January 2014

Autonomous Spacecraft Control During Close-Proximity Near-Earth Object Operations

Joshua Thomas Johnson

Follow this and additional works at: https://commons.und.edu/theses

AUTONOMOUS SPACECRAFT CONTROL DURING CLOSE-PROXIMITY NEAR-EARTH
OBJECT OPERATIONS

by

Joshua Thomas Johnson
Bachelor of Science, University of North Dakota, 2013

A Thesis

Submitted to the Graduate Faculty

of the

University of North Dakota

in partial fulfillment of the requirements
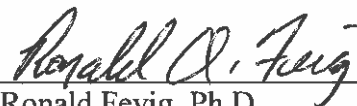
for the degree of

Master of Science

Grand Forks, North Dakota
December
2014

This thesis, submitted by Joshua T. Johnson in partial fulfillment of the requirements for the Degree of Master of Science from the University of North Dakota, has been read by the Faculty Advisory Committee under whom the work has been done and is hereby approved.
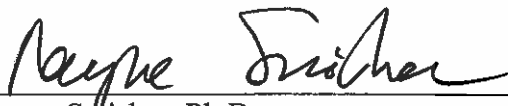
William Semke, Ph.D.

Ronald Fevig, Ph.D.

Marcellin Zahui, Ph.D.

This thesis is being submitted by the appointed advisory committee as having met all of the requirements of the School of Graduate Studies at the University of North Dakota and is hereby approved.

Wayne Swisher, Ph.D.
Dean of the School of Graduate Studies

December 10, 2014
Date

ii

PERMISSION

Title   Autonomous Spacecraft Control during Close-Proximity Near-Earth Object Operations

Department   Mechanical Engineering

Degree   Master of Science

In presenting this thesis in partial fulfillment of the requirements for a graduate degree from the University of North Dakota, I agree that the library of this University shall make it freely available for inspection. I further agree that permission for extensive copying for scholarly purposes may be granted by the professor who supervised my thesis work or, in his absence, by the Chairperson of the department or the dean of the School of Graduate Studies. It is understood that any copying or publication or other use of this thesis or part thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to the University of North Dakota in any scholarly use which may be made of any material in my thesis.

<div align="right">

Joshua T. Johnson

December 11, 2014

</div>

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would first like to thank my advisor, Dr. William Semke, for giving me the opportunity to pursue this project as well as his support and guidance throughout my graduate career at the University of North Dakota. I would like to thank my committee member, Dr. Ronald Fevig, for his advice and expertise in the field of orbital mechanics. I would also like to thank my other committee member, Dr. Marcellin Zahui, for sparking my interest in computer programming and control systems.

Finally, I would like to thank my parents, Charles and Claudia Johnson, my family, and my friends for supporting and believing in me. Without them, none of this would have been possible.

ABSTRACT

A control scheme is proposed for a satellite orbit controller around a small, irregularly

shaped near-Earth object (NEO) combining classical control theory and orbital mechanics into a

continuous hybrid control system that achieves and maintains a circular orbit in a perturbed

environment. NEOs are asteroids and comets that approach Earth's orbit around the Sun. They

are currently being studied for resource allocation and threat mitigation, while providing unique

opportunities for control systems. The NEO environment consists of a weak and complex

gravity field, as well as other perturbations such as solar radiation pressure (SRP) and third-body

gravitational disturbances. This project focuses on the gravity field of the NEO and

characterizes orbital stability within the NEO's gravity field. A three-term Proportional, Integral,

and Derivative (PID) controller is utilized in order to achieve and maintain a circular orbit in

close-proximity to the NEO 25143 Itokawa. The proposed control scheme merges a simple

controller with orbital mechanics to maximize the effectiveness and efficiency of the thrusters. It

uses the PID controller to thrust in the radial direction in order to maintain the proper orbital

radius, which is found to be an effective method of correcting perturbed orbits in the NEO

environment. This is followed by a change in the orbital velocity of the spacecraft in order to

match the specific mechanical energy for the desired circular orbit, which is typically the most

efficient method of correcting perturbed orbits. Systems Tool Kit (STK) is used to run the

simulation and a MATLAB-STK interface was developed that allows for sophisticated orbit

control development. Using the STK simulation software allows for the ability to test multiple

orbit parameters for stability. This was applied in studying the interaction between the complex gravity model and its effect on the satellite using a harmonic excitation analysis. It was found that when the ratio of the excitation frequency to the natural frequency (ω/ω_n) is greater than seven, the orbit is stable. This thesis provides methods for simulating and predicting satellite orbit control as well as providing guidelines for regions of stability for NEO missions.

CHAPTER I

INTRODUCTION

The primary purpose of this project is to develop an autonomous control system for satellite orbit control around a small asteroid, 500 m or less in size. This study makes use of a hybrid controller that merges orbital mechanics with a continuous three term Proportional, Integral, and Derivative (PID) controller, and only considers the spherical harmonic degree and order four complex gravity model of asteroid 25143 Itokawa. Performing a Hohmann-like change in velocity (ΔV) orbit transfer is found to be ineffective in the complex asteroid environment due to the perturbations caused by the gravity model. A more effective method of navigating around the small-body asteroid is to thrust along the direction of the radial vector. The previous two methods can be combined into a hybrid controller to exploit the efficiency of orbital mechanics and the effectiveness of simple controllers. Other factors such as Solar Radiation Pressure (SRP) and third-body gravity cause perturbations that need to be considered in future control schemes. The secondary purpose of this project is to develop a MATLAB and Systems Tool Kit (STK) interface to be utilized in complex mission design and analysis. MATLAB code is used to build the STK scenario and to read the satellite state data from the STK simulation. Then MATLAB computes and sends back maneuver data to the scenario while STK propagates the simulation. The final piece of the project is to launch an ongoing interdepartmental collaboration project between the Department of Mechanical Engineering and the Department of Space Studies at the University of North Dakota.

It is necessary to maintain a high degree of control while performing operations in close proximity to Near-Earth Objects (NEOs) for many reasons. First, the gravity field is tenuous since gravitational force is directly proportional to the mass of the objects interacting. With a central body having many orders of magnitude less mass than Earth, its gravitational influence on a satellite will be many orders of magnitude less than that of Earth. Therefore, it is relatively easy for the satellite to escape the gravitational force holding it in orbit around an NEO versus that of Earth. It is critical to make sure that the satellite maintains a precise trajectory and velocity at the desired orbital altitude so that it does not escape the NEO's tenuous gravity field.

Next, the shape of the gravity field is complex and is correlated to the shape and mass densities of the NEO. Since Earth is mostly round its gravity field can be approximated as a round shape, which makes it easy to predict and plan for in mission preparations. But with an NEO that may be shaped like a football, with varying mass densities throughout, the gravity field surrounding the NEO can be even more complexly shaped than the body itself. So if a satellite is put into a circular orbit in a complex gravity field, it will experience changing magnitudes of gravitational force as it traverses through its intended orbit path. This is especially true in close-proximity missions where the gravity field cannot be assumed to be that of a point mass. A point mass assumption is valid at high altitudes, where the asteroid interacts with the satellite as a point mass would. The perturbations from the gravity field at close range can cause dramatic changes to the trajectory of the satellite that can potentially cause the craft to either escape the system, or crash into the NEO within a relatively small amount of time. In order to extend the operational lifetime of the satellite, carefully planned correction maneuvers need to be executed in order to maintain its intended trajectory.

Even further opportunities for potential problems arise in the NEO environment. The Sun is constantly emitting light energy in the form of photons. These photons have no mass, but quantum mechanics explains that they have both energy and momentum in what is called the wave-particle duality. Each photon strikes the surface of the spacecraft and imparts a small change in momentum. What you end up with is millions of photons crashing into the entire surface area of the satellite facing the Sun, which all add up to a force that can push the spacecraft off its intended trajectory. This phenomenon is known as solar radiation pressure (SRP). In deep space missions, SRP can perturb the trajectory of the satellite and even overpower the force of gravity in some cases.

Yet another complexity is the other bodies in space having a gravitational influence on the spacecraft besides the central body it is orbiting. In the Solar System, everything is held together by the gravitational force of the Sun. But as you traverse through the system, the other planets can influence the satellite's trajectory as well, depending on the satellite's proximity to the planet. These are called third-body perturbations. As the asteroid, and the satellite with it, travels through its orbit within the Solar System, both are experiencing changes in gravitational force as they pass by the planets. The asteroid is on an orbit around the Sun dictated by its velocity and mass, but the satellite can be perturbed from its orbit around the asteroid as it nears some of the larger planets.

And finally, communication time lag is an issue as well. For example, a satellite is approaching Mars while the planet is at its average distance from the Earth (228.5×10^6 km) [1]. If a maneuver signal is sent from the mission control center on Earth at the speed of light (2.998×10^8 m/s), it takes nearly thirteen minutes for that signal to reach the satellite. If trying to avoid a crash, thirteen minutes is a long time as the multi-million dollar satellite quickly approaches

disaster. Then it takes yet another thirteen minutes for a signal to be sent back to mission control to communicate whether or not the maneuver even worked in avoiding the crash. That is if communications is maintained the whole time, which isn't always a guarantee. Therefore it is important for the satellite's systems to be able to detect and correct for any major problems without the intervention of the mission controllers stationed on Earth. In deep space missions, this time lag can be much more substantial depending on the distance of the satellite from Earth.

It can be understood that orbiting a small-body asteroid is a complex problem. Items that need to be considered include the tenuous and highly-variable gravity field, SRP, and third-body gravitational influences, as well as communications issues. Orbiting a small asteroid is a problem that is yet to be fully solved, and this project takes a look at some of the orbital mechanics involved while testing a few different control methods that can be used to maintain a stable orbit.

Motivation

Space travel and habitation has become a topic of ever-increasing interest and continues to gain in popularity. Some of the larger topics include continued space exploration, resource mining, and threat mitigation.

It is human nature to be curious and try to understand the universe around us. Throughout the centuries, scientists have sought to explore the heavens in an attempt to survey that which lies beyond Earth's atmosphere. The quest for knowledge has lead us to explore the outer reaches of our galaxy and beyond in search of other intelligent life, and to answer one of the most debated topics in history: How did we all get here? Scientist have been able to piece together evidence of The Big Bang, but continue to search for clues that lead to answers of the

many questions that still remain unanswered. Studying asteroids and comets at close range could provide greater incite and fill in the missing pieces of the puzzle, especially the formation of our Solar System.

A second reason to have a close encounter with an asteroid is to gather resources. As technology continues its integration into our lives, there is a need for the raw materials required to produce the circuit boards inside of all of our gadgets. Many modern phones and computers contain circuitry that requires what is known as semi-precious metals. They are considered semi-precious because they have only been found in specific areas of the world, and the countries that have them control the prices of these materials.   For example, the cost of cell phones has increased as the demand for certain metals has increased, but the known supply of those metals remains limited. It is likely that large deposits of these metals can be found floating around in space [2]. A more ambitious use of asteroids and comets is to mine them for raw materials to be used in space colonization. It could be more expensive to launch steel and other structural materials into space than to harvest such materials from bodies already in space.

Another precious resource is water. During manned operations in space, the mission cannot continue without water. Asteroids and comets have been observed to contain huge pockets of ice that can be harvested for water [3]. Furthermore, the hydrogen contained in the water molecules can be extracted and used as fuel to power spacecraft. This would make asteroids and comets "gas stations" for interplanetary travel.

More importantly, one of the most talked about topics regarding space in the news today is threat mitigation. The meteor that exploded above Chelyabinsk, Russia reawakened many people to the possibility of a deadly meteor or asteroid strike. Luckily the meteor exploded high above the surface of the Earth, but the shockwave from that explosion caused many windows to

5

shatter forcefully and injured a reported 1,210 people [4]. The roughly 19 m in diameter meteor exploded at a height of 24 - 30 km above Chelyabinsk with a blast energy of roughly 530 kilotons of trinitrotoluene (TNT) [5]. Pieces of the meteor made it to the surface of the Earth, but the reported damage was caused by the shockwave created from the explosion high above the city. Throughout history there have been many recorded impacts on Earth, including the meteorite impact that led to the extinction of the dinosaurs. It is important that we monitor and study celestial bodies flying near us so that one day we may be able to intercept and destroy the next killer asteroid before it reaches Earth.

Finally, this project stems from previous work done by Church and Fevig as they investigated the feasibility of creating a detailed gravity model of small body asteroids which allows for determination of the internal structure of the asteroid [6]. The proposed process involved having two satellites in orbit, flying in formation around the body. Then observations are made, from one satellite to another, of their changes in trajectory. By observing these perturbations, a detailed gravity model of the asteroid can be derived. This method was proven to be successful during the GRACE and GRAIL missions.

It was found that two satellites put into orbit next to each other around a simulation model of asteroid 25143 Itokawa quickly diverge in their trajectories [6]. The satellites have the same initial conditions except that they are separated by five degrees in true anomaly, the angle between their initial radius vectors. Figure 1 shows that quickly the satellite with the yellow trajectory crashes and the satellite with the red trajectory is ejected from the system, despite the fact that their initial conditions are quite similar (the white trajectory is their unperturbed circular orbit, displayed for reference). Therefore, it is necessary for a control algorithm to stabilize the orbits so that observations can be made in close proximity to the body.
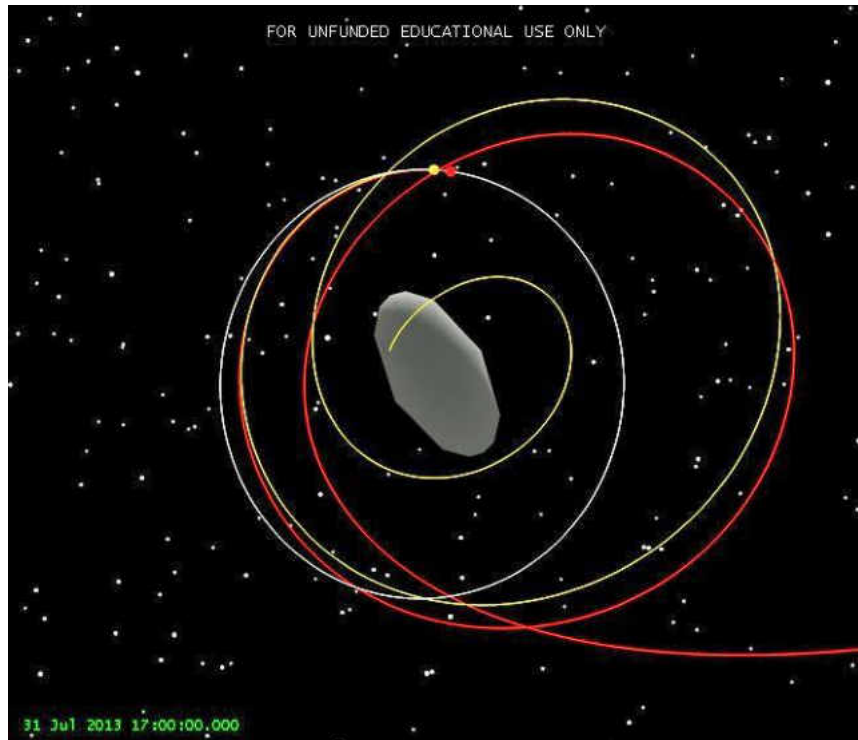
Figure 1. Diverging trajectories around near-Earth asteroid 25143 Itokawa. Two satellites are placed in a circular orbit next to each other. One crashes, while the other is ejected from the system.

Background

NASA has defines Near-Earth Objects as asteroids or comets with a perihelion distance less than 1.3 Astronomical Units (AU) [7]. The perihelion distance is the asteroid or comet's distance to the Sun at its closest point to the Sun in its orbit (Fig. 2). One AU is the average distance between the Sun and the Earth, and is used as a standard of measurement ($\approx$1.4960 x 10^{11} m). This means that an asteroid or comet is considered an NEO if the closest point in its orbit around the Sun is less than 1.3 times the average distance between the Sun and Earth. Figure 3 shows the NEO types recognized by NASA, and has a group of scientists dedicated to discovering and tracking NEOs in the Solar System.
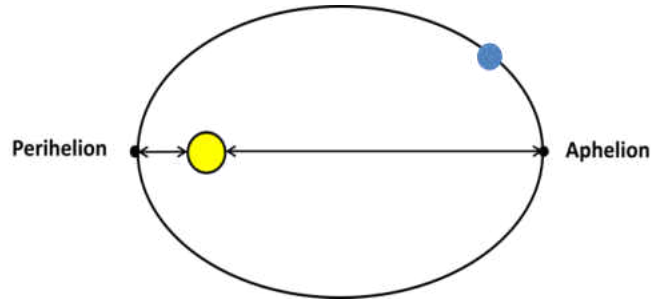
Figure 2. Perihelion and Aphelion points for a body orbiting the Sun.

Apollo
Semimajor Axis ≥ 1.0 AU
Perihelion ≤ 1.02 AU
Earth Crossing

Aten
Semimajor Axis < 1.0 AU
Aphelion ≤ 1.0167 AU
Earth Crossing

Amor
1.02 AU < Perihelion ≤ 1.3 AU

Inner Earth Objects (IEOs)
Aphelion < 0.983 AU
Always inside Earth's orbit
(aka Apohele)

Type	Near-Earth Population
Apollo	62% of known asteroids
Aten	6% of known asteroids
Amor	32% of known asteroids
IEO	6 known asteroids
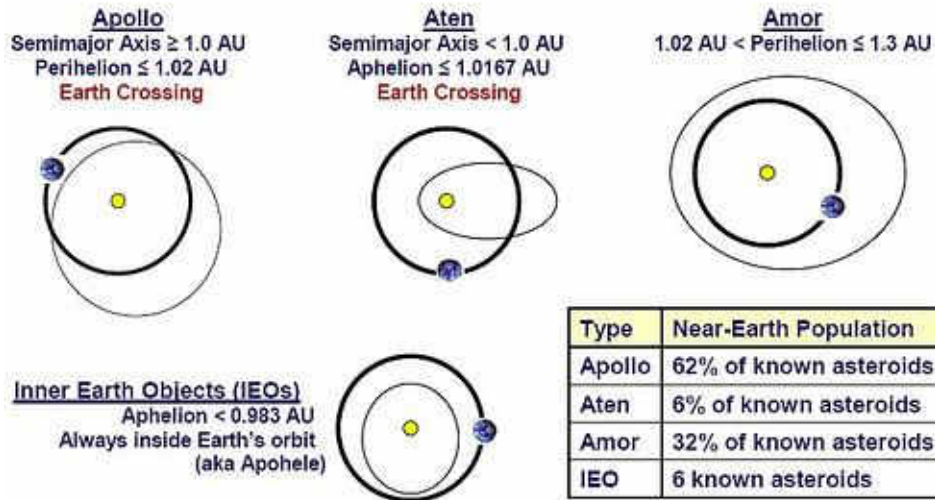
Figure 3. NEO groups as categorized by NASA [7]

At the time of this writing, there are over 1,500 Potentially Hazardous Asteroids (PHAs) that are being tracked and monitored by NASA [8]. Potentially hazardous does not mean that an impact is imminent, only that the orbit and size of the object are cause for concern. As new observations of these PHAs become available, scientists can better predict their orbit path and assess the likelihood of a future close approach to Earth.

As previously mentioned, orbiting an asteroid provides a unique control problem and it is yet to be completely solved. There has been success in close proximity NEO missions as NASA landed on an asteroid during the NEAR Shoemaker mission, and most recently the European

Space Agency (ESA) landed on a comet during the Rosetta mission. Also, Japan has had limited

success so far with their Hayabusa asteroid sample return missions. This project focuses on the

Hayabusa missions, and uses the data captured in order to simulate a real-world mission profile.

In 2003, the Japanese Aerospace Exploration Agency (JAXA) launched the Hayabusa

mission. The Hayabusa satellite travelled for over two years, using four xenon ion engines and

rendezvoused with the Apollo type asteroid 25143 Itokawa, shown in Fig. 4 [9].
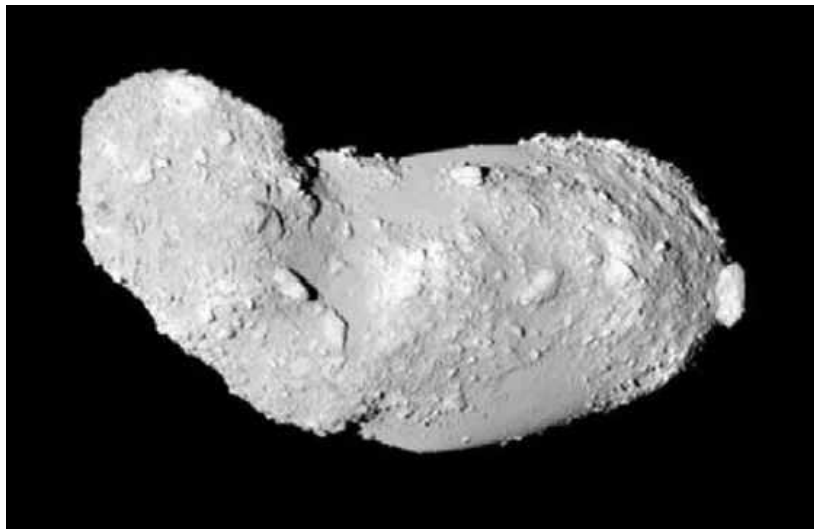
Figure 4. Asteroid 25143 Itokawa as seen from Hayabusa [10]

Upon arrival at Itokawa, the Hayabusa satellite observed Itokawa from a distance of 20

km and collected data about Itokawa's gravity field. The original plan was to move in close and

perform a few soft landings in order to collect samples of Itokawa's surface using a collection

horn, much like the "touch-and-go" maneuver that airplane pilots sometimes practice. The

collection horn was to skim the surface during the maneuver and collect samples from dust

clouds created by a series of projectiles launched from Hayabusa at Itokawa's surface [11].

