Development and Implementation of the SPHERES-Slosh Experiment

by

Dustin Luke Hayhurst

B.S., Astronautical Engineering (2012)
United States Air Force Academy

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of

Master of Science in Aeronautics and Astronautics

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2014

Signature redacted

Author .....
Department of Aeronautics and Astronautics
May 22, 2014

Signature redacted

Certified by .....
Alvar Saenz-Otero
Principal Research Scientist
Thesis Supervisor

Signature redacted

Accepted by .....
Paulo C. Lozano
Associate Professor of Aeronautics and Astronautics
Chair, Graduate Program Committee

Disclaimer: The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, the United States Department of Defense, or the United States Government.

Development and Implementation of the SPHERES-Slosh Experiment

by

Dustin Luke Hayhurst

Submitted to the Department of Aeronautics and Astronautics
on May 22, 2014, in partial fulfillment of the
requirements for the degree of
Master of Science in Aeronautics and Astronautics

Abstract

Understanding fluid slosh in a microgravity environment has the potential to make liquid-fueled spacecraft more effective. The SPHERES-Slosh Experiment gathers benchmarking data that can be used to validate computational fluid dynamics models for microgravity environments. This thesis discusses the design of the SSE hardware and the control theory used on the International Space Station during test sessions. This thesis proposes future control theory and design work.

Thesis Supervisor: Alvar Saenz-Otero
Title: Principal Research Scientist

Acknowledgments

Thanks to my family for their love and support.

Contents

List of Figures

List of Tables

List of Acronyms and Initialisms

AFS - Aurora Flight Sciences
ASPR - Almost Strictly Positive Real
CFD - Computational Fluid Dynamics
DARPA - Defense Advanced Research Projects Agency
DOF - Degree of Freedom
DSP - Digital Signal Processor
EKF - Extended Kalman Filter
ESA - European Space Agency
FIT - Florida Institute of Technology
FTDI - Future Technology Devices International
GMS - Global Metrology System
GSP - Guest Scientist Program
GUI - Graphical User Interface
InSPIRE - International Space Station SPHERES Integrated Research Experiments
ISS - International Space Station
JEM - Japanese Experiment Module
LED - Light Emitting Diode
LIIVe - Low Impact Inspection Vehicle
MIT - Massachusetts Institute of Technology
NASA - National Aeronautics and Space Administration
PADS - Position and Attitude Determination System
PD - Proportional-Derivative
PID - Proportional-Integral-Derivative
RGA - Reduced Gravity Aircraft
RINGS - Resonant Inductive Near-field Generation System
SAB - Slosh Avionics Box
SPHERES - Synchronized Position Hold Engage and Reorient Experimental Satellites
SAC - Simple Adaptive Controller
SSE - SPHERES-Slosh Experiment
SSL - Space Systems Laboratory
STAMP - Systems-Theoretic Accident Model and Processes
STPA - Systems-Theoretic Process Analysis
UART - Universal Asynchronous Receiver/Transmitter
UMD - University of Maryland
US - Ultra Sound
USB - Universal Serial Bus

VAS - VERTIGO Avionics Stack
VERTIGO - Visual Estimation for Relative Tracking and Inspection of Generic Objects

Chapter 1

Introduction

1.1 Motivation

Fluid slosh can have a markedly negative impact on the control of spacecraft in a microgravity environment. Specifically, Gangadharan points out that "Energy loss through the movement of the liquid fuel in the fuel tank affects the gyroscopic stability of the spacecraft and leads to nutation (wobble) which can cause devastating control issues." [1] Examples of the detrimental effects of fluid slosh on spacecraft include the mission failure of the ATS-V spacecraft in 1969[2], the unexpected behavior of the Intelsat IV series spacecraft, and multiple problems with ESA spacecraft. Even under nominal circumstances, such as during the orbital injection of a spacecraft using a liquid-fueled upper stage, fuel slosh has a considerable impact on attitude control. Despite a history of problems caused by fluid slosh and its longstanding influence on many control challenges, predicting slosh motion continues to present a considerable difficulty. [3] Burke further emphasizes the relevance of fluid slosh when she states, "Propellant fluid slosh in spacecraft can couple into the dynamics of the vehicle leading to performance degradation in the propulsion and attitude stabilization subsystems. [...] Understanding the dynamics of this fluid slosh is essential to the design of mitigation techniques such as attitude control, fluid baffles, etc."[4]

One way to work towards a better understanding of the behavior of fluids in microgravity and the mitigation of the associated detrimental behavior is to create

models of fluid slosh in microgravity. These models may take the form of scaled model testing, heuristic methods such as neural networks and other learning algorithms, or computational fluid dynamics (CFD) models. One promising aspect of CFD models is that they could potentially be used in both simulations and operational controllers. While CFD models have been applied to fluid slosh, they have not been fully matured in their application to capillary-dominated regimes. [3] One reason for this lack of maturity is that the scientific and engineering community lacks a substantial set of pertinent benchmarking data with which to validate microgravity CFD models. According to the Phase III 'Slosh' proposal submitted by FIT to a.i. solutions, the SPHERES-Slosh Experiment (SSE) will "provide the first data set from long duration tests in low-gravity that can be directly used to benchmark CFD models, including the interaction between the sloshing fluid and the tank/vehicle dynamics." [5]

1.2 Literature Review

Consider the interface layer between some fluid and gas. The fluid molecules would have some attraction to the other like-molecules in the fluid, cohesion, as well as some attraction to the unlike-molecules in the gas, adhesion. The interplay of these attractive forces forms the surface tension of the fluid which is a property of the interface. Surface tension may be defined as the magnitude, F, of the force exerted parallel to the surface of a liquid divided by the length, L, of the line over which the force acts, shown as

$$\gamma = \frac{F}{L} \tag{1.1}$$

and measured in $\frac{N}{m}$. For example, consider the surface tension of water when interfaced with air at the room temperature of 20 degrees Celsius. In this case, the surface tension, γ, is $.073\frac{N}{m}$.

Now, consider some soap bubble of radius, R, with an internal pressure, P_i, and an external pressure, P_o. Assume that the static soap bubble is acted upon by no external

forces such that the equation of motion of the soap bubble may be represented by

$$\Sigma F = 0 \tag{1.2}$$

If this theoretical bubble could be cut in half and the right hemisphere made into a free body diagram, it would be seen that the difference of the internal pressure and the external pressure acts as an outward radial force in which all of the force components not pointing to the right are canceled out. Therefore, the aggregate force would be the product of the inner pressure and the projected area of the right hemisphere. Along the circular edge of the hemisphere, the attractive force of the left hemisphere would act on the inner and outer side of the soap bubble shell. Therefore, the aggregate force pulling to the left would be the product of the soap bubble circumference and the surface tension of the fluid - this term must be multiplied by two to account for the inner and outer side of the soap bubble and will be given a negative value for convention since the force points to the left. The balance of forces equation may therefore be shown as

$$\Sigma F = F_{surface\ tension} + F_{pressure} = 0 \tag{1.3}$$

$$-2\gamma(2\pi R) + \Delta P(\pi R^2) = 0 \tag{1.4}$$

Therefore, the pressure difference may be shown as

$$\Delta P = \frac{4\gamma}{R} \tag{1.5}$$

In the case of a droplet instead of a bubble, only the outer side of the circular edge need be considered. Therefore, for a droplet, the pressure difference may be shown as

$$\Delta P = \frac{2\gamma}{R} \tag{1.6}$$

[6] Now, consider a droplet of water interfaced with air at room temperature and having the volume, V, of $1.7672L$ or $.0017672m^3$. According to the volume of a

sphere,

$$V = \frac{4}{3}\pi r^3 \tag{1.7}$$

this droplet would have a radius of $R = .075m$ and a diameter of $D = .150m$. Therefore, the pressure difference between the inside and the outside of the droplet would be $1.9467Pa$. In seeking the minimal energy configuration, this pressure difference will attempt to hold the droplet in the shape of a sphere. Given that the surface area of a sphere is

$$A = 4\pi r^2 \tag{1.8}$$

the surface area of this sphere will be $.0707m^2$ and the total force imparted by the pressure difference over the entire shell will be $.1376N$. Given that the mass of this sphere of water is $1.7672kg$, an Earth-like gravity of $9.81\frac{m}{s^2}$ will impart a force of $17.3362N$ on this drop of water. It is easy to see how a sphere of water this size will not hold its shape in a 1G environment. It is also easy to see how smaller droplets, with less mass and smaller radii, could hold their spherical shapes more effectively.

There exist three hydrodynamic regimes: gravity dominated, inertia dominated, and capillary dominated. The SSE will conduct research in the capillary dominated regime. In this regime, even large drops of water will naturally form spheres. Previous studies have used drop towers, reduced gravity aircraft (RGA) flights, or magnetic effects to induce a weightless environment. These approaches have limited effectiveness because they do not produce weightlessness for a long duration of time or because the zero-g simulation they create is not perfect. [7] The satellite Sloshsat FLEVO has conducted research in the microgravity environment over the course of two weeks for a total experiment time of of approximately 24 hours.[8]

1.3 Outline

1.3.1 Chapter Two: SPHERES-Slosh Experiment Design

This chapter will discuss the requirements presented to the SSE Team and the reasoning behind those requirements. This chapter will also discuss the design features

of the SSE and explain why these design features were integrated into the creation of the SSE. Additionally, this chapter will present the application of a new safety analysis technique on the SSE.

1.3.2 Chapter Three: ISS Operations

This chapter develops system identification and open-loop control methods for the SSE and relates how those methods were implemented on the ISS.

1.3.3 Chapter Four: Proposed Control Theory and Design

This chapter develops the framework for closed-loop control on the SSE and proposes how that closed-loop control could be achieved.

1.3.4 Chapter Five: Conclusions and Future Work

This chapter will discuss the takeaways from this thesis and give ideas for the path ahead.

Chapter 2

SPHERES-Slosh Experiment Design

This chapter will discuss the requirements presented to the SSE Team and the reasoning behind those requirements. This chapter will also discuss the design features of the SSE and explain why these design features were integrated into the creation of the SSE. Additionally, this chapter will present the application of a new safety analysis technique on the SSE.

2.1 Requirements

This section discusses the requirements that were presented to the SSE Team at the onset of Phase III (the beginning of substantial MIT involvement) of the SSE project. This section will relate the reasons for why these requirements were established. The overall requirement of the project was to acquire long-duration, low-gravity slosh data which could be used to calibrate computational fluid dynamics (CFD) models for coupled fluid-vehicle behavior. Beyond this overall mission requirement, several other functional and form requirements had also been established by the beginning of Phase III.

2.1.1 Functional Requirements

Video Image Collection

The collection of video images will allow for the validation of the CFD models. These images need to be high definition quality and collected under constant lighting so that the exposure of the cameras remains the same. Although, it should be noted that some of the tests were conducted with the light box hood removed and the cameras set to auto exposure. The requirement for the frequency of images taken was not specified based on capturing a certain order of effects or a certain level of fidelity of the fluid. Rather, FIT simply wanted to maximize the frequency based on the limitations of the hardware, specifically, the VERTIGO avionics stack and the associated operating system with its software. FIT wanted to maximize the quantity of images taken based on the available storage space on the SSE hard drives. The cameras used for the flight units were different from the earlier ground prototypes of earlier phases.

Collection of Inertial Measurement Unit Data

The collection of gyroscope and accelerometer data provides critical information for the development of the Computational Fluid Dynamics (CFD) models. Additionally, this data can be used to conduct system identification actions such as characterizing the forces of the thrusters and determining the values of the inertia tensor of the SSE.

Transfer of Data to Earth

The data being collected from the SSE and sent to Earth has no real time or near real time use as this data is used for post-processing. However, the post-processing of this data gives insight into the effectiveness of the associated test session. Lessons can be learned from the post-processing and applied to the next test session in order to increase the effectiveness of the SSE operations. For this reason, the data must reach the appropriate party on Earth, namely, a.i. solutions within the operational time frame of the SSE project as a whole. Therefore, this data should be received by

a.i. solutions no later than one week after the test session.

Moving Liquid in Known Way

To provide effective benchmarking data for the creation of accurate CFD models, the SSE tests should be conducted in such a way that the inputs to the system, the system itself, and the outputs from the system are thoroughly understood. Unknown entities will likely influence known entities and will probably be wrongly accounted for in analysis. For example, if there exist variable wind currents of significant magnitude in the ISS JEM, the SSE may be pushed further than nominal distance during the thruster characterization tests. This will lead to inaccurate system identification and will ultimately cause trouble during the analysis tasks by associating an inaccurate force with an observed liquid behavior. Another example, which comes from the system sphere, might be structural flexibility that introduces unwanted dynamics into the experiment and corrupts the science results. Therefore, the design and implementation of the system must ensure that the inputs, system, and outputs are known so that the liquid slosh behavior is being caused and captured in a known way.

2.1.2 Form Requirements

Liquid Capacity

The SSE has various fill levels and compositions (liquid water versus solid mass replicator) in three different tanks. These fill levels and compositions (20% liquid-filled, 40% liquid-filled, and a 40% solid mass replicator) were chosen so that fluid slosh could be observed at different partial fill levels for the validation of CFD models, fluid-vehicle coupling behavior could be observed and analyzed, and the solid mass tank could be utilized if desired in order to better isolate the impact of the fluid slosh. The tank designs had already been established at the onset of Phase III.

Interface with SPHERES Platform

The SPHERES system provides a unique platform of computing, actuation, sensing, and estimation that can be used, in conjunction with hardware augmentations, to gather the desired data.

Operate in the International Space Station

The ISS provides a unique operating environment integral to the successful completion of this science endeavor. The singular value of the ISS as a testing environment may be summarized by Figure 2-1.

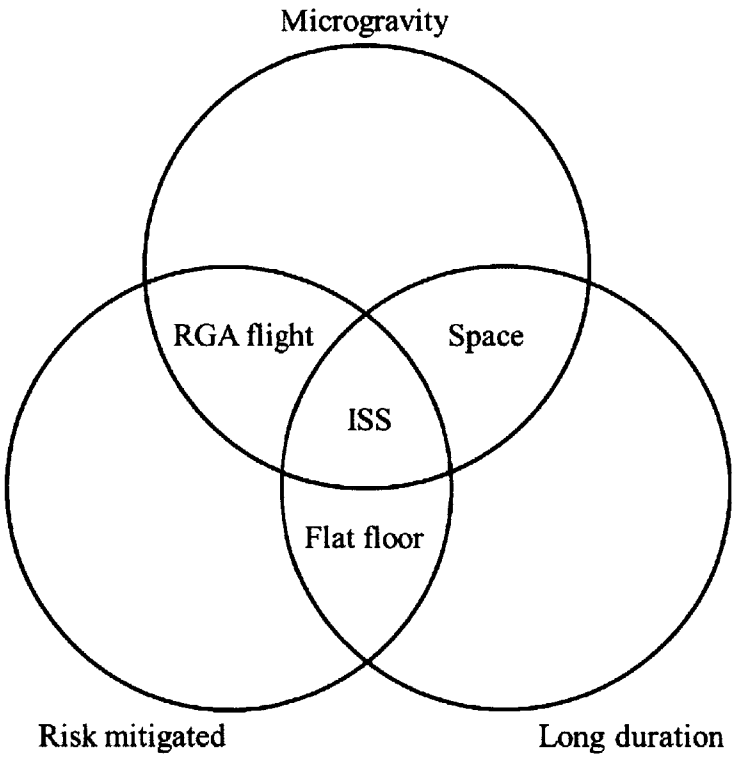

Figure 2-1: Characteristics of various testing environments which have been used for SPHERES projects. Credit: Bryan McCarthy.

The three characteristics outlined in Figure 2-1 are each very important. Microgravity gives the testing legitimacy in terms of providing useful benchmarking data for the development of CFD models. Risk mitigation preserves the SSE and allows

for repeated tests thus driving down the costs of development and replacement while also ensuring a large amount of data is collected. Long duration testing allows for the comprehensive capture of the fluid dynamics. There are certain time constants which determine the fluid behavior. Therefore, in order to adequately cause and observe a complete set of behaviors under a certain condition, long duration testing is necessary. It is worth noting that this necessity cannot be overcome by increasing the capability of the actuators.

2.2 System Design

This section will discuss the design features of the SSE and will explain why these design features were included; this discussion will provide traceability back to the requirements. This section will give a holistic overview of the SSE, will identify and explain interfaces to established systems, and will examine each of the individual parts unique to the SSE.

2.2.1 System Overview

The SSE uses a distributed computing architecture wherein parts of the computation are contained within the SPHERES satellites and parts of the computation are contained within the Vertigo avionics stacks (VAS). The SPHERES computation handles communications with the SPHERES graphical user interface (GUI), commands the actuation of the SSE, and executes the estimation processes of the SSE. The VAS computation handles the payload sensors, the storage of the associated data, and sends information to the VERTIGO GUI.

Each SPHERES satellite is mechanically interfaced with its respective VAS using four thumbscrews which connect the version 2 SPHERES expansion port to the VAS. Power and data exchange capability is achieved through a fifty pin connector. The two components are designed not to exchange power and do not exchange power in any meaningful way. However, the VERTIGO battery has powered up certain light emitting diodes (LEDs) on its respective SPHERES satellite when the satellite was

powered off - a curious anecdote for future users and designers. The communication data between the SPHERES satellites and their respective VASs takes the form of serial communication. This communication is used primarily to populate the VERTIGO GUI through backdoor communication and so that the SPHERES satellites and VASs can effectively coordinate the execution of tests.

The user can communicate with the VAS through the backdoor communication provided by the SPHERES satellites or by plugging an Ethernet cable directly into the Slosh Avionics Box (SAB) for direct 1 Gbps communication. As of the time of this writing, WiFi has not yet been enabled on station which would provide a third avenue by which to communicate with the VAS.

Each of the SPHERES satellites is independently powered by its own set of two packs of eight AA batteries. The rest of the electronics in the SSE are powered by the two Nikon Li-ion VERTIGO batteries (EN-EL4a). These batteries have a power capacity of 2500mAh at 11.1 V. Each VERTIGO batteries power half of the electronics aside of the SPHERES satellites including one VAS, one SAB, one camera, and one LED power strip.

All of the actuation capability is contained in the 24 cold gas carbon dioxide thrusters of the SPHERES satellites.

The radiation protection and thermal management systems of the SSE are comparable to ground systems because the International Space Station (ISS) has Earth-like properties with respect to temperature (about 21.5 degrees Celsius[9]) and radiation exposure. The VAS does have a fan to keep its computer cool. The main thermal consideration on the SSE encompasses the carbon dioxide tanks. As carbon dioxide is expended to actuate the SSE, the pressure inside the carbon dioxide tank drops. As the pressure of the tank drops, the temperature drops as well. If enough carbon dioxide is expended in a given amount of time, the ensuing temperature drop will cause the liquid carbon dioxide inside of the carbon dioxide tank to freeze. This will result in a loss of actuation capability.

The Japanese Experiment Module (JEM) of the ISS has five ultra sound (US) beacons at known locations. These US beacons are triggered by infrared transceivers

on the SPHERES satellites and emit US pulses at different known times after the infrared pulse. These US pulses are detected by the 24 US sensors on each satellite and used in conjunction with an Extended Kalman Filter (EKF) and body mounted gyroscopes to determine the states of the SPHERES satellites. Each satellite also has three accelerometers which are not used in the estimation process. For more information on the SPHERES satellites, the reader should refer to the SPHERES Guest Scientist Program documentation.[10] The primary hardware instruments added as part of the SSE are the two cameras mounted to the structure and the additional Inertial Measurement Units (IMU) components inside of the SAB.

The structure of the SSE houses a tank partially filled with water dyed green. With the exception of this water and the carbon dioxide, the SSE is a rigid structure. This rigidity is provided by the anodized aluminum SSE frame.

