the polar (s, ϕ) coordinates. The equations of motion for the contact point velocity thus become:

$$\begin{cases} \frac{ds}{d\tau} = s \left(-m_{11}^* c^2 \phi - (m_{12}^* + m_{21}^*) c\phi s\phi \right. \\ \quad \left. - m_{12}^* s^2 \phi + \frac{m_{13}^*}{\mu} c\phi - \frac{m_{23}^*}{\mu} s\phi \right) \\ \frac{d\phi}{d\tau} = -m_{21}^* c^2 \phi + m_{12}^* s^2 \phi + (m_{11}^* - m_{22}^*) c\phi s\phi \\ \quad + \frac{m_{23}^*}{\mu} c\phi - \frac{m_{13}^*}{\mu} s\phi \\ \frac{dv_3}{d\tau} = \frac{dv_3}{dp} \cdot \frac{dp}{d\tau} = s \left(-m_{31}^* c\phi - m_{32}^* s\phi + \frac{m_{33}^*}{\mu} \right) \end{cases} \quad (36)$$

The corresponding impulse equations, using τ as time-like variable, are found with the chain rule as:

$$\begin{cases} \frac{dp_1}{d\tau} = \frac{dp_1}{dp} \cdot \frac{dp}{d\tau} = -s \cos \phi \\ \frac{dp_2}{d\tau} = \frac{dp_2}{dp} \cdot \frac{dp}{d\tau} = -s \sin \phi \\ \frac{dp_3}{d\tau} = \frac{dp_3}{dp} \cdot \frac{dp}{d\tau} = \frac{s}{\mu} \end{cases} \quad (37)$$

While this technique was introduced by [11] in order to study the behavior of colliding objects near the point of stick, we find that it is highly efficient at dealing with the singularity at $s \rightarrow 0$. As it is impossible to integrate until $\tau \rightarrow \infty$, we must select some limiting value at which to terminate the integration. This is most easily done by selecting some small ϵ , such that the onset of stick occurs when $s \leq \epsilon$. The choice of a particular value for ϵ will affect the computational burden of carrying out the collision, but also the accuracy of the impulse \mathbf{P} that is required to drive the slip velocity to zero. In practice, we find that a value of $\epsilon \sim 10^{-10}$ produces accurate

results at low computational effort, when this methodology is implemented numerically in C++.

While the use of τ as a time-like variable is highly effective in increasing the accuracy with which the point of stick is reached, it can no longer be used once the contact point is in stick. This is easily understood by inspecting Eq. 35, which becomes undefined when $s = 0$, as in that case $dp/d\tau = 0$ and $d\tau/dp$. As such, we must continue integration of the collision with p as time-like variable, once stick has indeed been achieved. This requires changing back from polar to Cartesian coordinates.

Using this model, we can robustly handle collisions between a lander with an arbitrary shape and the surface of a small body. While collisions make up the bulk of surface interactions between a lander and a small body, it is also possible for the lander to be in continuous contact with the surface, which occurs when the normal contact point velocity is zero. This regime of motion has not yet been implemented, in other words, simulations are terminated when the normal contact velocity drops below some $v_{3,min}$ following a collision. It is noted that for arbitrary lander shapes, this regime only occurs for a very brief time at the very end of a deployment sequence, and therefore does not notably affect the final settling position and time of a lander.

6. RESULTS

By combining all of the elements described in the previous sections, we obtain the capability of performing simulations of lander/rover spacecraft in the small body environment and account for all its intricacies. This simulation capability has a wide range of applications, the most important of which is the verification of a release strategy.

Due to the challenges in spacecraft navigation around small bodies, as well as variations in any release mechanisms, the position and velocity of a lander/rover at the moment of release from its mothership is inherently uncertain. As our simulation software is capable of generating large numbers of simulations at moderate computational cost, it is possible to carry out sets of Monte Carlo simulations in which the release conditions are randomly distributed in agreement with the expected uncertainties. Analysis of the resulting deployment trajectories will then reveal whether the selected strategy (*e.g.* nominal release altitude and velocity) results in successful deployments. Additionally, we can obtain statistics about trajectory characteristics such as the duration of deployment, the reachable landing zone, and the magnitude of the impulsive forces acting on the lander; elements that may in turn affect hardware selection of the lander/rover.

In addition to uncertainties in the release conditions, these simulations also allow for uncertainties in the surface and lander properties. For example, it is possible to study the effect of variations in the coefficients of restitution and friction on the resulting surface spread and deployment time. Addition-

