ABSTRACT

Title of Document: AN EXPERIMENTAL EVALUATION OF SPACE MANIPULATOR DYNAMICS

Katherine Elizabeth Strickland
Master of Science, 2012

Directed By: Associate Professor David L. Akin
Department of Aerospace Engineering

The dynamics of a system define the expected loads on the system and are critical for the design and evaluation of a control strategy. This thesis aims to experimentally validate the dynamics of a free-floating or free-flying spacecraft testbed with a manipulator using the theoretical model developed by Papadopoulos. A 2-DOF manipulator was added to an existing air-bearing vehicle to serve as a testbed. Papadopoulos' dynamics model was adapted for the testbed using the simplifications afforded by a planar system. The system's physical parameters were determined, some analytically and some experimentally. Three experiments were performed using different manipulator maneuvers, and the observed accelerations were compared to the model's predicted accelerations. The experimental data followed the general trends of the model prediction, but did not align in magnitude. The observed position motions were then compared to an integrated model using a simulated controller, yielding much better results.

AN EXPERIMENTAL EVALUATION OF SPACE MANIPULATOR DYNAMICS

By

Katherine Elizabeth Strickland

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, College Park, in partial fulfillment
of the requirements for the degree of
Master of Science
2012

Advisory Committee:

Associate Professor David L. Akin, Chair

Associate Professor Robert M. Sanner

Associate Professor Ray Sedwick

Acknowledgements

First and foremost, many thanks to Professor David Akin for providing me the opportunity to work at the SSL, funding to support my studies, and guidance throughout the process of this thesis. I have learned more than I ever expected to from my experience here.

I am indebted to many members of the SSL for their insight and assistance over the last two years. Thanks to Connie Ciarleglio for teaching me about her class's ABV and helping with wiring and software issues throughout the course of this project. Thanks to Nick Limparis for his electronics expertise and assistance in designing the arm's electrical system. Thanks to Chris Carlsen for his assistance in the joint design, for machining necessary parts, and for his Pro-E tutorials. Thanks to Sharon Singer for reviewing this document and to Kate McBryan for general guidance on navigating the perils of thesis-defending. I am most thankful for Nick D'Amore and his willingness to assist in times of crisis and frustration. Throughout this project, he provided dynamics and control expertise, software debugging abilities, and general reassurance that I will in fact graduate eventually.

I am endlessly grateful for my family and their support throughout my graduate school years, especially in the form of free shelter and the frequent home-cooked meal. Lastly, thanks to Adam for his love and enduring patience when the rigors of graduate school made life a little crazy.

Contents

List of Tables

List of Figures

Nomenclature

Acronyms

ABV	air-bearing vehicle
ADC	analog-to-digital converter
ADCS	attitude determination and control subsystem
AR	augmented reality
ARDL	Advanced Robotics Development Laboratory
C&DH	command and data handling
CAD	computer-aided design
CAN	computer area network
CM	center of mass
DC	direct current
DEM	Dynamically Equivalent Manipulator
DH	Denavit-Hartenberg
DOF	degree(s) of freedom
DYMAFLEX	DYnamic MAnipulation FLight EXperiment
EOM	equations of motion
ESA	European Space Agency
HITLab	Human Interface Technology Laboratory
IDE	integrated development environment
IMU	inertial measurement unit
MISO	Master In Slave Out
MOI	moment of inertia
MOSI	Master Out Slave In
NBRF	Neutral Buoyancy Research Facility
PD	proportional-differential
RTOS	real-time operating system
SBC	single board computer
SOF	start-of-frame
SPI	serial peripheral interface
SSL	Space Systems Lab
UDP	user datagram protocol
VM	Virtual manipulator

Greek Symbols

δ_{im}	Kronecker delta (1 if $i = m$, 0 otherwise)
θ	Spacecraft attitude
$\dot{\theta}$	Spacecraft angular velocity
$\ddot{\theta}$	Spacecraft angular acceleration
μ_i	Mass distribution for Link i
$\underline{\rho}_k$	Vector from system CM to k^{th} link's CM

$\underline{\dot{\rho}}_k$	Time derivative of Link k CM relative to system CM
$\underline{\tau}$	joint torques
ω_k	Angular velocity of Link k

Roman Symbols

$\underline{c}_k$	Vector from k^{th} link's CM to its barycenter
$\underline{c}_k^*$	Vector from k^{th} link's barycenter to its CM
$\underline{C}^+$	Vector of non-linear terms representing system
C_1^+	Non-linear terms representing spacecraft
$\underline{C}_2^+$	Vector of non-linear terms representing each link
$\hat{d}_{ij}$	Cross-product inertia term for Links i and j
D	System inertia dyadic about system CM
D_{ij}	Mixed inertia dyadic for Links i and j
D_j	Sum of mixed inertia dyadics for Link j
D_q	Inertia of manipulator joints
D_{qq}	Mixed inertia of manipulator joints
$\underline{f}_k$	Vector of forces applied to Link k
${}^{\mathbf{0}}\mathbf{F_k}$	Projection matrix of Link k rotation into Link 0 frame
$\mathbf{H}^+$	System inertia matrix
$\mathbb{I}_n$	$n \times n$ identity matrix
${}^kI_k$	Moment of inertia of Link k about its CM
J^+	Jacobian matrix of the system
J_S^+	Jacobian matrix of the system about the Link 0 CM
$J_{11,S}$	Jacobian that maps spacecraft angular velocity about the Link 0 CM into Cartesian velocity
J_{11}	Jacobian that maps spacecraft angular velocity into Cartesian velocity
$J_{12,S}$	Jacobian that maps joint velocities about the Link 0 CM into Cartesian velocities
J_{12}	Jacobian that maps joint velocities into Cartesian velocities
$J_{22,S}$	Jacobian that maps the joint velocities about the Link 0 CM into angular velocities
J_{22}	Jacobian that maps the joint velocities into angular velocities
$\underline{l}_k$	Vector from k^{th} link's CM to its "left" joint
$\underline{l}_k^*$	Vector from k^{th} link's barycenter to its "left" joint
m_k	Mass of Link k
M	Total system mass

n_k	Torque applied to Link k
N	Number of degrees of freedom of manipulator
O_k	Origin of k^{th} joint's coordinate frame
q_i	Joint i angle
$\dot{q}_i$	Joint i angular rate
$\ddot{q}_i$	Joint i angular acceleration
$\underline{Q}_q$	Vector of generalized forces applied to Links k, $k = 1, ..., N$
Q_x	Generalized forces applied in the x-direction
Q_y	Generalized forces applied in the y-direction
Q_θ	Generalized torques applied to the spacecraft
$\underline{r}_{cm}$	Position of system CM in inertial frame
$\dot{\underline{r}}_{cm}$	Velocity of system CM in inertial frame
$\underline{r}_k$	Vector from k^{th} link's CM to its "right"joint
$\underline{r}_k^*$	Vector from k^{th} link's barycenter to its "right"joint
$\underline{r}_{k,p}$	vector from k^{th} link CM to point p
$\underline{R}_k$	Vector from inertial frame origin to k^{th} link's CM
$\mathbf{T_i}$	Rotation matrix from frame $i$into inertial frame
$\mathbf{{}^jT_i}$	Rotation matrix from frame $i$into frame j
$\underline{v}_{ik}$	Barycentric vectors
$\underline{v}_{ik,p}$	Barycentric vectors for a point p

1

Introduction

1.1 Motivation

Thorough analysis of a system's dynamics provides an essential foundation for any design. The mathematical representation of how a system moves and reacts to perturbations in its intended environment, in this case space, is captured in a system's equations of motion (EOM). The EOM define the system's dynamics, and enable design of a functional control algorithm. Most systems have natural instabilities and nonlinearities that cause undesirable motions. Understanding the EOM's allows the engineer to keep any beneficial dynamic properties and effectively remove any undesirable dynamic properties to successfully execute the mission.