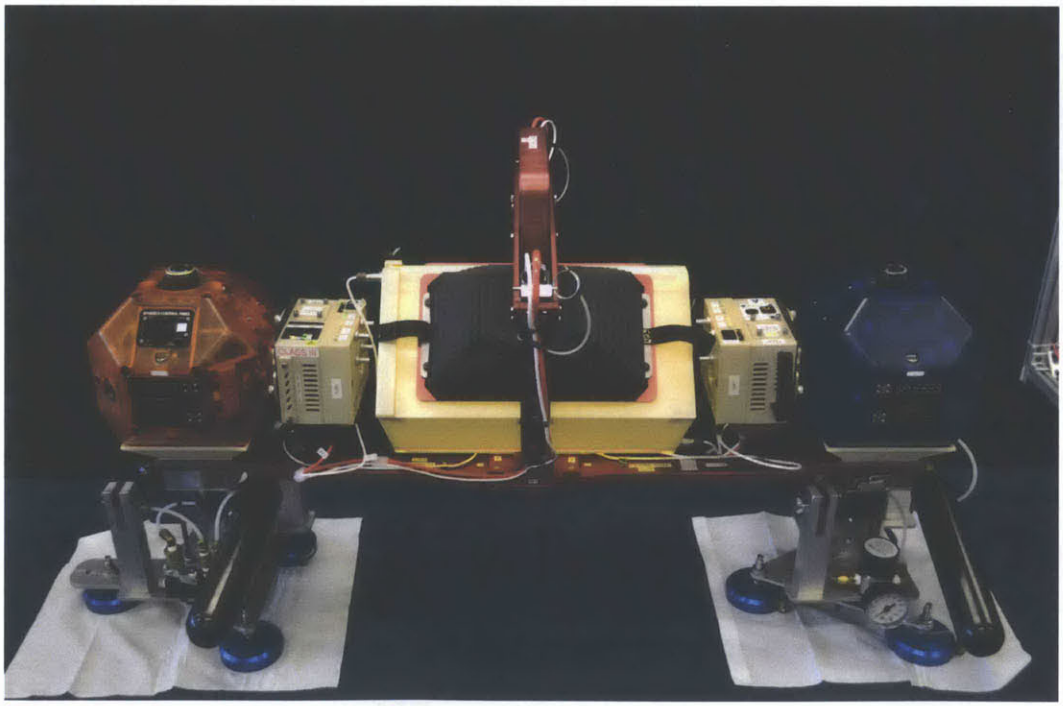

Figure 2-2: The SSE mounted on air carriages. Credit: FIT.

2.2.2 Camera System

Cameras

The iDS UI-5580CP cameras allow the SSE to collect the video images of the fluid slosh inside the tank. The cameras are set to constant exposure when the light box is closed and are set to auto exposure when the light box hood has been removed. The cameras pass these images to the Slosh avionics box (SAB) in the form of Ethernet data. The cameras are augmented with Navitar lenses and mounted onto the vertical frame arms of the center hub and secured using sliding dovetail mechanisms and pins. Table 2.1 shows the specifications of these cameras which meet the requirements. [11]

Family	CP
Interface	GigE
Sensor type	CMOS
Manufacturer	Aptina
Frame rate	14.1 fps
Resolution	2560 x 1920
Shutter	Rolling Shutter/Global Start Shutter
Optical class	1/2 inch
Pixel class	5 MPixel
Protection class	IP30
HDR	No

Table 2.1: Camera specifications.

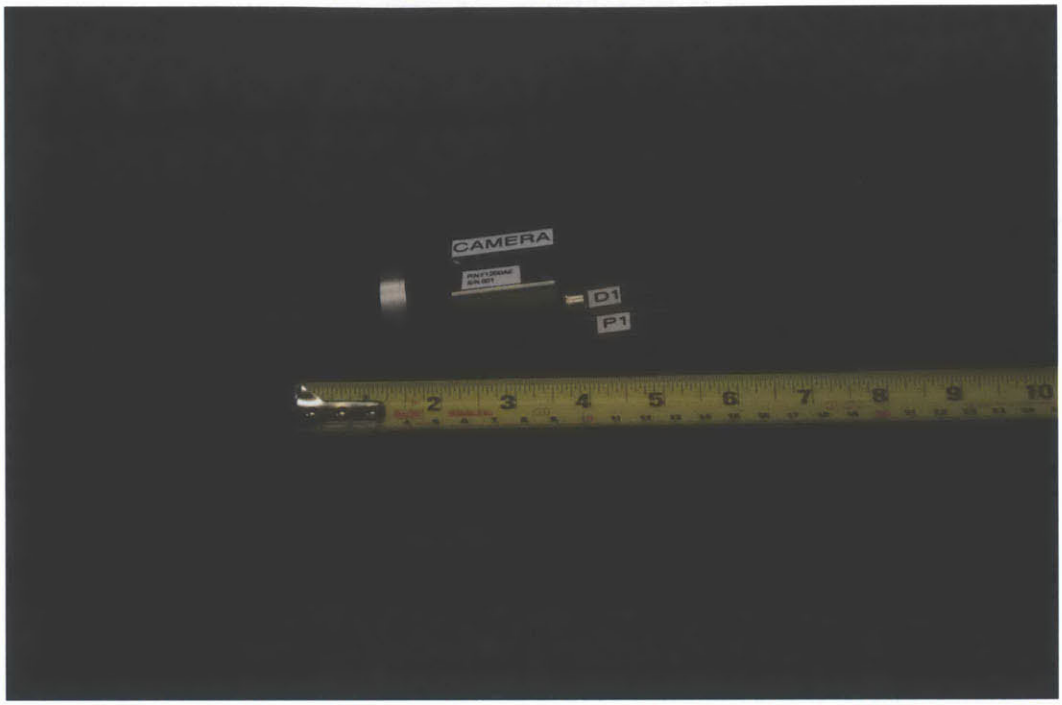

Figure 2-3: iDS UI-5580CP camera augmented with Navitar lens and mount - side view. Credit: FIT.

Figure 2-4: Camera mounted on SSE structure using sliding dovetail and pin with attached Ethernet and power cables. Credit: FIT.

Baffles

The baffles provide a light seal between each camera and the light box so that the lighting conditions remain constant when the video images are being collected. That is, the baffles prevent external light sources from lighting the fluid.

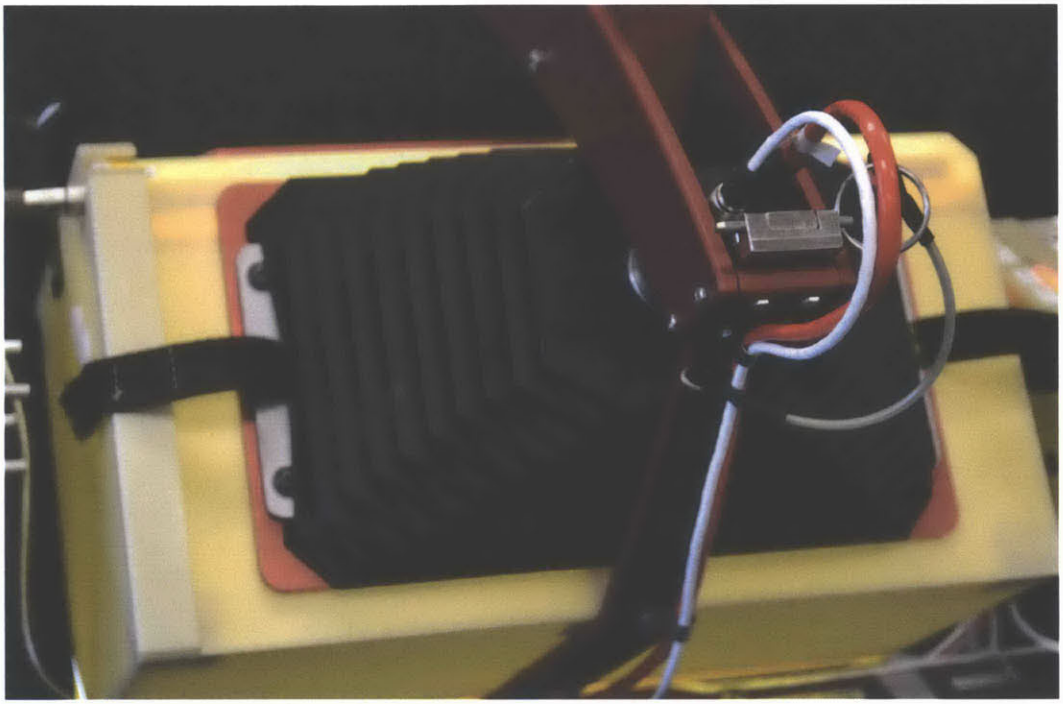

Figure 2-5: Deployed baffles - side view. Credit: FIT.

Light Emitting Diodes

The light emitting diodes (LEDs) provide the internal source of constant lighting so that video images may be collected of the water under constant exposure. There are two strips of LED lights that are covered with a semi opaque covers which increase the uniformity of the distribution of light within the light box. One of the strips is located at the top of the hood and runs along the corner. The other strip is located at the bottom of the backdrop and runs along the corner.

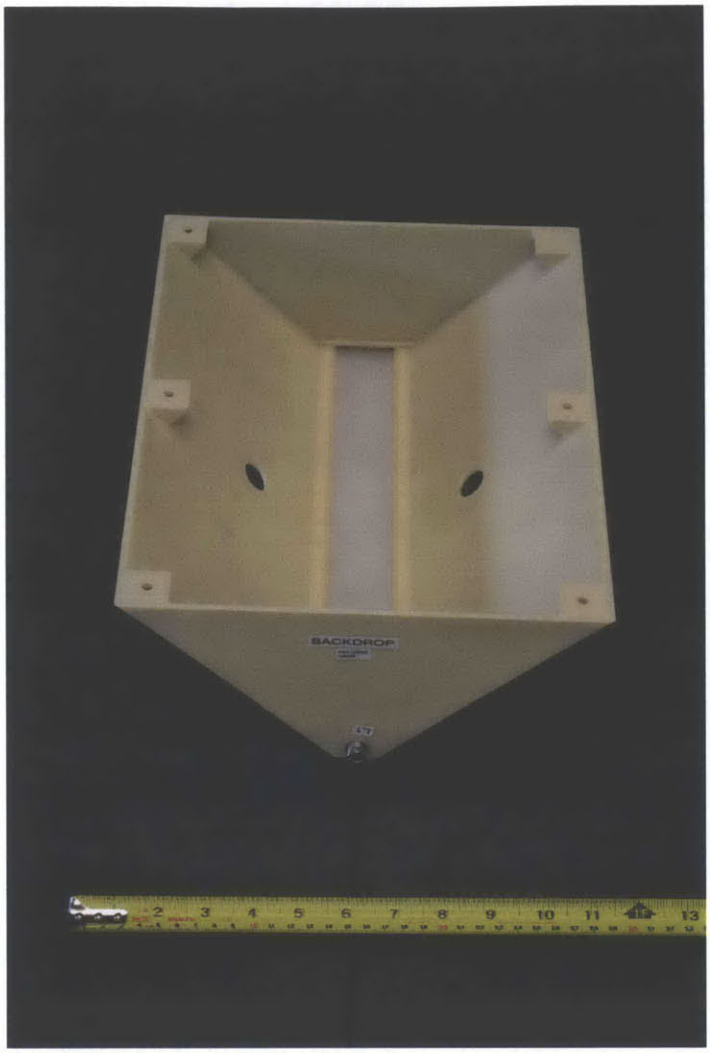

Figure 2-6: Backdrop with covered LED strip shown - front view. Credit: FIT.

Light Box

The light box is composed of two major pieces including the hood and the backdrop. The hood forms the top half of the light box and the backdrop forms the bottom half of the light box. They are fitted together with cylindrical protrusions on one piece which are inserted into cylindrical holes of the other piece and held together by friction/clamping force. This light box protects the relatively weaker tank from structural damage, serves as a flammability barrier which protects the tank, and prevents external light source from impacting the lighting conditions of the water when

video images are being collected. The light box pieces are created with 3D printed ULTEM which gives the light box considerable durability and thermal resistance.

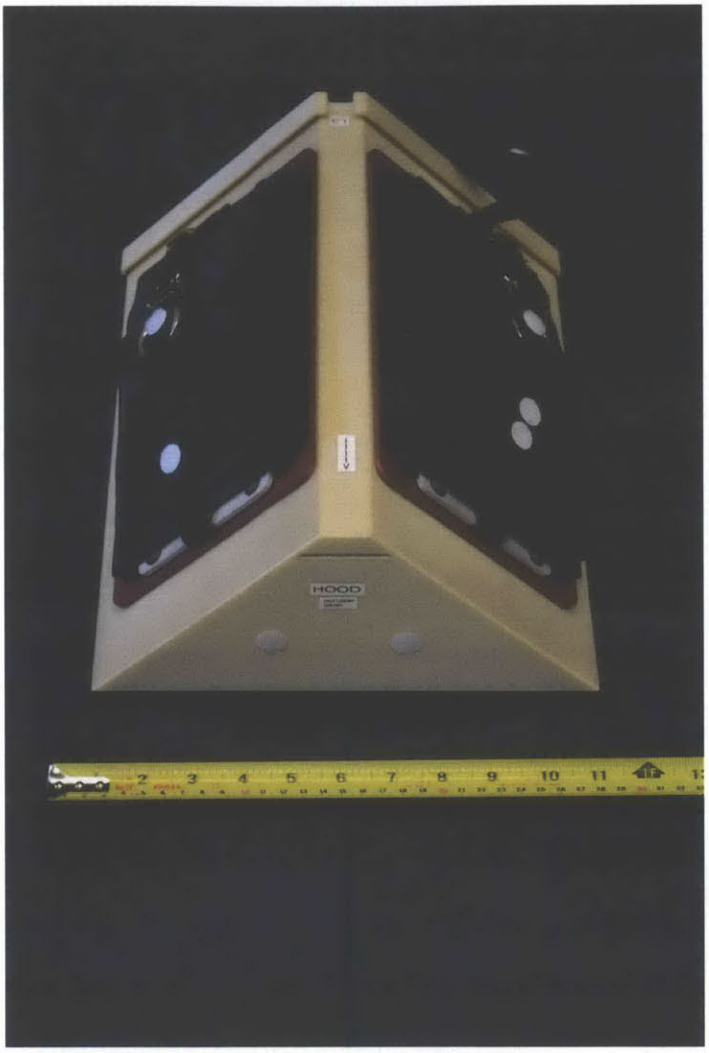

Figure 2-7: Hood with non deployed baffles - front view. Credit: FIT.

2.2.3 Tank

The SSE has three tank options: a 40% liquid filled tank, a 20% liquid filled tank, and a 40% solid mass replicator. The mass of the liquid/solid mass in each of the 40% tanks is 1.7672 kg (1.7672 liters for volume in the case of the liquid) and the mass of the liquid in the 20% filled tank is .8836 kg (.8836 liters for volume). The dry mass of the SSE is 21.1359 kg. Therefore, in the 40% filled scenario the liquid/replicator

material accounts for 7.72% of the mass in the SSE. In the 20% filled scenario, the liquid accounts for 4.00% of the mass in the SSE.

Each tank consists of two pieces of 3D printed Lexan which have been polished, glued together, filled with the appropriate level of liquid, and sealed. The transparency of the polished Lexan allows video images of the fluid to be taken. The tank is cylindrical with spherical end-caps.[12] The salient dimensions of the tank are the dimensions which provide the boundaries of the fluid. These dimensions include the inner diameter of the cylinder (156.35 ± 0.25mm), the length of the cylinder (150.01 ± 0.25mm), and the inner radius of the half-spheres (78.17 ± 0.25mm).[13]

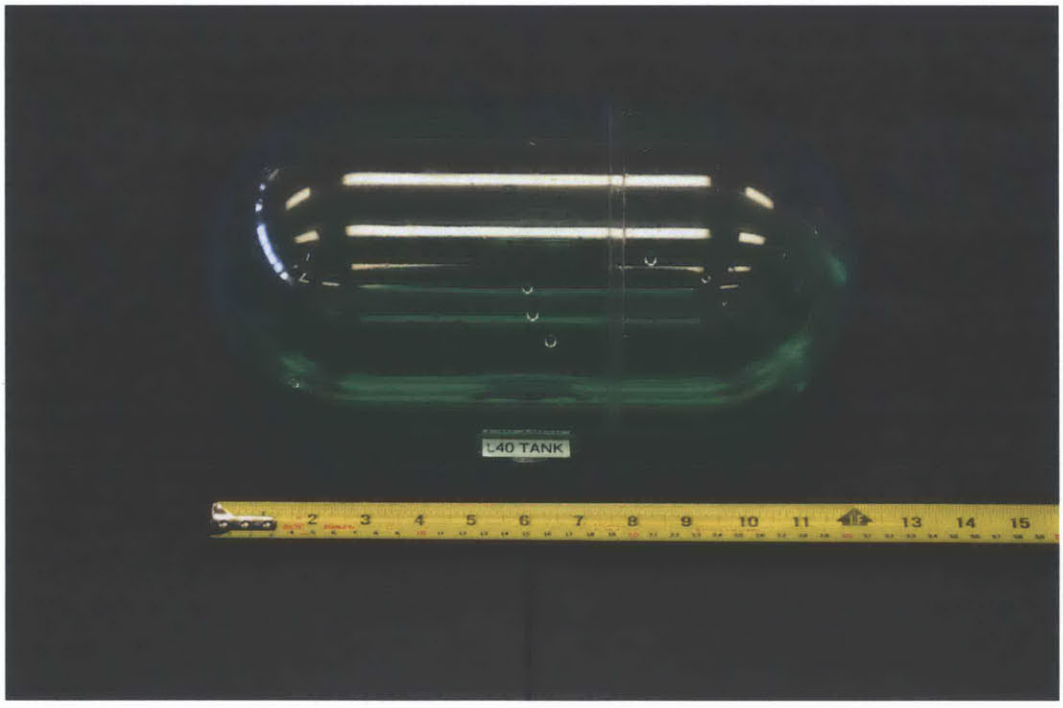

Figure 2-8: 40%-filled SSE tank - front view. Credit: FIT.

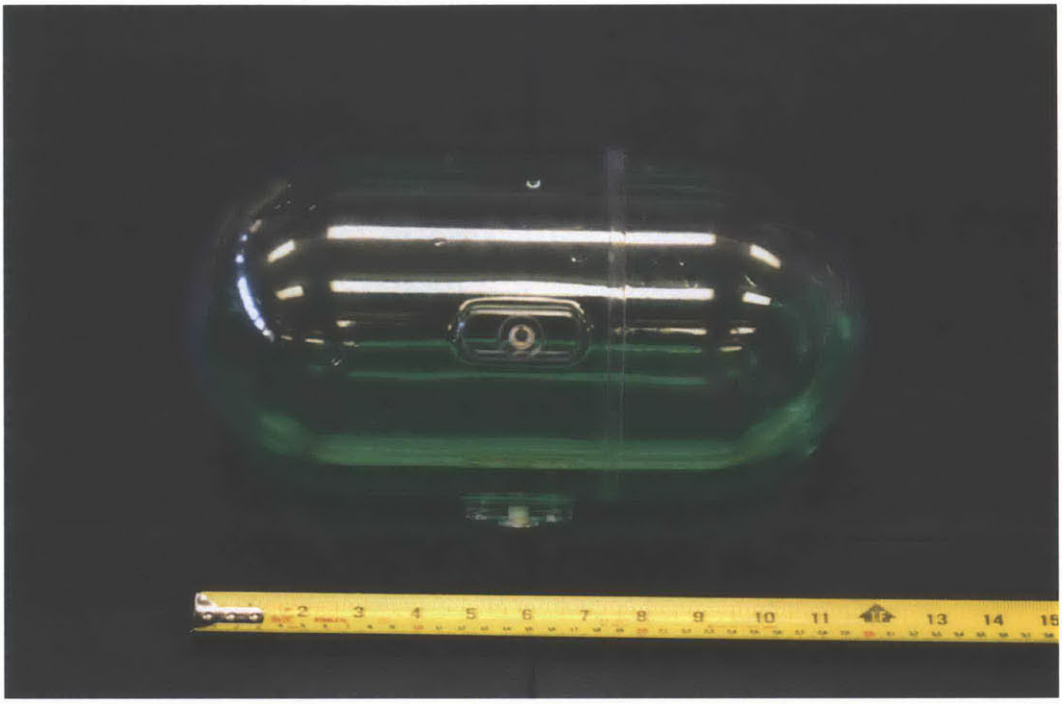

Figure 2-9: 40%-filled SSE tank - bottom view. Credit: FIT.