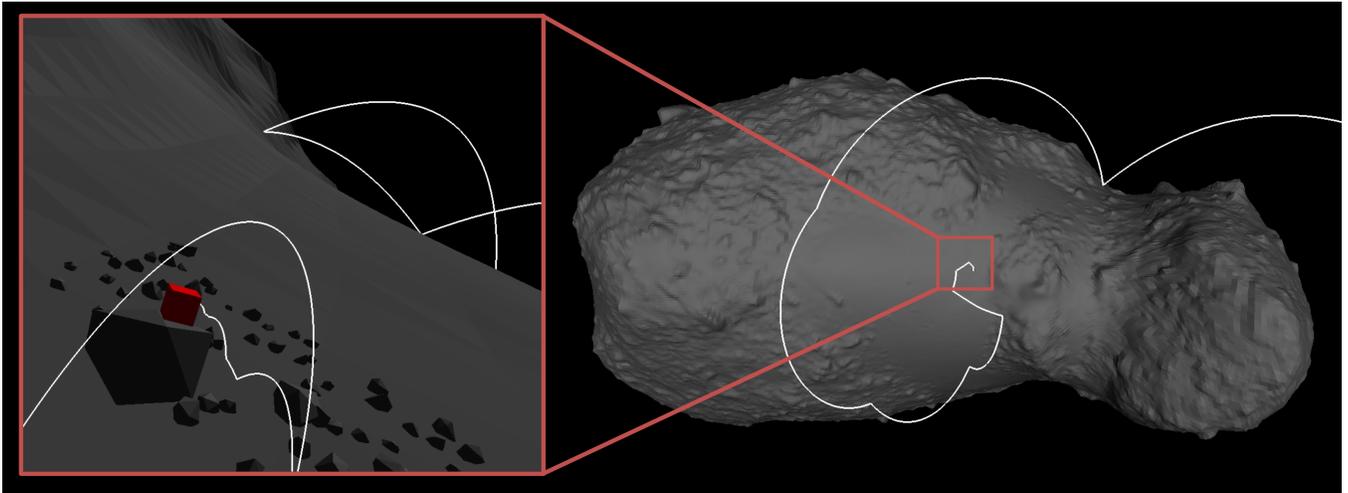


Fig. 5. Sample deployment of a cube-shaped lander to asteroid Itokawa.

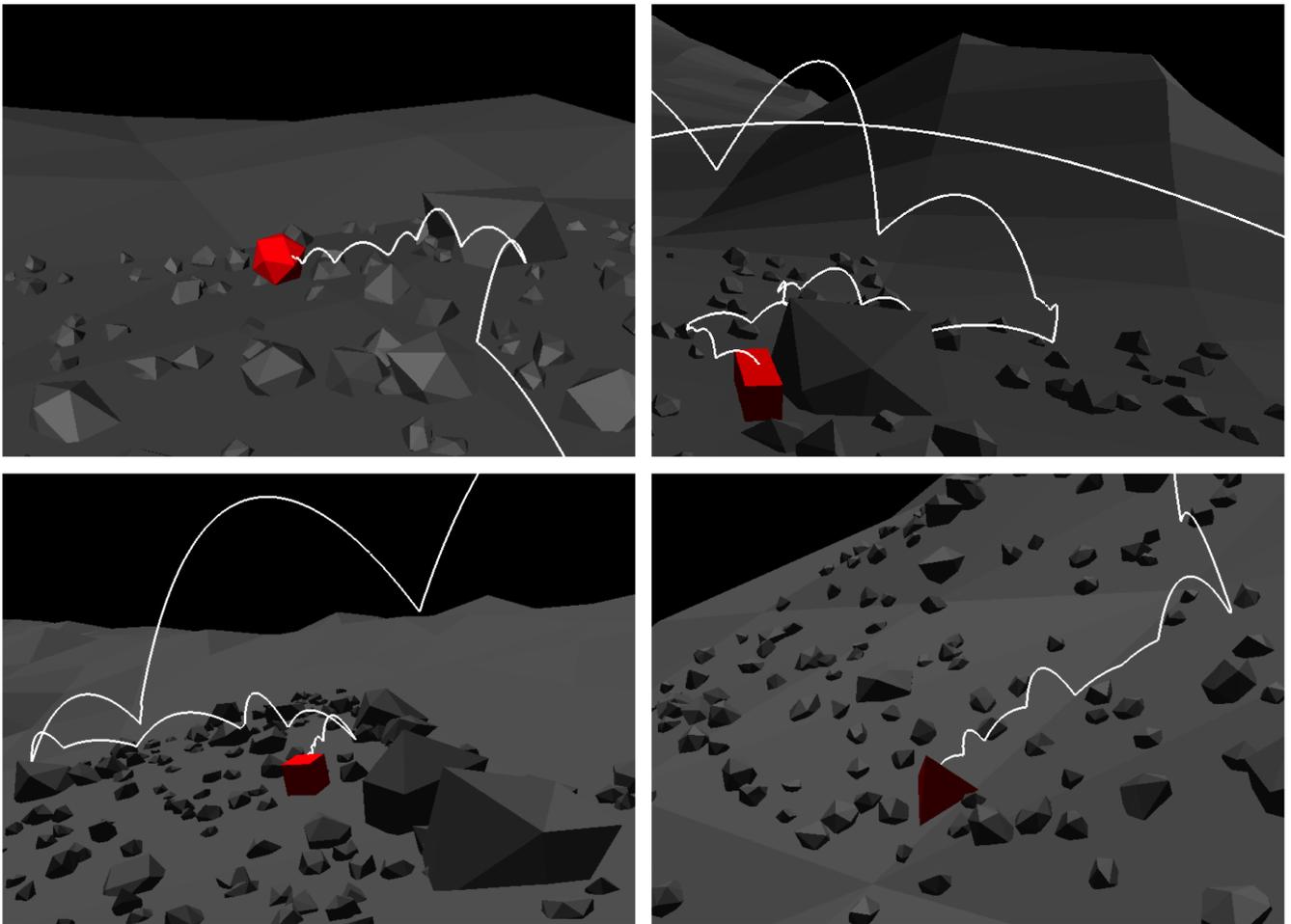


Fig. 6. Sample deployment of various lander shapes.