Validating a space system on Earth is a crucial step in proving its design. Space flight is terribly expensive, making ground testing an essential piece of every flight program. Simulation is a useful tool, especially as simulation fidelity increases to account for errors, perturbations, and unexpected events. However, simulations do not provide validation of the physical design. Developing new technology is an iterative process, and building prototypes and Earth analogues of a space system provide invaluable opportunities to discover flaws in the design in early iterations. Working through the manufacturing process will highlight any hardware limitations and eliminate infeasible designs. Refining communication between components is impossible without hardware, and early determination of communication protocols makes software development progress more smoothly. A physical representation of the space system allows control algorithms to be tested on Earth, lowering the possibility of an ineffective control scheme that would degrade and possibly prevent the success of the mission.

The dynamics of a satellite in space are fairly well defined, however the addition of a robotic manipulator changes these dynamics significantly. Several approaches have been defined and simulated, but few have been experimentally validated. The research in this thesis develops a testbed for the University of Maryland Space Systems Lab's DYnamic MAnipulation FLight EXperiment (DYMAFLEX) vehicle and examines its expected dynamics experimentally. An air-bearing vehicle, referenced as ABV in this thesis, is modified to represent a planar version of DYMAFLEX, and is used to understand a planar approximation to the dynamics of the DYMAFLEX system. The air-bearing vehicle has a two degree of freedom (DOF) robotic manipulator, and additional DOF's can be added as DYMAFLEX control algorithm development progresses. The manipulator can be modified to carry different link masses, allowing variety in the inertia ratios between the links and base. The modular design facilitates sensitivity testing and evolving designs.

1.2 Testbed Overview

There are several ways to simulate the microgravity environment of space before launch. Some of the most common and accessible methods are parabolic flight, neutral buoyancy vehicles, and air-bearing vehicles. Parabolic flights provide the most space-like environment for testing. Cyclic periods of microgravity only seconds long are achieved on the top arc of a parabolic flight path. This is enough weightlessness to do only very simple experiments and data collection. Control algorithms in particular must be tested over a much longer period of time to ensure they remain stable and perform as expected. Parabolic flight is a good option for preliminary evaluation, but more meaningful validation must be sought on a different test platform.

The underwater environment of a neutral buoyancy analogue system provides a three-dimensional space-like testbed that to the vehicle strongly resembles microgravity. This testbed allows more in-depth validation of a system's design as there is no time limit imposed by the environment. However, a neutral buoyancy vehicle requires many features that are not flight-like. All electronics must be in a tightly sealed interior to prevent leaks, and the propulsion system must be adapted for the aqueous environment. Additionally, the environment creates hydrodynamic drag that naturally stabilizes the vehicle, obscuring some of the vehicle's natural instabilities and damping any energy injected by the control algorithm.

The air-bearing approach employed in this thesis provides a two-dimensional emulation of space by allowing a vehicle to move across a planar surface with negligible friction. A slight perturbation causes an air-bearing vehicle to glide across the surface, much as it would in space. The primary challenge of this approach is achieving truly planar motion. Precision-leveled surfaces are best, but very expensive. More common alternatives, such as glass, are susceptible to bowing and other irregularities that could impact air-bearing motion and therefore degrade the quality of the simulation. However, an air-bearing vehicle is typically inexpensive to construct and provides an excellent prototype for proof testing and demonstration. In the case of a cold-gas propulsion system, such as the one described in this thesis, an air-bearing vehicle allows implementation of a flight-like system.

1.3 Dynamics Overview

The two primary methods of deriving a system's EOM are the Newton-Euler approach and the Langrangian approach. The Newton-Euler approach uses a generalized form of Newton's Second Law of Motion ($\underline{\mathbf{F}} = m\underline{\mathbf{a}}$) and conservation of linear and angular momentum to construct EOM for a system. The Lagrangian approach, used in Chapter 3 of this thesis, uses kinetic (T) and potential (V) energy equations of the generalized coordinates (q_j) and their associated external forces (Q_j) to formulate EOM. The kinetic energy equation includes translational and rotational motion. For space applications, potential energy V is often zero as the system is rigid and the gravitational potential energy is zero. In matrix form, the kinetic energy is calculated by:

$$T = \frac{1}{2}M\underline{\dot{\mathbf{r}}}_{\mathbf{cm}} \cdot \underline{\dot{\mathbf{r}}}_{\mathbf{cm}} + \frac{1}{2}\sum_{k=0}^{N}\left\{\underline{\omega}_{\mathbf{k}}{}^{T}\mathbf{I}_{\mathbf{k}}\underline{\omega}_{\mathbf{k}} + m_k\underline{\dot{\rho}}_{\mathbf{k}} \cdot \underline{\dot{\rho}}_{\mathbf{k}}\right\}$$

where M is the total system mass, $\underline{\dot{\mathbf{r}}}_{\mathbf{cm}}$ is the velocity of the system CM in the inertial frame, N is the number of degrees of freedom of the manipulator, $\underline{\omega}_{\mathbf{k}}$ is the angular velocity of Link k, $\underline{\mathbf{I}}_{\mathbf{k}}$ is the inertia dyadic of Link k with respect to its CM, m_k is the mass of Link k, and $\underline{\dot{\rho}}_{\mathbf{k}}$ is the time derivative of the Link k CM relative to the system CM.

The Lagrangian L is defined by:

$$L = T - V$$

And the equations of motion are defined as:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \underline{\dot{\mathbf{q}}}}\right) - \frac{\partial L}{\partial \underline{\mathbf{q}}} = \underline{\mathbf{Q}}$$

where $\underline{\mathbf{q}}$ and $\underline{\mathbf{Q}}$ are vectors of the generalized coordinates and forces, respectively. For a more detailed review of these two approaches, see [3] or [4].

1.4 Previous Work

The aerospace field is always looking to evaluate how well their simulations and testbeds emulate the space environment. This is an iterative process as successful missions allow engineers to improve their models with flight data. Additionally, there has been extensive theoretical work in the field of manipulator dynamics. One crucial variable is the type of base the manipulator is mounted on; fixed (e.g. bolted to a table), free-floating (e.g. moving freely in space), and free-flying (e.g. controlled via thrusters in space) are the three most common.

1.4.1 Testbeds

Parabolic flights are frequently performed by organizations such as NASA, and often open to university research projects. However, in order to maximize weightlessness, the trajectory must be a perfect parabola. Human pilots are not going to achieve this alone, so the Delft University of Technology [5] developed a "flight director"control system to help the pilot perform more precise parabolic maneuvers. A typical parabolic flight in the Delft experimental aircraft provides 12 seconds of zero gravity per maneuver and up to 30 seconds of partial gravity. The pitch angle of the aircraft changes from $+50$ degrees to -50 degrees over the course of each maneuver, exerting an acceleration of up to $3.0g$ on the passengers between weightless periods. These strong cyclic accelerations are very uncomfortable for the passengers on board, earning parabolic flight aircraft the nickname "vomit comets."

On a European Space Agency (ESA) flight, the University of Padua conducted and experiment with a spacecraft-mounted manipulator that achieved $0g$ flight for approximately 20 seconds of each parabola [6]. The vehicle base was mounted on a rack inside the aircraft to restrict its motion, however significant human interaction was required to secure it during the high-acceleration phases of the parabola. While the manipulator experienced zero gravity during this flight, the base experienced some microgravity because it was fixed to the aircraft.

Parabolic flights typically have acceleration errors on the order of $0.05g$ along the vertical axis and $0.01g$ along the horizontal axes [7, 8]. Additionally, since the pilot executes each parabolic maneuver manually the results are very difficult to repeat - each maneuver is slightly different [6]. Often a scaled prototype is required to be able to conduct experiments in the small interior of the aircraft.