Unfortunately, during the course of the sampling maneuver, communication with the

satellite was lost. Once communications were restored, Hayabusa was found to be in a safe

mode and was spinning above the surface of Itokawa in order to stabilize itself. The data collected from the onboard systems showed that Hayabusa had landed on Itokawa, but a sensor aborted the sample collection and put the satellite into safe mode. The safe mode initiated a burn to regain altitude back to the 20 km observation distance, and initiated the stabilization sequence. Luckily, the non-scripted landing had allowed for a very small amount of surface samples to enter the collection horn anyway, and the horn was sealed for its return trip to Earth.

The rest of the mission had many problems as a solar flare damaged the craft's solar panels, two reaction wheels used in attitude control failed, and some of the thrusters failed due to fuel leaks. Through this however, Hayabusa managed to make it back to Earth in 2010 and the re-entry capsule's samples were recovered intact [12].

The takeaways from this mission, in relation to the current research project, are as follows:

- The Hayabusa spacecraft did not attempt a close-proximity orbit, but used a hovering approach in order to maintain stability.
- Even with the stability of the spacecraft's flight maintained, a loss of communications with the craft was the source of undesired performance during the sampling portion of the mission.

Therefore, the focus of this project is to create a control algorithm that can achieve a stable, close-proximity orbit around a small-body asteroid with the use of an autonomous control system that requires no external input from a mission control center on Earth.

JAXA has learned a lot of important lessons from the first Hayabusa mission. It is recognized as the first unmanned asteroid sample return mission. The amount of samples captured was much less than the desired amount but, remarkably, the craft was successful in

returning those samples of Itokawa back to Earth despite technical issues during the return trip. JAXA has scheduled to launch follow-up mission, Hayabusa 2, on November 30, 2014 which will travel to asteroid 1999 JU_3 to study the environment and collect samples from the surface of that asteroid [13].

Taking all of the data from the Hayabusa mission, there is a significant amount of data to get started with our study and simulations of a known asteroid. First and foremost, the gravitational field about Itokawa is very tenuous. A particle, evaluated at Itokawa's perihelion point, can escape the surface of Itokawa if travelling at a speed of 0.08 m/s – 0.22 m/s (the variation depending on the location on the body of the asteroid) [14]. This is extremely low, considering a leisurely walk is roughly 1.4 m/s.

Next, solar radiation pressure has a dominating influence over gravity at certain altitudes. It was found that for a satellite with the surface area of Hayabusa, the semi-major axis of the orbit must be 1 km – 1.75 km in order for the craft to be bound to the asteroid's gravitational field without being overcome by SRP [14]. The range of this value is due to the SRP force changing with the asteroid's proximity to the sun in its orbit through the Solar System. Finally, a region of natural stability exists around Itokawa that lies between a radius of 1.0 km and 1.5 km from the center of the body [14]. In this region, a stable orbit can be maintained with no correction being made to the orbit trajectory. This is both interesting and helpful in that an orbit can be maintained in semi-close proximity with little effort. This region can potentially be used as a "staging area" in the mission design when preparing for other close-proximity maneuvers such as sampling or landing.

In Itokawa's natural stability region, gravity and SRP balance each other out in high-inclination orbits that lie near the sun-terminator plane [14]. This is the plane that splits the body

11

in half, and one half is lit by the sun while the other half is in total darkness. Outside of this plane it has been found that SRP causes the eccentricity of the orbit to increase towards unity, ultimately resulting in a hyperbolic trajectory.

A study was conducted by Matt Zimmer from the Department of Space Studies to find out how much effect 3rd body gravity has on the satellite while it is in close-proximity to Itokawa. As previously mentioned, SRP is the result of millions of photons crashing into the surface of the satellite. Over time, the changes in momentum caused by these photons can add up to a dominating force.

$$F_{SRP} = \left(\frac{F_S}{c}\right) A_s (1 + r) \cos I \qquad (1)$$

where

F_{SRP} = SRP force (N)

F_s = solar constant at Itokawa's orbit around the Sun (W/m²)

c   = speed of light = 3×10^8 m/s²

A_s = illuminated surface area (m²)

r   = surface reflectance (unitless)

I   = incidence angle to the sun (deg).

The PID velocity controller was tested around asteroid 1999 JU3 as the shape of that asteroid is more round than that of Itokawa, making the gravity field less complex. It was found that the PID velocity controller was able to handle large orbit transfers, from 3 km to 5 km, without the influence of SRP in the simulation, but the same controller was dominated by the effects of SRP, which is illustrated in Fig. 5. With this, it was seen that SRP has a huge impact on this type of mission design.
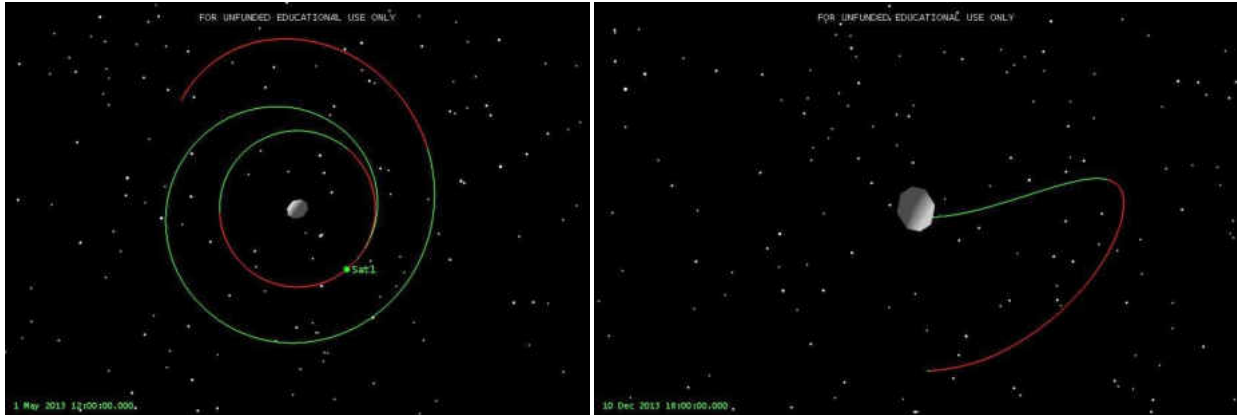
Figure 5. PID controlled velocity orbit transfer around asteroid 1999 JU3. Without SRP (left) and the same scenario with SRP (right).

Furthermore, Zimmer found that with Itokawa's elliptical orbit around the Sun, the force of SRP changes greatly with its position relative to the sun. Figure 6 shows the SRP force compared to the gravitational force of a satellite in a circular orbit around Itokawa with a radius of 5 km. At the asteroid's perihelion distance (closest to the Sun), the SRP force is actually greater than the gravitational attraction to Itokawa. At the asteroid's aphelion distance (furthest from the sun), the gravitational force becomes slightly stronger than SRP.
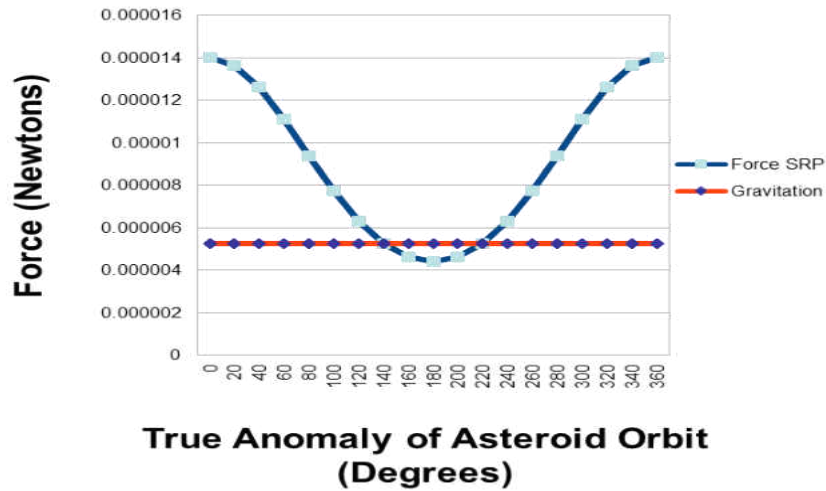
Figure 6. SRP and gravitational strength versus Itokawa's orbital position around the Sun.

Further complicating the control problem is the gravitational influence of the other planets in the Solar System. Figure 7 shows the average gravitational strength experienced by a spacecraft orbiting Itokawa from various bodies in the Solar System. It is important to note that these values are constantly changing throughout Itokawa's orbit around the Sun as its proximity to the planets changes. But on average, gravity from Venus, Earth, and Jupiter all perturb the satellite's orbit of Itokawa. Not shown in this graph is the Sun, as it is the dominating gravitational body in the Solar System, so it also has a gravitational influence on the spacecraft as well.
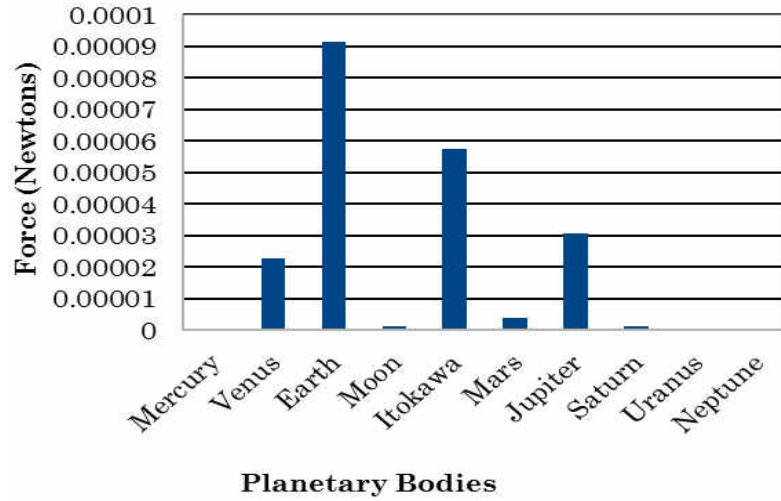
Figure 7. Average gravitational force on a spacecraft from the planets.

A control system needs to be able to deal with all of these perturbations and make corrections to the trajectory of the spacecraft in order to prevent crashing into the NEO or being ejected from the vicinity of the body.

CHAPTER II

THEORETICAL ANALYSIS

The spherical harmonic model of asteroid Itokawa has been constructed from data collected during the Hayabusa mission, and 3D simulation models were built using that data [14, 15]. However, those models would be a rather complex starting point. Thus, a simplified system dynamics model is required. From there, a control scheme can be selected and simplified models can be used to test the effectiveness of the orbital controller against some pre-determined criterion. Then other complexities can be added into the model and dealt with in the control algorithm one-by-one.

Analytical Modeling

Gravity can be thought of as a spring, in simple terms. As you pull on a spring, it pulls back with some force

$$F_s = kx \qquad (2)$$

where

F_s = force of the spring (N)

k = the spring constant (N/m)

x = distance the spring is stretched (m).
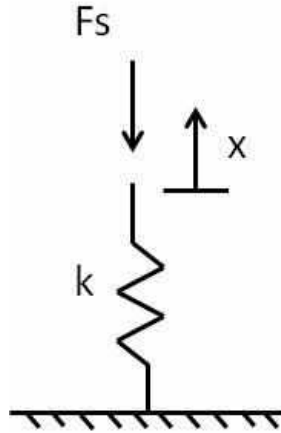
Figure 8. A simple spring.

The force of a spring acts to pull the spring back to its original equilibrium state. Gravity acts the same way. If you jump into the air, gravity tries to pull you back to your original starting position. The difference is that the force of gravity is a non-linear force. From Newton's law of universal gravitation

$$F_g = G \frac{m_1 m_2}{r^2} \qquad (3)$$

where

F_g = force of gravity (N)

G = universal gravitational constant ($\approx$6.674 x 10^{-11} N $(m/kg)^2$)

m_1, m_2 = the masses of the two bodies interacting (kg)

r = distance between the two bodies' centers of mass (m).

Though similar, the force of gravity is proportional to the inverse of the square of the distance between the two bodies, where the force of a spring is linearly proportional to distance.

Now if given the opportunity, a mass on top of an ideal spring that is pulled from equilibrium and released will continue to oscillate infinitely about the system's equilibrium point.

In order to stop this from occurring, mechanical systems use a damper to slow down the oscillations over time, shown in Fig. 9.
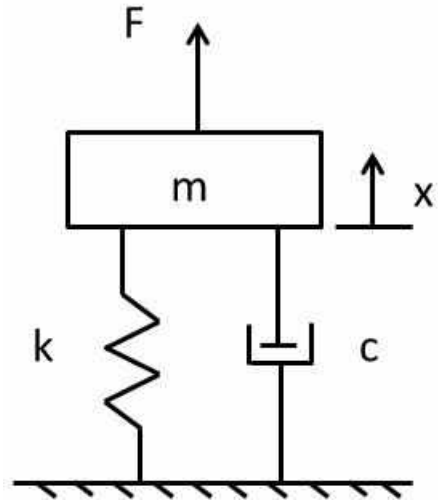
Figure 9. A simple spring-mass-damper system.

If a free-body diagram is considered and applying a summation of forces in the x-direction using Newton's Second Law of Motion, we end up with the following equation of motion for a simple spring-mass-damper system:

$$\Sigma F_x = ma_x \tag{4}$$

$$\Rightarrow F - kx - cv_x = ma_x \tag{5}$$

which simplifies to:

$$F = ma_x + cv_x + kx \tag{6}$$

where

a_x = acceleration in the x-direction (m/s^2)

v_x = velocity in the x-direction (m/s)

c = damping coefficient (N*s/m).

In the gravity analogy, the damper applies a resistive force that slows down the spring's oscillation. In a gravitational system, this force is typically atmospheric drag. Drag is a resistive force that slows down a body in motion and is defined as:

$$F_D = \frac{1}{2}\rho C_D A v^2 \tag{7}$$

where

F_d = force of drag (N)

ρ = mass density of the fluid (kg/m^3)

C_D = drag coefficient

A = the body's cross-sectional area (m^2)

v = the velocity of the object relative to the fluid (m/s).

Drag experienced by an Earth-orbiting spacecraft is caused by air resistance but, in deep space missions there is no atmosphere so there is negligible drag. The equation of motion for the satellite-asteroid system then becomes the equation of the simplified two-body problem seen in orbital mechanics:

$$F = ma_x + \frac{m\mu}{x^3}x \tag{8}$$

where

m = mass of the spacecraft (kg)

μ = G * $m_{asteroid}$, known as the gravitational parameter (km^3/s^2).

This equation is similar to Eq. 5 above, with zero damping. However, the spring constant (k) is replaced with a non-linear spring constant based on the gravitational force.

Up until this point, the force of gravity has been thought of as a spring force and Eq. 7 shows that the analogy works to a certain extent. However, there is a major difference in the

behavior of the two systems. The mechanical spring force is linearly dependent on the distance the free end of the spring is moved, and the gravitational force is highly non-linear in relation to the same term. Figure 10 shows the force of a mechanical spring vs. distance the free end is moved compared to the force of gravity vs. distance between the two objects. It's not easy to imagine a non-linear spring because a physical spring with non-linear properties doesn't exist, given that no plastic deformation occurs. This proves to be a challenge for the controller as the force of gravity changes dramatically based on the spacecraft's proximity to the central body.
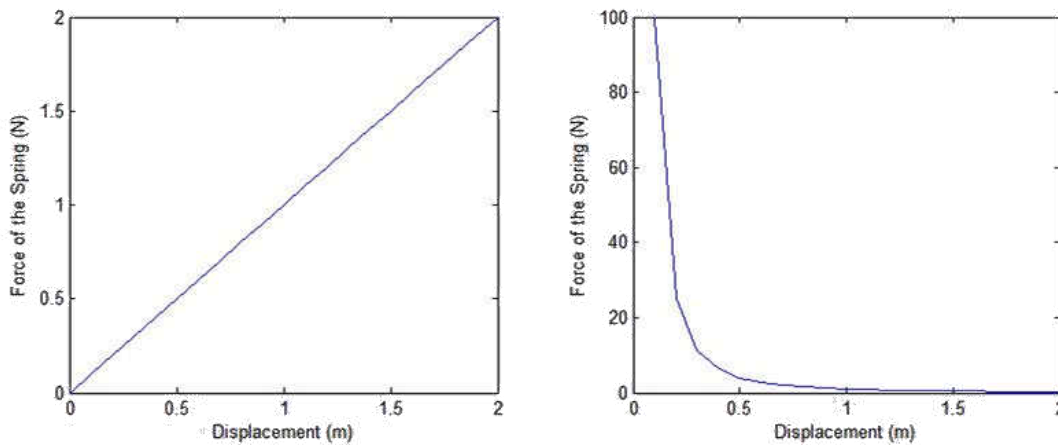
Figure 10. Linear and non-linear springs. The force of a linear spring (left) and the force of a non-linear gravity "spring" (right) for arbitrary values.

The conventional way to handle non-linearity in a control system is to linearize about an equilibrium point for the system. This allows for a small window of operations that maintains a local stability for the system. An example of this is the inverted pendulum, like the Segway Scooter (Fig. 11). The system is stable for small angles about the vertical equilibrium point due to the small angle approximation. The small angle approximation allows for an approximate solution to non-linear differential equations provided that for small angles, $\sin \theta \approx \theta$. This approximation is valid up until roughly 15°. If the system goes beyond that, the differential equation goes into the non-linear region and the system becomes unstable [16].

20
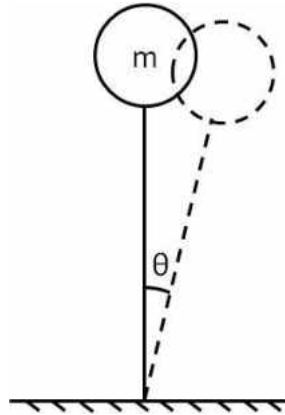
Figure 11. An inverted pendulum.

Other methods exist in solving non-linear control problems such as feedback linearization and optimal control strategies which are beyond the scope of this project [17].

The Hohmann Transfer

If a satellite is perturbed from its initial orbit, a maneuver needs to be performed in order to get back to the desired orbit. In orbital mechanics, typically the most efficient way to transfer from one orbit to another is to use the Hohmann transfer, shown in Fig. 12 [18].
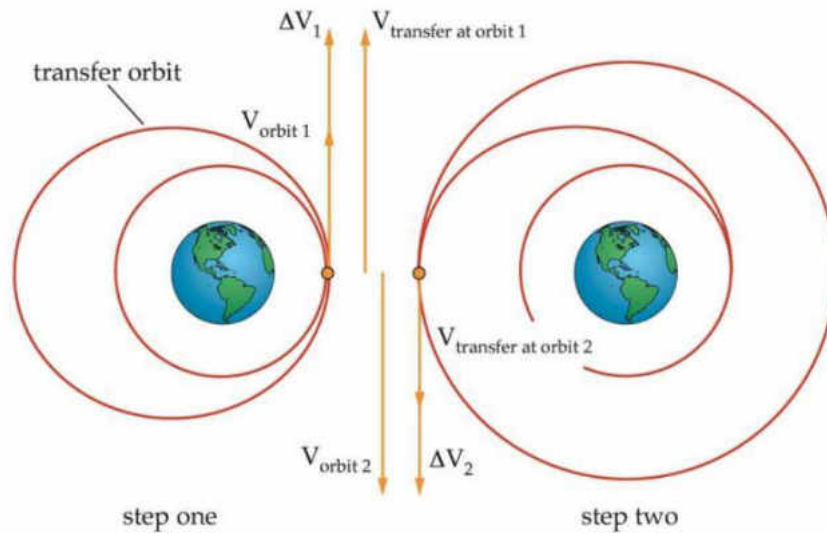
Figure 12. The Hohmann transfer [18].