2.2.4 SSE Avionics Box

The SSE avionics box (SAB) is secured mechanically to the VERTIGO avionics stack (VAS) using four thumb screws. The SAB and VAS exchange power and data through a 50 pin connector. Each SAB passes along video images through Ethernet communication from one of the cameras to the VAS where the video images are stored on the SSE hard drives. Each SAB contains a set of IMUs which include accelerometers and gyroscopes. Specifically, the IMUs are contained inside of a CHR-UM6 or CH Robotics UM6 Ultra-Miniature Orientation Sensor. This UM6 device creates IMU data in serial Universal Asynchronous Receiver/Transmitter (UART) form and passes the data along to an FT232RLSSOP Future Technology Devices International (FTDI) chip which converts the data from serial to Universal Serial Bus (USB) form. This data is passed through the 50 pin connector and stored in the SSE hard drives inside of the VAS. Each SAB has two UM6 devices and two FTDI chips.

The SAB contains Ethernet (RJ45) and USB (SHLD2) access points to the VAS

operating system. The SAB contains ultra sound (US) sensors which are disconnected and unused. Accompanying these US sensors are AD8567 operational amplifiers which would be used to do signal conditioning. The SAB contains an Si4946 transistor and an LTC4365 overvoltage, undervoltage, and reverse supply protection controller that help manage the 12 V power supply. The SAB contains power filters for the 5 V lines which power the chips in the SAB and for the 12 V lines which power external components such as the LED lights housed in the light box and the cameras. The 12 V lines also use a DC to DC converter. The SAB draws all of its power from the VERTIGO battery mounted inside of the VAS.[14]The SAB shell was created by assembling pieces of 3D printed ULTEM.

The SAB also contains its own set of external LEDs which communicate to human users the behavior of the SAB. The LEDs are designated as 1-8. Table 2.2 shows the meanings and color of the eight LED lights.

Number	Meaning	Color
1	The SAB has received 5 V regulated power	Green
2	The SAB has received 12 V regulated power	Green
3	The SAB has received 5 V unregulated power	Red
4	The SAB has received 12 V unregulated power	Red
5	IMU 1 has sent a packet of data	Yellow
6	IMU 1 has received a packet of data	Blue
7	IMU 2 has sent a packet of data	Yellow
8	IMU 2 has received a packet of data	Blue

Table 2.2: Meanings and colors of SSE SAB LED lights.

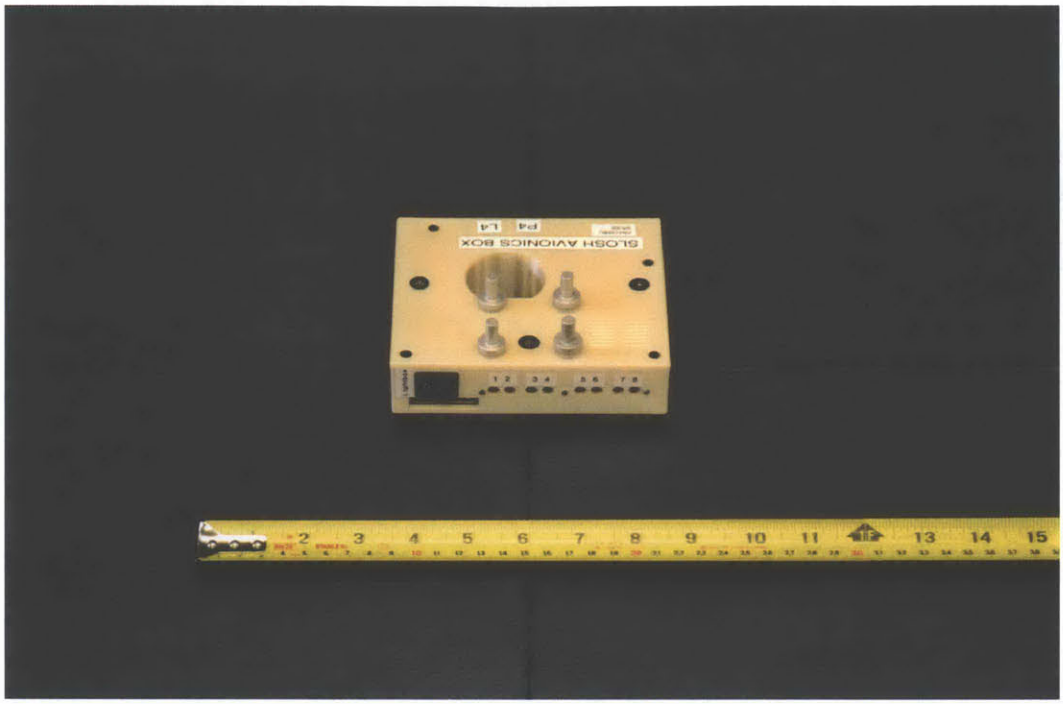

Figure 2-10: SSE avionics box - top-front view. Credit: FIT.

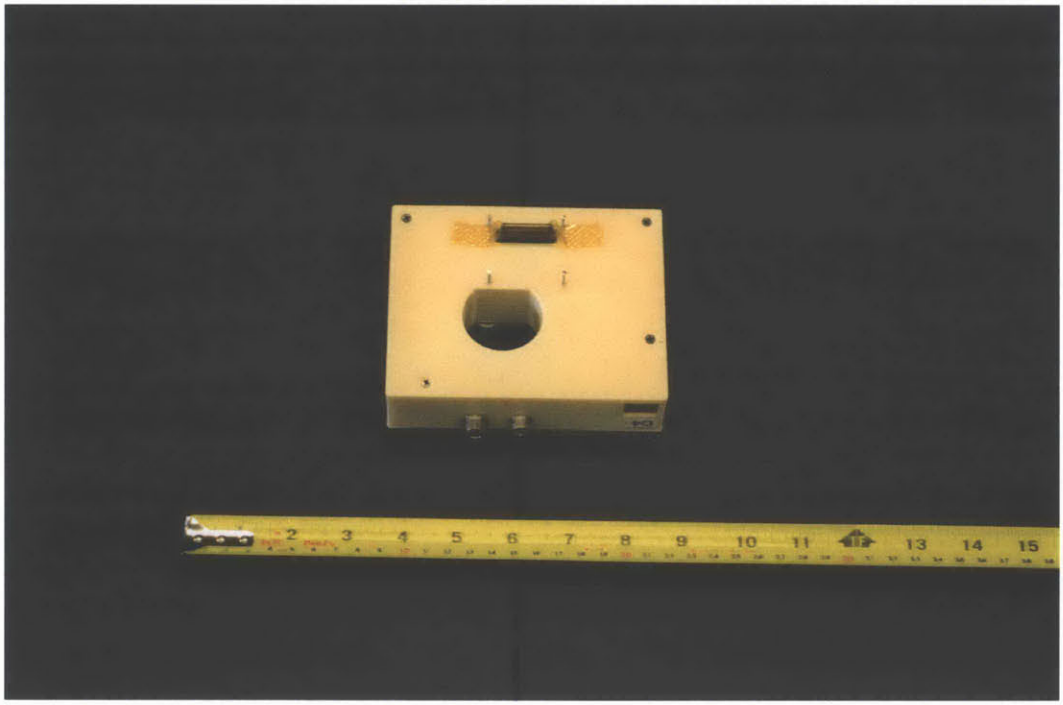

Figure 2-11: SSE avionics box - bottom-back view. Credit: FIT.

2.2.5 Structure

The structure of the SSE provides rigidity, proper alignment, the ability for astronauts to safely hold and move the system, limited protection, and mass symmetry to the experiment. The rigidity is sufficient to ensure that (with the exception of the water and the carbon dioxide) the SSE may be modeled as a perfectly rigid system. The proper alignment ensures, among other things, that the thrusters fire in the desired direction. The physical interface allows for manual maneuvers to be conducted by the astronauts. The mass symmetry allows for key simplifications during control design. Within the structures subsystem exists the center hub and two frame arms.

Center Hub

At the bottom of the center hub exists the base. This center base provides the physical interface to the two frame arms which can be attached in standard or nonstandard configuration to the bottom of the base. The center hub base also supports the two vertically aligned arms which are attached to the top of the base.

Figure 2-12: SSE center hub - bottom view. Credit: FIT.

These vertically aligned arms contain the thumb screws which are screwed into the protrusions of the SSE tank and keep the tank and light box secure. These vertical arms also contain the two dovetail bases and two pins which secure the cameras to the SSE structure. Fasteners are attached to these vertical arms which secure the camera Ethernet and power cables to the SSE structure. The vertical arms support the two arches.

These two arches support the counterweight. Together these arches and counterweight provide mass symmetry for the SSE. This symmetry ensures that the center of gravity of the SSE aligns with the designated geometric center of the SSE. Addition-

ally, this symmetry produces, within the inertia tensor, on-diagonal values which are significantly greater than the off-diagonal values. Several screws of various sizes are located throughout the entire SSE structure and have not been explicitly mentioned.

Figure 2-13: SSE center hub - side view. Credit: FIT.

Frame Arms

The two frame arms attach to the center hub using two thumb screws each. The frame arms are mounted with 3D printed ULTEM saddles which are used to secure the SPHERES satellites in place. At the end of each frame arm there exists a screw which can be used to tighten the component which holds the carbon dioxide tank of

the satellite. Each frame arm has fasteners which secure the power cable that is used to power the LED strips inside the light box.

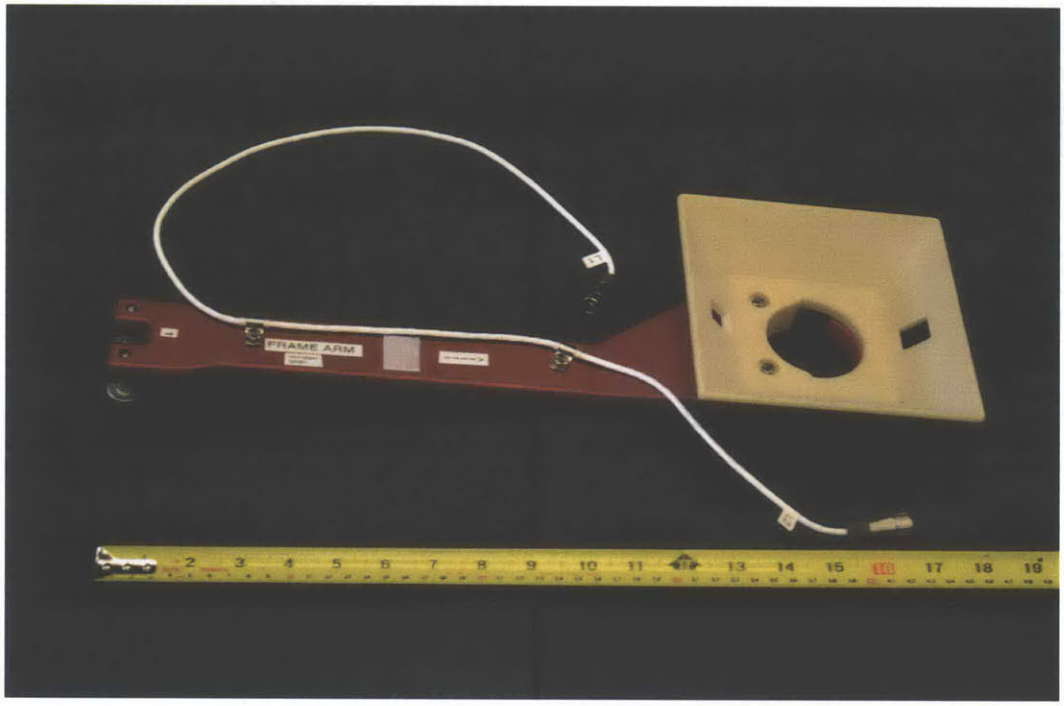

Figure 2-14: SSE frame arm - side view. Credit: FIT.

2.3 Safety Analysis

This safety analysis used the Systems-Theoretic Process Analysis (STPA) which is a new hazard analysis technique based on Systems-Theoretic Accident Model and Processes (STAMP).[15] STPA and STAMP were developed by Professor Nancy Leveson. This section encompasses the application of the STPA to the SSE. This section contains a subset of the comprehensive analysis which can be conducted on the SSE using this method and provides a framework for developing a more exhaustive analysis. High level functional goals of the SSE:

1. Execute predetermined open-loop maneuvers in a microgravity environment to simulate fuel slosh in the partially filled fuel tank of a spacecraft.

2. Determine the angular velocity and angular acceleration of the SPHERES-Slosh

system during the open-loop maneuvers.

3. Collect images of the fluid slosh during the open-loop maneuvers.

4. Post process the collected data to provide benchmarking results for models which simulate the behavior of fluids in microgravity.

Potential losses (Accidents):

1. The SSE collides with astronaut or physical structure in the ISS hurting the astronaut, damaging the Slosh system, or damaging the ISS.

2. The SSE experiences electrical damage resulting from an improperly controlled current input.

3. The SSE experiences fluid damage after being impacted by water.

4. The SSE experiences fluid damage after being impacted by excessive levels of carbon dioxide.

5. The SSE experiences thermal damage resulting from a rapid drop in temperature in the system.

Hazards:

1. (Maps to accident 1) The SSE translates and/or rotates at a rate which would hurt a human or cause physical damage on impact.

2. (Maps to accident 2) A battery pack is inserted into a SPHERES satellite while its power switch is turned on.

3. (Maps to accident 2) A battery pack is removed from a SPHERES while its power switch is turned on.

4. (Maps to accident 3) The fluid tank becomes unsealed and releases its water into the JEM or Node 2 of the ISS.

5. (Maps to accidents 4 and 5) The carbon dioxide tank which provides fuel for the SSE cold gas thrusters rapidly outputs a large quantity of carbon dioxide outside of the nominal pathways.

High-level functional and safety requirements/constraints:
High-level safety constraints:

1. (Maps to hazard 1) The SSE must not translate or rotate too quickly.
2. (Maps to hazard 2) The battery pack will not be inserted into the SPHERES while the power switch is on.
3. (Maps to hazard 3) The battery pack will not be removed from the SPHERES while the power switch is on.
4. (Maps to hazard 4) The fluid tank will not be removed from its protective shell during maneuvers unless supervised.
5. (Maps to hazard 5) The o-ring and valve thread will not be damaged during handling.

High-level safety requirements:

1. (Maps to constraint 1) The SSE will maintain an angular velocity lower than 1 radian per second and will be kept inside of the JEM test volume during translation to prevent momentum accumulation.
2. (Maps to constraints 2 and 3) The power subsystem door will remain closed while the power switch is 'on'.
3. (Maps to constraint 4) The fluid tank will be kept inside of the protective shell consisting of the backdrop and hood during maneuvers or will be consistently supervised by an astronaut.
4. (Maps to constraint 5) The carbon dioxide tanks will be secured into the SPHERES regulators (without inducing a kick load) at a 90 degree angle such that the threads mesh.

High-level control structure:

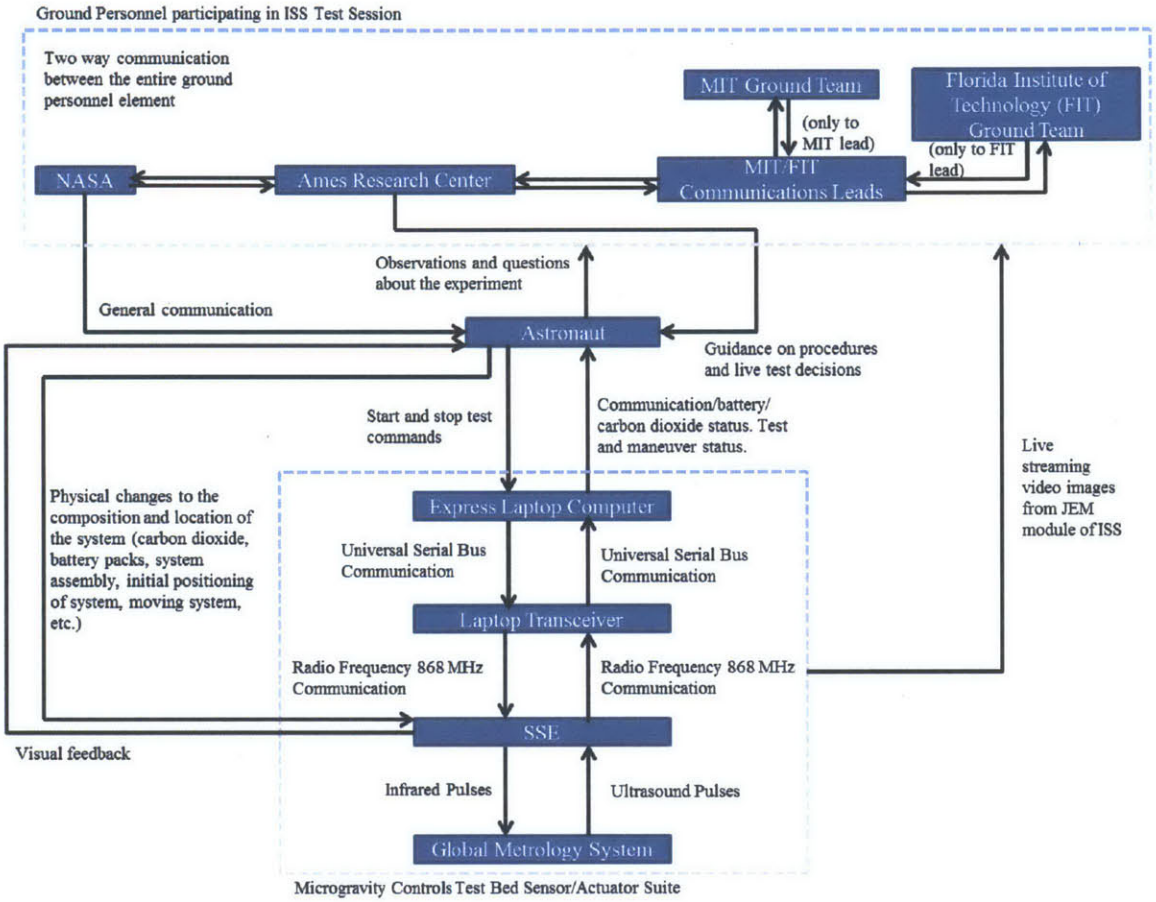

Figure 2-15: High-level control structure for SSE.