NASA built the first neutral buoyancy simulator at Marshall Space Flight Center in Huntsville, Alabama, where the Space Systems Lab's (SSL) Experimental Assembly of Structures in EVA (EASE) was tested in 1986 [9]. Currently, Johnson Space Center's Neutral Buoyancy Laboratory is used for International Space Station training [10]. The SSL's Neutral Buoyancy Research Facility (NBRF) is the only of its kind at a university. Future stages of the DYMAFLEX project will construct a neutral buoyancy prototype for 3-dimensional controls testing in the NBRF.

The advantage to neutral buoyancy testing is that there is essentially no time limit, as batteries can easily be replenished. However, neutral buoyancy testing requires an analogue vehicle that is designed specifically for the underwater environment. Robotic experiments are subject to forces exerted by the water, and these forces must be accounted for in the subsequent data analysis. The thrusters on a neutral buoyancy vehicle can create disturbances in the environment that impact results. The University of Padua [6] investigated several underwater propulsion approaches and determined that an impeller within the neutral buoyancy vehicle shell to cycle and pump water created the least environmental disturbance.

Air bearing vehicles have been used in spacecraft simulation and testing since at least 1960 [1]. Planar systems, such as the ABV used for this thesis, are the most common. A planar system provides 3 DOF, two translational and one rotational, and simplifies the mathematics of the dynamics and control for application. For a historical review of air-bearing vehicles, including various styles and complexities, refer to [1]. The primary advantages of the air-bearing test approach are low cost and simplicity. The planar case in particular allows the researcher to apply new concepts and technology in a simpler 2-dimensional environment for initial experimentation and discovery.

The EFFORTS-I and II simulators [11] at the Tokyo Institute of Technology validated Yoshida's Generalized Jacobian Matrix approach for resolved rate control and target capture. The EFFORTS simulators communicated wirelessly with the test computer serving as ground equipment. The EFFORTS-I base vehicle was 30 cm × 30 cm (1 ft × 1 ft) and only weighed 6.3 kg. Its manipulator consumed 1.4 kg, or 22% of the total vehicle mass [12], and its joints were driven by DC motors with planetary gearboxes.

Soh, Hamzah, and Steyn [13] used an air-bearing vehicle to perform hardware-in-the-loop testing for the attitude determination and control subsystem (ADCS) and flight software for a nanosatellite. Their vehicle had two levels: one for propellant tanks of compressed nitrogen, and one for batteries and sensors. The vehicle was hard-wired to the test computer via a computer area network (CAN) bus, and wires were secured to the ceiling to reduce the impact on the system dynamics.

The University of Padua [6] built two air-bearing vehicles each with a 3 DOF manipulator. One was a free-floating spacecraft base and one included a compressed-air thruster system for a free-flying spacecraft base. Their vehicle was 45.6 cm (18 in) square in the plane and weighed 27 kg. The manipulator consumed 4.5% of the total vehicle mass, or approximately 1.23 kg. Their vehicle experienced microgravity forces on the order of $10^{-4}g$ during air-bearing operations. A camera system mounted above the air-bearing table tracked their vehicle's motion.

Several air-bearing testbeds have achieved higher complexity than the planar case. The tabletop and umbrella configurations in Figure 1.1 have three rotational degrees of freedom each, and are used for spacecraft attitude control testing. Both types have two concentric spheres at the base with air in between them, and typically they allow ±90 degrees in pitch and roll (axes parallel to the table) and 360 degrees in yaw (axis perpendicular to the table) [1]. The dumbbell shape shown in Figure 1.1 provides two unrestricted rotational DOF's in the roll and yaw axes. These systems must be precisely balanced on the bearing in order to avoid introducing torques due to the testbed. York University [14] has a tabletop air bearing system where the vehicle is secured to the platform for testing. This configuration allows attitude control testing within the physical limits of the platform - 360 degrees in yaw and 45 degrees in roll and pitch.

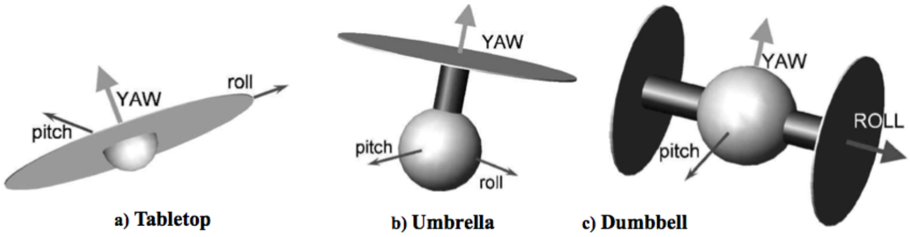


Figure 1.1: The tabletop and umbrella shapes provide 3 rotational DOF's. The dumbbell shape provides 2 rotational DOF's. [1]

The Harbin Institute of Technology [15] combined the planar air-bearing system with the spherical system to create a 5 DOF simulator. The lower level provides two translational DOF's in the plane of the air-bearing surface, and the upper level is mounted on a hemispherical bearing to provide three rotational degrees of freedom for a more comprehensive attitude control simulator. The hemispherical bearing allows 360 degrees of yaw rotation and ±25 degrees of pitch and roll. Compressed air was used as propellant. This vehicle is quite heavy at 105 kg due to the additional complexity.

Air-bearing vehicles with translational DOF's can float on a variety of surfaces. The flatness of a surface can impact results if the instrumentation is precise enough. Surface variations, including scratches, can impede the floating motion of the vehicle as the distance between the bearing and the surface changes. Different surface materials have different levels of variation. Usually the surface is chosen based on cost, as custom-made granite and marble slabs are significantly more expensive than glass tabletops. Glass, which is used in this thesis and [12, 13], has the most variation as it is susceptible to bowing. Marble [15] and granite [6, 16] surfaces are much smoother, with typical flatness variations on the order of μm, and are therefore more reliable air-bearing surfaces. The downside to these two materials is their high cost.

1.4.2 Dynamics

Dubowsky and Vafa [17] developed a virtual manipulator (VM) approach to kinematics analysis, representing an actual space manipulator as a massless kinematic chain with a fixed base at the system center of mass, also called the virtual ground. The first joint between the virtual ground and the first link of the manipulator is modeled as a passive spherical joint. This joint represents the rotational motion of the spacecraft. The virtual manipulator approach has three very useful properties: (1) the axis of the i-th virtual manipulator joint is parallel to the axis of the i-th space manipulator joint; (2) the rotation or displacement of the i-th virtual joint is identical to the rotation or displacement of the i-th physical joint; and (3) the end point of the virtual manipulator can be designed to be coincident with the space manipulator's end-effector. These properties allow a virtual manipulator to be mathematically equivalent to a space manipulator, and therefore the virtual manipulator can be used for kinematics and workspace analysis. The downside of the virtual manipulator approach is that the model is massless, and therefore cannot be physically built for controller testing.

Papadopoulos [2] established a generalized formula for dynamics of spacecraft with a manipulator that is used in Chapter 3 of this thesis. Appendix E of [2], in which the author works through a 1-DOF example and a 2-DOF example, is particularly useful. Vance [18] used Papadopoulos's methodology extensively in the examples in Appendix A of her dissertation. Yoshida [11] develops the dynamics of a free-floating base with n manipulator arms using a Lagrangian approach with a non-holonomic restriction.

Carnegie Mellon's Robotics Institute expanded the VM concept to create the Dynamically Equivalent Manipulator (DEM) for free-floating bases. The kinematics were derived using the virtual manipulator method, and as a result the DEM is kinematically identical to the physical manipulator. The added benefit of the DEM is that it is also dynamically identical to the physical manipulator, and has mass, therefore can be built with a fixed base for controller testing. The base vehicle is modeled as a passive spherical first joint, representing the three rotational DOF's of a space vehicle. The DEM has been shown to be dynamically equivalent to both free-floating [19] and free-flying vehicles controlled with reaction wheels [20]. This model has not been extended to free-flying bases with translational thruster control.