The first maneuver in the Hohmann transfer is a boost that increases the satellite's velocity (ΔV_1 in Fig. 12). This raises the orbit's apoapsis (the point in the orbit furthest from the orbited body) and puts the satellite into an eccentric orbit. Once the satellite reaches the point of apoapsis, a second maneuver further increases the satellite's velocity (ΔV_2 in Fig. 12). This raises the orbit's periapsis (the point in the orbit closest to the orbited body) and circularizes the orbit. More detail and the equations involved can be found in Appendix A.

The Hohmann transfer is a good starting point to use for applying corrective maneuvers. If gravity pulls the spacecraft inward from its desired trajectory, a Hohmann transfer can be performed to regain the lost altitude. The reverse is also true in that if the spacecraft is pushed outwards from its desired trajectory, a Hohmann transfer can be performed to decrease the added altitude.

PID Control Theory

This project makes use of simple onboard controllers that are able to perform corrections and reject further disturbance input from the variable gravity field that causes a change to the measured variable, which is the orbital radius.

The PID controller is a good place to start as it is a simple controller, yet it is highly effective. The three term controller (Proportional, Integral, and Derivative) helps to control the transient (short-term) and the steady-state (long-term) behavior of a mechanical system and is part of a negative feedback control loop (Fig. 13). The PID controller has been used as an industry standard since the 1940's and accounts for more than 90% of the control loops being used in process control today [19]. The popularity of the PID controller comes from its ease of use and cost-efficiency. The PID isn't an optimum controller but it is good enough that the added cost and complexity of an optimum controller usually outweighs the small increase in performance [20]. Another advantage is that the PID controller can be tuned, by adjusting the control gains, to handle disturbance rejection without requiring any added complexity [20].
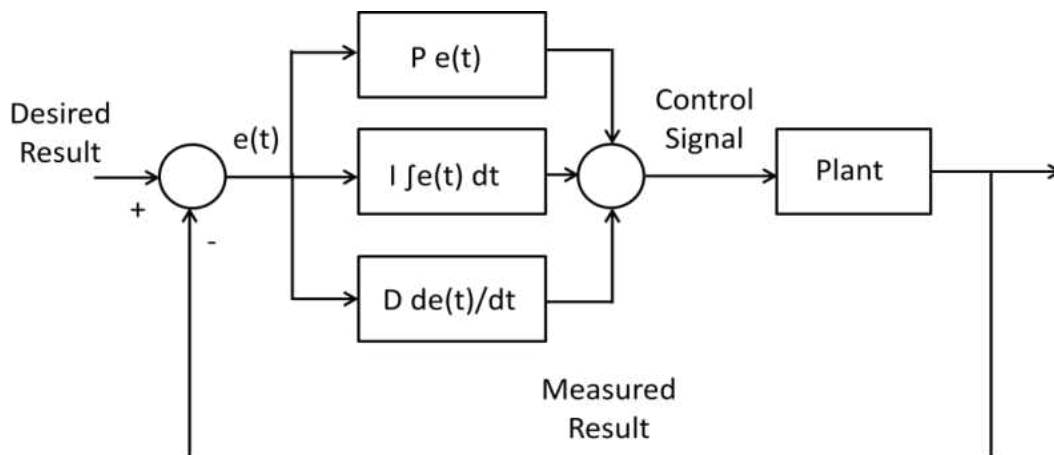
Figure 13. Feedback control loop with a PID controller.

The PID control equation is defined as:

$$u(t) = Pe(t) + I \int_0^t e(\tau)d\tau + D\frac{d}{dt}e(t) \qquad (9)$$

where

u(t) = the controller output

P   = the proportional gain, a tuning parameter

I   = the integral gain, a tuning parameter

D   = the derivative gain, a tuning parameter

e   = error, the difference between the desired value and the actual measured

value of the system

t   = time

τ = variable of integration.

The error in the system is dependent on the system parameter being controlled which for this project is the orbit radius. The desired radius is set by the user and fed into the system as a reference value. The actual radius of the spacecraft is measured and sent to the controller via the feedback loop. The difference between the desired radius and the measured radius is calculated and this value is the error. A correction maneuver is executed at each time interval until the error reaches zero. In the case of the satellite, a thruster is fired until the spacecraft reaches the desired orbit radius. Then that radius is maintained as the controller works to reject the perturbations caused the variation in gravitational force as the satellite traverses through the complex gravity field.

The control gains are used to tune the system to meet pre-determined performance criteria. As shown in Eq. 8, the control gains are multipliers P, I, and D that allow for an

increase or decrease in the effect of each term on the controller output. Going back to the spring-

mass-damper system (Fig. 14) will give a better idea of what the P, I, and D control gains do.
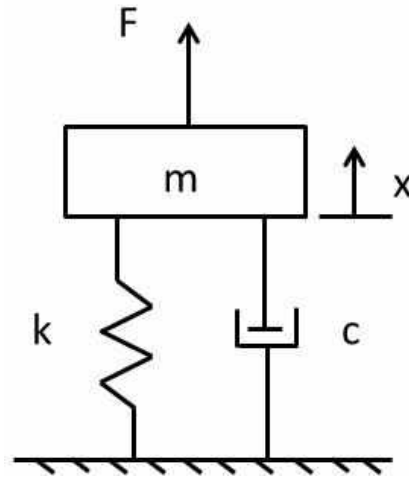
Figure 14. A simple spring-mass-damper system.

If the spring-mass-damper system is left alone, nothing will happen. If, however, the

mass is moved quickly to a new position the mass, having been disrupted, will oscillate about its

new equilibrium position, shown in Fig. 15. A controller can be applied to the system in order to

achieve a more desirable result by eliminating the oscillations and any long-term error.

Eliminating the oscillations provides precision and stability to the system. In close-proximity

orbits, a small oscillation in the spacecraft's radius may result in a crash or an ejection from the

system.
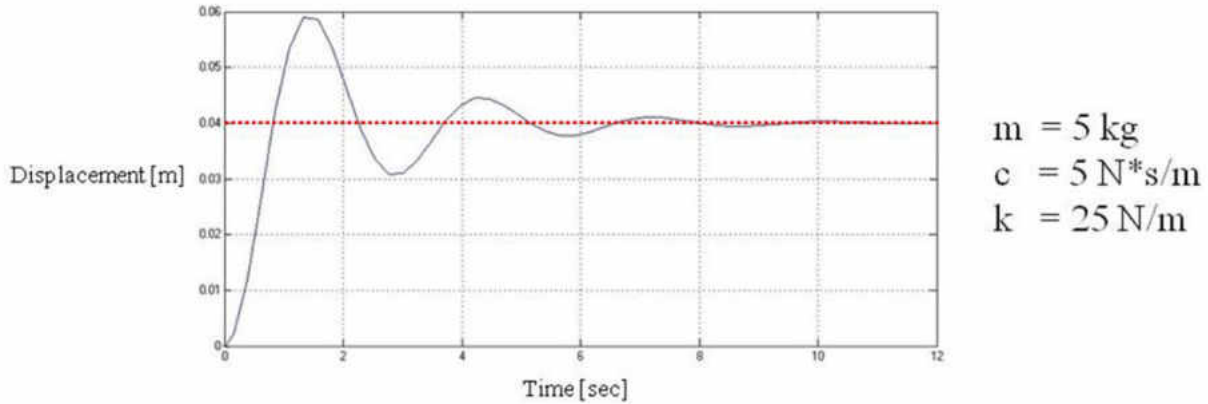
Figure 15. Time history for a spring-mass-damper freely oscillating about its equilibrium position.

The PID controller is useful for controlling both the transient and steady-state response. The Proportional gain and the Derivative gain work together to control the transient response. The P-gain applies an output directly proportional to the error. Essentially, if the error is five and the proportional gain is one, the controller output from the P-term will be five units of output (force, velocity, etc.). The D-gain monitors the change in the error, or how fast the error is changing. If the error is decreasing too quickly because the controller is making an aggressive correction, chances are the controller will overshoot (Fig. 16). In contrast, if the error is changing too slowly, the system is slow to respond (Fig. 17).
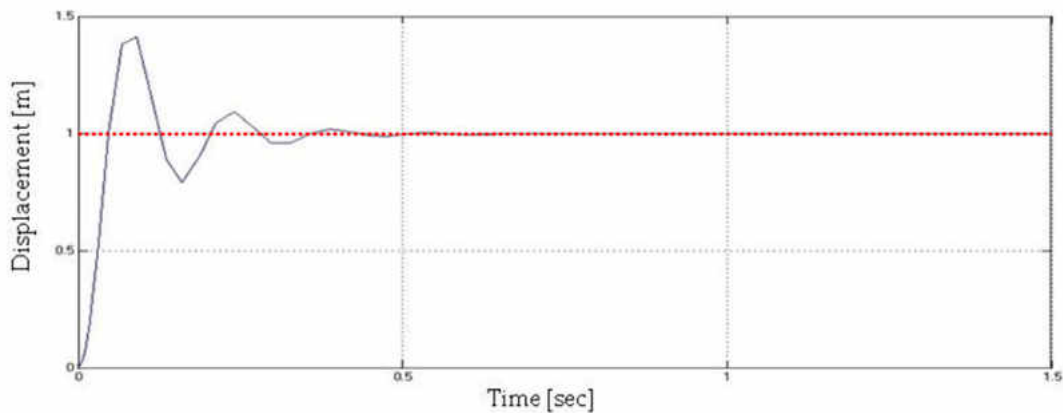
Figure 16. PID controlled system with high proportional gain (P).

26
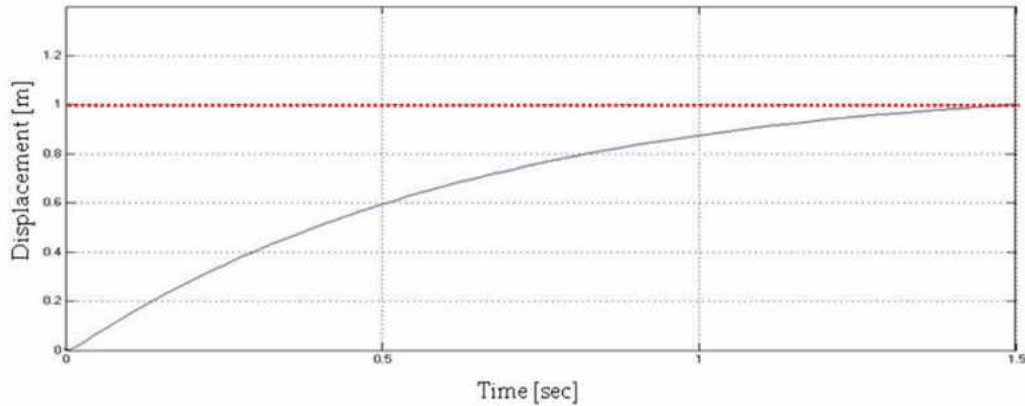
Figure 17. PID controlled system with high derivative gain (D).

The integral gain corrects the steady-state error of the system. The integral term adds up the error over the entire time of the scenario and compensates so that the long-term error becomes zero (Fig. 18).
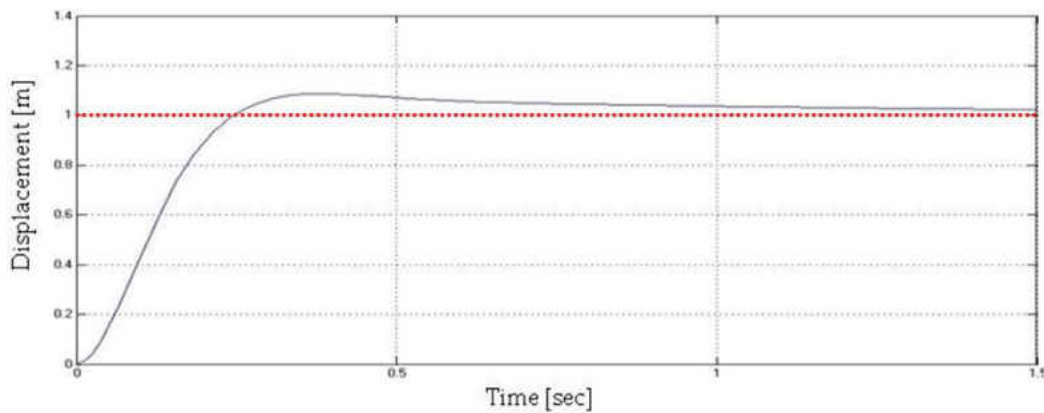
Figure 18. PID controlled system with high integral gain (I).

The P, I, and D gains all interact with each other, and when a balance between the three is met; peak time, overshoot, and steady-state error are minimized (Fig. 19). There are many known methods of tuning a PID controller, with one of the most popular being the Ziegler-Nichols tuning method [19].

27
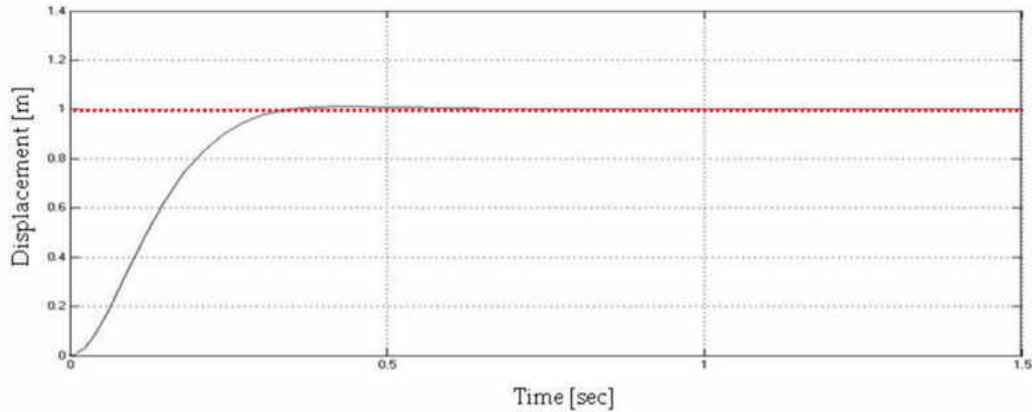
Figure 19. Tuned PID controlled system.

Finally, an objective set of criteria needs to be established in order to measure the performance of the feedback control system. As seen previously, tuning a feedback control system can come down to balancing opposing objectives. The P term decreases response time, but increases overshoot while the D term decreases overshoot, but increases response time. Because of this, control system design specifications are often not rigidly set standards to be adhered to, but an attempt at listing desired performance before starting the design process. There is room for revision and compromise to adjust the cost and performance of the system [21]. For this project, the following general controller design specification is formed, which matches the performance of the Hohmann transfer. This is the minimum desired performance of the controller.

Table 1. Control system design specifications.

Response Characteristic	Desired Behavior
Steady-state Error	No Steady-state Error
Peak Time	$\leq 1/2$ the Orbit Period
Percent Overshoot	No Overshoot
Settling Time	$\leq$ One Orbit Period

CHAPTER III

TESTING PROCEDURE AND RESULTS

The following sections display numerical simulations of all of the scenarios used to test the different controllers. This project utilizes Analytical Graphics, Inc.'s program Systems Tool Kit (STK) to run the simulations. STK has a myriad of built in orbit propagators and maneuver tools, but in order to create our PID orbit controller, MATLAB is used to send complex iterative commands to STK. This was done so that satellite position data can be read from STK, be put through a control algorithm in MATLAB, and then maneuver data is sent back to STK. This sequence occurs over a thousand times in each scenario, at designated time intervals. Figure 20 is used to illustrate the interface between MATLAB and STK. In reality, the program doesn't end when the desired orbit is achieved, but the PID continues to reject disturbance and if a perturbation is detected the process starts over again from the beginning. More detail on the STK-MATLAB interface and an example of the code can be found in Appendix B.
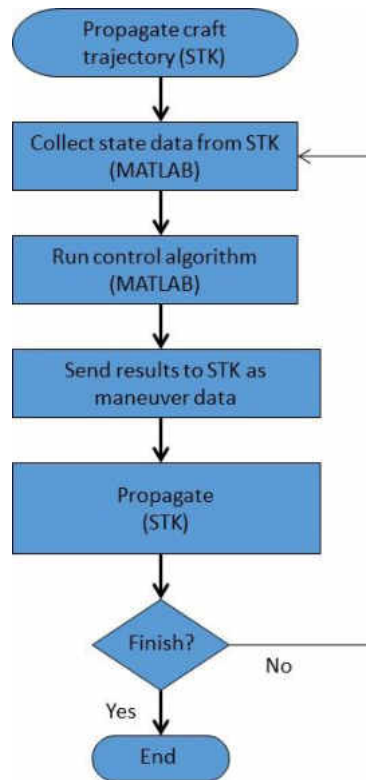
Figure 20. Simplified workflow diagram for STK/MATLAB interface.

Two models of asteroid Itokawa were used as the central body for the simulations: a

simple, two-body gravity model; and a complex, degree and order 4 spherical harmonic gravity

model built by Church using data from the Hayabusa mission [15]. The spherical harmonic

gravity model is used to create the complex shape of the gravity field caused by the odd shape of

the asteroid, coupled with the varying mass densities throughout. Figure 21 shows how the

degree and order coefficients play a role in the complexity of the spherical harmonic model. The

higher the degree and order, the higher the complexity of the gravity model. The GRACE

mission is processing data to make a highly detailed gravity model of Earth that has many orders

of complexity more than the ones this project is investigating. For example, a degree and order

200 model of Earth is shown in Fig. 22.
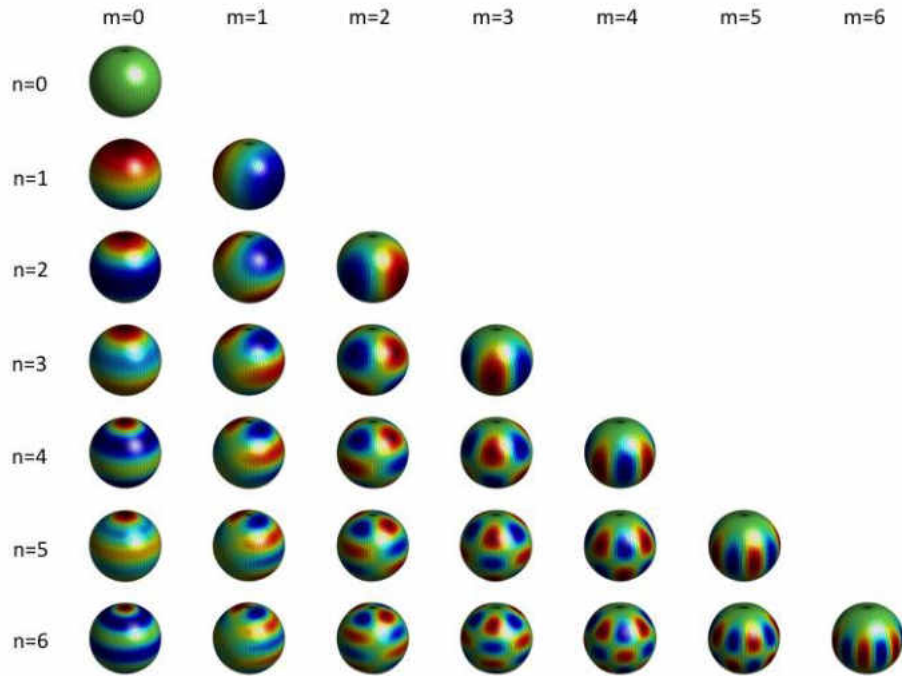
Figure 21. Spherical harmonic degree (n) and order (m) models [15].
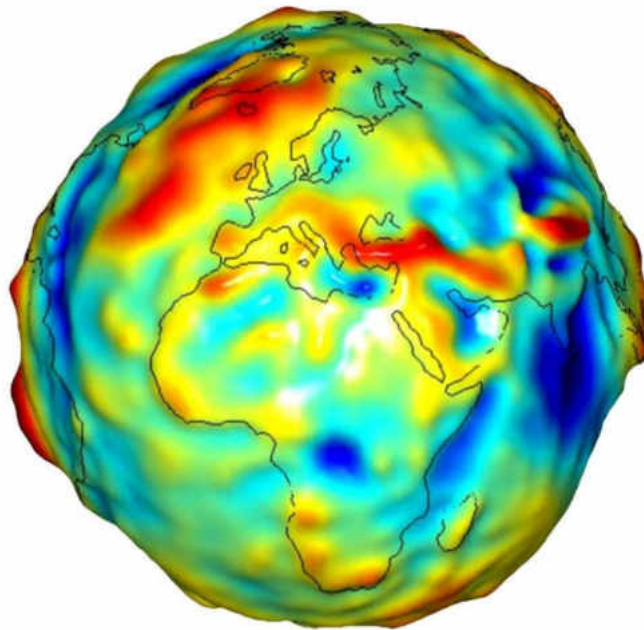
Figure 22. GRACE mission early gravity model of Earth [22]. The red portions illustrate higher gravitational force, where the blue represents lower gravitational force. These can be due to changes in surface elevation and mass density.