Selected loop in technical system:

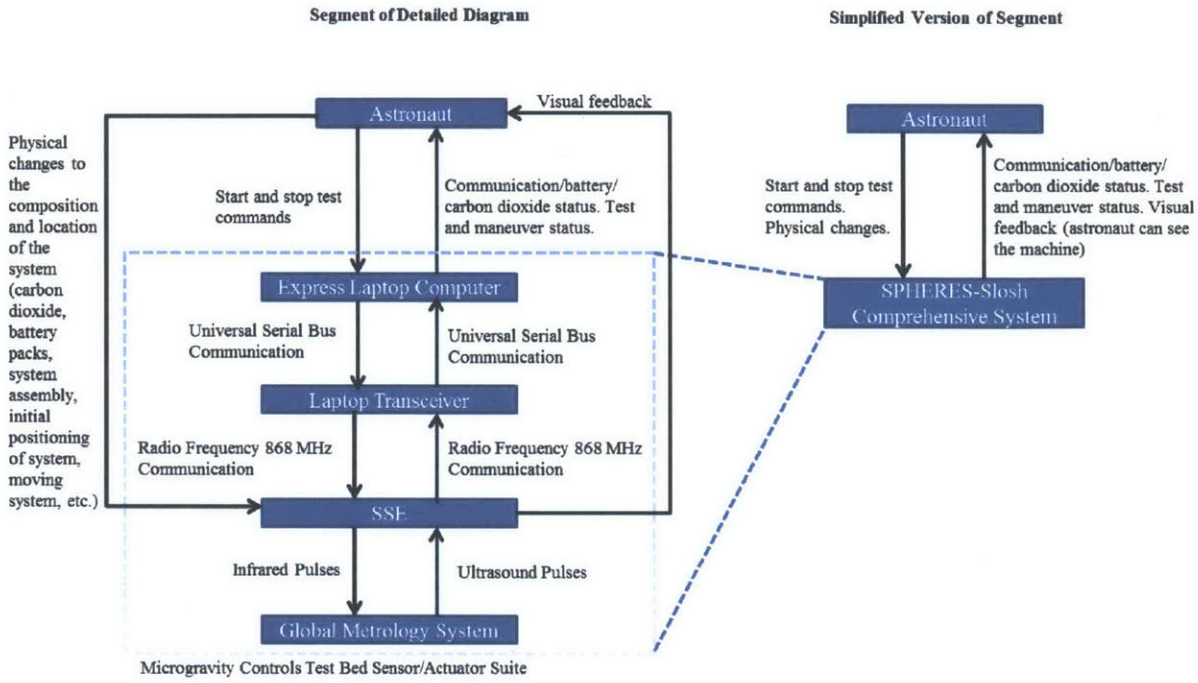

Figure 2-16: Selected loop in technical system.

Responsibilities of the astronaut:

1. Read test procedures and set up SSE.
2. Read the test plan and select which test to run.
3. Command the test to start using the graphical user interface (GUI).
4. Monitor the SSE as it executes the test (Manual tests require additional astronaut input).
5. Let the test finish or stop the test when appropriate to avoid damage or wasting time on a useless test.
6. Reposition the SSE and repeat the testing process as many times as necessary.
7. Refill the consumables on the SSE as necessary.
8. Disassemble and stow the SSE.

Process model of the astronaut:

1. State of SSE.
    (a) Position: x, y, z axes.
    (b) Velocity: x, y, z axes.
    (c) Quaternionions: q1, q2, q3, q4.
    (d) Angular rates: x, y, z axes.
2. Carbon dioxide levels in the SPHERES satellites.
3. Battery levels of the SPHERES satellites.
4. Battery levels of the VERTIGO batteries.
5. SPHERES test status.
6. SPHERES maneuver status.
7. SSE communication status.
8. Guidance from MIT/FIT.
9. Communication with NASA.
10. Test procedure instructions.
11. Test plan instructions.
12. Test overview instructions and notes.

Responsibilities of the SSE:

1. Execute open-loop maneuvers.
2. Collect IMU data from gyroscopes and accelerometers.
3. Collect video images of water slosh.
4. Send system information to astronaut.

Process model of the SSE:

1. State of the SSE.
   (a) Position: x, y, z axes.
   (b) Velocity: x, y, z axes.
   (c) Quaternions: q1, q2, q3, q4.
   (d) Angular rates: x, y, z axes.
2. Avionics status.
3. Video images collection status.
4. Consumable status
5. Target state, state error, forces/torques, thruster on/off times

Information sent on communication links:

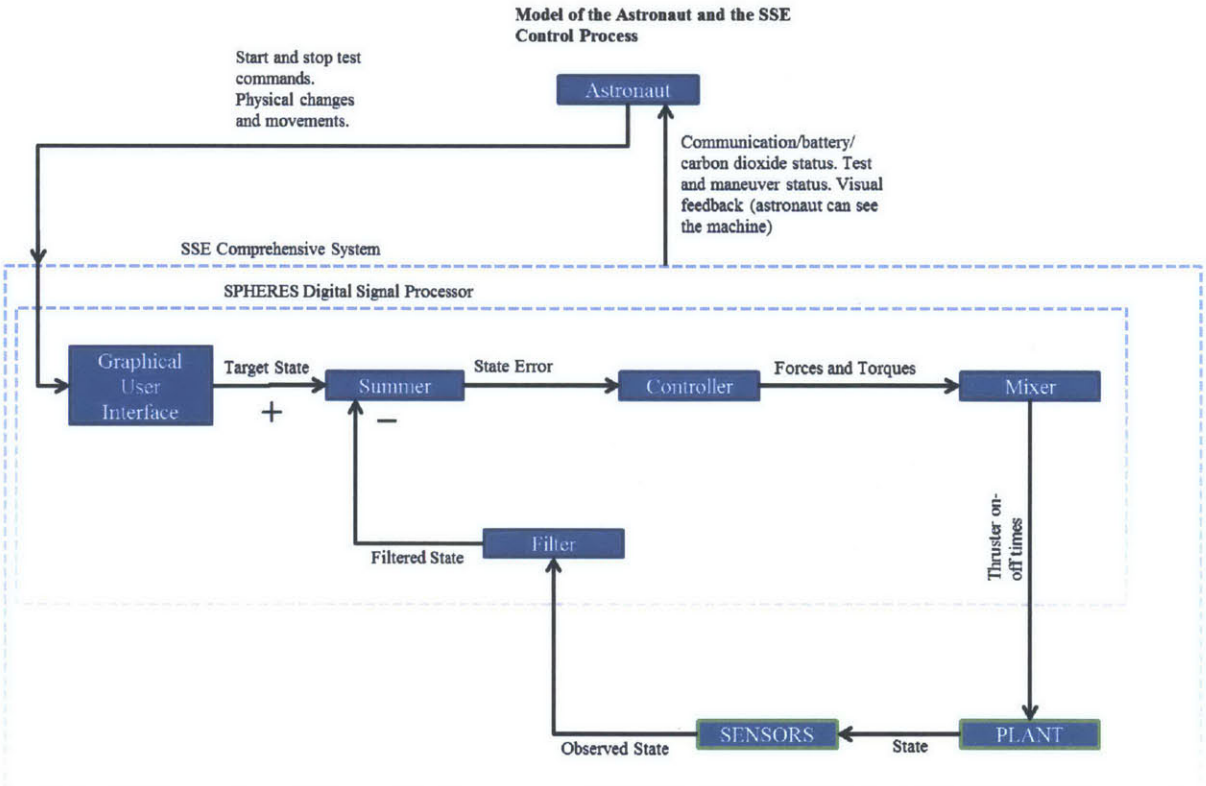

Figure 2-17: Information sent on communication links.

Control action	Not providing causes hazard	Providing causes hazard	Too early, too late, wrong order	Stopped too soon/ applied too long
Run test	N/A	Test is started when system is in the wrong location	The run command is sent before the system is properly set up and positioned	Commands are discrete: N/A
Stop test	A test which is going to cause the system to hit something is not stopped	The test is stopped before the system can stop itself with its actuators	The SSE impacts a human or physical structure before the stop command is sent	Commands are discrete: N/A

Table 2.3: Control actions and hazards.

Safety constraints/requirements:

1. The test must be stopped before the maneuver causes the Slosh system to impact a human or physical structure.

2. The test must be started only when the system is in the correct location.

3. The test must not be stopped while the system is using its actuators in a stable loop to stop itself.

4. The system must be completely set up before the test is started.

Causes, fixes, and requirements:

1. Unsafe control action 1: The SSE impacts a human or physical structure before the stop command is sent.

   (a) Cause 1: The laptop transceiver breaks after the start command is sent but before the stop command is sent.

      i. Fixes: Determine the mass of the system, the force of the thrusters, and the number of thrusters which can produce the same force or torque. Calculate the acceleration of the system. Determine the amount of time the thrusters would have to be on in order to create enough impulse for the system to be dangerous on impact. Limit

the run time of the test maneuvers as well as the fuel stored in the carbon dioxide tanks to prevent the system from reaching dangerous velocities.

ii. Requirement: The test maneuvers shall be no longer than 180 seconds.

iii. Requirement: The carbon dioxide tank shall hold no more than 300 grams of carbon dioxide.

(b) Cause 2: The astronaut does not pay attention to the experiment and does not stop the test in time.

i. The two fixes and requirements identified above will provide a fix/control for this cause as well.

2. Unsafe control action 2: The run command is sent before the system is properly set up and positioned.

(a) Cause 1: The test procedures were ineffective in communicating safe instructions to the astronaut.

i. Fixes: Place watchdogs in the software that will prevent a maneuver from starting when the hardware is not assembled correctly. Place additional instructions in the test overview located in the GUI of the SSE.

ii. Requirement: The watchdog will terminate the test whenever the SPHERES DSP cannot make contact with the VAS computer system.

iii. Requirement: The test overview will quantitatively and qualitatively illustrate the initial location of the system for the start of the test.

(b) Cause 2: The astronaut makes a mistake in following the instructions in the test procedures.

i. The two fixes and requirements identified above will provide a fix/control for this cause as well.

The software design of the SSE carried out some of these requirements to improve the effectiveness of the SSE: the watchdog was used, the test overviews provided

information to the astronauts that allowed the astronauts to carry out the tests, and the run time of the SPHERES satellite thrusters was limited.

2.4 Design Summary

The SSE aggregates hardware from different projects including SPHERES and VERTIGO and uses its own equipment to achieve its objective. It uses the SPHERES thrusters or manual maneuvers from the astronauts to move the SSE around and induce motion in the fluid. The motion of the SSE is captured in the gyroscopes and accelerometers and the motion of the fluid is captured by the cameras.

Figure 2-18: The SSE. Credit: FIT.

Chapter 3

Control Theory and Design Implemented during International Space Station Operations

This chapter will discuss control theory and design that has been developed for the SSE. These developments were implemented on the International Space Station (ISS) during two ISS test sessions. The first test session served as the checkout session for the SSE and the second session served as 'Science 1' for the SSE.

3.1 System Identification

System identification was one of the chief objectives of the SSE ISS checkout test session.

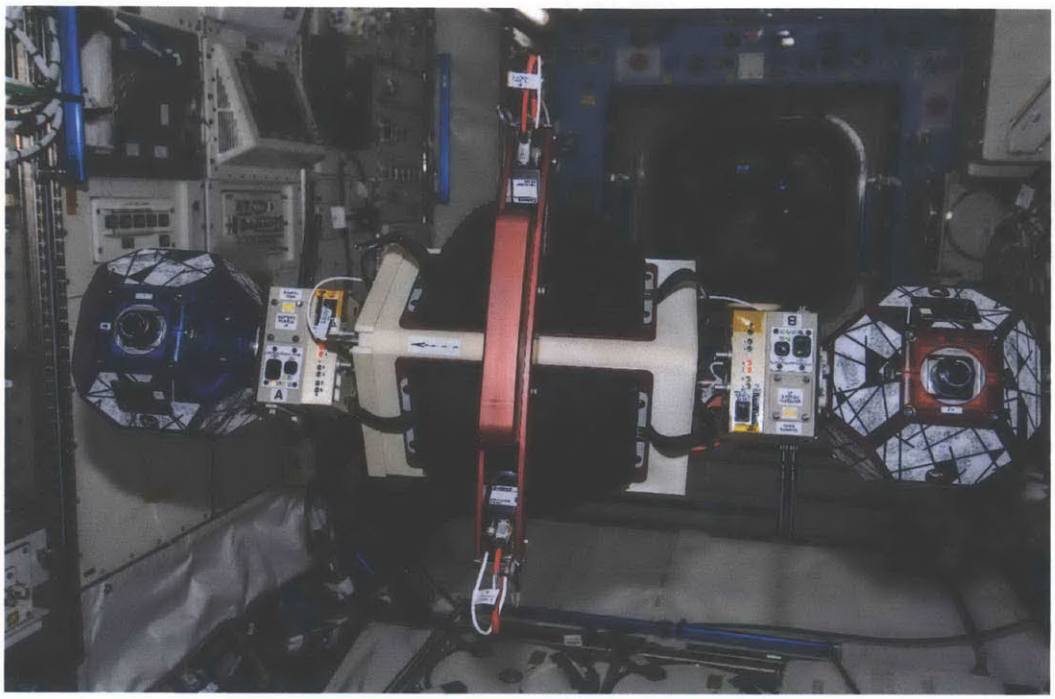

Figure 3-1: The SSE floats in the microgravity environment of the ISS during the successful checkout test session. Credit: NASA.

3.1.1 Purpose

System identification characterized the SSE in terms of actuator capability and principal axis inertia values. This characterization allows the correct amount of impulse to be applied to the SSE when using set thruster firing times during open-loop control. System identification also allows for the development of effective control gains when using closed-loop control. Finally, this characterization allows for the creation of effective baseline control gains when developing a simple adaptive controller. These baseline control gains are based on nominal operating conditions according to the results of the system identification.

3.1.2 Theory

In conducting system identification on the SSE, the first step is to determine the force each thruster produces. There exist two paths to achieving this goal - each with

advantages and disadvantages.

The first path begins by choosing a combination of thrusters which will attempt to produce pure translation (preferably along one axis) with approximately zero rotation of the SSE. These thrusters should not be impinged or at least equally impinged. This situation is governed by Newton's second law:

$$F = ma \tag{3.1}$$

The mass of the system should be known by weighing the SSE flight hardware on the ground. The acceleration can be determined using data from the accelerometers on the SPHERES satellites which are capable of measuring the forces produced by the SPHERES thrusters and saturate at .03 G's. Alternatively, the velocity components can be pulled from the state vector (when using the global metrology system) and differentiated to determine linear acceleration. This time differentiation can be done by simply taking the slope of velocity plotted against time.

The total force is then divided by the number of thrusters to determine the force each thruster produced. This approach is simpler, relies on fewer measurement instruments, and inertia values need not be included in the calculations. However, the force of an individual thruster may vary based on how may other thrusters are open. (This may actually be an advantage since characterizing one thruster and then using multiple thrusters might produce unintended results due to thruster variation. That is, it may be effective to characterize only thruster combinations and then use those combinations during maneuvers. This, however, will severely constrain the flexibility of the maneuvers.) Also, it may be difficult to produce purely linear translation.

The second path tests each thruster individually and produces both a translational and rotational motion for each thruster. As Deb states, "The motion of a rigid body is a combination of translational motion of the center of mass and rotation around the center of mass", that is, "Every possible motion of a rigid body can be represented as a combination of translational motion of the center of mass and rotation about an axis through the center of mass. This is true even when the center of mass accelerates,

so that it is not at rest in any inertial frame."[16] MITOpenCourseWare provides the the derivations and explanations for this concept.[17] This situation is governed by

$$F_{Total} = F_{Linear} + F_{Torque} \tag{3.2}$$

where

$$F_{Linear} = ma_{GC} \tag{3.3}$$

such that

$$a_{GC} = \sqrt{a_{GC:\,x}^2 + a_{GC:\,y}^2 + a_{GC:\,z}^2} \tag{3.4}$$

$$a_{GC:\,x} = \frac{a_{1:\,x} + a_{2:\,x}}{2} = \frac{d^2}{dt^2}\left(\frac{p_{1:\,x} + p_{2:\,x}}{2}\right) \tag{3.5}$$

$$a_{GC:\,y} = \frac{a_{1:\,y} + a_{2:\,y}}{2} = \frac{d^2}{dt^2}\left(\frac{p_{1:\,y} + p_{2:\,y}}{2}\right) \tag{3.6}$$

$$a_{GC:\,z} = \frac{a_{1:\,z} + a_{2:\,z}}{2} = \frac{d^2}{dt^2}\left(\frac{p_{1:\,z} + p_{2:\,z}}{2}\right) \tag{3.7}$$

and

$$F_{Torque} = \sqrt{F_x^2 + F_y^2 + F_z^2} \tag{3.8}$$

such that

$$F_x = \frac{I_{xx}\dot{\omega}_x + (I_{zz} - I_{yy})\omega_y\omega_z}{r_x} \tag{3.9}$$

$$F_y = \frac{I_{yy}\dot{\omega}_y + (I_{xx} - I_{zz})\omega_z\omega_x}{r_y} \tag{3.10}$$

$$F_z = \frac{I_{zz}\dot{\omega}_z + (I_{yy} - I_{xx})\omega_x\omega_y}{r_z} \tag{3.11}$$

The mass, m, is known by weighing. The acceleration values, a, are determined using the accelerometers. Alternatively, the position values, p, are determined using the global metrology system. The time differentiation can be done by simply taking the slope of position plotted against time to determine velocity and then taking the slope of that velocity plotted against time to determine acceleration. If the analysis requires a more technically rigorous approach, the second difference may be taken since the

measurements are discrete.[18] Use

$$\frac{d^2x}{dt^2} \approx \frac{\Delta^2 x}{\Delta t^2} = \frac{x(t+\Delta t) - 2x(t) + x(t-\Delta t)}{(\Delta t)^2} \tag{3.12}$$

where

$$x(t+\Delta t) = x(t) + \Delta t x'(t) + \frac{1}{2}\Delta t^2 x''(t) + \frac{1}{6}\Delta t^3 x'''(t) + ... \tag{3.13}$$

and

$$x(t-\Delta t) = x(t) - \Delta t x'(t) + \frac{1}{2}\Delta t^2 x''(t) - \frac{1}{6}\Delta t^3 x'''(t) + ... \tag{3.14}$$

The angular velocity values, ω, can be determined using the gyroscope measurements. The angular acceleration values, denoted by $\dot{\omega}$, may be determined by time differentiating the gyroscope measurements. The distance values, denoted by r and based on the local reference frame, are known based on thruster geometry. The subscripts of the F and r terms require some discussion since they are, to a degree, misnomers. The subscripts are meant only to associate their variables with the respective torque value. Obviously, the T_z value is associated with r_x and F_y or r_y and F_x. The particular combination is unique to each thruster and for the sake of brevity, this paper has used the aforesaid misnomers. Fortunately, the analysis process is simplified by the fact that, due to thruster geometry, two-thirds of the thrusters produce a torque about only one axis, one-third of the thrusters produce a torque about two axes, and none of the thrusters produce a torque about all three axes. The inertia values are based on computer models. It is worth noting that any use of the global metrology system in this analysis will add a degree of difficulty since the thrusters cannot fire while the beacons are being used since the thruster firings will severely degrade the capability of the ultrasound beacons. The user will need to create a thruster firing scheme that allows the estimation and actuation phases to work together effectively and does not invalidate the above process.

Once the thruster forces have been characterized, the inertia tensor values may be determined empirically. Euler's equations of motion may be simplified by conducting pure rotation about one axis. This will be further discussed in Section 4.1.3 - the same

simplification is made but by assuming that the on-diagonal inertia tensor values are approximately equal. These linearized versions of Euler's equations of motion can be used to determine the inertia tensor values because the force values and thruster geometry have been determined empirically and the angular acceleration values can be determined by time differentiating the gyroscope measurements. Of course, the empirically determined inertia tensor values may be different from the inertia tensor values determined using computer models and utilized in the force characterization equations. Therefore, the inertia tensor values gathered from empirical results must be plugged back into the force characterization equations to produce new force values. These new force values will then be plugged into the inertia determination equations to determine new inertia values. This process must be iterated until the inertia tensor values converge.

3.1.3 Results and Analysis

a.i. solutions used the first method to achieve thruster values and inertia tensor values. Shown below is the range of values calculated for the thruster forces by a.i. solutions.

$$F = [.066 - .17]N \tag{3.15}$$

Shown below are the ranges of inertia values calculated for each principal axis of inertia by a.i. solutions.

$$I = \begin{bmatrix} [.145 - .410] & . & . \\ . & [1.186 - 3.360] & . \\ . & . & [1.096 - 3.104] \end{bmatrix} kgm^2 \tag{3.16}$$

Different results over different test runs caused the large range of thruster values and the large ranges of inertia tensor values. a.i. solutions believed that these different results were caused by thruster variations or failed thrusters. a.i. solutions instructed FIT and MIT to use the numbers predicted by the computer models since those numbers were approximately the same as the average of the values achieved by a.i.

solutions. [19]

3.2 Open-Loop Controller

The execution of effective open-loop control was one of the chief objectives of the first SSE ISS science test session.