1.5 Organization

Chapter 2 of this thesis describes in detail the construction of the Space Systems Laboratory's air-bearing vehicle (ABV) base vehicle and manipulator, discussing each of the components involved. The ground station and testing environment are also discussed, and coordinate frames are defined. Chapter 3 outlines the physical characteristics of the system, such as center of mass, and how they were determined. It develops equations of motion specific to this system in accordance with Papadopoulos [2]. These equations of motion are what the thesis aims to validate experimentally. Chapter 4 establishes the test methodology and presents the results of the experiment. It then determines correlation with the expected values found in Chapter 3. Lastly, Chapter 5 presents the conclusions of the project as well as a discussion of future work possible with the ABV and the dynamics analysis performed.

2

Testbed

The testbed developed for this thesis has several components. The vehicle itself is a planar air-bearing vehicle representative of the DYMAFLEX design. It communicates with a ground station over a wireless bridge. Ground station commands instruct the vehicle to operate in closed loop or open loop control mode. Lastly, an augmented reality vision system establishes an inertial frame for the testbed and supplies the vehicle with its position and orientation in this frame. The following sections describe each of these components in detail.

2.1 Test Vehicle

The Space Systems Laboratory's air-bearing vehicle (ABV), seen in Figure 2.1, was originally designed and constructed by the Spring 2011 ENAE 484 senior capstone design class as part of the term project. The vehicle has undergone some modifications and upgrades since then, including the addition of a robotic manipulator, but the basic design is still in place. The frame is 13 inches square and 12.25 inches tall. It is made of one inch thick $80/20^{TM}$ aluminum. It has three shelves of 1/4-inch thick nylon that support all the necessary hardware.

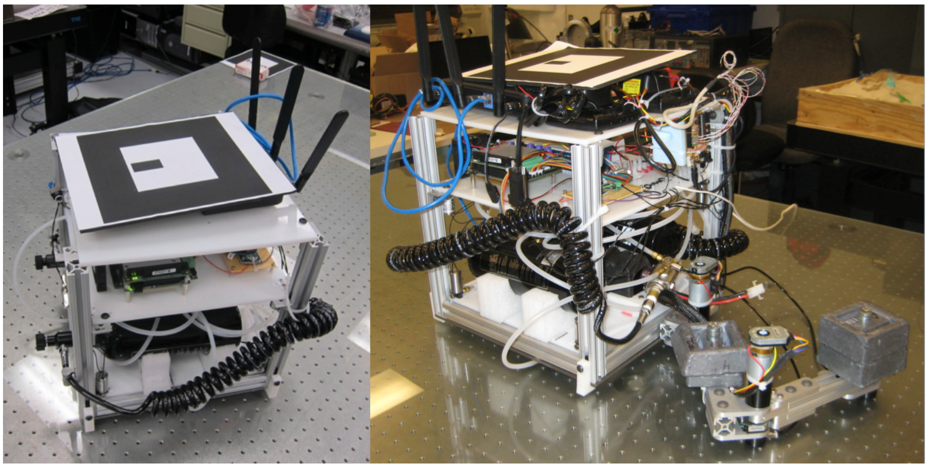


Figure 2.1: On the left is ENAE 484's ABV. On the right, with the robotic manipulator installed.

2.1.1 ABV Base

The vehicle stands on five pucks, which are flat pad air-bearings made by Nelson Air Corporation. Each puck creates a 0.0002 inch cushion [21, 22] underneath it that allows the vehicle to glide nearly frictionlessly across the air-bearing surface. Three pucks support the base vehicle and two smaller pucks support the manipulator. The three base vehicle pucks are 1.5 inches in diameter, with 50 pounds of lift capacity each. They are mounted to a plastic platform underneath the base in a triangular shape. Each puck is held in place by three screws, and the bearing ball is seated in a hole in the plastic to allow for changes in the air-bearing plane. See Figure 2.2. The pucks have a nominal operating pressure of 60 psi, but for this thesis they were supplied 40 psi carbon dioxide.

The ABV's bottom level, shown in Figure 2.3, holds components for the power, thruster, and puck systems. The battery is seated between the tanks and underneath the thruster manifold. It is a five ampere-hour nickel-metal hydride battery that outputs 24 volts. It supplies electricity to the vehicle through a power switch mounted on the aluminum frame. Two 20-ounce carbon dioxide tanks supply propellant to the puck system and the thruster system. The tank supplying the puck system splits

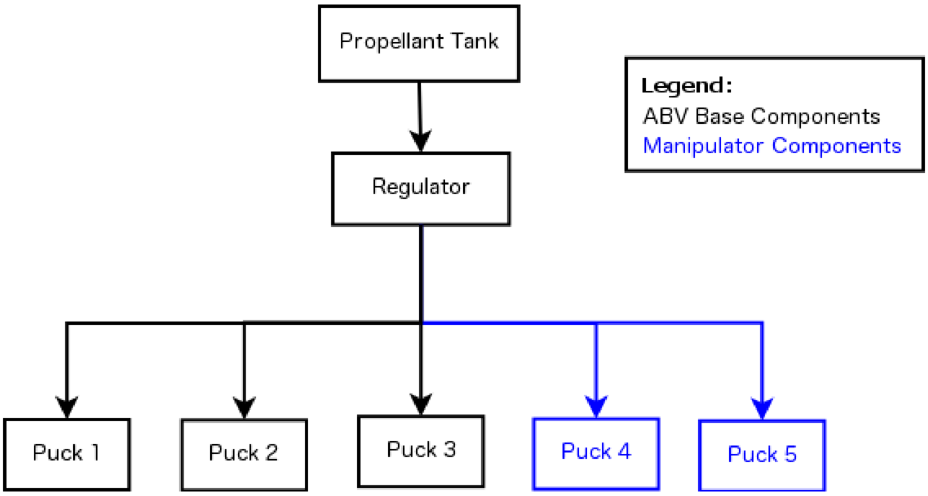


Figure 2.2: Pucks are mounted on the ABV's base in an equilateral triangle for balance in the plane

propellant flow among 5 lines: three 1/4-inch tubing lines run to the pucks under the base vehicle, and two 5/32-inch tubing lines run to the pucks supporting the elbow and endpoint of the manipulator. An adjustable regulator brings the supply pressure down to 60 psi, allowing the supply pressure to be lowered if less lift is required.

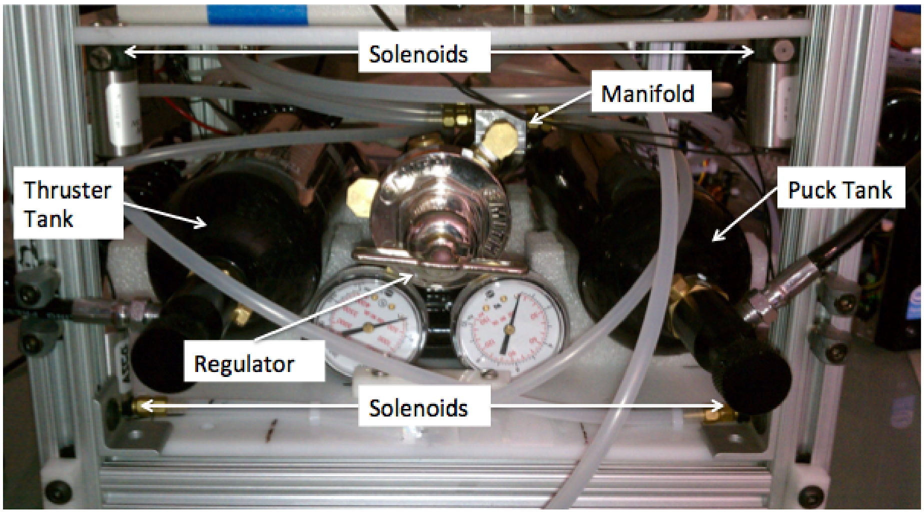


Figure 2.3: ABV Bottom Platform

The second 20-ounce carbon dioxide tank supplies propellant to the thruster system used to control position and orientation in the plane. Propellant is split among 8 lines, one to each solenoid valve. The solenoid valves are mounted to the the bottom platform and underneath the middle platform, two on each side of the vehicle. They serve as “bang-bang”thrusters as they are either open or closed. There is no throttling capability. The solenoid valves are designed for an operational pressure up to 100 psi, however to maintain the integrity of the seals in the thruster system the regulator is typically set to output propellant at 80 psi. The propellant is distributed within an aluminum manifold custom built in the SSL machine shop for more uniform fluid flow and distribution. This is an upgrade, particularly in reliability, from the ENAE 484 system that distributed propellant through a series of tubing joints. O-rings were also added to all joints in the thruster system to minimize leakage. A second adjustable regulator, labeled in Figure 2.3, controls the thruster supply pressure separately from the puck pressure.