Six parameters are needed in order to fully describe the size, shape, and orientation of an orbit along with the spacecraft's position within the orbit. Johannes Kepler defined the six orbital elements, known as classical orbital elements (COEs) or Keplerian orbital elements. A more detailed description can be found in any orbital mechanics textbook, as well as the equations involved in computing the orbital elements [18]. But, in general they describe

- Orbital size, using semimajor axis, a

- Orbital shape, defined by eccentricity, e

- Orientation of the orbital plane, using

    o inclination, i

    o right ascension of the ascending node, Ω

- Orientation of the orbit within the plane is defined by argument of perigee, ω

- The spacecraft's location in the orbit is given by true anomaly, v

Figure 23 shows each of the elements in relation to the geocentric-equatorial coordinate system which is an Earth-centered inertial coordinate system that uses the Earth's equator as the fundamental plane and the vernal equinox as the principal direction. The vernal equinox direction is found by drawing a line from the center of the Earth to the Sun's position on the first day of Spring, which remains constant.

For an elliptical orbit, the semimajor axis and eccentricity describe the size and shape of the ellipse. The inclination of the orbital plane is the angle between the Earth's Z-axis and the axis normal to the orbital plane, the h-axis. The right ascension of the ascending node is the angle between the vernal equinox direction and the point at which the orbital path crosses the equatorial plane as the spacecraft travels into the Northern Hemisphere. The argument of

perigee is the angle between the ascending node and the point of perigee. Finally, true anomaly

is the angle between the point of perigee and the spacecraft's current position within the orbit.
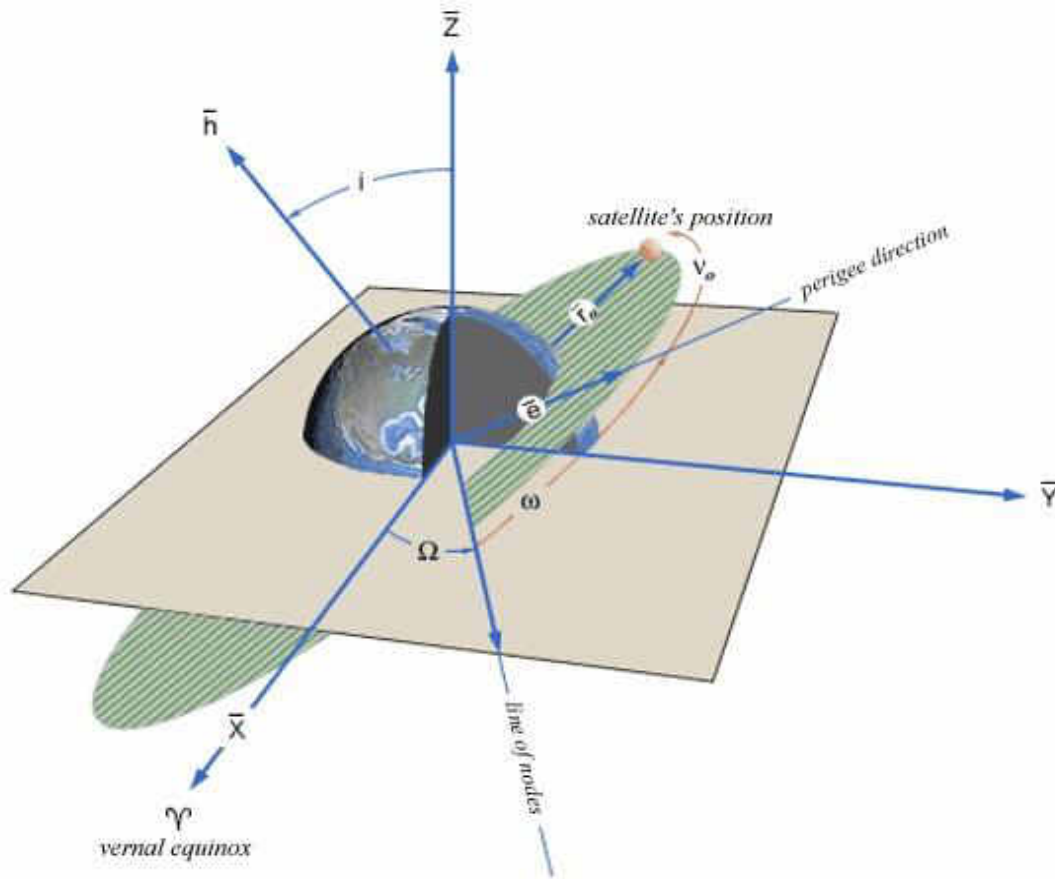
Figure 23. Keplerian orbital elements [23].

All of the following scenarios have the same initial conditions, apart from semi-major

axis (which is the same as the radius in a circular orbit) (Table 2). Three different values are

used for semi-major axis: 0.65 km, 1.10 km, and 3.00 km. These were chosen so that the

performance of the system could be evaluated in a highly variable gravity region (low-altitude),

the region of natural stability as observed by Scheeres (mid-altitude), and a "two-body" region

(high-altitude).

Table 2. Initial Keplerian orbital elements for the simulations.

Keplerian Element	Value
Semi-major Axis	0.65 km (low), 1.10 km (mid), and 3.00 km (high)
Eccentricity (e)	0.005
Inclination (i)	0°
Right-Ascension of the Ascending Node (Ω)	0°
Argument of Perigee (ω)	0°
True Anomaly (V_o)	0°

The low-altitude region is a close-proximity orbit that is highly perturbed by the spherical harmonic gravity model. The region of natural stability (mid-altitude) lies within 1.0 km and 1.5 km above the center of mass of Itokawa [14].   The two-body region is an altitude that is great enough such that the spherical harmonic gravity effects are not felt by the spacecraft. At this altitude the asteroid is effectively a point mass, so there is only a uniform gravitational force.

Note that the following sections contain figures that are screen captures of the simulations performed using STK. In each figure, a scenario using a simple two-body gravity model is shown on the left and a scenario using a complex spherical harmonic gravity model is shown on the right. This allows a direct comparison between the effects of the gravity models with all else being the same.

Finally, two objectives are being tested in the scenarios: (1) a small orbit transfer and (2) orbital stability, or disturbance rejection. Simply put, is the controller able get to the desired orbit, and then maintain that orbit over time. A small initial offset of 50 meters is given so that the controller engages at the start, since if the spacecraft starts at the desired altitude, the difference between the desired altitude and the actual altitude, or the error, is zero. After the

desired orbit is reached, the controller acts to reject perturbations caused by the complex gravity field in order to maintain the new circular orbit.

The satellite model used in all of the scenarios has a dry mass of 50 kg with 5kg of fuel. The satellite is equipped with an 8-cm aperture XIPS ion engine, with a minimum thrust of 2 mN and a maximum thrust of 14 mN.

NOTE: These scenarios do not take Solar Radiation Pressure (SRP) or third-body gravitational influences into account, only the spherical harmonic gravity perturbations.

Hohmann Transfer Simulation

A control sample is needed to test the effectiveness of the control methods being investigated. The Hohmann transfer is chosen as the control as it is a basic maneuver that works well in a simple system.

Figure 24 shows that at close range to the asteroid the Hohmann transfer successfully performs the altitude correction of the initial offset in the two-body gravity model, but cannot handle the perturbations from the complex gravity model as the satellite is ejected from system.

Figure 24. Low-altitude Hohmann transfer. Shown with the two-body gravity model (left) and the spherical harmonic gravity model (right).

Figures 25 and 26 illustrate that in the region of natural stability and at high altitude, the

Hohmann transfer is effective at performing the correction of the initial offset from the desired

orbit and is able to circularize and maintain the new orbit.

Figure 25. Mid-altitude Hohmann transfer. Shown with the two-body gravity model (left) and
the spherical harmonic gravity model (right).

Figure 26. High-altitude Hohmann transfer. Shown with the two-body gravity (left) and the
spherical harmonic gravity (right).

As expected, the Hohmann transfer performs well as a method for changing orbit radius

in all regions when using the two-body point mass gravity model. With the complex gravity

model, the Hohmann transfer is unstable at the low altitude and the satellite is ejected from the

system, but it successfully perform the transfer and maintains a circular orbit in the mid altitude

region of natural stability and the high altitude point mass region. This further illustrates the need for a controller in close-proximity to the NEO as the orbit becomes disrupted by the perturbations caused by the complex gravity model.

Controller Simulations

Three variations of the PID controller have been selected for testing: a classical PID velocity controller, a non-linear P velocity controller, and a classical PID radial position controller. The classical PID velocity controller and the non-linear P velocity controller are similar to the Hohmann transfer in that they provide ΔV burns along the direction of the orbital velocity vector. The controller is used to adjust the magnitude of the ΔV burns depending on the difference between the current and desired orbit radii. The classical PID radial position controller is a more direct approach in that the thrusters fire along the direction of the radial vector, which pushes the spacecraft directly to a higher or lower orbit radius.

PID Velocity Controller

The first control algorithm to be tested is the classical PID controller used to adjust the velocity of the satellite. The Hohmann transfer uses the ΔV burns in order to increase or decrease the size of the orbit by speeding up or slowing down the satellite, respectively. This results in different orbit altitudes due to the speed of the satellite. Similarly, the PID controller is used to set the magnitude of the ΔV burns along the direction of the orbital velocity vector in order to get to and maintain the desired orbit radius by speeding up or slowing down the satellite.

Figure 27 shows that at close range to the asteroid the PID velocity controller takes longer than the Hohmann transfer, but successfully performs the altitude correction of the initial

37

offset in the two-body gravity model. The PID velocity controller isn't able to respond fast enough in order to handle the perturbations in the complex gravity model and the satellite crashes into the asteroid.
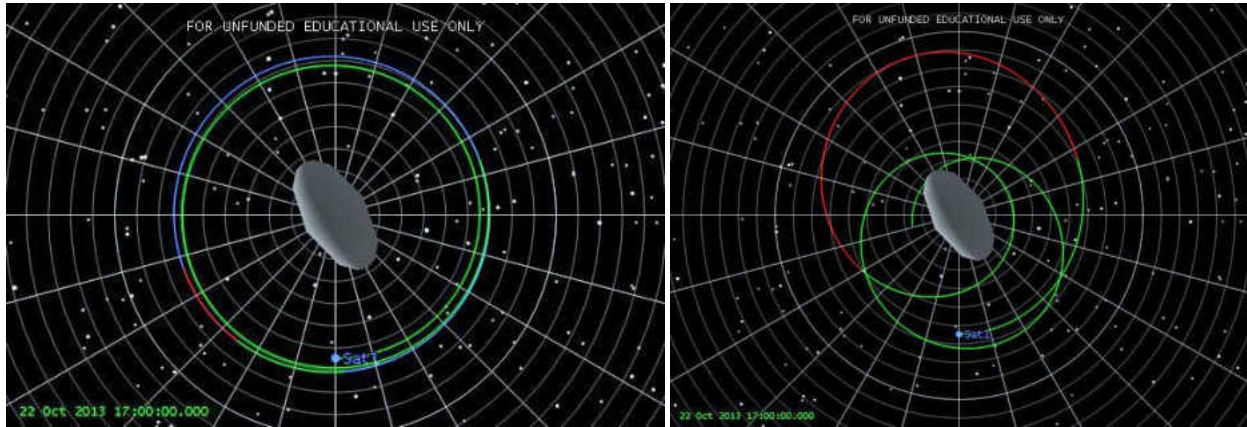
Figure 27. Low-altitude PID velocity transfer. Shown with the two-body gravity model (left) and the spherical harmonic gravity model (right).

Figures 28 and 29 illustrate that in the region of natural stability and at high altitude, the PID velocity transfer again takes longer than the Hohmann transfer but is effective at performing the correction of the initial offset from the desired orbit, and is able to circularize and maintain the new orbit.
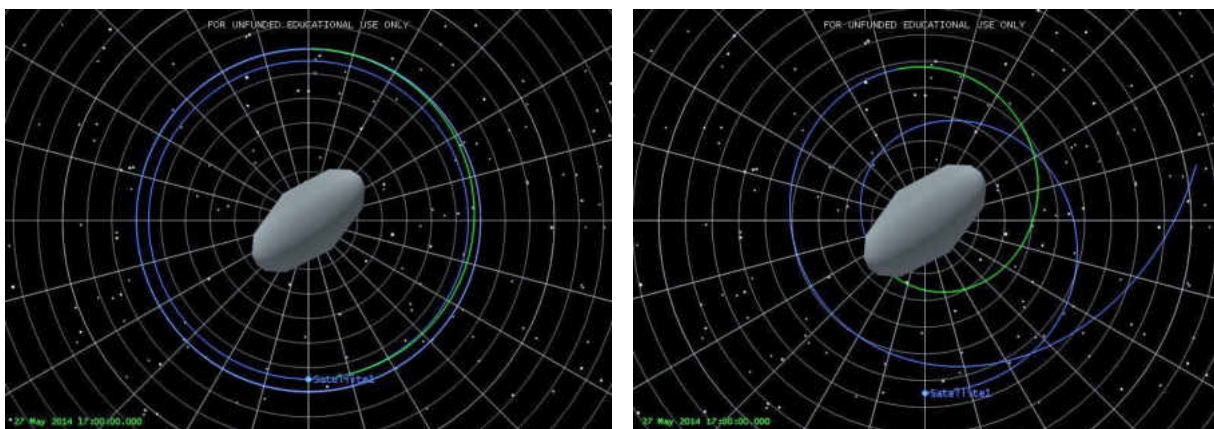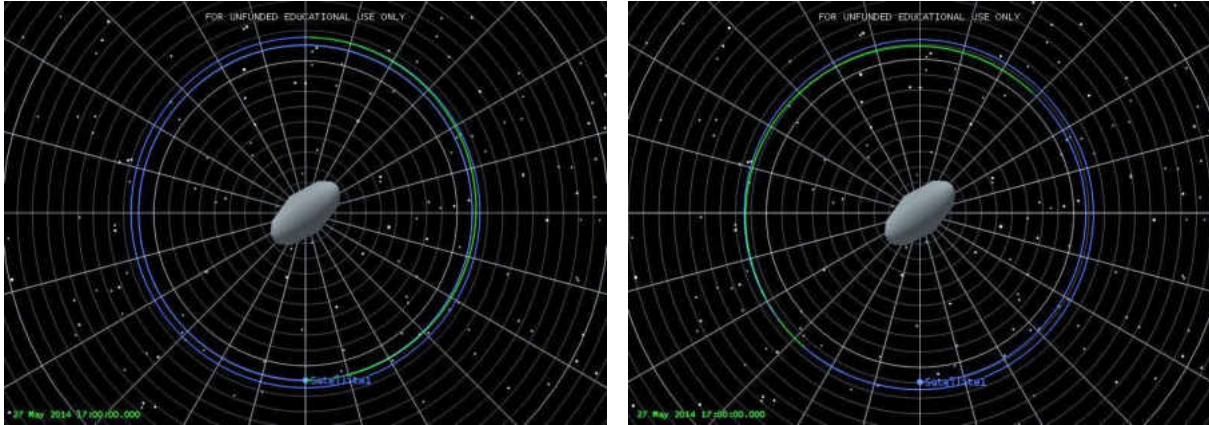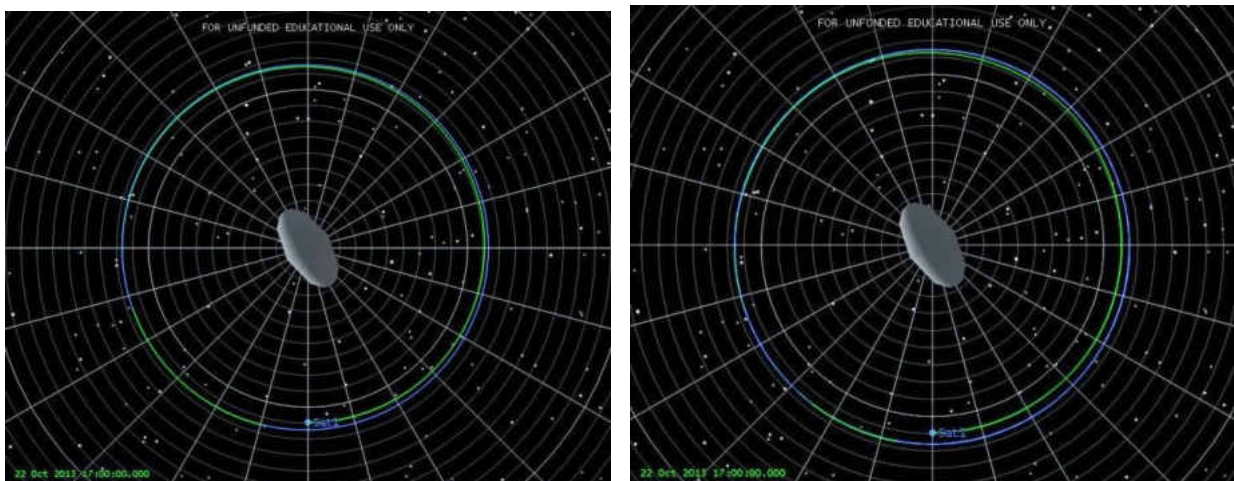

Figure 28. Mid-altitude PID velocity transfer. Shown with the two-body gravity model (left) and the spherical harmonic gravity model (right).
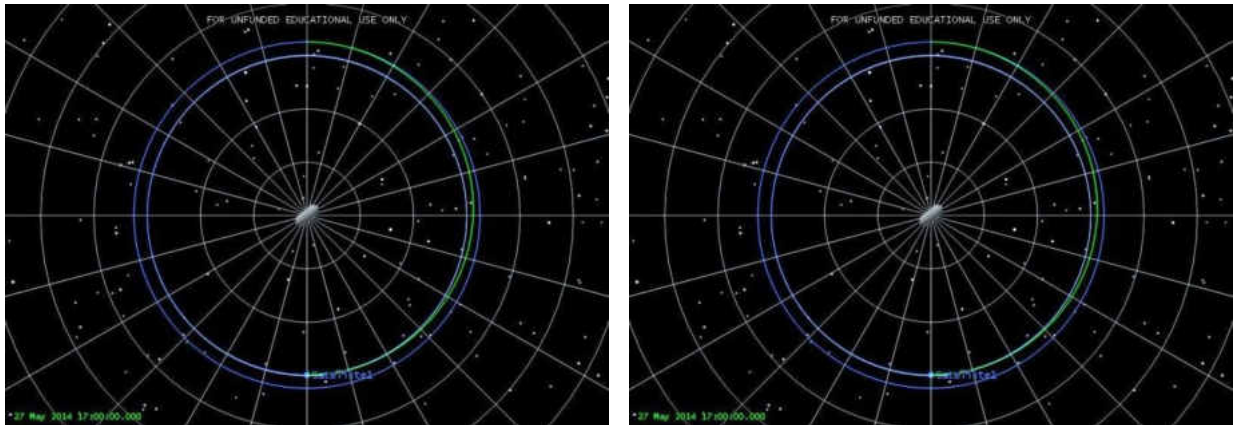
38
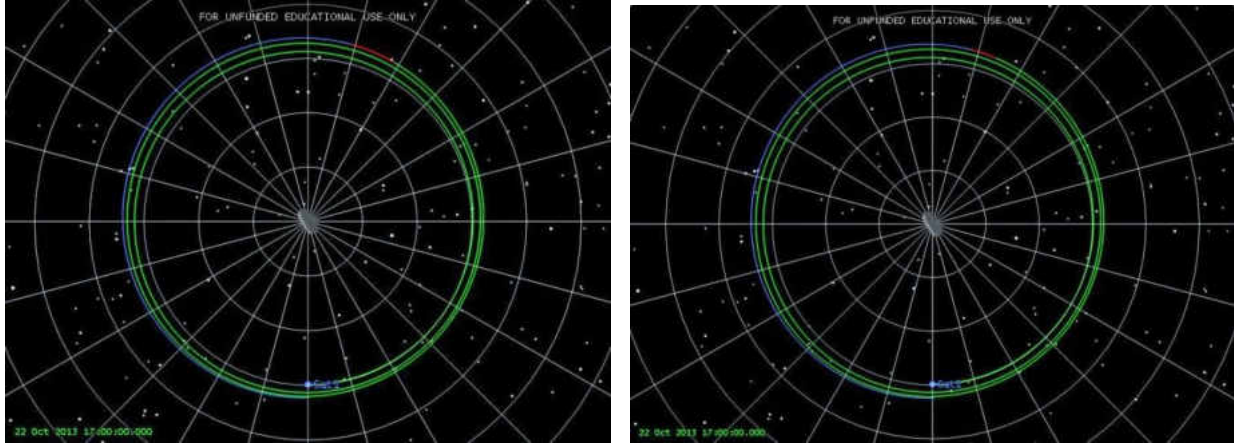
Figure 29. High-altitude PID velocity transfer. Shown with the two-body gravity model (left) and the spherical harmonic gravity model (right).