Figure 3-2: The SSE floats in the microgravity environment of the ISS during the successful Science 1 test session. Credit: NASA.

3.2.1 Purpose

Open-loop control provides arguably the simplest method by which to control the motion of the SPHERES satellites. This low-cost solution allows the user to know the force outputs ahead of time (excepting the mechanical imperfections of the plant). Open-loop control, in the context of the SPHERES satellites, can be accomplished by feeding forces and torques into the mixer or by feeding thruster firing times directly into the plant. In the case of the SSE project, the thruster times were fed directly into the plant. This method was implemented with the global metrology enabled and with the global metrology disabled using different software schemes. This approach assumes that the designer understands the plant and environment sufficiently well so as to create control inputs that produce predictable and desirable plant behavior.

3.2.2 Theory

In the context of the SPHERES satellites, the force of an individual thruster cannot be modulated in any useful way. Each thruster can only be controlled in an on-off (bang-bang) way. Therefore, much of the SPHERES control must be understood in terms of impulse modulation.

The translational motion of the SSE can be understood in its analytical form using Equation 3.17. The respective variables are scalar values.

$$p = \int v\,dt = \int\int a\,dt\,dt = \int\int \frac{F}{m}\,dt\,dt \tag{3.17}$$

The rotational motion of the SSE can be understood in its analytical form using Equation 3.18. This equation shows the linearized form and its variables are scalar.

$$\theta = \int \omega\,dt = \int\int \alpha\,dt\,dt = \int\int \frac{\tau}{I}\,dt\,dt \tag{3.18}$$

These analytical form must be changed into numerical forms that are computationally effective. In order to make this change for the translational equations, use Equations

3.19 - 3.21 which have variables in the form of vectors and matrices.

$$a(t) = \frac{F(t)}{m} \tag{3.19}$$

$$v(t) = v(t-1) + a(t)\Delta t \tag{3.20}$$

$$p(t) = p(t-1) + v(t)\Delta t \tag{3.21}$$

Equations 3.19 - 3.21 are variants of equations used in the SSE motion MATLAB simulation developed by FIT (the equations presented here have had the coding jargon removed). In order to make the corresponding change for the rotational equations, use Equations 3.22 - 3.24. These equations use vectors and matrices for variables and are cleaned up versions of the equations used in the SSE motion MATLAB simulation.[20]

$$\alpha(t) = \frac{\tau - \omega(t-1) \times I\omega(t-1)}{I} \tag{3.22}$$

$$\omega(t) = \omega(t-1) + \alpha(t)\Delta t \tag{3.23}$$

$$\theta(t) = \theta(t-1) + \omega(t)\Delta t \tag{3.24}$$

3.2.3 Results and Analysis

Table 3.1 shows the test runs which were performed during the SSE science 1 test session.

Index num	Test num	Sat 1 result	Sat 2 result	Test description
1	1	1 (Normal)	1 (Normal)	Quick checkout
2	17	3 (Stopped via GUI)	103 (Stopped via GUI plus IR noise)	Manual pitch (no hood)
3	17	101 (Normal plus IR noise)	1 (Normal)	Manual rotation (no hood)
4	17	1* (Normal)	1* (Normal)	Manual translation (no hood)
5	17	1* (Normal)	1* (Normal)	Manual pitch (no hood)
6	17	1* (Normal)	1* (Normal)	Manual rotation (no hood)
7	2	1 (Normal)	1 (Normal)	Open-loop x translation
8	3	1* (Normal)	1* (Normal)	Manual x translation
9	8	253 (General VERTIGO error)	1 (Normal)	Open-loop x rotation
10	8	1 (Normal)	253 (General VERTIGO error)	Open-loop x rotation
11	10	1 (Normal)	1 (Normal)	Open-loop y rotation
12	11	103 (Stopped via GUI plus IR noise)	3 (Stopped via GUI)	Manual y rotation
13	15	3 (Stopped via GUI)	3 (Stopped via GUI)	Manual Pitch

Table 3.1: Table of test runs during the SSE science 1 test session. * denotes an unconfirmed value.

a.i. solutions provided MIT with the position components from the state vector data achieved for these thirteen test runs.[21] The motion induced by the manual maneuvers relies on the astronauts moving the SSE and does not rely on thruster firings. For this reason, the manual maneuvers represent a different kind of control. Furthermore, the estimator is less likely to be effective when the SSE is moving at velocities encompassed within the manual maneuvers. For these reasons, this analysis will focus on an open-loop test run with normal test results, namely, the test run denoted by index number 7. Figure 3-3 shows the intended movement of the SSE during test number 2 referenced by index number 7.

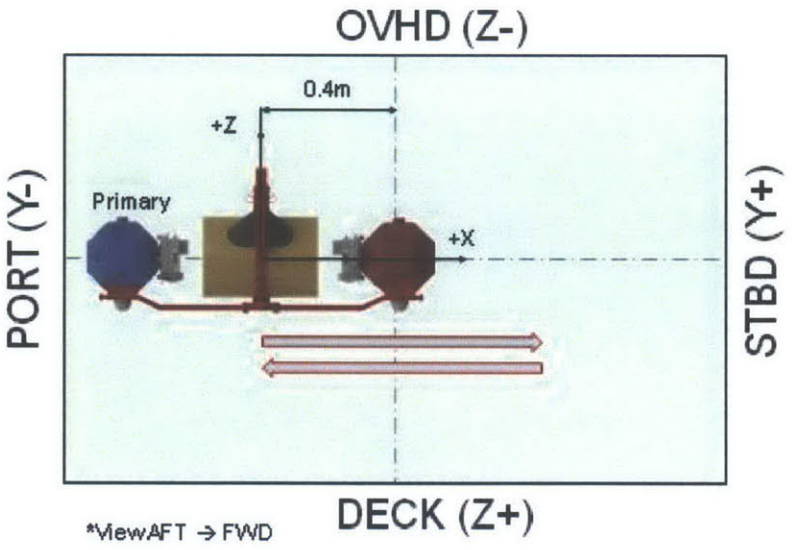

Figure 3-3: Open-loop x translation maneuver from aft view. Credit: FIT.

Figure 3-4 shows the empirical results achieved by the estimator for the index number 7 test run for the primary SPHERES satellite. Time $t = 0$ represents three seconds after the test has started - data before this point has been removed to avoid showing corrupted data. Figure 3-5 shows the empirical results achieved by the estimator for the same test run for the secondary SPHERES satellite. Time $t = 0$ represents three seconds after the test has started - data before this point has been removed to avoid showing corrupted data.

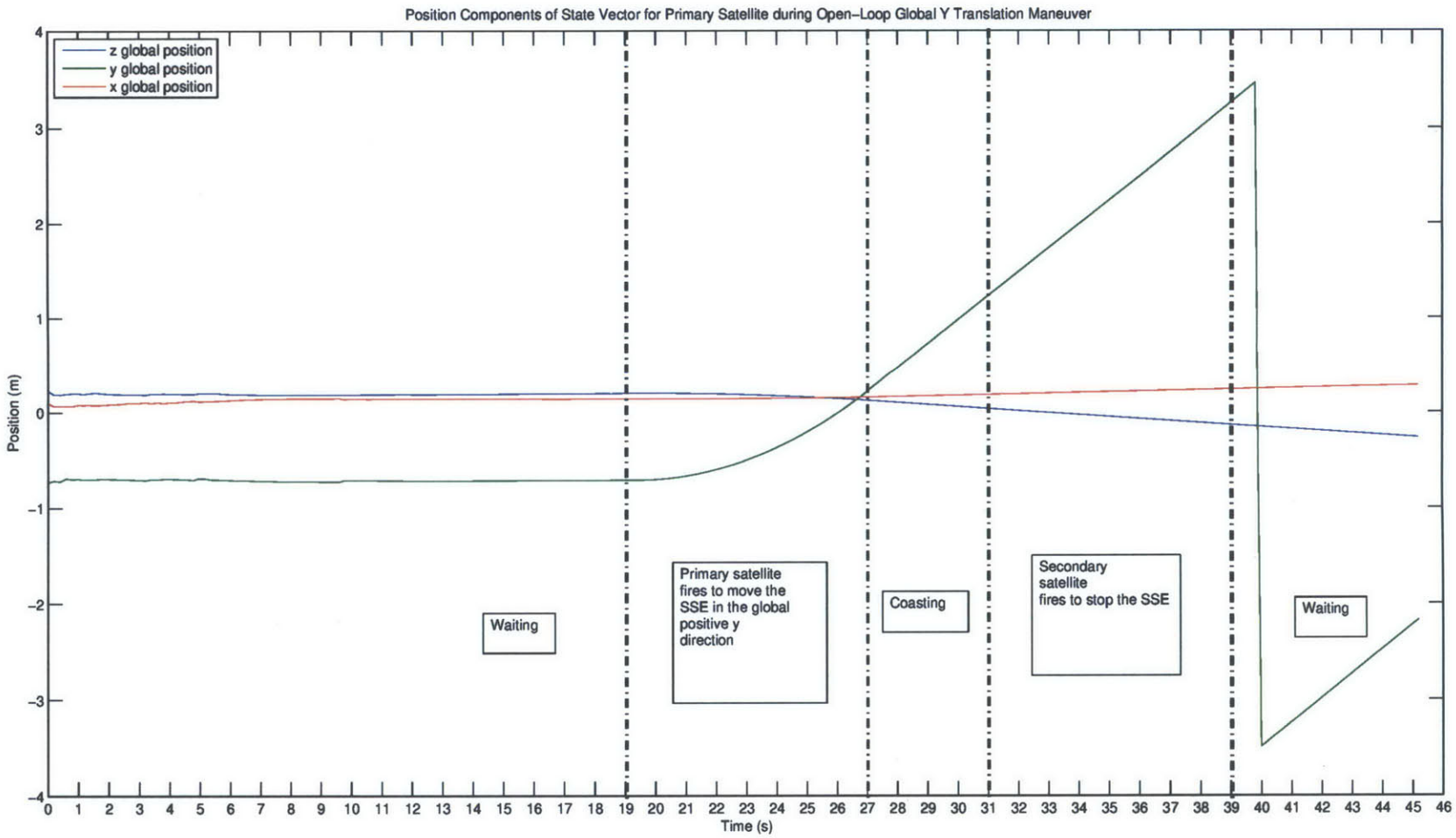

Figure 3-4: Empirical results from SSE science 1 test session: index number 7; primary SPHERES satellite.

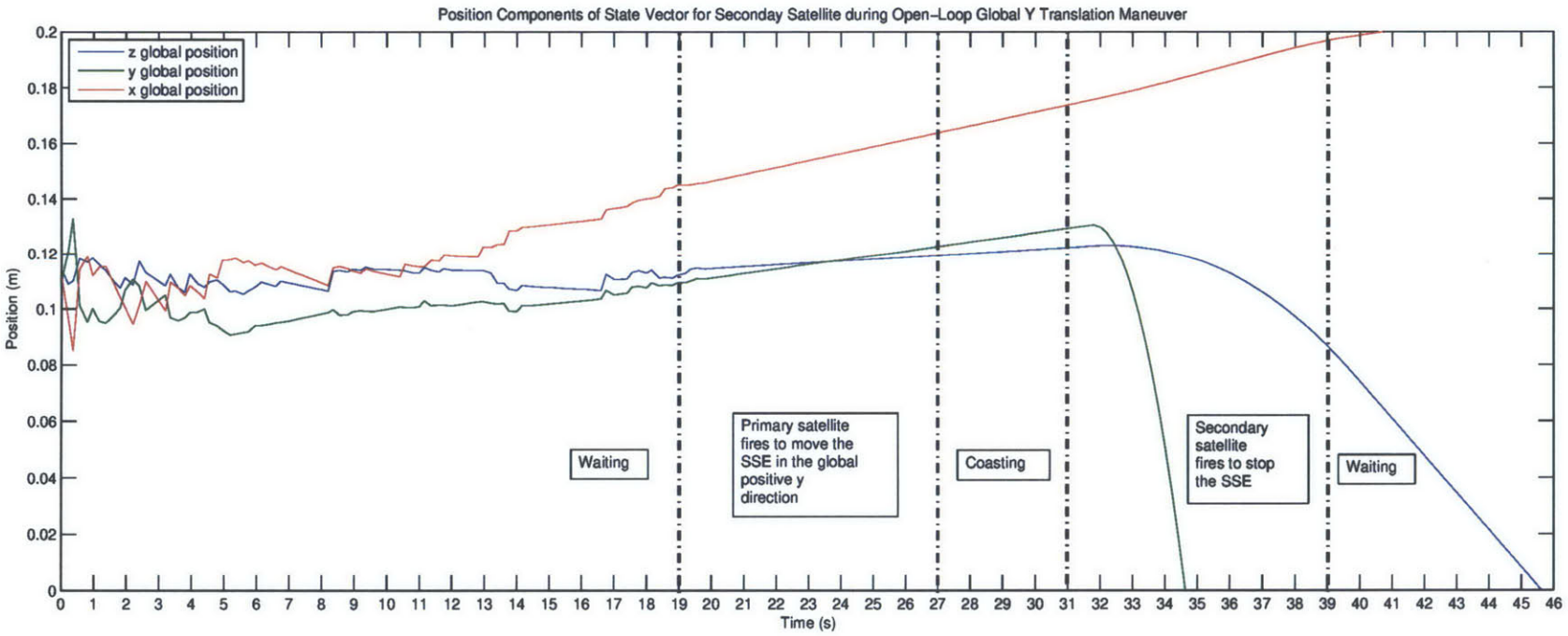

Figure 3-5: Empirical results from SSE science 1 test session: index number 7; secondary SPHERES satellite.

These graphs provide insight into some of the developmental steps which need to be taken to gather effective global metrology data for the SSE and, ultimately, to use closed-loop control. First, it must be understood that the firing phases represent a 100% actuator duty cycle with no global metrology and the coasting and waiting phases represent a 0% actuator duty cycle with global metrology being collected at 5 Hz. This intensive actuation was requested by a.i. solutions so that the relatively massive SSE system could be accelerated to a greater extent - this greater acceleration caused by longer thruster firings provided more effective data to a.i. solutions during the checkout session in the form of IMU data.

Global metrology cannot be used while the system is actuating. The estimator uses an Extended Kalman Filter (EKF) with a model of the plant. If the plant moves slightly away from where the model expects the plant to be, the estimator will trust the sensors. However, if the plant moves too quickly in a way that is not expected by the model and diverges too far from the expected position, the estimator will disregard the sensors and trust the model. Unfortunately, the model that is used for a SPHERES satellite is very different from the actual plant of the SSE. Some of these differences, such as mass, can be changed relatively easily. Others, such as adding 12 new actuators (cold gas thrusters), are much more difficult to change.

Looking at Figure 3-4, it is clear that the plant model used by the EKF does not account for the thruster firings of the other satellite. The metrology system correctly converges on the position of the primary satellite in the first waiting phase. Then, as the firing phase begins, the global metrology is turned off and the estimator relies on the model which says that the position should be changing quadratically due to constant acceleration. By the time the primary satellite enters the global metrology phase of coasting, the primary satellite has diverged too far and the estimator trusts only the model. Therefore, the position changes linearly due to constant velocity. When the secondary satellite begins to fire, the primary satellite does not account for this firing in the model and the position continues to grow linearly beyond any realistic bound. By the time the final waiting phase comes, the satellite has diverged so far that the estimator has been rendered ineffective.

The same dynamic can be seen by looking at Figure 3-5. The secondary satellite correctly converges during the first waiting phase. When the first actuation phase begins, the secondary satellite begins to rely on the model which has inflicted the state with a small amount of velocity due to either drift of the satellite or the noise of the estimator. The position progresses linearly according to a constant velocity and continues to do so during the coasting phase because, by this time, the estimator has disregarded the sensors. Upon the secondary satellite firing, the model accounts for this firing and shows the position of the secondary satellite rapidly changing in a quadratic manner.

When MIT began working on the SSE project, the full scope of the work needed to develop and implement the SSE system was not understood. This analysis demonstrates that changing the estimator at a fundamental level will likely be necessary to achieve closed-loop control for the SSE.

3.3 Summary

This chapter discussed several methods of control that were applied to the SPHERES satellites and the SSE and described how of those methods were implemented on the ISS. This chapter discussed system identification methods that could be used for the SSE including a purely linear method and an iterative method based on firing one thruster at a time.

Chapter 4

Control Theory and Design

This chapter proposes control theory and design work that can be implemented on the SSE. This chapter lays the groundwork for the proposed work.

4.1 Proportional, Proportional-Derivative, and Proportional-Integral-Derivative Controllers

4.1.1 Purpose

Proportional, Proportional-Derivative (PD), and Proportional-Integral-Derivative (PID) controllers provide relatively simple and effective ways of implementing closed-loop control. The interaction of an effective closed-loop controller with its environment can be invaluable to achieving certain system behavior and performance. In the case of the SPHERES system, the performance capability of closed-loop control can manifest itself in the ability of the SPHERES satellites to follow a specific path through the test volume at a specific velocity and attitude. In the case of the SSE, the SSE can be made to change its position, velocity, and attitude as necessary to achieve the required maneuvers. The salient idea encompassed within closed-loop control is that the system will change its output to the environment to achieve the desired behavior.

This point has implications for the execution of the SSE test sessions. Since the users have imperfect knowledge of the system and the environment, using effective

closed-loop control can give them a higher degree of certainty of accomplishing the desired maneuvers with the SSE. Therefore, closed-loop control provides a powerful tool for ensuring that settling maneuvers and slosh-inducement maneuvers are executed effectively.

The merit of closed-loop control must be balanced against its disadvantages. Developing a closed-loop controller presents a greater technical challenge than developing an open-loop controller and consequently will consume a greater amount of the organization's resources. Additionally, since the user has only a posteriori knowledge of the system output (thruster firings), data analysis will be more difficult; this component of data will be messier and known only after the test session. The need to develop more sophisticated data analysis tools will use additional resources.

4.1.2 Theory

Dynamical systems can be affected by offline parameters which are determined by design as well as online parameters which can be adjusted while the system is operating. Control is the use of these online parameters to make a dynamical system behave in a certain way. The two main approaches to control include feedforward and feedback. Feedforward control is a function of desired behavior only whereas feedback is a function of both the desired behavior and the actual observed behavior. Feedback often uses the difference between the desired behavior and the actual observed behavior, the error, to apply control. Control systems often use a combination of feedforward and feedback to achieve the desired behavior as effectively as possible.

The application of control theory and design often begins with the creation of a differential equation which characterizes the behavior of a system. This differential equation can always be reduced to an ordinary differential equation initial value problem of the form represented by the equation below.

$$\frac{dx(t)}{dt} = f(t, x(t), u(t), w(t)),\ x(t_0) = x_0 \tag{4.1}$$

In this equation, t is time where time is a real number, $x(t)$ is an n-dimensional state

vector where n is also the order of the system, $u(t)$ is an m_u-dimensional control vector, and $w(t)$ is an m_w-dimensional disturbance vector, for all real time, t.