The middle platform, shown in Figure 2.4, holds a relay for the solenoid valves, power components, and computing equipment. The KTA-223 USB/RS485 Arduino-based relay controller actuates the solenoid valves when commanded to provide thrust in pulses approximately 10 ms long. The power switch connects directly to the power and ground terminals on this platform. Additionally, there are two power regulators that supply 5V power to the bridge and 12V power to the relays and the IMU.

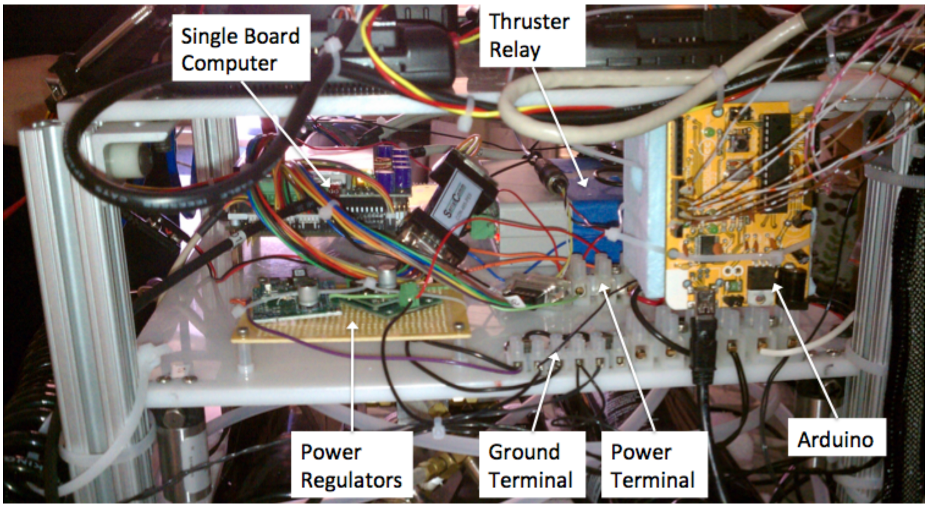


Figure 2.4: ABV Middle Platform

An AMPRO Core Module 745 is the brain of the ABV. It is an Atom$^{\mathrm{TM}}$PC/104 single board computer with a 1.3 GHz clock rate. Figure 2.5 shows the command

and data handling (C&DH) architecture, adapted from [23] to represent the current system. Components added for this thesis are shown in red. The SBC runs on the QNX Neutrino real-time operating system (RTOS). It has three serial ports: one communicating with the relay, one communicating with the inertial measurement unit (IMU) over RS-422, and one communicating with the Joint 1 motion controller for the manipulator over RS-232. A 1GB Ethernet connection links the SBC and the router for communication with the ground station.

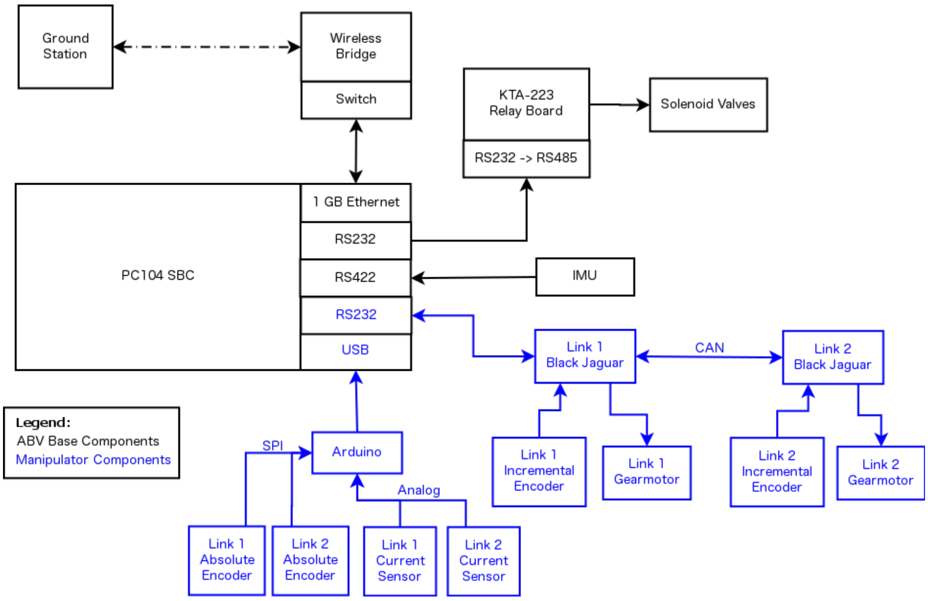


Figure 2.5: ABV C&DH Architecture (adapted from [23])

One USB 2.0 port receives absolute encoder feedback from a Freeduino v1.22 microcontroller. The SBC does not send any commands or data to the Freeduino. This Freeduino version is analogous to the Arduino Duemilanove. The microcontroller hangs from the top platform and is supported by rigid foam backing. It communicates with the absolute encoders via a serial peripheral interface (SPI) bus and receives analog data from the current sensors. The two encoders are connected to the same SPI bus (digital pins on the Freeduino) and the board uses chip select pins to differentiate communications. Additionally, there is a serial clock pin, a Master In Slave Out (MISO) pin, and a Master Out Slave In (MOSI) pin. The encoders are also connected to +5V power and ground pins on the Freeduino. Absolute encoder position is sent over the SPI bus 100 times per second and relayed to the SBC such that each control loop has fresh data. The current sensors output analog readings to the Freeduino's analog-to-digital converter (ADC) at a rate of 80 kHz. The Freeduino reads from the ADC (analog pins on the Freeduino) 100 times per second and relays that data to the SBC such that each control loop has fresh data.

The top platform, shown in Figure 2.6, holds the IMU, communication hardware, and manipulator components. The Memsense Nano-IMU houses an accelerometer, a magnetometer, and a gyroscope that together provide acceleration, orientation, and angular rate data for three axes. As the ABV is a planar system, Z-axis data from the accelerometer and magnetometer are irrelevant, and X- and Y-axis data from the gyroscope are irrelevant. A comprehensive explanation of the ABV coordinate frames is provided in Section 2.1.3. This platform also holds a 2.4 GHz router that communicates with the ground station as a bridge. It supplies data to the SBC over Ethernet.