This approach works in the two-body gravity models, but is less time efficient when compared to the Hohmann transfer. It takes the PID controlled system roughly one and one-third orbits to reach the desired altitude at low altitude, three-quarters of an orbit at mid altitude, and the high-altitude scenario takes multiple orbits to reach the desired altitude. The Hohmann transfer performs the same maneuver in half an orbit in the two-body gravity scenarios.

Now, it is reasonable to assume at this point that increasing the gain levels of the controller will achieve a faster, more time efficient result for the altitude transfer. This would be the case for a linear gravity model. But with the non-linear decrease of the force of gravity as altitude increases, the velocity control system is very sensitive to the tuning process and increasing the gains any further would cause the system to be unstable. In tuning the above systems to get the proper P, I, and D values, slight variations in the parameters (on the order of $\pm 1.00 \times 10^{-8}$) causes instabilities in the system. Furthermore, each region tested requires a new set of P, I, and D gain values. With this in mind, in order to tune the PID controller, when used to control ΔV burns, the satellite-asteroid system must be very well known. This method would

not be effective around a new, unknown asteroid. Table 3 shows the PID gains used for the previous scenarios, which were calculated using a modified Ziegler-Nichols tuning method.

Table 3. PID velocity control gains.

	Low-Altitude	Mid-Altitude	High-Altitude
P	7.88E-08	2.59E-08	1.91E-08
I	1.58E-12	2.76E-13	1.28E-14
D	3.94E-05	2.43E-05	2.87E-04

At a low altitude, with the complex gravity model, the PID velocity controller used here does not handle the perturbations from the complex gravity field very well and the satellite crashes after two and a half orbits.

Furthermore, using a linear time-invariant (LTI) controller in a non-linear time-varying system introduces complications to the problem. The LTI controller is designed to handle LTI systems. As mentioned previously, a linear approximation can be made for small regions of a non-linear system. This may be an explanation for why the controller works well for a small two-body transfer, since the force of gravity is only changing slightly as the orbital radius changes. However, when the linear controller exits the region of the system that is linearly approximated, it becomes unstable. In the complex gravity model, the gravitational force is non-linear and the controller is unable to maintain stability.

Another potential issue may be in the system itself. Classical control theory explains that there may be some non-zero steady-state error, depending on the system. Consider the feedback control system shown in Fig. 30.
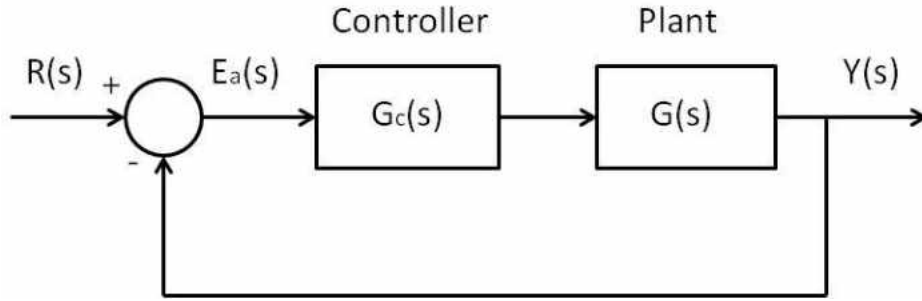
Fig. 30. Feedback Control System

All of the terms are taken as a Laplace transform of a time-dependent system, where the tracking error of the system ($Y(s)$ - $R(s)$) is defined as:

$$E(s) = \frac{1}{1+G_c(s)G(s)} R(s) \tag{10}$$

The final value theorem gives the steady-state (long-term) error of the system as:

$$\lim_{t \to \infty} e(t) = e_{ss} = \lim_{s \to 0} sE(s) \tag{11}$$

With this in mind, the loop transfer function is written in general form as:

$$G_c(s)G(s) = \frac{K \prod_{i=1}^{M}(s+z_i)}{s^N \prod_{k=1}^{Q}(s+p_k)} \tag{12}$$

where $\prod$ denotes the product of the factors and $z_i \neq 0$, $p_k \neq 0$ for any $1 \leq i \leq M$ and $i \leq h \leq Q$.

Now, the steady-state error depends on N. For example, a step input of magnitude A is given as

$$R(s) = \frac{A}{s} \tag{13}$$

This gives

$$e_{ss} = \lim_{s \to 0} \frac{s(\frac{A}{s})}{1+G_c(s)G(s)} = \frac{A}{1+\lim_{s \to 0} G_c(s)G(s)} \tag{14}$$

If N is greater than zero, then $\lim_{s \to 0} G_c(s)G(s)$ approaches infinity, and the steady-state error approaches zero. If $N = 0$, there is a constant stead-state error for a step input of magnitude A that is:

41

$$e_{ss} = \frac{A}{1+K_p} \tag{15}$$

where K_p is defined as the position error constant and is equal to $\lim_{s\to 0} G_c(s)G(s)$. A similar error

exists for a ramp (velocity) input which is given as:

$$R(s) = \frac{A}{s^2} \tag{16}$$

When $N = 0$, the steady-state error in infinite. When $N = 1$, the error is:

$$e_{ss} = \frac{A}{\lim_{s\to 0} sG_c(s)G(s)} = \frac{A}{K_v} \tag{17}$$

where K_v is designated as the velocity error input.

This is all true for a linear system, whereas the system in this project is a nonlinear

system. The stability and system error may be assessed using this procedure; however, this

analysis of the nonlinear system is beyond the scope of this thesis and it is recommended to

perform this study in future work. More information can be found in texts by Dorf & Bishop and

Nise [21, 24].

Since the PID velocity controller used here is so sensitive to tuning, and does not handle

disturbance rejection, it is not an ideal choice for an autonomous orbital control system. It is

more desirable to have a controller that can operate at many different altitudes, or one that can

adapt to the environment, as well as one that can respond fast enough to handle the gravitational

perturbations without introducing instabilities to the system.

Non-linear P Controller

The classical PID controller has constant gains and performs well in a system that does

not change over time. It's been shown that the force of gravity changes non-linearly with

altitude, so it is necessary for a controller to be able to adjust itself to deal with those changes.

Likewise, it is known that the Hohmann transfer uses a thruster burn at the beginning of the maneuver and a thruster burn at the end of the maneuver. It is then necessary to have a velocity controller with a lot of force when the error is large, and a lot of force as the error approaches zero. A type of non-linear PID controller being used in robotics has the ability to change the amount of output provided, based on changes to the input as well as providing varying amounts of force based on the degree of error experienced [25]. One such application of this type of controller is for robotic arms that contact surfaces of varying surface hardness. The robotic arm can adjust the amount of force applied based on whether it is interacting with a hard or soft surface. Without this ability to adapt to its environment, a fixed-gain PID controller would have unpredictable dynamic performance and may become unstable.

For the controller proposed, a P-gain that follows one period of the cosine curve is used. This gain has the form shown in Eq. 18 where A and B are constants and e is the measured error, which is the difference between the desired and current orbital radius as observed by the satellite. Using arbitrary values, Fig. 31 shows the non-linear P-gain for the sequence. At the beginning of the maneuver, when the error is large, the gain is high which translates into large burn values at the start of the maneuver like in the Hohmann transfer. Similarly, as the error approaches zero, the gain is again high and the burns have large magnitudes of thrust.

$$Gain = A(cosBe + 1) \tag{18}$$
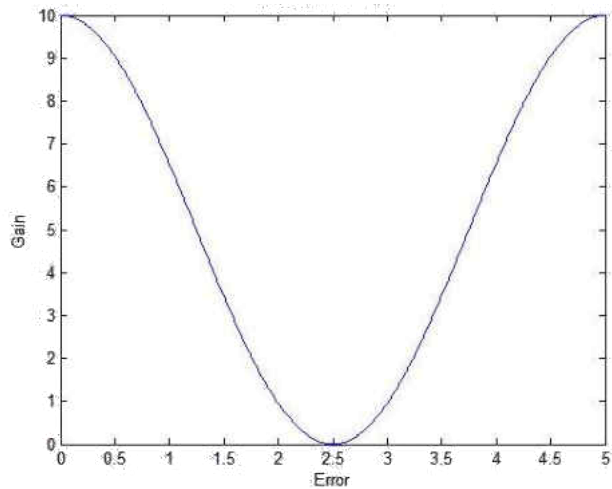
Figure 31. Non-linear gain using arbitrary values.

Figure 32 shows that in the two-body gravity scenario the non-linear P velocity controller is able to perform the altitude correction maneuver in the same amount of time as the Hohmann transfer, or in half an orbit. At close range in the complex gravity scenario, the non-linear P velocity controller isn't unable to handle the perturbations in order to maintain a stable orbit.
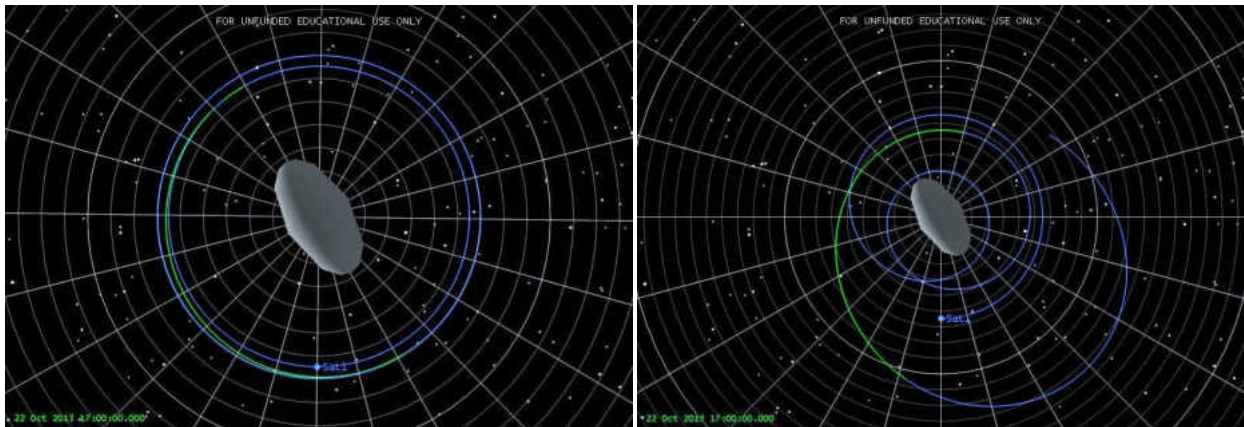
Figure 32. Low-altitude non-linear P transfer. Shown with the two-body gravity model (left) and the spherical harmonic gravity model (right).

Figures 33 and 34 illustrate that in the region of natural stability and at high altitude, the non-linear P velocity controller transfer performs like that of the Hohmann transfer in correcting the initial offset from the desired orbit, and is able to circularize and maintain the new orbit.

44
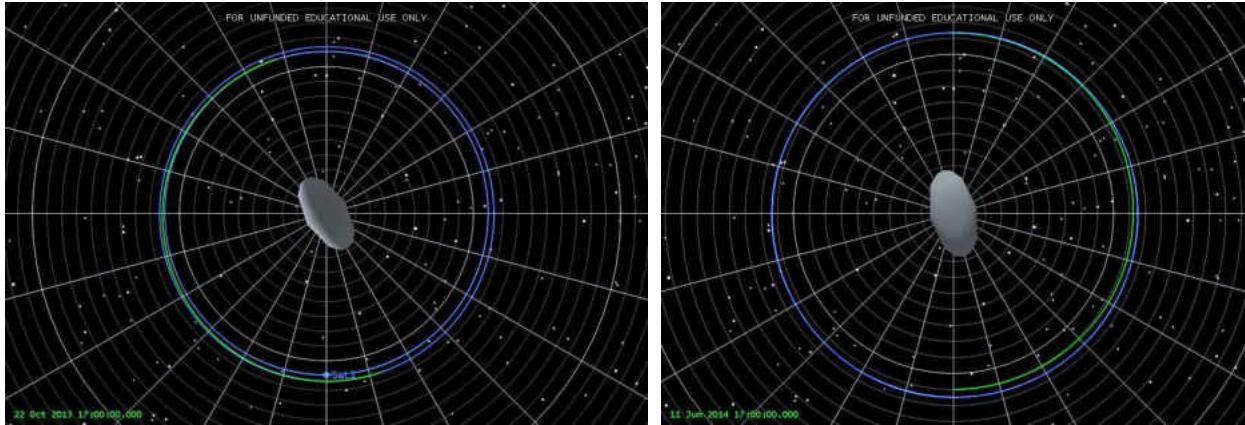
Figure 33. Mid-altitude non-linear P transfer. Shown with the two-body gravity model (left) and the spherical harmonic gravity model (right).
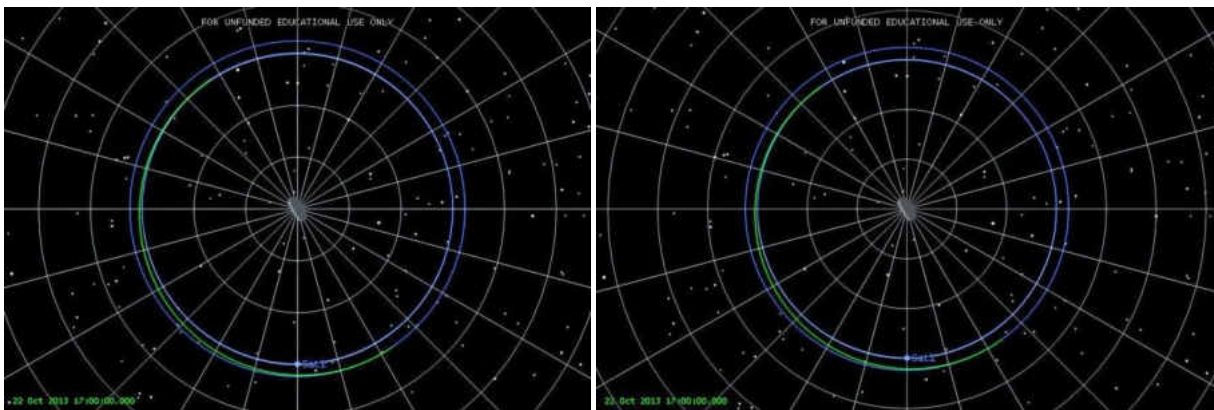
Figure 34. High-altitude non-linear P transfer. Shown with the two-body gravity model (left) and the spherical harmonic gravity model (right).

The results above show that in a two-body system, the non-linear P velocity controller is more time efficient than the standard PID velocity controller and matches that of the Hohmann transfer. In all three regions, the non-linear P velocity transfer portion (green) is half an orbit whereas the classical PID velocity transfer took one and one-third orbits to complete in the low region and nearly three orbits in the high-altitude region. Table 4 shows the percentage decrease for each region in the amount of time required to perform the altitude correction. These are only applicable to the two-body gravity models.

Table 4. Two-body gravity model velocity control performance.

	Low		Mid		High	
	Transfer Time	% Change	Transfer Time	% Change	Transfer Time	% Change
PID velocity control	22 hours	-63.6	38 hours	-44.7	465 hours	-80
Non-linear P vel.	8 hours		21 hours		93 hours	

So the non-linear P velocity controller is more time efficient than the classical PID velocity controller in the two-body gravity models. However, in the low-altitude region with complex gravity turned on, the orbit is again unstable with the non-linear P velocity controller. Furthermore, the non-linear P controller is a continuous controller with orbital mechanics principles that can be used in an autonomous system, whereas the Hohmann transfer is not continuous and cannot be used in an autonomous system.

Both of the velocity control methods tested here are ineffective at disturbance rejection when attempting to maintain a stable orbit at close range. A more direct approach at controlling orbital radius is necessary in order to compensate for the complex gravity model perturbations as the Hohmann-like methods of controlling orbits by way of ΔV burns is ineffective in a complex environment.

Radial Vector Thrust

The previous two methods are referred to by the author as "indirect" control methods, meaning that the orbit radius is controlled by means of adjusting the velocity. From an orbital mechanics standpoint this make sense, but from a controls standpoint, this seems counterintuitive and it has been shown to be ineffective in the highly-variable gravitational environment (Figs. 27 & 32). The last controller to be tested uses the classical PID to directly control the radius of the

orbit by thrusting along the radial vector. That is to say that if the orbit is shrinking, thrusters push the satellite outwards, and if the orbit is growing, thrusters push the satellite inwards.

In the environment of the asteroid being tested, this direct method of control is feasible with the equipment onboard the satellite model. The ion engines provide us with a very low amount of thrust, between 2 mN and 14 mN. But with the levels of gravitation produced by the interaction between the asteroid and satellite, the 8-cm aperture XIPS ion engine gives enough thrust to perform more direct maneuvers without being overcome by gravity. This may not be the case for larger asteroids that would require higher levels of thrust, such as 433 Eros which is 34 km along its long axis versus Itokawa's 500 m. Another consideration for using the indirect control approach was fuel usage.

As mentioned, the Hohmann transfer is typically the most efficient way to transfer from one orbit to another. In a study on fuel usage, a transfer from a circular orbit with a radius of 1.0 km to a circular orbit of 1.1 km about Itokawa was analyzed. The transfer was performed using a Hohmann transfer and the classical PID velocity controller described in the previous sections. Table 5 is a summary of the results observed. The classical PID velocity controller took a much longer time to perform the maneuver, but used a similar amount of fuel when compared to the Hohmann transfer. The satellite model has 5 kg of fuel onboard so extrapolating the data, the PID velocity controller could run continuously at this rate for 6.48×10^7 hours, or roughly 7,399 years before running out of fuel. At this point, it was safe to assume that a more direct control method would not put the mission in danger from a fuel usage perspective.

Table 5. Fuel usage data for a Hohmann transfer vs. a PID controller transfer

	Hohmann Transfer	PID Controller	% Change
Fuel Used (kg)	3.3E-06	3.7E-06	13.6
TOF (hrs)	19.3	49	153.6

Again, two things are being tested in these scenarios: (1) a small orbit transfer that is representative of an orbital perturbation, and (2) disturbance rejection after the transfer is performed.

Figure 35 shows that in both the two-body gravity scenario and the complex gravity model, the altitude correction happens quickly at the bottom of each frame. In the complex gravity model, the PID radius controller is able to respond quickly and compensate for the perturbation caused by the changes in the force of gravity.
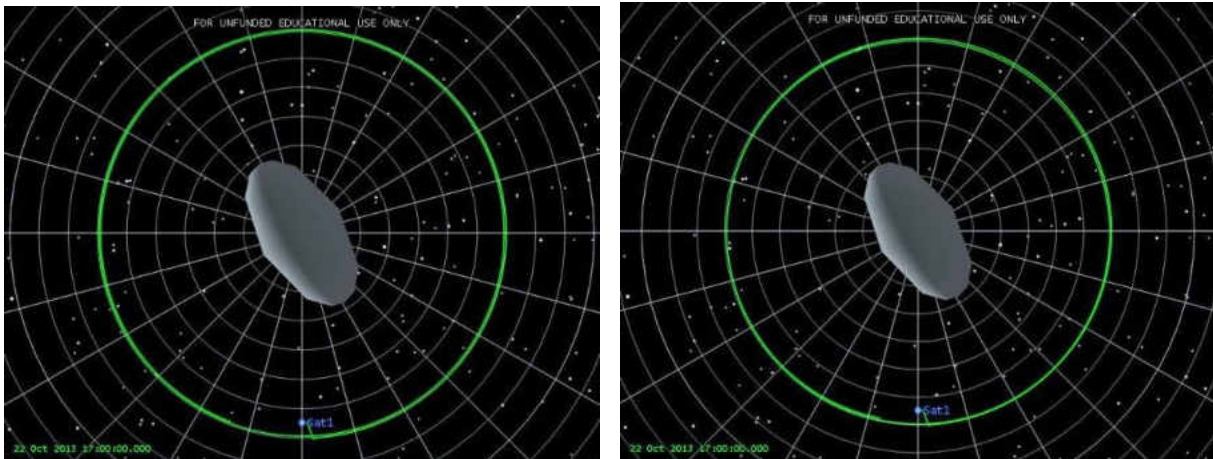
Figure 35. Low-altitude radial thrust transfer. Shown with the two-body gravity model (left) and the spherical harmonic gravity model (right).