Through the use of design on the respective system or appropriate assumptions, this expression may be linearized wherein any linear combination of inputs produce the same linear combination of outputs. This concept is also called the superposition of effects and may be represented by the equation below.

$$x(w+z) = x(w) + x(z) \tag{4.2}$$

Through the use of design on the respective system or appropriate assumptions, the system may also be made time invariant. A system is said to be time-invariant if time-shifted inputs produce time-shifted outputs, that is, the output of the system does not explicitly depend on time. This concept is represented by the expression below.

$$x(t) \rightarrow y(t),\ x(t+\delta) \rightarrow y(t+\delta) \tag{4.3}$$

If a system is such that it is both linear and time-invariant, then the system may be represented in the general form by the following two state-space equations.

$$\dot{\bar{x}}(t) = A\bar{x}(t) + B\bar{u}(t) \tag{4.4}$$

$$\bar{y}(t) = C\bar{x}(t) + D\bar{u}(t) \tag{4.5}$$

Let n represent the order of the original differential equation, r represent the number of inputs, and p represent the number of outputs. For the two equations above, $\bar{x}$ is the state vector of dimensions $(n \times 1)$, $\bar{u}$ is the input vector of dimensions $(r \times 1)$, $\bar{y}$ is the output vector of dimensions $(p \times 1)$, A is the system matrix of dimensions $(n \times n)$, B is the input matrix of dimensions $(n \times r)$, C is the output matrix of dimensions $(p \times n)$, and D is the coupling matrix of dimensions $(p \times r)$.

For a linear time-invariant system with perfect state knowledge, the optimal form of control to use is full-state feedback.

Given that full-state feedback will be used in the development of a controller,

there exist several approaches to creating said controller and multiple ways of making these approaches successful. One way to design a controller is by using direct pole placement based on the desired characteristics of the system. If only one stable pole needs to be considered than this pole will take the form represented by the equation below. Let σ represent the value associated with the horizontal axis of the s-domain, ζ represent the damping ratio, and ω_n represent the natural frequency.

$$\lambda = -\sigma = -\zeta\omega_n \tag{4.6}$$

If two stable poles need to be considered than these poles will take the form represented by the equation below. Let ω_d, the damping ratio, represent the value associated with the vertical axis in the s-domain.

$$\lambda = -\sigma \pm \omega_d j = -\zeta\omega_n \pm \omega_d j \tag{4.7}$$

In order to determine an appropriate place to put the poles using the method of direct pole placement, it is necessary to decide what performance characteristics the system should have in terms of percent overshoot, rise time, t_r, peak time, t_p, and settling time, t_s. Once these performance characteristics have been decided on, the equations below can be used to place boundaries within the s-domain that limit the area that poles can be placed. The following two equations characterize how the percent overshoot limits the angle from the negative real axis.

$$Percent\ Overshoot = e^{\frac{-\zeta\pi}{(1-\zeta^2)^{\frac{1}{2}}}}(100\%) \tag{4.8}$$

$$\theta = \cos^{-1}(\zeta) \tag{4.9}$$

The following four equations characterize the limitations placed on the s-plane by rise time.

$$\omega_d \approx \frac{\pi}{2t_r} \tag{4.10}$$

$$\sigma \approx \frac{3.358}{t_r} \tag{4.11}$$

$$t_{r,\,0\%\rightarrow 100\%} \approx \frac{1}{\omega_d}\left[\pi + \tan^{-1}\left(\frac{\omega_d}{-\zeta\omega_n}\right)\right] \tag{4.12}$$

$$t_{r,\,10\%\rightarrow 90\%} \approx \frac{1 + 1.1\zeta + 1.4\zeta^2}{\omega_n} \tag{4.13}$$

The following equation characterizes the limitation placed on the s-plane by settling time.

$$t_s = \frac{4}{\sigma} \tag{4.14}$$

The following equation characterizes the limitation placed on the s-plane by peak time.

$$t_p = \frac{\pi}{\omega_d} \tag{4.15}$$

Addressing a system where two stable poles are concerned, Equation (4.7) can be used to develop a desired characteristic equation. Having achieved a desired characteristic equation, direct algebraic substitution could be used by equating the desired characteristic equation and the closed-loop characteristic equation represented by the equation below.

$$determinant(\lambda I - (A - BK)) = 0 \tag{4.16}$$

This method, however, could become cumbersome for higher order systems that would require solving a system of equations with as many equations as the order of the system. For higher order systems, Ackerman's method is more effective. The Ackerman method is presented below.

1. Determine the controllability matrix, Q.

$$Q = [B, AB, A^2B, ..., A^{n-1}B] \tag{4.17}$$

2. Determine Q^{-1}.

$$Q^{-1} = \frac{1}{determinant(Q)} adjugate(Q) \tag{4.18}$$

3. Determine the desired characteristic equation from the poles.

4. Achieve the alpha values from the desired characteristic equation.

$$s^n + \alpha_{n-1}s^{n-1} + ... + \alpha_1 s + \alpha_0 = 0 \tag{4.19}$$

5. Determine the Cayley-Hamilton component.

$$\phi(A) = A^n + \alpha_{n-1}A^{n-1} + ... + \alpha_1 A + \alpha_0 \tag{4.20}$$

6. Determine the controller, K.

$$K = [0 ... 0\ 1]Q^{-1}\phi(A) \tag{4.21}$$

Another method for selecting the controller, K, is to use a linear-quadratic regulator. Linear-quadratic regulators provide a systematic method for balancing the importance of state accuracy and control effort. Essentially, this design approach seeks to minimize the cost function represented in the equation below.

$$J(x,u) = \int_0^T [x(t)^T Q x(t) + u(t)^T R u(t)]dt \tag{4.22}$$

The linear-quadratic regulator method is presented below.

1. Choose Q and R matrices based on trial and error or Bryson's rule.

2. Solve for the P matrix in the Riccati equation.

$$A^T P + PA + Q - PBR^{-1}B^T P = 0 \tag{4.23}$$

3. Solve for the controller, K.

$$K = R^{-1}B^T P \tag{4.24}$$

Another aspect of control theory and design concerns the use of proportional control, proportional-derivative control, and proportional-integral-derivative control.

Proportional control can be used to move the location of the poles on the root locus. This method can be very helpful when part of the root locus lies in the left-half plane of the s-domain such that a high enough gain will make the system stable. Proportional-derivative control adds a zero to the transfer function and changes the shape of the root locus. This method can be very helpful if the actual shape of the root locus needs to be changed in order to achieve stability for the system. Proportional-integral-derivative control adds a zero to the transfer function as well as a free s-term which serves as an additional pole. Adding the integrator can be very helpful in reducing or eliminating steady-state error because it increases the number of free s-terms in the denominator of the transfer function and therefore increases the type of the system. On a block diagram, this concept can be understood as integrating an error in the system and then sending that integrated error through a feedback loop so that the appropriate adjustments can be made to the online parameters so as to ensure that the value coming out of the integrator decreases to zero. For a step input, $u(t)$, the steady state-error for a type 0 and type 1 system is given by the equations below, respectively. Let K_B represent the Bode gain and N represent the type of system.

$$e_{SS,\,N=0} = \frac{1}{1+K_B} \tag{4.25}$$

$$e_{SS,\,N=1} = 0 \tag{4.26}$$

For a ramp input, $r(t)$, the steady-state error for a type 0, type 1, and type 2 system is given by the equations below respectively.

$$e_{SS,\,N=0} = \infty \tag{4.27}$$

$$e_{SS,\,N=1} = \frac{1}{K_B} \tag{4.28}$$

$$e_{SS,\,N=2} = 0 \tag{4.29}$$

Some systems require the use of a scaling matrix in the feedforward component of the system. In this case, the state-space equation must be augmented by multiplying

the B matrix by the term $\bar{N}$ where $\bar{N}$ is defined in the equation below. Let S be the scaling matrix.

$$\bar{N} = KS = \frac{-1}{C(A - BK)^{-1}B} \tag{4.30}$$

Of course, the effectiveness and feasibility of any of the aforementioned control theory and design methods depends greatly on the system for which the control is being implemented.

4.1.3 Application to SPHERES

This subsection deals with the SPHERES satellites without the SSE augmentations. This section was written for one SPHERES satellite to show how the SSE hardware augmentations do not fundamentally change the system in terms of PD control about a single axis. That is, the same control methods can be applied. As with many real-life operational systems, SPHERES is non-linear, time varying, and lacks perfect state knowledge. However, given the design of the system, there exist certain assumptions which can be made that simplify the model of the SPHERES system to one that is linear, time-invariant, and possesses perfect state knowledge.

Each of the SPHERES satellites uses a carbon dioxide tank to fuel its twelve cold gas thrusters which are used for actuation. The liquid slosh within each carbon dioxide tank changes the A matrix of the system with respect to time giving the SPHERES satellite the time-varying characteristic. However, the mass, inertia, and thrusting capability of each SPHERES satellite in relation to the carbon dioxide tanks as well as the duration of each SPHERES test causes the effect of this liquid slosh to be negligible. Given that the liquid slosh is negligible, the SPHERES system is time-invariant.

Without the SPHERES software to control the bang-bang thrusters and assuming that the motions of a SPHERES satellite are coupled in a non-negligible manner, the SPHERES system would need to be modeled as a non-linear system and have its motion be modeled by the general vector form of Euler's equation of motion

represented in the equation below.

$$I\dot{\omega} + \omega \times (I\omega) = T \tag{4.31}$$

However, there exist design features and software which allow the SPHERES system to be modeled as a linear system. Since the SPHERES system inertia matrix encompasses on-diagonal values which are much larger than its off-diagonal values, the motion a SPHERES satellite can be modeled using the three simplified Euler equations listed below.

$$I_1\dot{\omega_1} + (I_3 - I_2)\omega_2\omega_3 = T_1 \tag{4.32}$$

$$I_2\dot{\omega_2} + (I_1 - I_3)\omega_3\omega_1 = T_2 \tag{4.33}$$

$$I_3\dot{\omega_3} + (I_2 - I_1)\omega_1\omega_2 = T_3 \tag{4.34}$$

As a result of the geometry of the SPHERES hardware and the associated mass distribution, the SPHERES on-diagonal values are approximately equal such that

$$I_1 \approx I_2 \approx I_3 \approx I \tag{4.35}$$

therefore, the Euler equations may be simplified to

$$I\dot{\omega_1} = T_1 \tag{4.36}$$

$$I\dot{\omega_2} = T_2 \tag{4.37}$$

$$I\dot{\omega_3} = T_3 \tag{4.38}$$

Given that Equation 4.35 holds true, the equation of motion describing the rotation of a SPHERES satellite about any axis of rotation is shown below.

$$\ddot{\theta} = \frac{u}{I} \tag{4.39}$$

In describing the translational motion of a SPHERES satellite, Newtons second law

may be used and is represented by the below equation.

$$F = ma \tag{4.40}$$

Therefore, the equation of motion describing the translation of a SPHERES satellite along any axis of translation is shown below.

$$\ddot{x} = \frac{u}{m} \tag{4.41}$$

Given the aforesaid equations of motion represented by Equations 4.39 and 4.41, the SPHERES system may be considered a linear system.

While the SPHERES system does not possess perfect state knowledge, the metrology system (sensor) and the software filter (estimator) are effective enough such that this assumption may be made under nominal conditions.

Given that the SPHERES system may be assumed to be a linear time-invariant system with perfect state knowledge, the generalized state-space equations represented in Equations 4.4 and 4.5 may be modified to give the state-space representation

of the SPHERES system. This representation is shown in the equations below.

$$\begin{bmatrix} \dot{x}_x \\ \ddot{x}_x \\ \dot{\theta}_x \\ \ddot{\theta}_x \\ \dot{x}_y \\ \ddot{x}_y \\ \dot{\theta}_y \\ \ddot{\theta}_y \\ \dot{x}_z \\ \ddot{x}_z \\ \dot{\theta}_z \\ \ddot{\theta}_z \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_x \\ \dot{x}_x \\ \theta_x \\ \dot{\theta}_x \\ x_y \\ \dot{x}_y \\ \theta_y \\ \dot{\theta}_y \\ x_z \\ \dot{x}_z \\ \theta_z \\ \dot{\theta}_z \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{m} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{I} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{m} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{I} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{m} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{I} \end{bmatrix} \begin{bmatrix} F_x \\ T_x \\ F_y \\ T_y \\ F_z \\ T_z \end{bmatrix} \tag{4.42}$$

$$\bar{y} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_x \\ \dot{x}_x \\ \theta_x \\ \dot{\theta}_x \\ x_y \\ \dot{x}_y \\ \theta_y \\ \dot{\theta}_y \\ x_z \\ \dot{x}_z \\ \theta_z \\ \dot{\theta}_z \end{bmatrix} \tag{4.43}$$

The C matrix exists as the identity matrix because the estimator of the SPHERES system computes the state vector based on sensor outputs. While there does exist error in the estimator values, these values may be taken as truth because of the

aforementioned effectiveness of the sensor and estimator systems.

Given the form of the state-space equations above, the gain matrix K will take the form shown in the equation below.

$$K = \begin{bmatrix} K_1 & K_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & K_3 & K_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & K_5 & K_6 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & K_7 & K_8 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & K_9 & K_10 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & K_11 & K_12 \end{bmatrix} \tag{4.44}$$

Given that the A matrix is decoupled as it is and that the mass and inertia are equivalent for each axis of translation and rotation, respectively, it may be said that many subsets of K matrices are the same as described in the equations below.

$$[K_1\ K_2] = [K_5\ K_6] = [K_9\ K_10] \tag{4.45}$$

$$[K_3\ K_4] = [K_7\ K_8] = [K_{11}\ K_{12}] \tag{4.46}$$

Furthermore, for the development of these controllers and the transition of the state-space representation to transfer function representation, Equation 4.42 may be simplified into the following two equations.

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} F \tag{4.47}$$

$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{I} \end{bmatrix} T \tag{4.48}$$

Using the equation below, these state-space representations can be turned into transfer function representations.

$$Transfer\ Function = C(sI - A)^{-1}B \tag{4.49}$$

The transfer functions for both translational motion and rotational motion are shown below, respectively.

$$Transfer\ Function_{Translation} = \frac{1}{ms^2} \tag{4.50}$$

$$Transfer\ Function_{Rotation} = \frac{1}{Is^2} \tag{4.51}$$

Equations 4.47 and 4.48 may be augmented to represent an integrator state being added to the original states. These representations are shown below.

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{x}_I \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ x_I \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m} \\ 0 \end{bmatrix} F + \begin{bmatrix} 0 \\ 0 \\ x_T \end{bmatrix} \tag{4.52}$$

$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \\ \dot{\theta}_I \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ \theta_I \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{I} \\ 0 \end{bmatrix} T + \begin{bmatrix} 0 \\ 0 \\ \theta_T \end{bmatrix} \tag{4.53}$$

While adding an integrator can be very effective for many systems, adding an integrator into the SPHERES control (using this approach and without further augmentations to the controller) would be ineffective because the SPHERES system is already a type 2 system and adding an integrator, even with the addition of a zero, would drive the root locus into the right-half plane of the s-domain. Additionally, given that the SPHERES system is already a type 2 system, the steady-state error should not be an issue.

Therefore, use proportional-derivative control and let the controller take the form represented by the equation below.

$$u = -kx \tag{4.54}$$

Using the method of direct pole placement combined with Ackerman's method, the K matrix can be determined to be the values shown in the equations below for

translation and rotation, respectively.

$$K_{translation} = [\zeta^2 \omega_n^2 m + \omega_d^2 m,\ 2\zeta\omega_n m] \tag{4.55}$$

$$K_{rotation} = [\zeta^2 \omega_n^2 I + \omega_d^2 I,\ 2\zeta\omega_n I] \tag{4.56}$$

Let the damping frequency be defined by the relationship represented in the equation below.

$$\omega_d = \omega_n (1 - \zeta^2)^{\frac{1}{2}} \tag{4.57}$$

Therefore, Equations 4.55 and 4.56 can be simplified to the equations shown below.

$$K_{translation} = [\omega_n^2 m,\ 2\zeta\omega_n m] \tag{4.58}$$

$$K_{rotation} = [\omega_n^2 I, 2\zeta\omega_n I] \tag{4.59}$$

Therefore, the control input, u, takes the form shown in the equations below.

$$u_{translation} = -\omega_n^2 mx - 2\zeta\omega_n m\dot{x} \tag{4.60}$$

$$u_{rotation} = -\omega_n^2 I\theta - 2\zeta\omega_n I\dot{\theta} \tag{4.61}$$

Let the mass and inertia values be shown by the equations below, respectively.

$$m = 4.377 kgm^2 \tag{4.62}$$

$$I = .023 kgm^2 \tag{4.63}$$

The other values such as ω_n, ω_d, and ζ can be determined based on the where the poles are placed using direct pole placement. Where these poles are placed can be determined using Equations 4.8 through 4.15. The controllers may also be designed using the aforementioned linear-quadratic regulator method.

4.1.4 Proposal for the SPHERES-Slosh Experiment

The subsection deals with the entire SSE. The figure below shows the SSE setup and identifies the local reference frames of the primary SPHERES satellite, the secondary SPHERES satellite, and the entire SSE experiment.

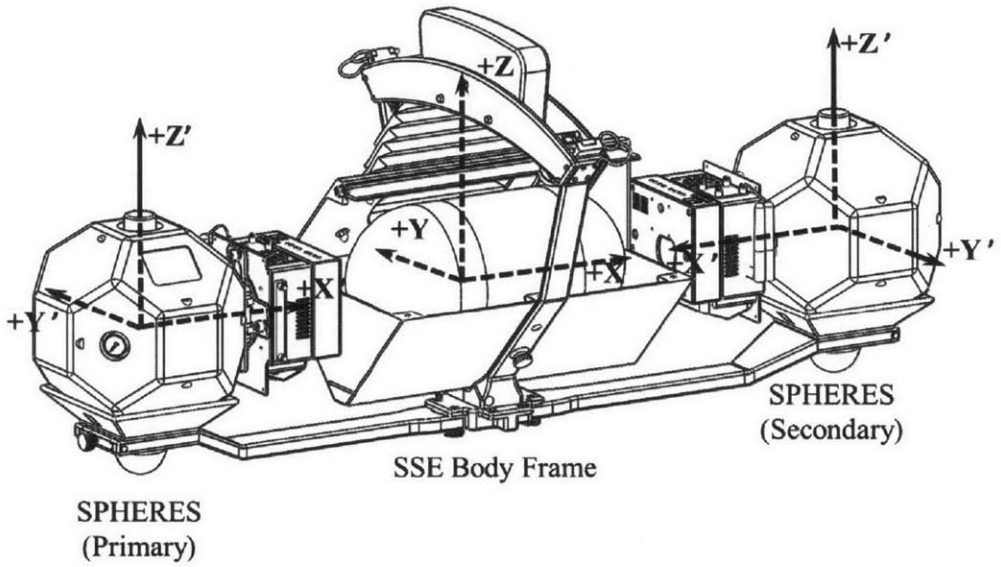

Figure 4-1: The local reference frames of the SPHERES satellites and SSE in standard configuration. Credit: FIT.

The design of the SSE attempts to ensure that the on-diagonal values of the SSE inertia matrix are substantially greater than the off-diagonal values. This feature is shown in the equation below which represents the dry mass of the SSE in standard configuration as calculated by a computer model.[22] These numbers have not been validated by a swinging parallelogram test. For a discussion on how these numbers were verified in a microgravity environment, please refer to Section 3.1.

$$I = \begin{bmatrix} .3007 & .0028 & 0 \\ .0028 & 2.5108 & 0 \\ 0 & 0 & 2.3864 \end{bmatrix} kgm^2 \tag{4.64}$$

This design feature is a necessary condition for the simplification of Euler's equation of motion to be carried out here such that Equations 4.32 through 4.34 remain

valid. The contribution of inertia by the water and carbon dioxide must be considered. This is especially true since the inertia values of the carbon dioxide tanks and the slosh tank will change with respect to time for different configurations of the fluids within their respective tanks. The tables below show the maximum and minimum inertia matrix values of the water in the 40%-filled and 20%-filled slosh tanks. Additionally, these tables show the range of the centers of gravity of the fluid along each axis as well as the mass of the fluid.[22] Let the x-axis be the axis oriented along the length of the slosh tank as shown in Figure 4-1. Understand that the values of each column must be considered in isolation as each cell represents an individual worst-case configuration.