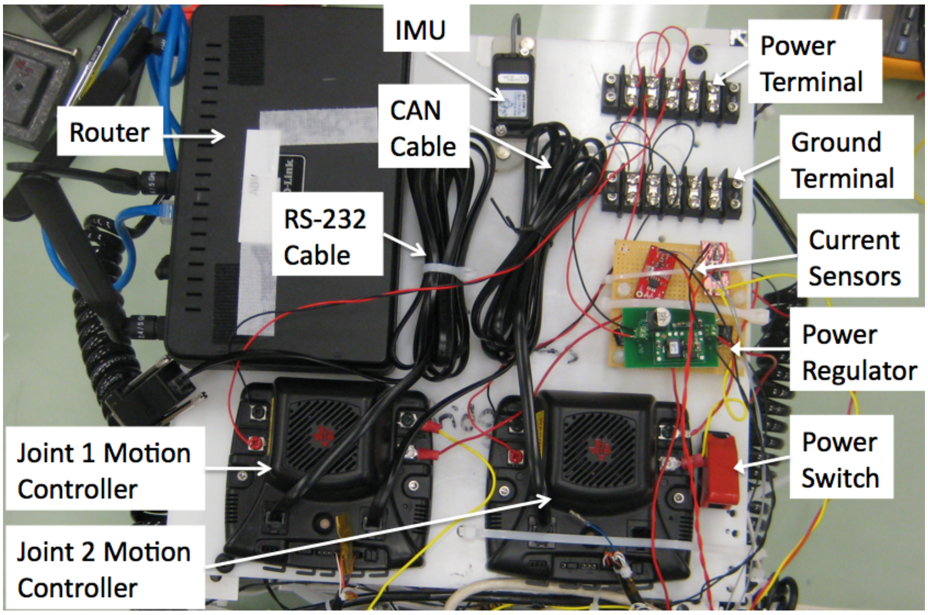


Figure 2.6: ABV Top Platform

There are several manipulator components on this platform. A power switch controls the power supply to all manipulator components. A power regulator brings the voltage down to 12V for the motion controllers and manipulator motors. Power is distributed to the motion controllers via power and ground terminals. The motion controllers are Texas Instruments Black Jaguar control modules. The Joint 1 controller is connected to the SBC via RS-232, and passes any Joint 2 commands over a computer area network (CAN) bus to the Joint 2 controller. Likewise, any Joint 2 feedback data is sent through the Joint 1 controller to the SBC. They supply power to the motors on their respective joints. Each Jaguar is connected to an incremental encoder using a 5-pin connector, with pins for +5V power, ground, A channel data, B channel data, and index output. The index output pin is not used. The Jaguars can operate in position, speed, current, or voltage control modes. The testing in this thesis uses position control mode, however changing control modes requires only a small update to the ABV software.

There are two current sensors on the wires between the Black Jaguar M+ terminals and their respective gearmotors. These Allegro Microsystems, Inc. ACS712 sensors are in an integrated circuit (IC) for ease of use. Positioned over top of all these instruments is the target figure for the visual tracking system, ARToolkit (see Section 2.2.2).

2.1.2 ABV Manipulator

The manipulator pucks are flat-pad air bearings 1 inch in diameter with 20 pounds of lift capacity each, to accommodate the smaller mass of the manipulator. These smaller pucks have no mounting holes on their top surface, so the aluminum plates on the bottom of Joint 2 and the endpoint have a seating for the bearing ball that keep the pucks in place. The manipulator's two links are manufactured out of one inch thick 80/20^{TM} aluminum, held together and to the base vehicle with brackets custom-made in the SSL machine shop of 6061-T6 aluminum alloy. These brackets also accommodate the bearing balls for the pucks as discussed in Section 2.1.1.

Each joint, shown in Figure 2.7, has a 12V brushed direct current (DC) motor from Midwest Motion Products with a 51:1 planetary gearbox on its output shaft. Their maximum current draw is 0.9A. On the input side of the motor shaft is the incremental encoder that supplies motor position data in revolutions to the Black Jaguar motion controllers. At the output end is the absolute encoder that supplies link position data to the Freeduino. As the incremental encoder data must be extrapolated across the gearmotor to get joint position, the absolute encoder provides more precision and accuracy. Not only does the absolute encoder give position of the gearmotor output shaft, and therefore the adjoining link, but it also has more counts per revolution (4096) than the incremental encoder (2048).

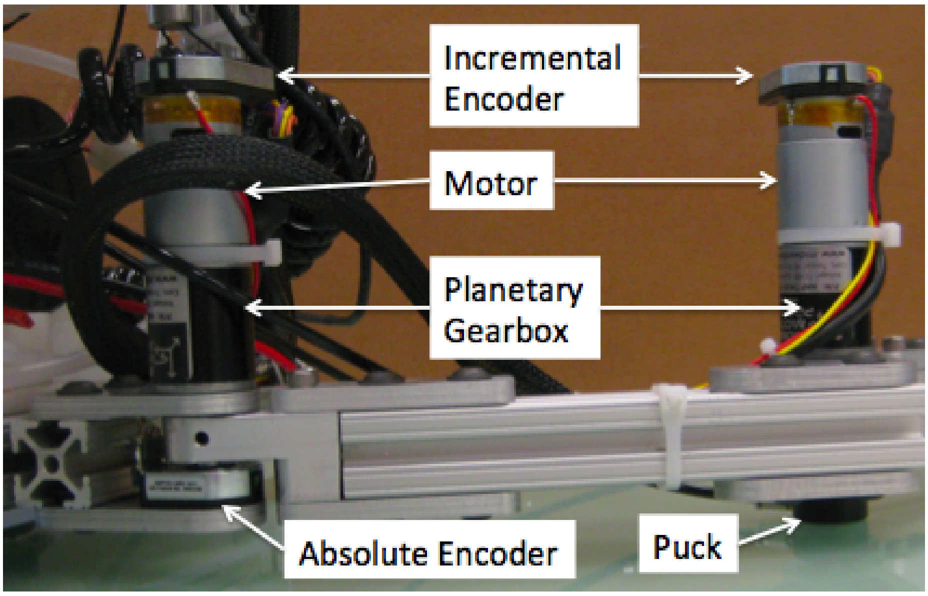


Figure 2.7: ABV Manipulator joint structure

80/20^{TM} aluminum makes it easy to mount weights to the manipulator if more mass or a different mass distribution is desired. Some additional weight is required to achieve the required performance from the manipulator pucks. The weights employed are lead weights with holes drilled through the middle so they can be secured on the link with a simple bolt and fastener.

The Denavit-Hartenberg (DH) parameters [3] for the manipulator were determined from its Pro-Engineer CAD model. They are summarized in Table 2.1. θ_1 and θ_2 are variables as both joints are revolute.

Table 2.1: ABV Denavit-Hartenberg Parameters

Link	$\mathbf{a_{i-1}}$ (m)	$\alpha_{\mathbf{i-1}}$ (deg)	$\mathbf{d_i}$ (m)	$\theta_{\mathbf{i}}$ (deg)
1	0.193	0	0.041	θ_1
2	0.210	0	0	θ_2

2.1.3 Coordinate Frames

All coordinate frames defined for this system are right-handed frames. They are shown in Figure 2.8. Pucks are numbered 1-5, with Puck 1 located directly underneath the IMU. All X-axes are marked in blue, Y-axes in green, and Z-axes in red. As this is a planar system, all Z-axes will be going either into or out of the page. Out of the page is indicated by a circle around the origin ○, and into the page by a circle with an x through it around the origin ⊗. The origin of the Link 0 frame is at the geometric center of the base. The origin of each link frame is on the axis of rotation, the motor shaft. The IMU frame is defined by the manufacturer. The X-axes of the IMU, Link 0, and Link 1 frames are coincident when the manipulator is in its home position, shown in the Figure. The system's inertial frame is called the Table frame, and is defined by ARToolkit as discussed in Section 2.2.2.