Figures 36 and 37 illustrate that in the region of natural stability and at high altitude, the PID radius controller performs the altitude correction quickly at the bottom of each frame and maintains a circular orbit.
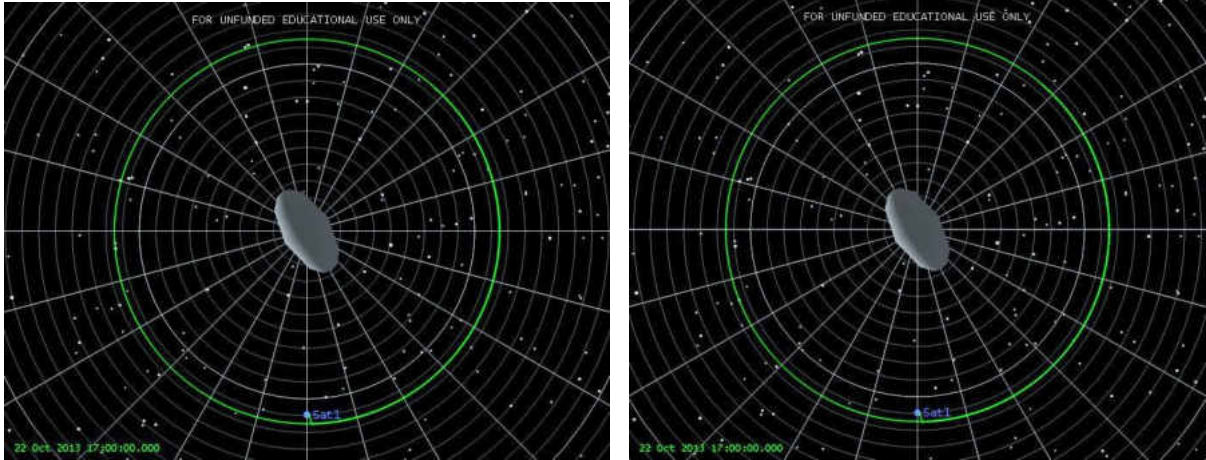
Figure 36. Mid-altitude radial thrust transfer. Shown with the two-body gravity model (left) and the spherical harmonic gravity model (right).
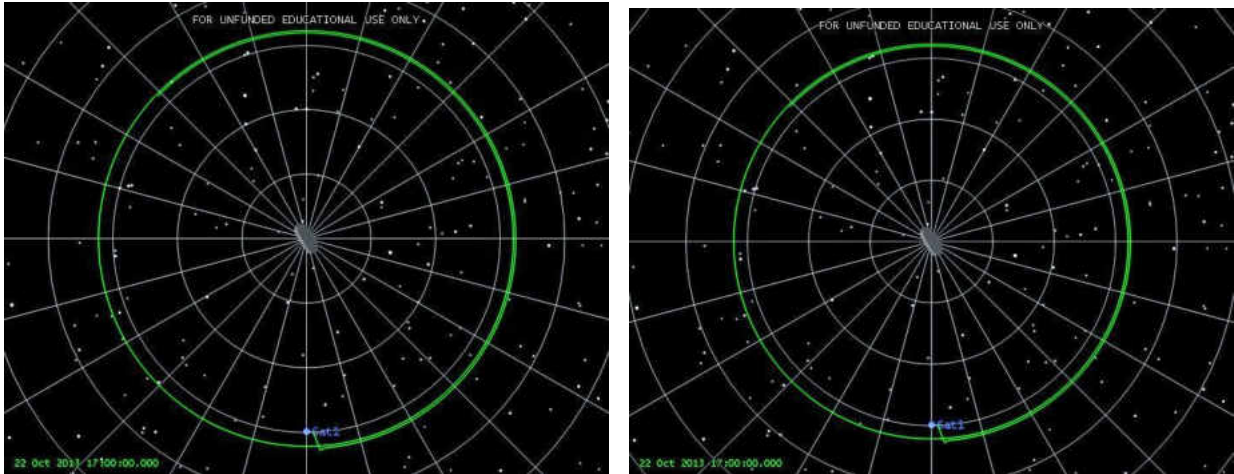
Figure 37. High-altitude radial thrust transfer. Shown with the two-body gravity model (left)

and the spherical harmonic gravity model (right).

Although it's not easy to see, the orbit transfer takes place almost immediately in all of these scenarios. There is a small section at the very bottom of each screen shot that shows the satellite being pushed outwards to the desired altitude. Each propagation step is five minutes long so the satellite is moving outwards a distance of 50 meters in a time of five minutes, or 0.17 m/s. The caveat is that the satellite's velocity in the direction of the orbit is not what is required

at the new orbit, so the controller has to keep pushing outwards to maintain the higher radius. The controller is thrusting outwards all the time at the new altitude.

For circular orbits, there is a specific orbital velocity required in order to maintain the orbital radius. This comes from the specific mechanical energy of the orbit:

$$\varepsilon = \frac{V^2}{2} - \frac{\mu}{R}$$

(19)

where

ε =   spacecraft's mechanical energy (km^2/s^2)

V   =   magnitude of the spacecraft's velocity vector (km/s)

μ =   G * (mass of the asteroid), known as the gravitational parameter (km^3/s^2)

R   =   magnitude of the spacecraft's position vector (km).

so when the altitude increases, the velocity must also decrease from the Law of the Conservation of Energy. Since this doesn't happen in the simulated scenarios, the controller has to continually thrust outwards in order to make up for the discrepancy.

At low altitude with complex gravity turned on, the controller is able to maintain a circular orbit. It rejects the disturbance provided by the influence of the complex gravity field.

As a final note on this controller, it needs to be said that this controller is robust in the environment being tested. The previous two control schemes are very sensitive to the tuning parameters and require a new set of control gains for each of the low, mid, and high-altitude regions. This requires the satellite-asteroid system to be very well known so that the controller can be properly tuned before the mission launch. The PID radius controller does not follow suit. It only requires one set of P, I, and D gain values for all three regions of the Itokawa model and the initial tuning is a lot less sensitive. Table 6 displays the control gains used in all three

regions. This controller, once tuned, can be used for circular orbits at any altitude in the complex

gravity model of Itokawa.

Table 6. Radial vector thrust control gains

	All Regions
P	2.05E-06
I	1.59E-12
D	1.06

Experimental Results

The table below, Table 7, gives the results from the scenarios shown previously.

Displayed is the transfer time of each controller and its performance with disturbance rejection,

or its ability to maintain the circular orbit with the perturbations from the complex gravity field.

It can be seen that the PID radial vector thrust controller is the fastest at all regions and the most

effective in rejecting the disturbance caused by the complex gravity field perturbations.

Table 7. Summary of results.

		Low		Mid		High	
		Transfer Time	Disturbance Rejection	Transfer Time	Disturbance Rejection	Transfer Time	Disturbance Rejection
Two-Body	Hohmann	8 hours	N/A	21 hours	N/A	93 hours	N/A
	PID velocity	22 hours	N/A	38 hours	N/A	465 hours	N/A
	Non-linear P vel.	8 hours	N/A	21 hours	N/A	93 hours	N/A
	PID radius	5 minutes	N/A	5 minutes	N/A	5 minutes	N/A
Complex Gravity Model	Hohmann	Unstable	Poor	21 hours	N/A	93 hours	N/A
	PID velocity	Unstable	Poor	38 hours	N/A	465 hours	N/A
	Non-linear P vel.	Unstable	Poor	21 hours	N/A	93 hours	N/A
	PID radius	5 minutes	Good	5 minutes	N/A	5 minutes	N/A

To recap, the Hohmann transfer is the most efficient way to transfer between circular, co-planar orbits in a two-body point mass system. The PID velocity controller is able to perform a transfer in a two-body point mass model, but it is less time efficient than the Hohmann transfer. The non-linear P velocity controller performs much like the Hohmann transfer, but it is controlled by the negative feedback loop instead of hand calculating each maneuver. The PID radius controller is a very fast controller but the satellite's velocity in the direction of the orbit isn't changed to match the requisite velocity of the new orbit so it has to push the satellite outwards all the time to make up for the discrepancy in specific mechanical energy.

The Hohmann transfer, the PID velocity controller, and the non-linear P velocity controller did not perform well with disturbance rejection in the complex gravity field. They all became unstable when perturbed by the changing force of gravity. The PID radius controller was able to maintain a circular orbit by rejecting the disturbance of the changing gravity.

Putting this all together, a low-altitude scenario was created using the PID radius controller and the PID velocity controller together. The PID radius controller gets the satellite to the desired orbit and the PID velocity controller adjusts the velocity of the satellite to match that of the orbital velocity of the new orbit. It can be seen in Fig. 38 that inside the complex gravity field, the PID radius controller (green portion) is still working to reject the disturbance of the complex gravity field but the fuel used to maintain the orbit is less than before since the satellite is travelling at the appropriate speed for the new orbit.
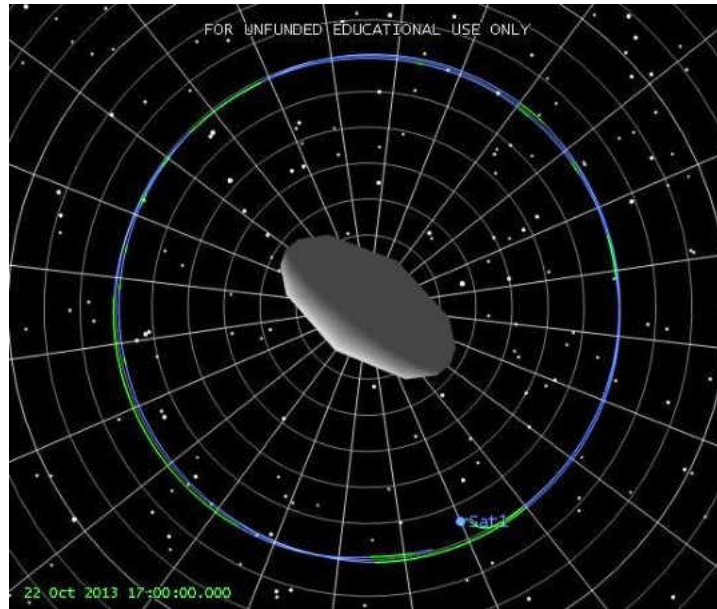
Figure 38. Combination of radius and velocity control.

Table 8 shows the ΔV and fuel usage comparison between the PID radius controller and the combination controller. The combination controller uses almost 2/3 the amount of fuel of the PID radius controller. Over the period of one day, or just under two orbits (shown), the amount of fuel used is pretty small but once used, there isn't a refueling tanker in deep space that can come by and replenish the satellite. Planning for fuel usage is critical in the mission's success. So one might think that the radius controller is only using 0.685 grams of fuel each day, so implementing a more complex controller isn't really worth the added effort. But over the course of a long mission, the fuel saved might be the difference between returning to Earth, and being lost in space forever.

Table 8. Controller comparison.

	PID Radius Controller	PID Radius and Velocity Controller	% Change
Total DV (km/s)	4.6E-04	2.8E-04	-38.9
Fuel Used (kg)	6.9E-04	4.2E-04	-38.9

CHAPTER IV

OBSERVATIONS AND ANALYSIS

It was observed through the course of the study that orbit stability depends on the excitation frequency provided by the spherical harmonic coefficients. Retrograde orbits are more stable than prograde orbits since the relative speed between the satellite and the surface of Itokawa is much higher, thus the satellite experiences a much higher excitation frequency due to the spherical harmonic gravity coefficients. The following is a technical discussion on this observation.

In order to capture the irregularities of the gravity model in three dimensions, a spherical harmonic model of the gravitational potential is used, as shown in Eq. 20.

$$U = \frac{\mu}{r} \sum_{n=0}^{\infty} \sum_{m=0}^{n} \left\{ \left\{ \left(\frac{R_{CB}}{r} \right)^n P_{nm} sin\theta cos(m\lambda) \right\} C_{nm} + \left\{ \left(\frac{R_{CB}}{r} \right)^n P_{nm} sin\theta cos(m\lambda) \right\} S_{nm} \right\} \quad (20)$$

where U is gravitational potential. The coordinates r, θ, and λ are the radial distance, latitude, and longitude of the spacecraft in a coordinate system fixed to the object's center of mass. R_{CB} is the mean radius for the body and μ is the object's gravitational parameter. The functions P_{nm} are the normalized Legendre polynomials, and C_{nm} and S_{nm} are the gravity coefficients of degree n and order m [26]. For Itokawa's gravity model, the C_{20}, C_{22}, C_{42}, and C_{44} coefficients are the most significant, as shown in Table 9 [14]. The C_{20} and C_{22} values correspond with the effects of the asteroid's oblateness and its ellipcity.

Table 9. Itokawa normalized gravity coefficients for a constant density gravity field [14].

Order l	Degree m	C Coefficient C_{lm}	S Coefficient S_{lm}
0	0	1.0	–
1	0	0.0	–
1	1	0.0	0.0
2	0	-0.145216	–
2	1	0.0	0.0
2	2	0.219420	0.0
3	0	0.036115	–
3	1	-0.028139	-0.006137
3	2	-0.046894	-0.046894
3	3	0.069022	0.033976
4	0	0.087852	–
4	1	0.034069	0.004870
4	2	-0.123263	0.000098
4	3	-0.030673	-0.015026
4	4	0.150282	0.011627

A body's oblateness and its effect on orbits has been characterized and studied extensively as we see this perturbation in Earth orbits. Due to the Earth's angular velocity as it spins about its polar axis, there is a bulge around the equator. The effect this bulge has on orbits (known as the J_2 effect) causes a precession in the orbital plane. Specifically, the orbit's right ascension of the ascending node (RAAN) (Ω) rotates westward for prograde orbits around the Earth, and the argument of periapsis (ω) rotates in the direction of the spacecraft's motion. Semi-major axis (a), eccentricity (e), and inclination (i) suffer no long-term perturbations from oblateness [27].

A body's ellipcity has more dramatic effects on the orbit and can cause the spacecraft to transition from a safe orbit into an impacting or ejecting orbit within a few periods. The ellipcity of the body causes changes in the orbit semi-major axis, eccentricity, and inclination while

55

effecting both the orbits energy $\varepsilon = -\mu/(2a)$ and angular momentum $h = [\mu a(1-e)]^{1/2}$ [28]. In previous studies, it has been observed that prograde orbits experience much larger changes in energy and angular momentum for each orbit, where retrograde orbits experience little, if any, changes per orbit [29].

To look at this further from a system dynamics viewpoint, we recall that the equation of motion for the simplified two-body system is:

$$F = ma_x + \frac{m\mu}{x^3}x \qquad (21)$$

where

m = mass of the spacecraft (kg)

a_x = acceleration in the x-direction (m/s²)

μ = G * (mass of the asteroid), known as the gravitational parameter (km³/s²)

x = distance between the asteroid and the spacecraft.

From mechanical vibrations, the undamped natural frequency of a linear spring-mass system is defined as:

$$\omega_n = \sqrt{\frac{k}{m}} \qquad (22)$$

where

k = spring constant (N/m)

m = mass of the spacecraft (kg).

This is the frequency at which an undamped linear system will naturally oscillate. This is an important parameter in vibrations because if a system is driven to oscillate at this frequency, small excitations grow into large amplitudes of oscillation.

This amplification ratio at which oscillations grow when an undamped system is excited at its natural frequency is given as:

$$\frac{X}{\delta_{st}} = \frac{1}{1-(\frac{\omega}{\omega_n})^2}$$

(23)

where X/δ_{st} is the ratio of the dynamic to the static amplitude of motion, and ω/ω_n is the ratio of the excitation to the natural frequency of the system [16]. Figure 39 is a plot of ω/ω_n versus X/δ_{st} for an undamped linear system. From this we expect to see instabilities in the spacecraft's orbit as the excitation frequency experienced by the spacecraft from the rotating asteroid approaches the natural frequency of the system.
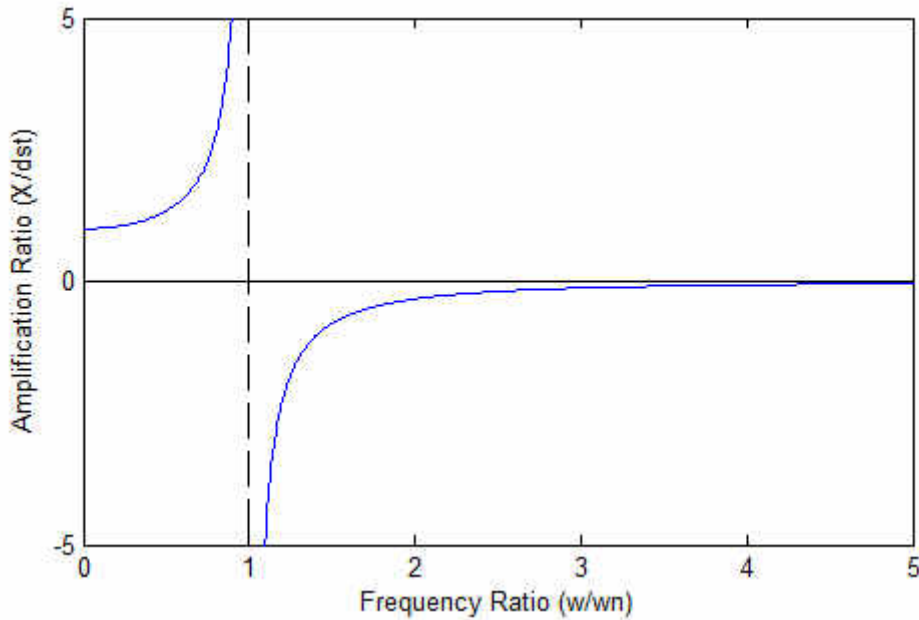
Figure 39. Amplification ratio of an undamped linear system.

Replacing the spring constant (k) in Eq. 22 with the "non-linear spring" constant, it becomes:

$$\omega_n = \sqrt{\frac{\frac{m\mu}{x^3}}{m}}$$

(24)

57

which simplifies to:

$$\omega_n = \sqrt{\frac{\mu}{x^3}} \qquad (25)$$

and this leaves us with the natural frequency of the system in radians per second (rad/s).

Scheeres provides that μ for Itokawa is 2.36×10^{-9} km³/s² [14]. This gives the natural frequency

of the spring-mass analogous system to be 1.05×10^{-4} rad/s when evaluated at the chosen initial

orbital radius of 0.60 km. As mentioned before this is the frequency at which a linear spring-

mass system will naturally oscillate, if disturbed, when considering gravity to be a spring.

This term is also seen in orbital mechanics and is called mean motion, n, which is the

mean angular velocity of the body in orbit. This equation for mean motion comes from Kepler's

third law and is useful in orbital analysis [30]. Furthermore, mean motion resonance can occur

when two bodies have periods of revolution that are integer ratios of each other. This can lead to

stabilization or destabilization of orbits, depending on the details. A notable occurrence of this is

in the main belt asteroids where some mean motion resonances with Jupiter have a significant

amount of asteroids (3:2), while others are practically devoid of asteroids (3:1, also known as a

Kirkwood gap) [31]. These ratios describe the number orbits of an asteroid compared to the

number of orbits of Jupiter around the sun. So, at the semimajor axis where an asteroid makes

three orbits around the Sun in the same period that Jupiter make two orbits, there is a cluster of

asteroids in that semimajor axis region. Conversely, at the semimajor axis where an asteroid

would make three orbits around the Sun for each of Jupiter's orbits, there are virtually no

asteroids in this semimajor axis region.

In order to study the effects of the excitation frequency, the STK simulation is utilized.

Through this, we can change the rotational rate of the asteroid. This speeds up or slows down

the frequency of excitation experienced by the spacecraft from the spherical harmonic gravity model of asteroid Itokawa.

With the spherical harmonic model, the relative frequency between the satellite and the asteroid can be expressed as:

$$\omega_{sat/asteroid} = (\omega_{sat} \pm \omega_{asteroid}) \tag{26}$$

where the frequencies add in the retrograde case, and subtract in the prograde case. With the C_{44} coefficient in the spherical harmonic model, we see that the gravitational force reaches a maximum value four times throughout a single rotation of Itokawa (Fig. 21). This results in the excitation frequency:

$$\omega = m\omega_{sat/asteroid} \tag{27}$$

where m is the highest order of the gravity model significantly impacting the excitation. This implies that the satellite experiences *m* points of peak gravitational force from the complex gravity field. This gives the frequency of harmonic gravitational excitation of the satellite-asteroid system as the asteroid spins.

Multiple scenarios were simulated in STK varying the rate of rotation of the asteroid and the direction of the orbit. This allows for the study of the interaction between the excitation frequency (ω), and the natural frequency (ω_n). Observations are given in Table 10. The asteroid spin rate multiplier is the number that Itokawa's rotational rate is multiplied by, resulting in a slower of faster spin of the asteroid in order to study the effect of the harmonic excitation frequency on orbital stability.