	CG_x	CG_y	CG_z	I_{xx}	I_{yy}	I_{zz}	I_{xy}	I_{xz}	I_{yz}	Mass
Maximum	.0766	.0366	.0366	.0075	.0216	.0216	.004	.004	0	1.7672
Minimum	-.0766	-.0366	-.0366	.0043	.011	.011	0	0	0	1.7672
Range	.1531	.0731	.0731	.0032	.0106	.0106	.004	.004	0	0

Table 4.1: Maximum and minimum inertia matrix values (kgm^2), centers of gravity (m), and mass (kg) of fluid in 40%-filled tank.

	CG_x	CG_y	CG_z	I_{xx}	I_{yy}	I_{zz}	I_{xy}	I_{xz}	I_{yz}	Mass
Maximum	.1031	.0505	.0505	.0043	.0135	.0135	.0031	.0031	0	.8836
Minimum	-.1031	-.0505	-.0505	.0018	.005	.005	0	0	0	.8836
Range	.2062	.1010	.1010	.0025	.0085	.0085	.0031	.0031	0	0

Table 4.2: Maximum and minimum inertia matrix values (kgm^2), centers of gravity (m), and mass (kg) of fluid in 20%-filled tank.

Since the inertia values of the fluid are orders of magnitude less than the associated values of the SSE, the impact of the water on the inertia of the system will be considered negligible. Since the mass of the carbon dioxide in each tank($.172kg$) is orders of magnitude less than the mass of the SSE, the contribution of the carbon dioxide slosh to inertia values will also be considered negligible. Therefore, the simplification of Euler's equation of motion remains valid. However, the inertia of

each principal axis has a different value. Therefore, Equations 4.36 through 4.38 do not remain valid for the SSE. However, there exists a different method of making the aforesaid simplification: If the SSE is rotated about only one axis at a time such that two of the three angular velocities are kept at approximately zero, then the system may be linearized such that Equation 4.39 holds true. The translational dynamics associated with the SSE are the same as the translational dynamics of the SPHERES satellites such that Equation 4.41 holds true. Therefore, if the angular velocities are appropriately constrained, the SSE is governed by the same motion dynamics and the SSE closed-loop controllers may be developed in fundamentally the same way as described in Section 4.1.3. If the rotational motion is not constrained, the user must develop a non-linear controller for attitude control of the SSE.

In order to create effective closed-loop control for the SSE, a new mixer and state merger must also be developed. In developing the mixer, the first step is to determine the distance (in x, y, z coordinates) from the geometric center of the SSE to each thruster of the system. The geometric center of the SSE is the origin of the SSE body frame shown in Figure 4-1 and is calculated as the midpoint between the origin of the two local SPHERES reference frames also shown in Figure 4-1. To do this, let $d_1 = distance\, between\, SPHERES\, reference\, frame\, origins$ and let $d_2 = (.5)d_1$. Let $d_3 = .965mm$ and let $d_4 = .516mm$. The d_3 and d_4 values represent the distances from the SPHERES reference frame origin to various thrusters. Reference the SPHERES Guest Scientist Program (GSP) for a detailed account of thruster placement on the SPHERES. Add values d_{1-4} appropriately for each thruster to establish accurate distances for the 24 thrusters. After distances have been established, determine the nominal force direction of each thruster with respect to the SSE. Table 4.3 shows the appropriate distances and resultant force directions for each of the 24 thrusters on the SSE. Let thrusters 1-12 represent thrusters 0-11 on the primary SPHERES satellite and let thrusters 13-24 represent thrusters 0-11 on the secondary SPHERES satellite as laid out in the SPHERES GSP.

Thruster number	Thruster position			Resultant force direction		
	x	y	z	x	y	z
1	$-d_2 - d_4$	0	d_3	1	0	0
2	$-d_2 - d_4$	0	$-d_3$	1	0	0
3	$-d_2 + d_3$	$-d_4$	0	0	1	0
4	$-d_2 - d_3$	$-d_4$	0	0	1	0
5	$-d_2$	d_3	$-d_4$	0	0	1
6	$-d_2$	$-d_3$	$-d_4$	0	0	1
7	$-d_2 + d_4$	0	d_3	-1	0	0
8	$-d_2 + d_4$	0	$-d_3$	-1	0	0
9	$-d_2 + d_3$	d_4	0	0	-1	0
10	$-d_2 - d_3$	d_4	0	0	-1	0
11	$-d_2$	d_3	d_4	0	0	-1
12	$-d_2$	$-d_3$	d_4	0	0	-1
13	$d_2 + d_4$	0	d_3	-1	0	0
14	$d_2 + d_4$	0	$-d_3$	-1	0	0
15	$d_2 - d_3$	d_4	0	0	-1	0
16	$d_2 + d_3$	d_4	0	0	-1	0
17	d_2	$-d_3$	$-d_4$	0	0	1
18	d_2	d_3	$-d_4$	0	0	1
19	$d_2 - d_4$	0	d_3	1	0	0
20	$d_2 - d_4$	0	$-d_3$	1	0	0
21	$d_2 - d_3$	$-d_4$	0	0	1	0
22	$d_2 + d_3$	$-d_4$	0	0	1	0
23	d_2	$-d_3$	d_4	0	0	-1
24	d_2	d_3	d_4	0	0	-1

Table 4.3: Thruster geometry and resultant force directions.

Following the creation of Table 4.3, establish a 24 by 3 matrix composed of the

thruster position values shown in Table 4.3 called R. Establish the matrix R_{CG} of the same dimensions such that

$$R_{CG} = \begin{bmatrix} . & . & . \\ . & . & . \\ R_{CG,X} & R_{CG,Y} & R_{CG,Z} \\ . & . & . \\ . & . & . \end{bmatrix} \tag{4.65}$$

where $R_{CG,X}$, $R_{CG,Y}$, and $R_{CG,Z}$ represent the distance from origin of the SSE body reference frame called the geometric center to the center of gravity of the SSE in x, y, and z coordinates, respectively. Let

$$R_{New} = R - R_{CG} \tag{4.66}$$

Create the matrix F such that it contains the 24 by 3 component of Table 4.3 which encompasses the resultant force directions. Let torque, T, be

$$T = R_{New} \times F \tag{4.67}$$

Aggregate the force and torque matrices to produce the inverse mixer.

$$Inverse\ Mixer = [\, F\ T \,] \tag{4.68}$$

Transpose this matrix to produce the prime inverse matrix.

$$Prime\ Inverse\ Matrix = (Inverse\ Mixer)' \tag{4.69}$$

Finally, let the mixer matrix be the right pseudo-inverse of the prime inverse matrix.[23]

$$Mixer = (Prime\ Inverse\ Matrix)^{-1}_{right} \tag{4.70}$$

State merging can be accomplished using multiple methods. This writing will discuss four approaches. The first approach uses a series of logical algorithms to discount invalid measurements and simple math to combine valid measurements. In this approach, state merging presents a relatively simple problem when the estimator presents the state merging algorithm with nominal values for the state vector that make sense for the physical system. Given that the estimator reports values which are accurate as determined by the state merging algorithm, the equations below can be used to merge the states. Let p represent the position components, let v represent the velocity components, let q represent the quaternion components, and let r represent the rate components.

$$p_{M:\,x,y,z} = \frac{p_{1:\,x,y,z} + p_{2:\,x,y,z}}{2} \tag{4.71}$$

$$v_{M:\,x,y,z} = \frac{v_{1:\,x,y,z} + v_{2:\,x,y,z}}{2} \tag{4.72}$$

$$q_{M:\,1,2,3,4} = q_{1:\,1,2,3,4} \tag{4.73}$$

$$r_{M:\,x,y,z} = r_{1:\,x,y,z} \tag{4.74}$$

Essentially, this process averages the position and velocity information from the two SPHERES satellites and takes the quaternion and rate information of the primary SPHERES satellite to be that of the SSE since both of the respective body frames are aligned in terms of attitude.

Of course, methods must be implemented to identify and address off-nominal scenarios. The beginning step is to identify off-nominal scenarios. Position will be addressed first. Take the distance between the two SPHERES:

$$d = \sqrt{|p_{1:\,x} - p_{2:\,x}|^2 + |p_{1:\,y} - p_{2:\,y}|^2 + |p_{1:\,z} - p_{2:\,z}|^2} \tag{4.75}$$

Determine if this distance falls within an acceptable limit as shown by the algorithm logic

$$if(d_1 - \Delta \leq d \leq d_1 + \Delta) \tag{4.76}$$

where Δ is some offset margin from d_1.

Velocity will be addressed next. Determine the position vector of the secondary SPHERES satellite with respect to the primary SPHERES satellite.

$$\bar{p}_L = [p_{2:x} - p_{1:x},\ p_{2:y} - p_{1:y},\ p_{2:z} - p_{1:z}]^T \tag{4.77}$$

Determine the velocity vector of the secondary SPHERES satellite with respect to the primary SPHERES satellite.

$$\bar{v}_L = [v_{2:x} - v_{1:x},\ v_{2:y} - v_{1:y},\ v_{2:z} - v_{1:z}]^T \tag{4.78}$$

Determine if $\bar{p}_L$ and $\bar{v}_L$ are orthogonal vectors. They must be perpendicular to one another to accurately characterize the motion of the SSE. To do this, take the dot product of the two vectors and determine if the result is sufficiently close to the value of zero. Use

$$dot = \bar{p}_L \cdot \bar{v}_L \tag{4.79}$$

$$if(-\Delta \leq dot \leq \Delta) \tag{4.80}$$

where Δ is some offset margin from zero.

Attitude will be addressed next. The z-axes should have the same alignment and the x and y-axes should be 180 degrees apart. In other words, it may be said that the secondary SPHERES satellite is rotated 180 degrees about the z-axis from the primary SPHERES satellite. A quaternion may be described using

$$\bar{q} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} = \begin{bmatrix} v_x \sin(.5\theta) \\ v_y \sin(.5\theta) \\ v_z \sin(.5\theta) \\ \cos(.5\theta) \end{bmatrix} \tag{4.81}$$

where

$$\bar{v} = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} \tag{4.82}$$

represents a three dimensional unit vector and θ represents the rotation about that vector. Therefore, the quaternion that would describe the attitude of the secondary satellite with respect to the reference frame of the primary satellite is

$$\bar{q} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \tag{4.83}$$

To determine if the estimator is providing potentially correct values, use the find-state-error method based on Hilstad's thesis.[23] Instead of using this method to determine the error quaternion, use this method to determine the relationship of the secondary satellite and primary satellite.

$$\bar{q}_1 = \begin{bmatrix} q_{1:1} \\ q_{1:2} \\ q_{1:3} \\ q_{1:4} \end{bmatrix} \tag{4.84}$$

$$\bar{q}_2 = \begin{bmatrix} q_{2:1} \\ q_{2:2} \\ q_{2:3} \\ q_{2:4} \end{bmatrix} \tag{4.85}$$

$$\bar{q}_e = \begin{bmatrix} q_{e:1} \\ q_{e:2} \\ q_{e:3} \\ q_{e:4} \end{bmatrix} = \begin{bmatrix} -q_{2:1}q_{1:4} - q_{2:2}q_{1:3} + q_{2:3}q_{1:2} + q_{2:4}q_{1:1} \\ q_{2:1}q_{1:3} - q_{2:2}q_{1:4} - q_{2:3}q_{1:1} + q_{2:4}q_{1:2} \\ -q_{2:1}q_{1:2} + q_{2:2}q_{1:1} - q_{2:3}q_{1:4} + q_{2:4}q_{1:3} \\ q_{2:1}q_{1:1} + q_{2:2}q_{1:2} + q_{2:3}q_{1:3} + q_{2:4}q_{1:4} \end{bmatrix} \tag{4.86}$$

$$if(q_{e:4} < 0) \rightarrow \bar{q}_e = -\bar{q}_e \tag{4.87}$$

$$\hat{q}_e = \frac{\bar{q}_e}{|\bar{q}_e|} \tag{4.88}$$

The resultant vector, $\hat{q}_e$, describes how the secondary satellite would need to rotate in order to be aligned with the reference frame of the primary satellite. For the SSE, this resultant vector should be represented by Equation 4.83. Of course, this comparison needs to account for some offset margin, Δ.

Finally, angular rate will be addressed. The z-axis gyroscopes used in the primary and secondary satellites should output the same measurements. The x and y-axis gyroscopes should output measurements which are equal in magnitude and opposite in sign such that

$$r_{1:x} = -r_{2:x} \tag{4.89}$$

$$r_{1:y} = -r_{2:y} \tag{4.90}$$

$$r_{1:z} = r_{2:z} \tag{4.91}$$

The offset margin, Δ, must be accounted for.

Once off-nominal conditions have been identified, the key is to isolate and discard the particular inaccurate measurements which would distort the state merging. This can be done by checking for divergent state components. In the cases where the inaccurate measurements cannot be attributed to a particular satellite, the respective state components may be discarded for both satellites. Once inaccurate measurements have been discarded, those discarded state components can be filled in using other state components and math or simply ignored throughout the state merging process.

The second approach uses the existing Extended Kalman Filter (EKF) which encompasses a state model of the plant and and includes additional processes which account for latency in communication between the two satellites. While the estimator could be redesigned to create a more elegant solution that started from the baseline of the measurement instruments, it would be more expedient to use the current estimator to feed individual state vectors into a new software process that merged the state vectors into one state vector. Ideally, each satellite's state merging software process would receive the same information at the same time, execute the same calculations using the same internal processes, and output the same information at the same time. However, the process must be created in a way that accounts for and addresses non

ideal situations.

The third approach fundamentally changes the plant estimator to account for new actuators in the plant model. This change presents a more difficult problem than simply accounting for the new physical parameters such as mass, inertia, and center of gravity. This change must also incorporate the changes discussed in the second approach; essentially, the distributed computing and resultant latency must be accounted for with new software processes that handle the state merging.

The fourth approach upgrades the sensor suite on the SPHERES platform so that the estimator does not rely so heavily upon the model and can work even when the actuators are being utilized. Again, this change must encompass the changes made in second and third approach.

These approaches have been described in a very general manner and represent point solutions on a spectrum of difficulty that encompasses a multitude of options. One of the key points for the reader to understand is that hardware and software aggregation, at some point, becomes more than just an augmentation to an existing system. The kind of aggregation carried out for the SSE represents the creation of a fundamentally different system. The SSE encompasses two SPHERES satellites as well as the accompanying global metrology system and, as a system, shares many similarities to the individual SPHERES satellite system. However, the SSE and the individual SPHERES satellite are fundamentally different creations.

4.2 Simple Adaptive Controller

This section deals specifically with SPHERES satellites without SSE augmentations but these methods could potentially be used for the SSE to deal with unknown perturbations or unknown mass. The simulations were conducted in MATLAB Simulink.

4.2.1 Purpose

This controller provides simple adaptive translational control of the SPHERES satellites. Simulation demonstrates that this controller is more effective when subjected

to an unknown mass or an unknown mass and unknown perturbation. This enhancement is possible because the controller has the capability to adjust the control gains online based on the error whereas linear time invariant PD controllers use fixed gains. This subsection provides the framework for developing a simple adaptive controller for the SSE.

4.2.2 Theory

Given that the SPHERES system may be assumed to be a linear time-invariant system with perfect state knowledge, state-space equations may be derived to represent the SPHERES system. Through a series of simplifications and calculations, a transfer function associated with the translational motion of SPHERES may be created and is shown in Equation 4.50 Analysis of the root locus associated with this transfer function demonstrates that a proportional controller can achieve only marginal stability. However, a proportional-derivative controller of the form

$$C(s) = K_d\left(s + \frac{K_p}{K_d}\right) \tag{4.92}$$

with a sufficiently high gain may be used to stabilize the plant. Specifically, the following proportional-derivative controller may be used to stabilize a SPHERES satellite.

$$C(s) = 3.88(s + .1) \tag{4.93}$$

While this controller will perform satisfactorily under nominal conditions with known mass, no unknown perturbations, and full state feedback, this proportional-derivative control method will be ineffective for an unknown mass, an unknown disturbance force, or when using output feedback. Adaptive control must be used to control a SPHERES system of unknown mass undergoing unknown disturbances using only output feedback.

Adaptive control cannot be directly applied to this problem with any stability guarantees because the plant is not minimum phase and its transfer function is not

almost strictly positive real (ASPR). However, it is known that if a plant is stabilizable (or maintains its stability) with some controller, augmenting the plant with the inverse of this controller as a causal parallel feedforward with relative degree of one makes it ASPR. Specifically, the following proportional-derivative controller is selected.

$$H(s) = K_d(s + 2.5) \tag{4.94}$$

The root locus of this controller with the plant shows that satisfactory behavior can indeed be obtained by selecting K_d to be a sufficiently large value, such as $K_d = 80$ (indicated by stars on the root locus).

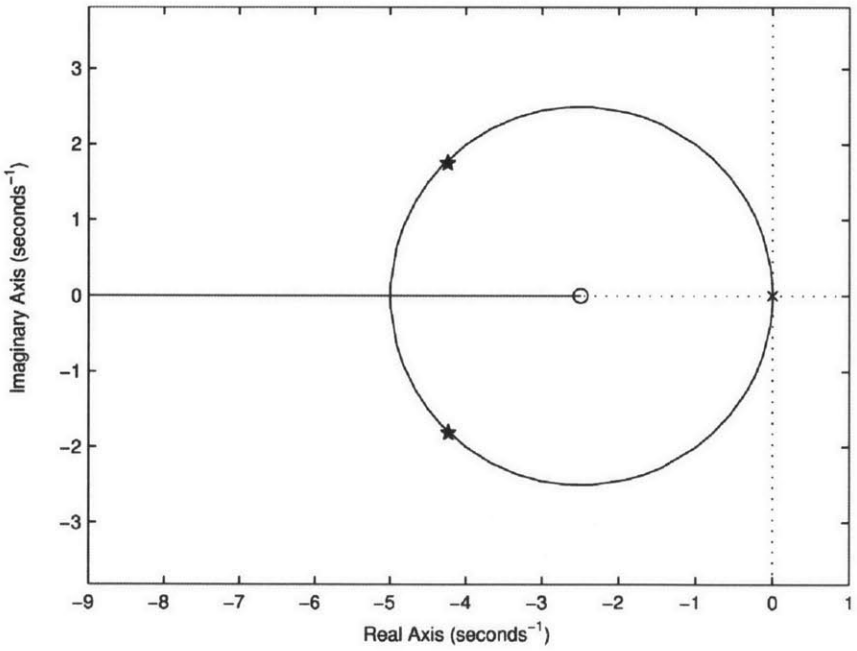

Figure 4-2: Root locus of plant with PD controller $H(s)$

This causal controller is now used as parallel feedforward around the plant

$$H^{-1}(s) = \frac{K_d^{-1}}{s + 2.5} = \frac{.0125}{s + 2.5} \tag{4.95}$$

Furthermore, as opposed to $C(s)$, the gain of $H(s)$ is large, such that its inverse, K_d^{-1}, is small. As a result, the contribution of H^{-1} (s) to the augmented output is expected

to be negligible relative to that of the actual plant output, such that $y_a(t) \cong y_p(t)$ for all practical purposes. The control input force to the augmented plant output transfer function is then

$$G_a(s) = G_p(s) + H^{-1}(s) = \frac{.0125(s^2 + 8.248s + 20.6186)}{s^2(s + 2.5)} \tag{4.96}$$

Note that $G_a(s)$, the augmented plant, is minimum phase and of a relative degree of one, having two zeros and three poles, which makes it ASPR. A root locus showing the high gain stability of the augmented open-loop ASPR plant is provided in Figure 4-3.