2.1.4 ABV Software

The ENAE 484 class wrote the basic functions of the ABV software, including communications with the ground station and IMU, thruster actuation, and a preliminary control strategy. All software related to the manipulator was developed as part of this thesis. See Figure 2.9 for a diagram of the ABV software architecture. All software

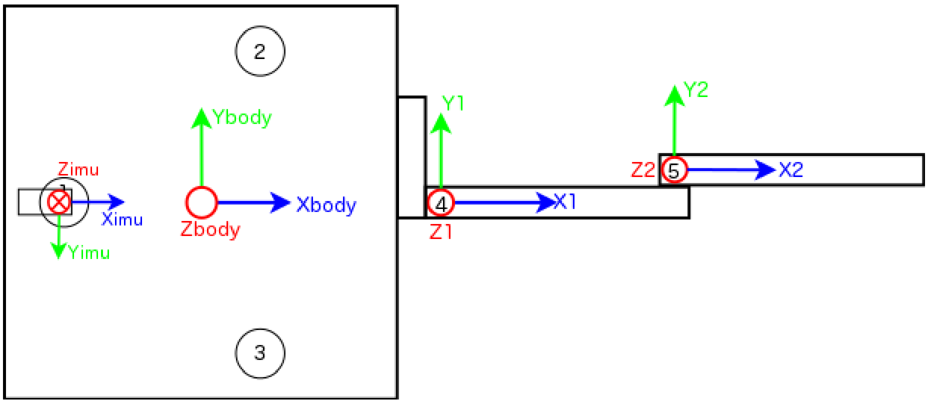


Figure 2.8: ABV Coordinate Frames

on the ABV is written in C++, with the exception of the Freeduino software which was developed using the Arduino integrated development environment (IDE).

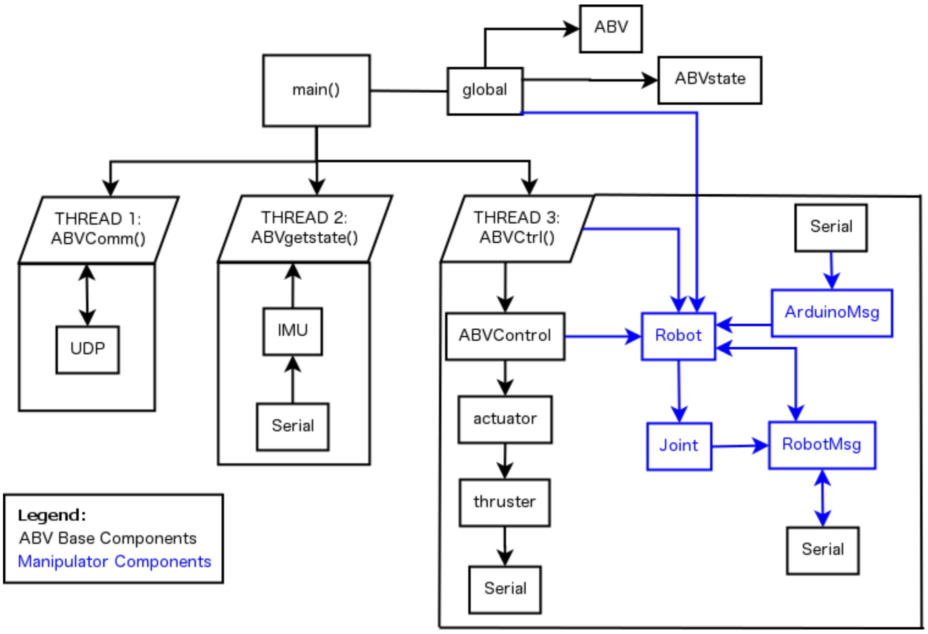


Figure 2.9: ABV Software Architecture

The main program initializes three essential global objects: the `ABV` object representing the vehicle, the `ABVstate` object representing the base vehicle state, and the `Robot` object representing the manipulator. It then spawns three threads: one handles communications with the ground station, one continuously collects feedback data from the IMU, and the third executes the controller and performs all manipulator operations. The `ABVComm()` thread reads user datagram protocol (UDP) messages that arrive over Ethernet and stores them in a buffer. The `ABVgetstate` thread reads from the IMU serial port and stores those bytes in a separate buffer. The `ABVCtrl` thread has access to both of these buffers, which gives it knowledge of the base position and orientation as well as all the IMU data. It locks the mutex while reading from each buffer to prevent reading and writing to the same memory address at once. If the vehicle is in open loop mode, the controller passes the commands from the joystick on to the actuator object which converts them into thruster commands. If the vehicle is in closed loop mode, the controller implements a proportional-differential (PD) controller for position and attitude. The "bang-bang"thrusters apply a mostly constant thrust per pulse, so the controller only issues a thruster command if the required force is above a threshold. Additionally, the thruster selection logic checks if two directly opposing thrusters are both set to "on"in the outgoing command, and if so turns both off. This conserves propellant by preventing counteracting thrusts. For this thesis, closed-loop control was activated in order to receive position and orientation feedback from ARToolkit, however thruster commanding was turned off to maintain a free-floating configuration.

The addition of the manipulator required several new functionalities to be added to the software. The `Robot` object is the primary interface between the controller and the manipulator. It contains a vector of `Joint` objects for a modular approach. It initializes each motion controller with configuration data and control mode settings through the `Joint` object, which uses `RobotMsg` to communicate with the Black Jaguar motion controllers. Once all the initialization commands have been sent, it asks for feedback to verify that the configuration was successful. The `Robot` object passes on position commands to each `Joint` as well. The `ArduinoMsg` object handles all data reporting from the Freeduino.

The Jaguars require a heartbeat message every 100 ms to maintain operations and valid communications. If the heartbeat is not received, the Jaguar will stop supplying current to the motors and return to a neutral state. The control loop tracks the time of the last heartbeat message, and resends the heartbeat if that interval is greater than 60 ms. The controller also polls each Jaguar for fault status and applied current data at each loop.

The `RobotMsg` object handles communications to and from the Jaguars. They require messages in the CAN interface format, with four bytes identifying the target controller and the type of message. Different messages accept and return different types of data, either integer or fixed point in 8-bit to 32-bit sizes. Some messages require two data values. Additionally, an `0xff` data or message byte is represented by two bytes as that is the start-of-frame (SOF) indicator in the Jaguar protocol. The message size byte is not incremented to accommodate the extra byte, so `RobotMsg` checks for it before parsing the message. Specific information pertaining to the Jaguar communications protocol can be found in [24], which is provided in the Black Jaguar Rapid Development Kit CD package that is available for free download from Texas Instruments. The `RobotMsg` object parses outgoing and incoming messages - outgoing messages are sent over serial, and incoming messages are stored in a “received”vector for processing by the `Robot` object.

QNX has a serial-to-USB conversion driver that allows the SBC to treat the USB port as a serial port. Thus, the `ArduinoMsg` object can use the `Serial` object for communications implemented by the ENAE 484 class. The `ArduinoMsg` object uses a simplified version of the Jaguar protocol to relay absolute encoder data to the SBC. The first byte of a message is the SOF byte `0xff`, followed by the message size, the joint identifier, and the message type. Encoder messages have type `0x0a` and current sensor messages have tupe `0x0b`. The last 2-3 bytes of the message are data bytes. The absolute encoders return 12 bits of valid data representing their position, and the ADC returns 10 bits of valid data representing the applied current. The `0xff` data byte handling is the same as in the Jaguar protocol: `0xff` data is represented by 0xfe 0xfe, and 0xfe data is represented by 0xfe 0xfd. However, the message size byte reflects the additional bytes to simplify parsing. See Table 2.2. The data byte in parentheses is only used in the case of an 0xff or 0xfe data byte in the least-significant byte, as the most significant byte has a maximum possible value of 0x0f. There is no send functionality in the ArduinoMsg object.

Table 2.2: Freeduino Message Protocol

Byte 0	**Byte 1**	**Byte 2**	**Byte 3**	**Byte 4**	**Byte 5**	**Byte 6**
SOF	Size	Joint	Type	Data	Data	(Data)

The ABVControl object logs all necessary data at the end of each control loop. Data measurement, recording, and analysis are discussed in Chapter 4. The ABV software is executed by logging into the SBC via Telnet. It runs until the user kills it using Ctrl-C.

2.2 Ground Station

The ENAE 484 class designed and implemented the ground station. The ground station sends control mode, inertial position, and inertial orientation information to the ABV via a wireless bridge. It communicates with the ARToolkit program for the inertial frame data. The SSL integrated the ARToolkit system, which calculates the inertial frame data and provides a video feed with augmented reality markers for the ground station user to see in real time.