Asteroid Spin Rate Multiplier	Prograde Frequency Ratios (ω/ω_n)	Orbit Quality	Event	Retrograde Frequency Ratios (ω/ω_n)	Orbit Quality	Event
1/32	-3.8	stable		4.2	unstable	Impact
1/16	-3.7	stable		4.3	semi-stable	High Precession
1/8	-3.3	stable		4.7	stable	
1/4	-2.6	stable		5.4	stable	
1/3	-2.2	stable		5.8	stable	
1/2	-1.2	unstable	Impact	6.8	stable	
3/4	0.1	unstable	Impact	8.1	stable	
7/8	0.8	unstable	Impact	8.8	stable	
Natural Rate 1	1.5	unstable	Impact	9.5	stable	
1 1/2	4.3	unstable	Impact	12.3	stable	
1 3/4	5.6	semi-stable	Eject	13.6	stable	
2	7.0	semi-stable	Eject	15.0	stable	
3	12.5	stable		20.5	stable	
4	18.0	stable		26.0	stable	
5	23.5	stable		31.5	stable	
6	29.0	stable		37.0	stable	
7	34.5	stable		42.5	stable	

It is observed that when the ratio of the excitation frequency to the natural frequency (ω/ω_n) is $-1.3 < \omega/\omega_n < 4.3$, the orbit is unstable. Examples of unstable and stable orbits are shown in Fig. 40. Figure 40 shows the asteroid at its natural rate of rotation with a satellite travelling with the direction of rotation (prograde). The satellite diverges from its intended circular orbit path and crashes into Itokawa. Figure 40 also shows a satellite travelling in the opposite direction (retrograde). The satellite maintains an orbit that is mostly circular, but the orbit rotates around the body as the Right Ascension of the Ascending Node undergoes precession caused by the changing gravitational magnitude. The orbit is perturbed, but the frequency ratio of the excitation frequency to the natural frequency of the system is outside of the amplified region that causes the orbit to become unstable.

Figure 40. Example of an unstable prograde orbit and a stable retrograde orbit. Both orbits shown have the same initial conditions, but are separated by 180° of inclination.

The time history of the prograde system at the asteroid's natural rate of rotation, seen in Fig. 41, shows that the force of gravity pulling the satellite inwards increases (negatively) as the satellite crashes into the surface of the asteroid. The retrograde time-history is interesting in that it shows that there is more than one excitation frequency. A higher frequency oscillation is superimposed over the lower frequency motion.

Figure 42 shows the Fast Fourier Transform (FFT) analysis of the radial acceleration of the system at the asteroid's natural rate of rotation. The FFT is useful because any peaks seen in the graph illustrate a dominant frequency at which the system was excited. The prograde system is unstable from the very beginning, so we don't expect to see any dominant frequencies. The retrograde FFT is again shows the two dominant excitation frequencies. These potentially correspond with the C_{22} and C_{44} coefficients, though further analysis is required.
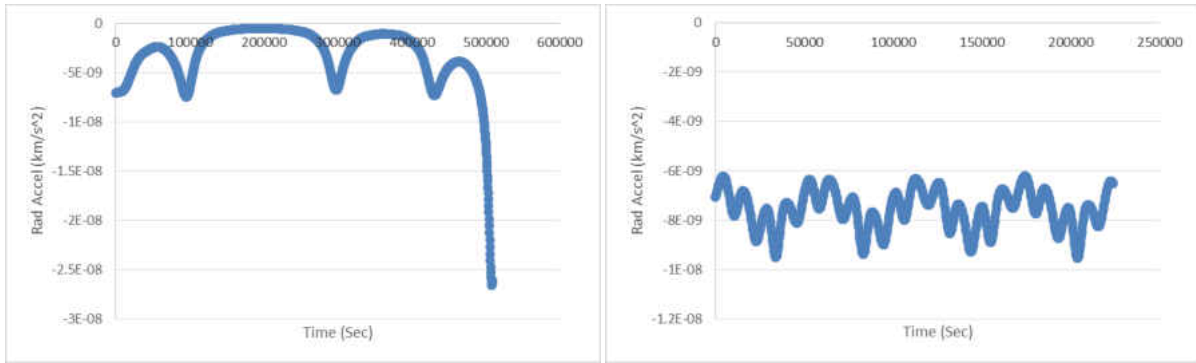
Figure 41. Time-history of the radial acceleration data. The data shown is for an unstable prograde orbit (left) and a stable retrograde orbit (right) at the asteroid's natural rotation rate.
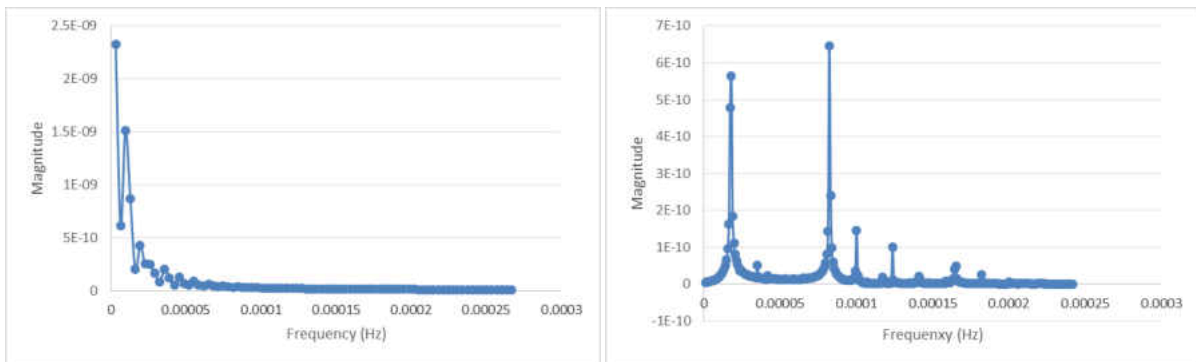
Figure 42. FFT of the radial acceleration data. The data shown is for an unstable prograde orbit (left) and a stable retrograde orbit (right) at the asteroid's natural rotation rate.

It is interesting to note that within $-1.3 < \omega/\omega_n < 4.3$ the orbit becomes unstable immediately and crashes into the asteroid within very few periods. However, at the values of 5.6 and 7.0, as seen in Table 10, the orbit is stable for some time and then at one point in the orbit is ejected from the system.
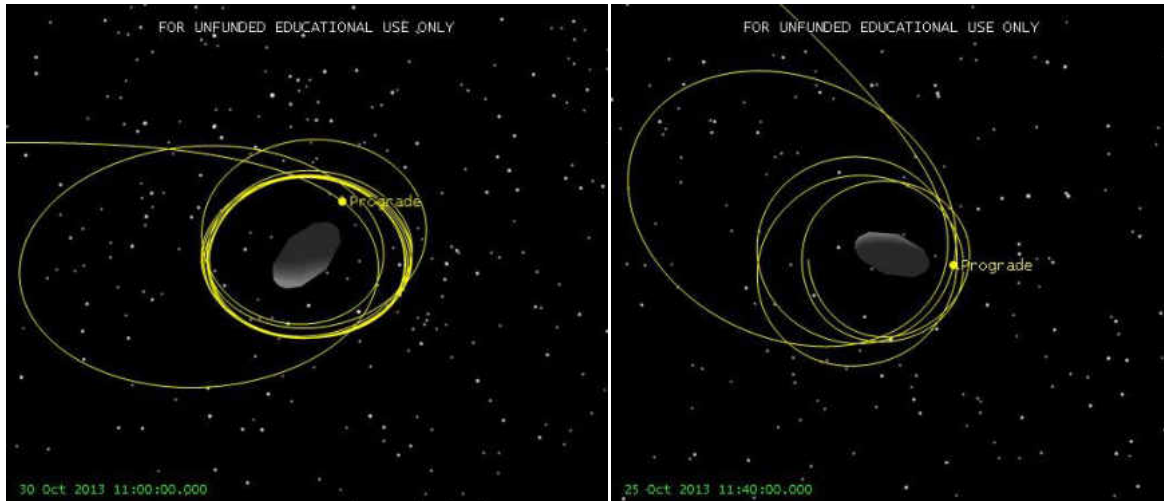
Figure 43. Escaping orbits. Rotation of Itokawa sped up by 1 3/4 times the natural rate (left) and by 2 times the natural rate (right).

Figure 43 shows the satellite at the point just before it is ejected from the system when ω/ω_n is 5.6 and 7.0, respectively. Just before the ejection, the satellite enters a highly eccentric orbit and has a close approach to the protruding end of the body at the periapsis. As the satellite passes by the asteroid, a small amount of energy is transferred to the satellite as it is then accelerated into a hyperbolic orbit [18].

CHAPTER V

CONCLUSIONS AND RECOMMENDATIONS

It was found that a PID controller can be used to control orbit radius under the specific

conditions tested in the study. These conditions are that the satellite is in a circular orbit within a

single orbital plane using a model that includes only a degree and order four spherical harmonic

gravity model of Itokawa and the rotational effects from the asteroid.

The controllers experimented with in this project are simple controllers and don't require

a lot of computing power. This is advantageous in satellite systems because it is necessary for

the system to respond quickly without sending up heavy-duty hardware. Furthermore, the aim is

to have software uploaded to the satellite before launch, and then send it off to perform the

mission. Work done by researchers at the University of Surrey have proposed a system that uses

optical navigation cameras to take pictures of the entire surface of a target asteroid in order to

build a complex 3D model. Then the data is sent back to Earth and analyzed on the ground while

the satellite is put into a safe orbit. On the ground, a simplified mass model is built and a

complex control scheme is sent back to the satellite [32]. This method has potential for high-

precision maneuvers like sampling and landing, but the work done in this project has shown that

a very simple radial vector thrust controller can be used to carry out basic maneuvers in a

complex environment without any further input from mission control.

The use of a harmonic excitation analysis on the linearized system can be used predict the

regions where orbital stability is expected. By computing the excitation frequency as a function

of the rotational spin rate and order of the gravity model, a comparison can be made to the natural frequency of the satellite-asteroid system. This provides important information regarding the likelihood of stable, long-duration orbits.

It was found that there is a range of excitation frequencies that causes an orbiting spacecraft around Itokawa to become unstable. There are other factors that need to be considered before attempting close-proximity orbits, but it is recommended that a spacecraft attempting to orbit small body asteroids (around 500 m in diameter) enter into an orbit that has a ω/ω_n ratio greater than seven in order to avoid unstable perturbations caused by the spherical harmonic gravity coefficients. For retrograde orbits, the ratio of ω/ω_n is increased due to the higher relative speed of the satellite to the surface of the asteroid as it rotates. This effect is similar to the impact harmonic excitations have on linear systems; relatively high frequency excitations have small impact on the dynamic response of the system, while excitations near the resonant frequency (ω_n) have a large impact on the dynamic response of the system. The same basic response is observed in satellite orbits around an asteroid where a non-linear spring restoring force is coupled with complex gravitational excitation.

Using the results from this project, the author proposes that in order to achieve a stable orbit about an asteroid the PID radial vector thrust controller be combined with the PID velocity controller in order to maximize thrust effectiveness and efficiency in the complex gravitational environment. These controllers can be tuned on the ground before launch using ground-based observational data and simulation of a small target asteroid. The mass of the asteroid can be estimated using spectroscopy data, which is a method of determining the material that the asteroid is made of by analyzing the light that it reflects. Once the mass is estimated, the gravitational parameter can be calculated and a simple model can be built for simulation. The

harmonic excitation analysis data can be utilized in to find regions of natural stability, as well as determine orbits with high ω/ω_n ratios that exhibit stability. From this the control gains can be determined and uploaded to the system before launch.

Opportunities for future work include to first add complexities into the asteroid model. Some of the complexities include, but are not limited to, SRP, third-body gravitational effects from the planets, thermal radiation effects, and solar reflectivity of the asteroid effects. It was seen in early work that these effects can have a large impact on the trajectory of the satellite. These effects are predictable phenomena that have been studied and can be compensated for with added complexity to the current control system. Next would be to study the impact of attitude control on the current system. Then satellite systems and packaging study can be performed to investigate different thruster and attitude control configurations, and their impact on fuel usage, flight dynamics, and mission feasibility. After a thorough study of a mission around Itokawa, the autonomy of the system can be expanded. Itokawa is a well-known asteroid and the control system built in this research was not tested around any other asteroids, known or unknown. The end goal is to create a control system that can approach a completely unknown asteroid, and enter into close-range stable orbit without any external input. And finally, a complete mission design from launch to re-entry and landing can be built and proposed to a government space agency or a commercial space agency for an asteroid mission with great opportunities for space science and exploration.

APPENDICES

Appendix A
Hohmann Transfer Orbit

The Hohmann transfer is the most fuel efficient way to go from one orbit to another. We limit the Hohmann transfer to:

- Co-planar orbits

- Circular orbits

- Instantaneous velocity changes (Delta-Vs or ΔVs) are tangent to the initial and final orbits

The Delta-V is provided by the ion thruster with a "burn", where a small amount of fuel is expended. Typically for an ion thruster, which has a low magnitude of thrust, the thruster is activated over an extended period and the ΔV is definitely not impulsive. However, with the weaker gravity field of a small NEO, the amount of time required for the thruster to be activated is many times smaller (21,021 times smaller) than the orbital period. Therefore, it can be assumed that the maneuver is impulsive. To simplify matters, the burns are considered to be instantaneous because the amount of time that the engine fires is very short compared to the orbital period. This assumption makes it so that it isn't necessary to integrate the thrust over the amount of time in order to calculate the ΔV.

Whenever the velocity is changed, the orbit's specific mechanical energy, ε, is changed

$$\varepsilon = \frac{V^2}{2} - \frac{\mu}{R} = -\frac{\mu}{2a} \tag{28}$$

where

ε   =   spacecraft's mechanical energy (km^2/s^2)

V   =   magnitude of the spacecraft's velocity vector (km/s)

μ   =   G * (mass of the asteroid), known as the gravitational parameter (km^3/s^2)

R   =   magnitude of the spacecraft's position vector (km)

a   =   semimajor axis (km).

The right-hand side of Eq. 28 shows that when we change the energy of the orbit, the size of the orbit's semimajor axis changes as well. From this basic principle, we can calculate the magnitude of the burns needed to perform the Hohmann transfer.
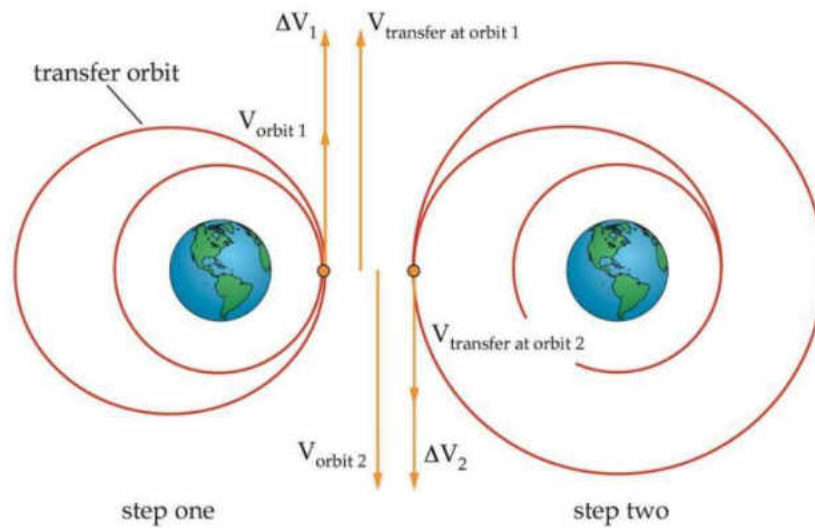
Figure 44. Hohmann transfer orbit [18].

The Hohmann transfer is performed in two steps:

1. ΔV1 raises the apoapsis of the orbit, putting the spacecraft into the elliptical transfer orbit

2. ΔV2 raises the periapsis of the transfer orbit, putting the spacecraft into the circular final orbit.

The process works the same, in reverse, to transfer from a high orbit to a low orbit. To find the ΔVs, we need to find the energy of each orbit. Since the orbits are circular, we can use the semimajor axis to find specific mechanical energy.

$$\varepsilon_{orbit\ 1} = -\frac{\mu}{2a_{orbit\ 1}} \tag{29}$$

$$\varepsilon_{orbit\ 2} = -\frac{\mu}{2a_{orbit\ 2}} \tag{30}$$

The transfer orbit's semimajor axis is the sum of the two orbital radii:

$$2a_{transfer} = R_{orbit\ 1} + R_{orbit\ 2} \tag{31}$$

It follows that:

$$\varepsilon_{transfer} = -\frac{\mu}{2a_{transfer}} \tag{32}$$

Knowing the energies, we can find the orbital velocities by re-arranging Eq. 28

$$V_{orbit\ 1} = \sqrt{2(\frac{\mu}{R_{orbit\ 1}} + \varepsilon_{orbit\ 1})} \tag{33}$$

$$V_{orbit\ 2} = \sqrt{2(\frac{\mu}{R_{orbit\ 2}} + \varepsilon_{orbit\ 2})} \tag{34}$$

$$V_{transfer\ at\ orbit\ 1} = \sqrt{2(\frac{\mu}{R_{orbit\ 1}} + \varepsilon_{transfer})} \tag{35}$$

$$V_{transfer\ at\ orbit\ 2} = \sqrt{2(\frac{\mu}{R_{orbit\ 2}} + \varepsilon_{transfer})} \tag{36}$$

Now it's a matter of evaluating the difference of the orbital velocities to get ΔV1, ΔV2, and the

ΔV$_{total}$ for the maneuver

$$\Delta V_1 = \left|V_{transfer\ at\ orbit\ 1} - V_{orbit\ 1}\right| \tag{37}$$

$$\Delta V_2 = \left|V_{orbit\ 2} - V_{transfer\ at\ orbit\ 2}\right| \tag{38}$$

$$\Delta V_{total} = \Delta V_1 + \Delta V_2 \tag{39}$$

Appendix B
MATLAB and STK Interface

STK is a powerful tool that can help simulate complex orbit problems through the use of its built-in High Precision Orbit Propagator. It also has many built-in functions that assist in mission design. However, it can be very tedious if you wish to use a more hands-on approach. Some of the scenarios in this work had over 1,000 maneuvers. This would require clicking to add each maneuver, calculate the burn for the maneuver, and then clicking to add a propagate segment after each maneuver. On top of that, it is required to go into each maneuver and propagation to change the central body to reference Itokawa, instead of Earth (which is the default). This process would be painfully inefficient. Thankfully, this is where MATLAB comes in.

MATLAB is used here to:

- Open STK

- Set up a new scenario with Itokawa as the central body

- Add a satellite with given parameters (mass, area, etc.)

- Start the orbit

- Implement the control algorithm being used

- Pull the satellite's state data at a defined time interval

- Calculate and apply the correction maneuvers, if needed

- Run the simulation to the next time interval

As mentioned above, each scenario can have up to 1,000 iteration steps. MATLAB just chugs away, calculating and applying the needed corrections. Some of the larger scenarios take a full day to run, but that is a fraction of the time it would take to do manually.

Getting Started

Three things are required to get the scenarios up and running:

1) A licensed version of STK with Astrogator installed.

2) A licensed version of MATLAB installed.

3) The STK-MATLAB connector installed from http://www.agi.com/.

Once you have those up and running, you can beginning running scenarios with the planets in the

Solar System, as they come with Astrogator.

There are some built-in tutorials that teach the basics of using STK that can be found

through the main menu bar under Help > User Resources. They can also be found at

http://www.agi.com/training/. There are also some code samples that can be found under User

Resources in the Help menu or at http://www.agi.com/resources/library/code-snippets-samples/.

For the code that follows, it was necessary to develop custom central body models for

Itokawa and the custom ion engine model that was used. Those custom models are referenced in

the code, so it will not run unless those models are in the proper STK data folders. It is

important that all of the models that are being used for the project are named properly. In the

code, it calls each object and since these are custom models the user designates the name for

each one when it is created. This is especially important when collaborating. Each user must

have all of the models named the exact same in order to run the code on different machines.

NOTE: It helps, but isn't required to have an understanding of Object-Oriented

Programming as well as calling external functions in MATLAB, particularly calling COM

objects. There is documentation with examples on these subjects available at

http://www.mathworks.com/help/matlab/ under the Advanced Software Development section.

AGI has a MATLAB connector that allows the use of STK via the TCP/IP connecter, which

allows the use of the Connect Command Library. To do this, however, your machine needs to be running a 32-bit version of MATLAB since STK is 32-bit. Using the COM interface removes compatibility issues between 32-bit and 64-bit software.