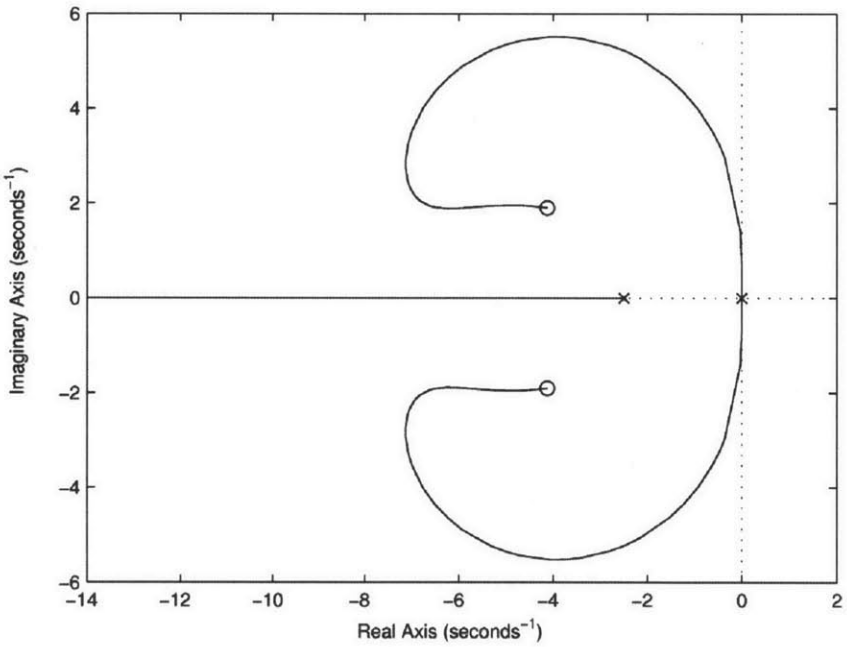

Figure 4-3: Root locus of the augmented plant $G_a(s)$

The plant is required to follow the output of a simple second-order model designed to provide a settling time similar to that of the original proportional-derivative controller $C(s)$, thus having a transfer function representation of the form

$$G_m(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{4.97}$$

where ω_n and ζ denote the natural frequency and damping ratio of the ideal closed-loop system. Selecting these parameters to be 0.18 rad/s and 0.9, respectively, results in

$$G_m(s) = \frac{.0316}{s^2 + .32s + .0316} \tag{4.98}$$

The simple adaptive control law is adopted such that

$$u = K_e(t)e_y + K_x(t)x_m + K_u(t)u_m \tag{4.99}$$

where, in the context of this study

$$e_y = y_m - y \tag{4.100}$$

$$x_m = y_m \tag{4.101}$$

and where u_m represents the desired plant output, as commanded by the user. The control gains are adapted as follows

$$K_e(t) = K_{P_e}(t) + K_{I_e}(t) \tag{4.102}$$

$$K_x(t) = K_{P_x}(t) + K_{I_x}(t) \tag{4.103}$$

$$K_u(t) = K_{P_u}(t) + K_{I_u}(t) \tag{4.104}$$

where

$$K_{P_e}(t) = e_y^2\Gamma_{P_e} \tag{4.105}$$

$$K_{P_x}(t) = e_y x_m \Gamma_{P_x} \tag{4.106}$$

$$K_{P_u}(t) = e_y u_m \Gamma_{P_u} \tag{4.107}$$

and

$$\dot{K}_{I_e}(t) = e_y^2\Gamma_{I_e} - \sigma K_{I_e}(t) \tag{4.108}$$

$$\dot{K}_{I_x}(t) = e_y x_m \Gamma_{I_x} \tag{4.109}$$

$$\dot{K}_{I_u}(t) = e_y u_m \Gamma_{I_u} \tag{4.110}$$

Let the Γ terms represent the tuning gains. Figure 4-4 shows the block diagram of the adaptive control system including the reference model, the plant, the parallel feedforward, and the adaptive controller.

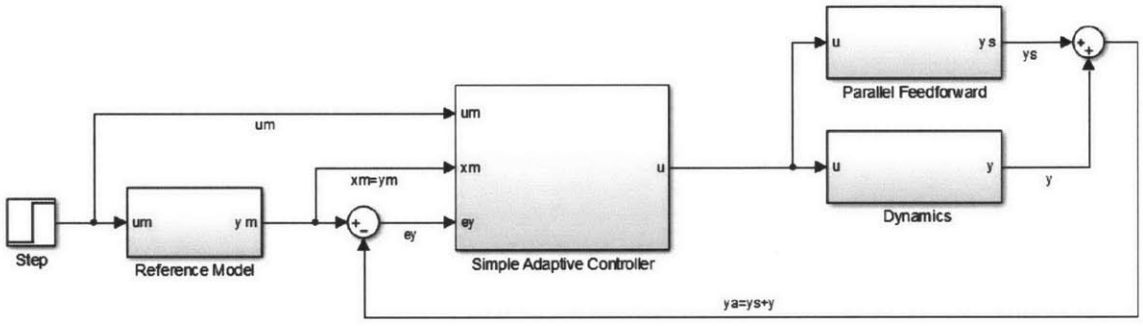

Figure 4-4: Block diagram representation of the adaptive control strategy

4.2.3 Simulated Results

The following graphs illustrate the performance of the adaptive controller and the proportional-derivative controller for nominal and unknown SPHERES mass in the absence of an unknown perturbation. Figure 4-5 shows the performance of the adaptive controller while Figure 4-6 shows the performance of the proportional-derivative controller.

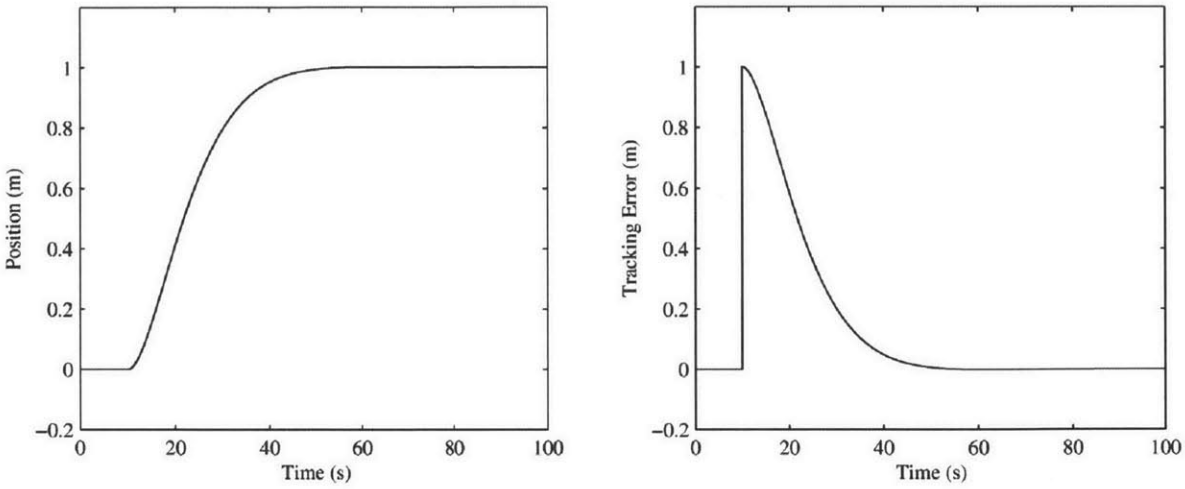

Figure 4-5: Step response of plant with adaptive controller. Solid lines = nominal, dashed = uncertain.

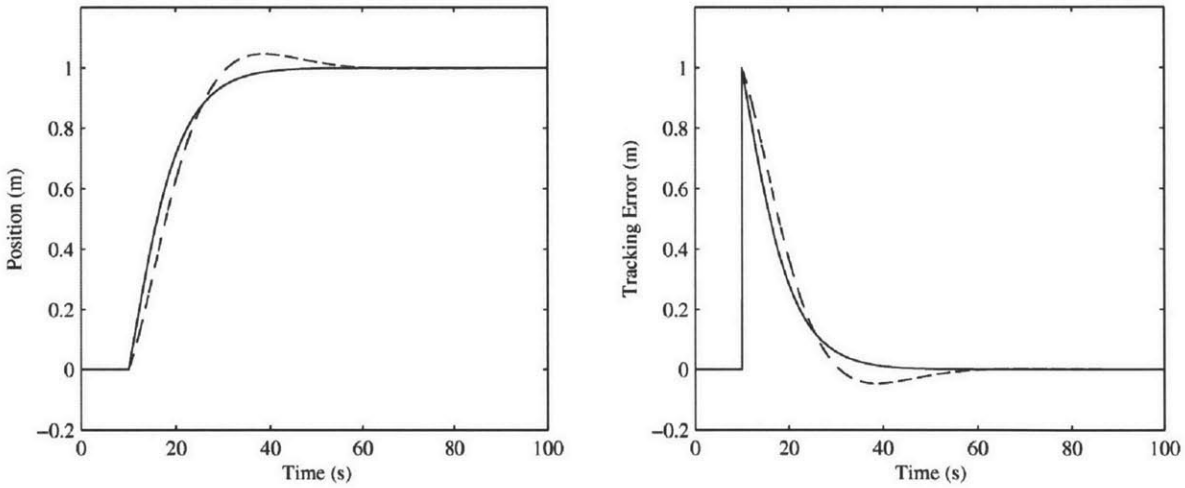

Figure 4-6: Step response of plant with PD controller $C(s)$. Solid lines = nominal, dashed = uncertain.

The adaptive controller clearly outperforms the proportional-derivative controller given an unknown SPHERES mass. The following graphs illustrate the performance of the adaptive controller and the proportional-derivative controller for nominal and unknown SPHERES mass in the presence of an unknown perturbation. Figure 4-7 shows the performance of the adaptive controller while Figure 4-8 shows the performance of the proportional-derivative controller.

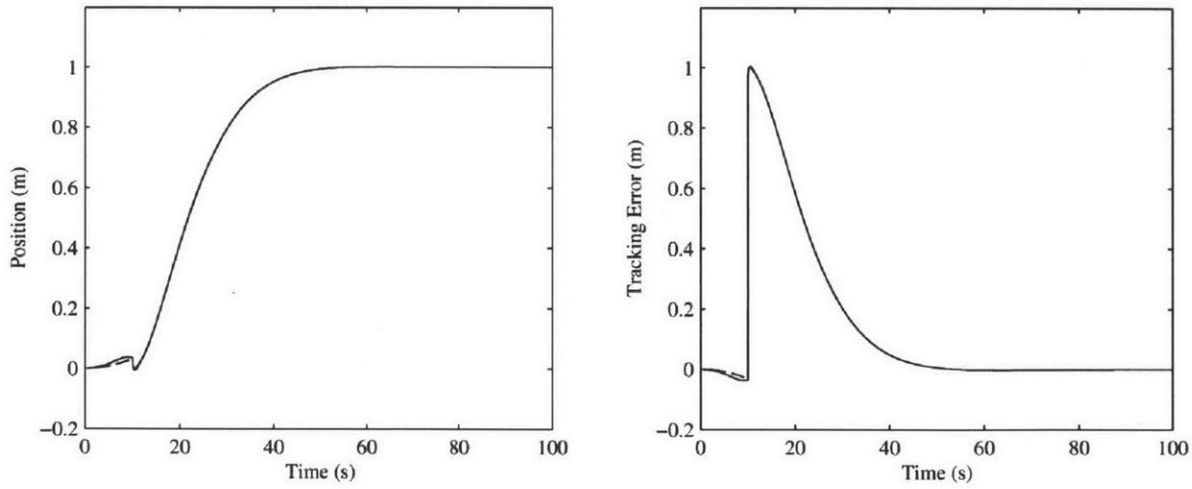

Figure 4-7: Step response of plant under an unkownn perturbation with adaptive controller. Solid lines = nominal, dashed = uncertain.

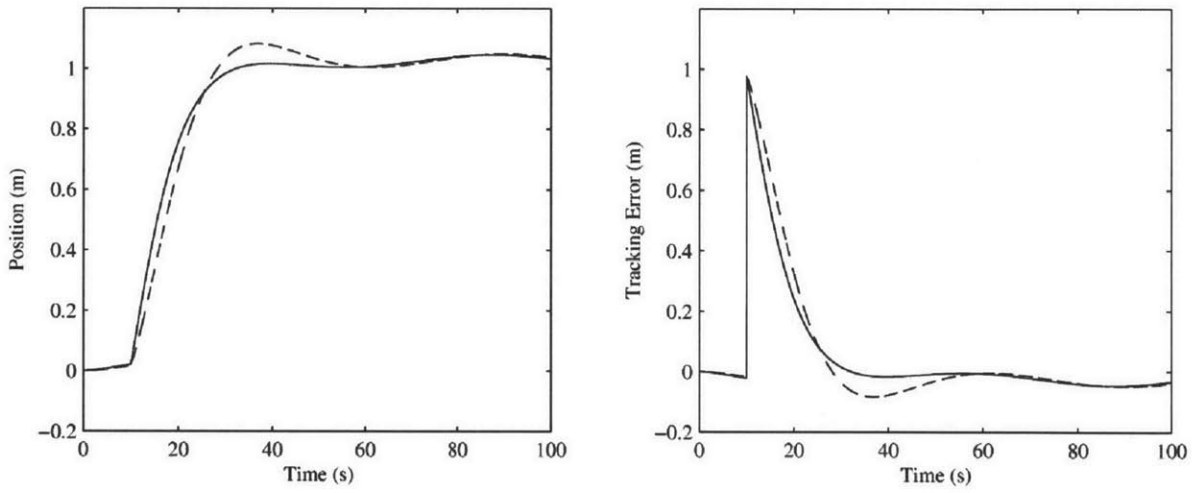

Figure 4-8: Step response of plant under an unknown perturbation with PD controller $C(s)$. Solid lines = nominal, dashed = uncertain.

The adaptive controller clearly outperforms the proportional-derivative controller given an unknown SPHERES mass or the presence of an unknown perturbation.[24]

4.3 Control Theory and Design Summary

This chapter discussed proportional control, PD control, PID control, and simple adaptive control. This chapter proposed a new mixer and state merging method.

Chapter 5

Conclusions and Future Work

Creating this thesis provided the author with the opportunity to learn many valuable lessons about system design, control theory and design, and project management. Furthermore, the work encompassed within this thesis and within the respective overall graduate experience allowed the author to make a meaningful contribution to this important science endeavor. Specifically, the high level contributions of this thesis include:

- A description of the requirements and design features of the SSE as well as an overview of how the SSE specific components were integrated with the existing SPHERES and VERTIGO platforms.

- A demonstration of how a novel systems safety approach can be applied to the SSE.

- An explanation of the system identification method used to characterize the thruster forces of the SSE and determine the values of the SSE inertia tensor as well as an account of the system identification results obtained by a.i. solutions. Additionally, this thesis provides an alternative method by which to conduct system identification.

- An explanation of the method used to conduct open-loop control with the SSE. This thesis shows the test data for one of these open-loop tests and provides

the analysis and implications of those results.

- A theoretical framework for the development of various types of closed-loop controllers that could be used on the SSE. Furthermore, this thesis develops other important aspects of control theory for the SSE such as the mixer and state merger and identifies the need for a new EKF.

These contributions lay the foundation for future work. This future work will probably require a new proposal and contract and may need to be carried out under a different project. The decision on whether or not to use existing hardware for continued research will cause a significant bifurcation of approaches. Here are some recommendations for the future system design of the SSE2 if the respective future proposal encompasses the creation of substantial new hardware:

- Unless the field of resource aggregation has significantly matured, it may be difficult to create an effective system which uses an existing hardware platform. Creating a new elegant system focused specifically on achieving the desired outcome may offer some advantages - certainly, these advantages would need to be weighed against potential negatives such as increased cost.

- Using a coding or scripting language with a higher level of abstraction than C may make the software development process easier for some people.

- Developing a simple and streamlined process that allows the user to go from writing or changing code to executing that code on the hardware in a short amount of time with minimal intermediate steps has the potential to save time. Furthermore, this feature can help to ensure that the development experience is enjoyable, has a low entry cost in terms of learning, and can be iterated with ease.

- Developing hardware which can fully use its estimation process while the system is actuating may give the controls engineer greater flexibility during development.

- Emphasizing simplicity of design can help the engineer to create a machine which is easier to operate and maintain. At the same time, creating a machine that allows for pertinent simplifying assumption to made can be immensely helpful during later parts of the process. For example, creating a machine with on-diagonal inertia tensor values which are much greater than the off-diagonal values can make controller development much easier when designing for a 6-DOF microgravity environment.

- Creating a machine with centralized computing helps the developer to avoid potential problems associated with communications (latency, interference, loss of signal, et cetera) and state merging.

If the new proposal utilizes the same SSE hardware, here are some future work items for this project.

- Implement the SSE mixer and state merger in C code and integrate this code with the existing SPHERES software architecture.

- Create a new EKF which encompasses a full and accurate state model of the SSE.

- Develop a nonlinear attitude controller for the SSE which can control concurrent rotational motion about multiple axes.

- Utilize open and closed-loop control to manipulate the motion and displacement of the fluid inside the fluid tank. Apply this control to useful challenges such as developing a fluid settling maneuver.

In completing this work, the future graduate student will have the opportunity to make meaningful contributions to an important field of study.

Bibliography

[1] Sathya Gangadharan, James Sudermann, Andrea Marlowe, and Charles Njenga. Parameter estimation of spacecraft fuel slosh model. Palm Springs, California, April 2004. 13

[2] Ed Grayzeck. ATS 5 description, August 2013. 13

[3] J.P.B. Vreeburg. Spacecraft maneuvers and slosh control. *IEEE Control Systems*, 25(3):12–16, June 2005. 13, 14

[4] Caley Burke. *Nutation in the Spinning SPHERES Spacecraft and Fluid Slosh.* PhD thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, June 2010. 13

[5] Daniel Kirk. Acquisition of long-duration, low-gravity slosh data utilizing existing ISS equipment (SPHERES) for calibration of CFD models of coupled fluid-vehicle behavior: Phase III, 2012. 14

[6] ucsc. Surface tension, 2014. 15

[7] Franklin Dodge. The new "Dynamic behavior of liquids in moving containers", 2000. 16

[8] European Space Agency. launchers, February 2005. 16

[9] Bruce Hochstetler. Environmental and thermal operating systems, May 2014. 24

[10] Mark Hilstad, John Enright, Arthur Richards, and Swati Mohan. The SPHERES guest scientist program, May 2014. 25

[11] IDS Imaging Development Systems GmbH. UI-5580CP, May 2014. 26

[12] Paul Schallhorn, Jacob Roth, Brandon Marsell, Daniel Kirk, Hector Guitierrez, Alvar Saenz-Otero, Daniel Dorney, and Jeffrey Moder. Acquisition of long-duration, low-gravity slosh data utilizing existing ISS equipment (SPHERES) for calibration of CFD models of coupled fluid-vehicle behavior, June 2012. 32

[13] C. Holicker. SPHERES SLOSH EXPERIMENT ASSEMBLY, June 2013. 32

[14] Gabriel Lapilli. Slosh avionics custom shape Rev12_GABRIEL, May 2013. 34

[15] Nancy Leveson. An STPA primer, August 2013. 39

[16] Satyajit Deb. *FUNDAMENTAL OF PHYSICS 9ed by Halliday Resnick Ralker-Wiley.* Satyajit Deb, December 2013. 52

[17] Walter Lewin, Peter Dourmashkin, Thomas Greytak, Craig Watkins, Andy Neely, Sahana Murthy, David Litster, and Matthew Strafuss. Physics i: Classical mechanics, May 2014. 52

[18] Gilbert Strang. *Computational Science and Engineering.* Wellesley-Cambridge Press, Wellesley, Massachusetts, second edition, 2012. 53

[19] Gabriel Lapilli. System parameters from slosh checkout, February 2014. 55

[20] Gabriel Lapilli. Slosh_Sim.m, February 2014. 57

[21] Brandon Marsell. your data, April 2014. 58

[22] Richard Schulman. inertia_separate, September 2013. 79, 80

[23] Mark Hilstad. *A Multi-Vehicle Testbed and Interface Framework for the Development and Verification of Separated Spacecraft Control Algorithms.* PhD thesis, MIT, Cambridge, Massachusetts, June 2002. 83, 86

[24] S. Ulrich and D. L. Hayhurst. Simple adaptive control for spacecraft proximity operations. 2014. 95