ally, we can study the effect of variations in the applied rock model. All these variations can be applied to the small body in both a global sense and a local sense. For example, the coefficient of restitution could be assumed to constant across the entire small body surface, or certain latitudes may be given a higher coefficient than others. These techniques enable mission designers to take into account all available information on a target body when designing a deployment strategy, and predict the effects of other parameters that are inherently difficult to estimate on ground without prior observations from spacecraft. As an illustration, Fig. 5 shows a sample deployment of a cube-shaped lander to asteroid Itokawa.

A similar but different application is the shape optimization of a lander/rover spacecraft with respect to certain mission goals. This can be done by varying this shape within certain constraints, such as maintaining a constant total volume, and investigating the resulting deployment trajectories. As an example, this allows us to choose a lander shape that will guarantee the fastest energy dissipation rate, *i.e.* that guarantees the lander comes to rest on the small body surface as quickly as possible. To illustrate, Fig. 6 illustrates sample deployments of landers with the same total mass and volume, but different shapes.

Finally, a similar approach can be used to design and verify control strategies for surface mobility operations where a rover uses momentum exchange mechanisms to perform hopping arcs across the small body surface. Once again, we can design both the hopping strategy and the spacecraft shape with respect to given mission goals.

7. CONCLUSIONS AND FUTURE WORK

In this paper, we have reviewed established methodologies to perform simulations of the deployment of lander/rover spacecraft, and expanded them with the capability to handle spacecraft with arbitrary shapes and procedurally generate surface rock distributions. The numerical burden of evaluating high-resolution gravity, surface, and rock models is reduced to acceptable levels using a variety of strategies, all of which grant the user direct control of the error (if any) inherent in these approaches. Significant attention has been given to the implementation of a robust collision model capable of handling the non-eccentric impacts between an arbitrary rigid lander and the small body surface. These simulations allow for the design and verification of deployment and surface mobility strategies, as well as the optimization of lander shapes with regard to given mission goals.

In future work, we plan to carry out such optimization, in particular with the goal of minimizing the settling time and surface displacement of lander spacecraft. Additionally, we hope to carry out analyses for the landers that are planned on future missions to small bodies, such as the ones listed at the start of this paper. With regards to further software development, we plan to include handling of the aforementioned

contact motion; the brief period of motion at the very end of a deployment in which the lander dissipates its final velocities and comes to a full stop on the small body surface.

8. REFERENCES

- [1] J. Biele, S. Ulamec, M. Maibaum, R Roll, L Witte, E Jurado, P Muñoz, W. Arnold, H.-U. Auster, and C. Casas, “The landing (s) of philae and inferences about comet surface mechanical properties,” *Science*, 2015.
- [2] J. Reill, H.J. Sedlmayr, P. Neugebauer, M. Maier, E. Krämer, and R. Lichtenheldt, “MASCOT-Asteroid Lander with Innovative Mobility Mechanism,” in *Proc. of the 13th Symposium on Advanced Space Technologies in Robotics and Automation, Noordwijk, The Netherlands*, 2015.
- [3] T. Kubota and T. Yoshimitsu, “Intelligent rover with hopping mechanism for asteroid exploration,” in *6th International Conference on Recent Advances in Space Technologies, Istanbul, Turkey*, 2013.
- [4] P. Michel, A.F. Cheng, and M. Küppers, “Asteroid Impact and Deflection Assessment (AIDA) mission: science investigation of a binary system and mitigation test,” in *European Planetary Science Congress, Nantes, France*, 2015.
- [5] S. Tardivel, C. Lange, S. Ulamec, and J. Biele, “The deployment of Mascot-2 to Didymoon,” in *26th AAS/AIAA Space Flight Mechanics Meeting, Napa, CA*, 2016.
- [6] S. Tardivel, D.J. Scheeres, P. Michel, S Van wal, and P. Sánchez, “Contact motion on surface of asteroid,” *Journal of Spacecraft and Rockets*, 2014.
- [7] S. Van wal, “The ballistic deployment of asteroid landers,” M.S. thesis, Delft Univ. of Technology, 2015.
- [8] 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,” *Cel. Mech. and Dyn. Astr.*, 1996.
- [9] A. Chatterjee and A. Ruina, “A new algebraic rigid-body collision law based on impulse space considerations,” *Journal of Appl. Mech.*, 1998.
- [10] W.J. Stronge, *Impact mechanics*, Cambridge University Press, 2004.
- [11] V. Bhatt and J. Koechling, “Three-dimensional frictional rigid-body impact,” *Journal of Applied Mechanics*, 1995.