The first step is establishing the connection between STK and MATLAB. This is done with the following MATLAB code:

```matlab
% This script establishes the link between STK and MATLAB

try
    % Grab an existing instance of STK 10 if one is available
    uiapp = actxGetRunningServer('STK10.application');

catch
    % If STK is not already running, launch a new instance of STK10 and
    % grab it
    uiapp = actxserver('STK10.application');

end

% Get the root from the personality.  There are two of them, use the one
% designated as "Personality2"
root = uiapp.Personality2;

% Set visible to true (shows the STK GUI)
uiapp.visible = 1;
```

Having run this, it's time to open a new scenario, set up the parameters, and run the control loop:

```matlab
%% Low Altitude Transfer Scenario Around Itokawa Using Radial Vector Thrust

% Close current scenario (if one is open) and open a new scenario with
% Itokawa as the central body
try
    root.CloseScenario();
    root.ExecuteCommand('New / Scenario Itokawa_VectorThrust CentralBody
    Itokawa');

catch
    root.ExecuteCommand('New / Scenario Itokawa_VectorThrust CentralBody
    Itokawa');

end
```

```matlab
%% Set up the Scenario Time and Then Change Units to Epoch Seconds

% Set units to UTCG before setting scenario time period
root.UnitPreferences.Item('DateFormat').SetCurrentUnit('UTCG');

% Set scenario time period
root.CurrentScenario.SetTimePeriod('22 Oct 2013 17:00:00.000','22 Dec 2013
17:00:00.000');

% Set units to epoch seconds (easiest for data analysis in MATLAB)
root.UnitPreferences.Item('DateFormat').SetCurrentUnit('EPSEC');

% Turn the ECI Radial Grid on for visual reference
root.ExecuteCommand('VO * Grids Space ShowECI On ShowRadial On');


%% Create satellite and Activate Astrogator

satObj = root.CurrentScenario.Children.New('eSatellite', 'Sat1');

% Change the propogator to Astrogator
satObj.SetPropagatorType('ePropagatorAstrogator');

% Define the Astrogator driver and MCS main sequence
driver = satObj.Propagator;
main = driver.MainSequence;

% Remove all default sequences from the MCS main sequence
main.RemoveAll


%% Add a Sattelite and Configure the Sattelite's Initial State

% Set sattelite epoch to scenario start time
Initial = main.Insert('eVASegmentTypeInitialState','Initial State','-');
Initial.Properties.DisplayCoordinateSystem = 'CentralBody/Itokawa Inertial';
Initial.OrbitEpoch = 0;

% Change sattelite coordinate system to reference Itokawa
% and change initial state parameter type to Keplarian
Initial.CoordSystemName = 'CentralBody/Itokawa Inertial';
Initial.SetElementType('eVAElementTypeKeplerian')

% Define satellite initial state using Keplerian Elements
keplerian = Initial.Element;
keplerian.SemiMajorAxis = 0.65;
keplerian.Eccentricity = 0.005;
keplerian.Inclination = 0;
keplerian.RAAN = 0;
keplerian.ArgOfPeriapsis = 0;
keplerian.TrueAnomaly = 0;
```

```matlab
% Define spacecraft parameters (setting all to zero turns off SRP)
SCParam = Initial.SpacecraftParameters;

SCParam.Cd = 0;
SCParam.Cr = 0;
SCParam.Ck = 0;
SCParam.K1 = 0;
SCParam.K2 = 0;

% Define spacecraft dry mass in kg
SatDryMass = 50;
SCParam.DryMass = SatDryMass;

% Define the fuel mass in kg
SatFuelMass = 5;
SCFuel = Initial.FuelTank;
SCFuel.FuelMass = SatFuelMass;

% Define the spacecraft mass
SatMass = SatDryMass + SatFuelMass;

%% Define control parameters

DesiredAltP = 0.65;      % Desired parking altitude in km
DesiredAltF = 0.70;      % Desired final altitude in km

NSteps = 500;            % Number of iteration steps
StepT = 300;             % Step time in seconds

% P, I, and D are the gains for the controller
% Uses the Ziegler-Nichols classical PID tuning method for no overshoot

Ku = 1.025e-05;      % Ultimate gain.  High enough gain to cause system
oscillation
Tu = StepT*5150;     % Period of oscillation in seconds (Step time * # of
iterations)

P = 0.2*Ku;
I = 2*P/Tu;
D = P*Tu/3;

% pre-allocate the arrays as zeros, to be filled in later
% E is the Error, dV is the delta V array, iE is the Riemann Sum of the
Error, and dE is the derivative of the Error
E = zeros(NSteps+1,1);
dV = zeros(NSteps+1,1);
iE = zeros(NSteps+1,1);
dE = zeros(NSteps+1,1);
```

```matlab
%% Initial Propogation Iterations and Data Collection

% Activation commands for the data providers
scen = root.CurrentScenario;

% Change for additional satellites
sat = scen.Children.GetElements('eSatellite').Item('Sat1');

% Add a single 'Propagate' segment
Propagate = main.Insert('eVASegmentTypePropagate','Propagate1','-');
Propagate.Properties.DisplayCoordinateSystem = 'CentralBody/Itokawa
Inertial';
Propagate.Properties.Color = 8;
Propagate.PropagatorName = 'Itokawa Prop';
StopCon = Propagate.StoppingConditions;
Duration = StopCon.Item('Duration');
Duration.properties.Trip = StepT;

% Run the Mission Control Sequence
driver.RunMCS;

% Pull the Classical Orbital Elements for the satellite and the Itokawa
position data
satDP = sat.DataProviders.Item('Classical
Elements').Group.Item('Inertial').Exec(scen.StartTime,scen.StopTime,StepT);
satTime = cell2mat(satDP.DataSets.GetDataSetByName('Time').GetValues);
satSMA = cell2mat(satDP.DataSets.GetDataSetByName('Semi-major
Axis').GetValues);
satEcc = cell2mat(satDP.DataSets.GetDataSetByName('Eccentricity').GetValues);
satInc = cell2mat(satDP.DataSets.GetDataSetByName('Inclination').GetValues);
satRAAN = cell2mat(satDP.DataSets.GetDataSetByName('RAAN').GetValues);
satAoP = cell2mat(satDP.DataSets.GetDataSetByName('Arg of
Perigee').GetValues);
satTA = cell2mat(satDP.DataSets.GetDataSetByName('True Anomaly').GetValues);

% Pull the Satellite orbital position vector data
PosDP =
sat.DataProviders.Item('Vectors(Inertial)').Group.Item('Position').Exec(scen.
StartTime,scen.StopTime,StepT);
SatRAsc =
cell2mat(PosDP.DataSets.GetDataSetByName('RightAscension').GetValues);
SatDec = cell2mat(PosDP.DataSets.GetDataSetByName('Declination').GetValues);
SatRMag = cell2mat(PosDP.DataSets.GetDataSetByName('Magnitude').GetValues);

% Define the Initial Error for the first iteration (placeholder for data
% extraction)
E(2) = DesiredAltP - SatRMag(2);


%% Begin the control loop for the transfer

for count=2:1:NSteps
```

```matlab
% Measure the error, the Riemann's Sum of the error,
% and the differential of the error
E(count) = DesiredAltF - SatRMag(count);
iE(count) = sum(E)*StepT;
dE = diff(E)/StepT;


% Define the correction Delta V
dV(count) = (P*E(count)) + (I*iE(count)) + (D*dE(count-1));



% Add Impuslive Thrust Vector 'Maneuver' segment
Maneuver1 = main.Insert('eVASegmentTypeManeuver','Maneuver2','-');
Maneuver1.Properties.DisplayCoordinateSystem = 'CentralBody/Itokawa
  Inertial';
Maneuver1.Properties.Color = 21;
Maneuver1.Maneuver.SetAttitudeControlType('eVAAttitudeControlThrust
Vector');
Maneuver1.Maneuver.SetPropulsionMethod('eVAPropulsionMethodEngineModel',
'New Custom Ion Engine');
ManeuverAttitudeControl = Maneuver1.Maneuver.AttitudeControl;
ManeuverAttitudeControl.ThrustAxesName = 'Itokawa Inertial';
ManeuverAttitudeControl.AllowNegativeSphericalMagnitude = 1;

if dV(count) < 0 && SatRAsc(count) < 0
    ManeuverAttitudeControl.DeltaVVector.AssignSpherical(SatDec(count),
    SatRAsc(count)+180, abs(dV(count)));

elseif dV(count) < 0 && SatRAsc(count) > 0
    ManeuverAttitudeControl.DeltaVVector.AssignSpherical(SatDec(count),
    SatRAsc(count)-180, abs(dV(count)));

elseif dV(count) > 0
    ManeuverAttitudeControl.DeltaVVector.AssignSpherical(SatDec(count),
    SatRAsc(count), dV(count));
end

% Add a single 'Propagate' segment
Propagate1 = main.Insert('eVASegmentTypePropagate','Propagate1','-');
Propagate1.Properties.DisplayCoordinateSystem = 'CentralBody/Itokawa
Inertial';
Propagate1.Properties.Color = 1;
Propagate1.PropagatorName = 'Itokawa Prop';
StopCon = Propagate1.StoppingConditions;
Duration = StopCon.Item('Duration');
Duration.properties.Trip = StepT;

% Run the Mission Control Sequence
driver.RunMCS;
```

```matlab
    % Pull the Classical Orbital Elements for the satellite
    satDP = sat.DataProviders.Item('Classical
    Elements').Group.Item('Inertial').Exec(scen.StartTime,scen.StopTime,
    StepT);
    satTime = cell2mat(satDP.DataSets.GetDataSetByName('Time').GetValues);
    satSMA = cell2mat(satDP.DataSets.GetDataSetByName('Semi-major
    Axis').GetValues);
    satEcc = cell2mat(satDP.DataSets.GetDataSetByName('Eccentricity').
    GetValues);
    satInc = cell2mat(satDP.DataSets.GetDataSetByName('Inclination').
    GetValues);
    satRAAN = cell2mat(satDP.DataSets.GetDataSetByName('RAAN').
    GetValues);
    satAoP = cell2mat(satDP.DataSets.GetDataSetByName('Arg of
    Perigee').GetValues);
    satTA = cell2mat(satDP.DataSets.GetDataSetByName('True
    Anomaly').GetValues);

    % Pull the Satellite orbital position vector data
    PosDP = sat.DataProviders.Item('Vectors(Inertial)').Group.Item
    ('Position').Exec(scen.StartTime,scen.StopTime,StepT);
    SatRAsc = cell2mat(PosDP.DataSets.GetDataSetByName('RightAscension').
    GetValues);
    SatDec = cell2mat(PosDP.DataSets.GetDataSetByName('Declination').
    GetValues);
    SatRMag = cell2mat(PosDP.DataSets.GetDataSetByName('Magnitude').
    GetValues);

end


%% Reset the Animation
root.ExecuteCommand('Animate * Reset');
root.CurrentScenario.Animation.AnimStepValue = 60;
```

The result is a simulation in STK that uses a small satellite to orbit Itokawa using the PID

radius controller at low altitude with complex gravity turned on and all other effects turned off.
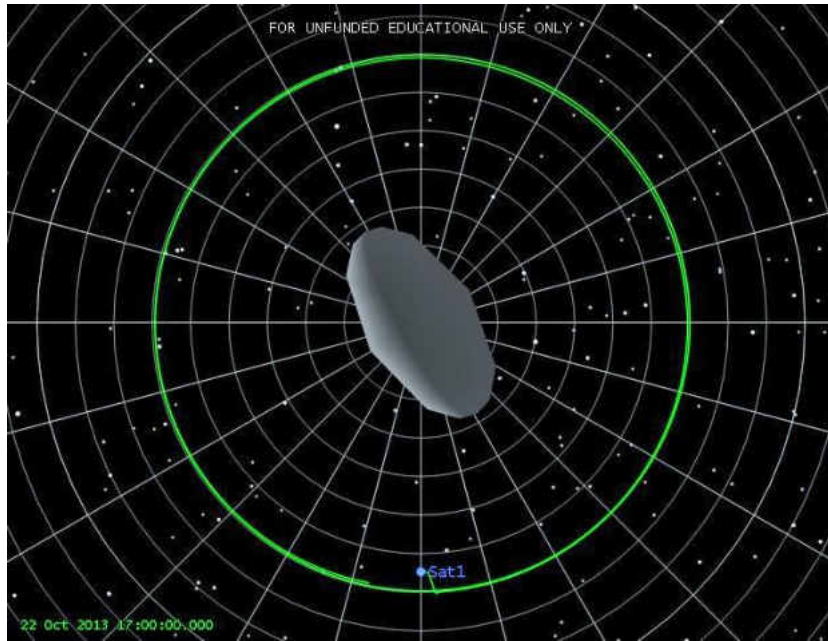
Figure 45. Low-altitude radial vector thrust transfer with spherical harmonic gravity.

REFERENCES

[1]   Williams, D. R., 2014, "Mars Fact Sheet," from
       http://nssdc.gsfc.nasa.gov/planetary/factsheet/marsfact.html

[2]   Lewis, J. S., 1996, *Mining the Sky: Untold Riches from the Asteroids, Comets, and
       Planets*, Addison-Wesley Pub. Co., Reading, MA

[3]   Dasso, R. D., 2010, "NASA Scientist Confirms Water Ice on Asteroid," from
       http://www.nasa.gov/topics/earth/features/water_ice_asteroid.html

[4]   Popova, O. P. , Jenniskens P., Emel'yanenko, V., Kartashova, A., Biryukov E.,
       Khaibrakhmanov, S., Shuvalov, V., Rybnov, Y., Dudorov, A., Grokhovsky, V. I.,
       Badyukov, D. D., Yin, Q., Gural, P. S., Albers, J., Granvik, M., Evers, L. G., Kuiper, J.,
       Kharlamov, V., Solovyov, A., Rusakov, Y. S., Korotkiy, S., Serdyuk, I., Korochantsev A.
       V., Larionovm, M. Y., Glazachev, D., Mayer A. E., Gisler, G., Gladkovsky, S. V.,
       Wimpenny, J., Sanborn, M. E., Yamakawa, A., Verosub, K. L., Rowland, D. J., Roeske,
       S., Botto, N. W., Friedrich, J. M., Zolensky, M. E., Le, L, Ross, D., Ziegler, K.,
       Nakamura, T., Ahn, I., Lee, J. I., Zhou, Q., Li, X., Li, Q., Liu, Y., Tang, G., Hiroi, T.,
       Sears, D., Weinstein, I. A., Vokhmintsev, A. S., Ishchenko, A. V., Schmitt-Kopplin, P.,
       Hertkorn, N., Nagao, K., Haba, M. K., Komatsu, M., Mikouchi, T., and the Chelyabinsk
       Airburst Consortium, 2013, " Chelyabinsk Airburst, Damage Assessment, Meteorite
       Recovery, and Characterization," Science, 342 (6162), pp. 1069 - 1073

[5]   Brown, P. G., Assink, J. D., Astiz L., Blaauw R., Boslough M. B., Borovicka J., Brachet,
       N., Brown, D., Campbell-Brown, M., Ceranna, L., Cooke, W., de Groot-Hedlin, C., Drob
       D. P., Edwards, W., Evers, L. G., Garces, M., Gill, J., Hedlin, M., Kingery, A., Laske, G.,
       Le Pichon, A., Mialle, P., Moser, D. E., Saffer, A., Silber, E., Smets, P., Spalding, R. E.,
       Spurny, E., Tagliaferri, E., Uren, D., Weryk, R. J., Whitaker, R., and Krzeminski, Z.,
       2013, "A 500-kiloton Airburst Over Chelyabinsk and an Enhanced Hazard From Small
       Impactors," Nature, 503 (7475), pp. 238 – 241

[6]   Church, C. and Fevig, R., 2013, "A Feasibility Study on the Characterization of the
       Internal Structure of Small NEOs with Small Spacecraft," Abstract #2999 at *LPSC XLIV,*
       Lunar and Planetary Institute, Houston, TX.

[7]   Yeomans, D., 2014, "NEO Groups," from http://neo.jpl.nasa.gov/neo/groups.html

[8]   Yeomans, D., 2014, "Orbit Diagrams," from http://neo.jpl.nasa.gov/orbits/

[9]   Kawaguchi, J., Uesugi, K., and Fujiwara, A., 2003, "The MUSES-C Mission for the Sample and Return - Its Technology Development Status and Readiness," *Acta Astronautica*, Vol. 52, pp. 117 - 123

[10]   Kuninaka, H., Nishiyama, K., Funaki, I., Shimizu, Y., Yamada, T., and Kawaguchi, J., 2006, "Assessment of Plasma Interactions and Flight Status of the Hayabusa Asteroid Explorer Propelled by Microwave Discharge Ion Engines," *IEEE Transactions on Plasma Science*, Vol. 35, No. 5, pp. 2125 - 2132

[11]   Kawaguchi, J., Fujiwara, A., and Uesugi, T., 2005, "Hayabusa (MUSES-C) – Rendezvous and Proximity Operation," Paper No. IAC-05-A3.5.A.01, International Astronautical Congress, International Astronautical Federation, Fouduoka, Japan

[12]   Kuninaka, H., 2008, "Microwave Discharge Ion Engines onboard Hayabusa Asteroid Explorer," *AIP Conference Proceedings*, American Institute of Physics, Melville, NY, Vol. 997, pp. 572 - 581

[13]   Tsuda, Y., Yoshikawa, M., Abe, M., Minamino, H., and Nakazawa, S., 2013, "System Design of the Hayabusa 2 – Asteroid Sample Return Mission to 1999 JU3," *Acta Astronautica*, Vol. 91, pp. 356 - 362

[14]   Scheeres, D. J., Gaskell, R., Abe, S., Barnouin-Jha, O., Hashimoto, T., Kawaguchi, J., Kubota, T., Saito, J., Yoshikawa, M., Hirata, N., Mukai, T., Ishiguro, M., Kominato, T., Shirakawa, K., and Uo, M., 2006, "The Actual Dynamical Environment About Itokawa," AIAA 2006-6661, *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, American Institute of Aeronautics and Astronautics, Inc., Keystone, CO

[15]   Church, C., 2014, "A Feasibility Study on the Implementation of Satellite-to-Satellite Tracking Around a Small Near-Earth Object," M.S. Thesis, Department of Space Studies, University of North Dakota

[16]   Rao, S. S., 2011, *Mechanical Vibrations*, Prentice Hall, Upper Saddle River, NJ, 5th Ed., Chap. 2 - 3

[17]   Coetsee, J. A., 1994, "Control on Nonlinear Systems Represented in Quasilinear Form," Massachusetts Institute of Technology, Cambridge, MA

[18]   Sellers, J. J., Astore, W. J., Griffen, R. B., and Larson, W. J., 2005, *Understanding Space: An Introduction to Astronautics*, McGraw-Hill Companies, Inc, New York, NY, 3rd Ed., Chap. 5-7

[19]   Astrom, K. J. and Murray, R. M., 2012, *Feedback Systems: An Introduction for Scientists and Engineers*, Princeton University Press, Princeton, NJ, Chap. 10

[20]   Rogers, R., 2013, "Building a Benchtop PID Controller," Paper No. 3243, Kiethley Instruments, Inc., Cleveland, OH

[21]   Dorf, R. C., Bishop, R. H., 2008, *Modern Control Systems*, Pearson Prentice Hall, Upper Saddle River, NJ, 11th Ed., Chap. 5

[22]   Kruizinga, A., 2005, from
http://www.csr.utexas.edu/grace/gallery/gravity/ggm01_euro2.html

[23]   Kauderer, A., 2012 "Orbital Elements," from
http://spaceflight.nasa.gov/realdata/elements/graphs.html

[24]   Nise, N. S., 2011, *Control Systems Engineering*, John Wiley & Sons, Inc., Hoboken, NJ, 6th Ed., Chap. 10

[25]   Seraji, H., 1997, "A New Class of Nonlinear PID Controllers for Robotic Applications," Jet Propulsions Laboratory, California Institute of Technology, Pasadena, CA

[26]   Montenbruck, O. and Gill, E., 2000, *Satellite orbits: Models, methods, and applications*, Springer Science & Business Media, Heidelberg, Germany, pp. 53 - 116

[27]   Scheeres, D. J., Ostro, S. J., Hudson, R. S., and Werner, R. A., 1996, "Orbits Close to Asteroid 4769 Castalia," *Icarus*, 121, pp. 67 – 87

[28]   Scheeres, D. J., Marzari, F., Tomasella, L., and Vanzani, V., 1998, "Rosetta Mission: Satellite Orbits Around a Cometary Nucleus," *Planetary Space Science*, 36(6/7), pp. 649 – 671

[29]   Scheeres, D. J., Williams, B. G., and Miller, J. K., 2000, "Evaluation of the Dynamic Environment of an Asteroid: Applications to 433 Eros," *Journal of Guidance, Control, and Dynamics*, 23 (3), pp. 466 – 475

[30]   Vallado, D.A., 2013, *Fundamentals of Astrodynamics and Applications*, Microcosm Press, Hawthorne, CA, Chap. 2

[31]   Nesvorny, D., Ferraz-Mello, S., Holman, M., and Morbidelli, A., 2002, "Regular and Chaotic Dynamics in the Mean-motion Resonances: Implications for the Structure and Evolution of the Asteroid Belt." *Asteroids III*, 1, pp. 379-394.

[32]   Turconi, A., Palmer, P., and Roberts, M., 2014, "Efficient Modelling of Small Bodies Gravitational Potential for Autonomous Approach," Paper IAC-14-C1.7.9 at *International Astronautical Congress*, International Astronautical Federation, Toronto, ON