2.2.1 Communications

The ground station consists of a desktop computer, a router, a joystick, and a camera. The computer runs on the Ubuntu 8.04 (Hardy Heron) OS, connecting to the joystick and the camera via USB 2.0 and to the router via Ethernet. The desktop runs not only the ground station software, but also the ARToolkit software (see section 2.2.2).

The desktop acts as a server with the ARToolkit program as client in order to pass position and orientation data to the ABV.

The ground station software spawns two threads - one for joystick and one for communications with the vehicle. A joystick button is used to control the ABV when in open loop mode or to turn on closed loop mode. The communications thread sends UDP messages to the vehicle over the bridge. These messages are either position and orientation data from ARToolkit, a closed-loop switch signal, or a directional command from the joystick. The ground station software runs until the user kills it using Ctrl-C.

2.2.2 ARToolkit

A camera mounted on the ceiling over the air-bearing surface provides video to the ARToolkit augmented reality (AR) vision system. ARToolkit was developed at the Human Interface Technology Laboratory (HITLab) at the University of Washington, originally intended to enhance videoconferencing [25]. It uses targets to determine a workspace and track objects within the workspace. There are two targets affixed to the optical bench on diagonally opposite corners, and one target on the top shelf of the vehicle (see Figure 2.10). ARToolkit provides position and attitude data to the vehicle relative to a coordinate system centered on the workspace. The origin of this coordinate system is roughly the center of the optical bench. The ARToolkit software runs until the user kills it using Ctrl-C.

2.3 Air-Bearing Surface

The Advanced Robotics Development Laboratory (ARDL), an SSL facility, has an 8 ft × 4 ft optical bench that has independent adjustment of each leg. It is adjusted with a precision level before testing. There is an 8 ft × 4 ft piece of tempered glass

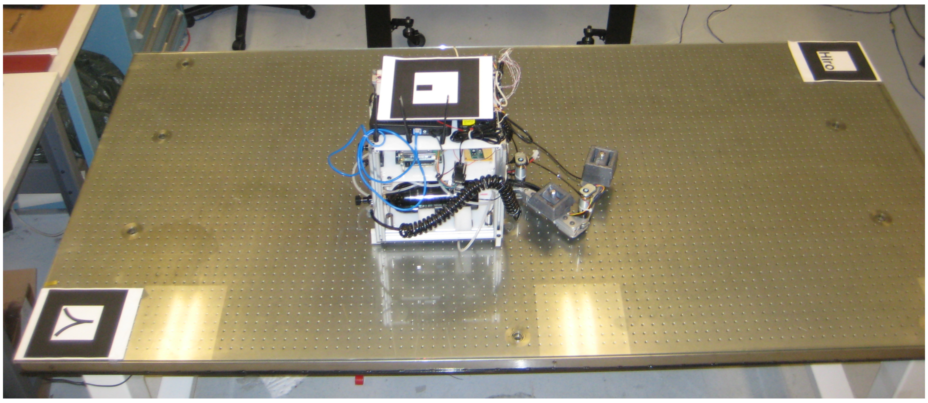


Figure 2.10: ARToolkit Targets

on top of this bench that serves as the air-bearing surface. There is some bowing in the glass that has been lessened by the addition of shims between the glass and the bench.

3

Dynamics Model

The dynamics derived in this chapter follow Papadopoulos's dynamics model developed in [2] and are adapted for the ABV. The ABV is a 3-DOF planar free-flying vehicle with a planar revolute 2-DOF manipulator. Example EOM derivations for planar systems are provided in Papadopoulos's Appendix E and Vance's Appendix A. The model developed here follows these examples, with a few noted exceptions, and uses the physical characteristics of the ABV to obtain numeric values for the required physical parameters. These equations will be evaluated against experimental data in Chapter 4.

The nomenclature and notation used in this thesis follow standard robotics terminology. A system is described in links and joints. Link 0 is the base vehicle of the ABV. Link 1 is the manipulator link closest to the base, and Joint 1 connects Links 0 and 1. Similarly, Link 2 is the manipulator link farthest from the base, and Joint 2 connects Links 1 and 2. Joints 1 and 2 are revolute joints with parallel joint axes.

Matrices are printed in bold and vectors are underlined. An underlined scalar is a column vector with all entries equaling that scalar (e.g. $\underline{1} = [1 \quad 1]^T$). Vectors with a left-superscript i are in the i-th link frame. Vectors without a left superscript are in the inertial Table frame. For example, ${}^0\underline{a}_2$ is a Link 2 vector expressed in the Link

0 frame, whereas $\underline{a}_2$ is the same vector expressed in the Table frame. The System frame has the same orientation as the Table frame, but its origin is located at the system center of mass (CM).

The dynamics model outlined in Papadopoulos's dissertation requires knowledge of these physical ABV parameters, shown for the general case in Figure 3.1:

- m_k: The mass of the k^{th} link
- I_k: The rotational inertia of the k^{th} link about its CM
- O_k: The origin of the k^{th} joint's coordinate frame, located at the point of rotation
- $\underline{r}_{cm}$: The position of the system CM in the inertial frame
- $\underline{\rho}_k$: The k^{th} link's CM with respect to the system center of mass
- $\underline{l}_k$: The vector from the k^{th} link's CM to its preceding ("left") joint, toward the base vehicle
- $\underline{r}_k$: The vector from the k^{th} link's CM to its following ("right") joint, away from the base vehicle
- $\underline{c}_k$: The vector from the k^{th} link's CM to its barycenter, defined below

The derivation in this section represents each link as an augmented body with mass equal to the total system mass [26]. The mass of all links to either side of the link in question is represented as a point mass located at the adjoining joint. For example, the augmented body representation of Link 1 has the mass of Link 0 at the Joint 1 axis and the mass of Link 2 at the Joint 2 axis. The barycenter of a link is the center of mass of its augmented body representation (see Figure 3.2). The kinematics for the ABV are defined in terms of each link's barycenter, calculated in Section 3.1.2.1.

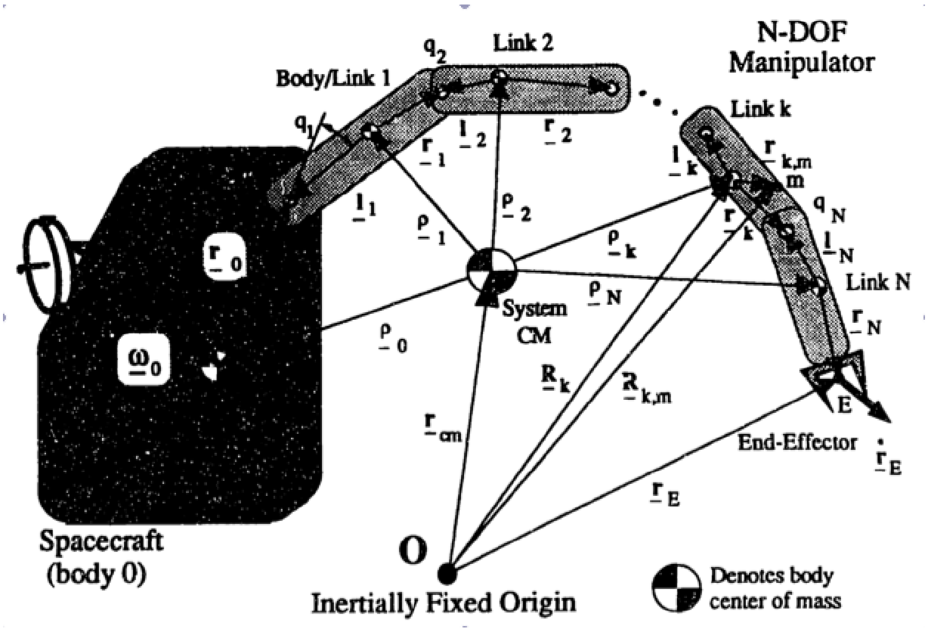
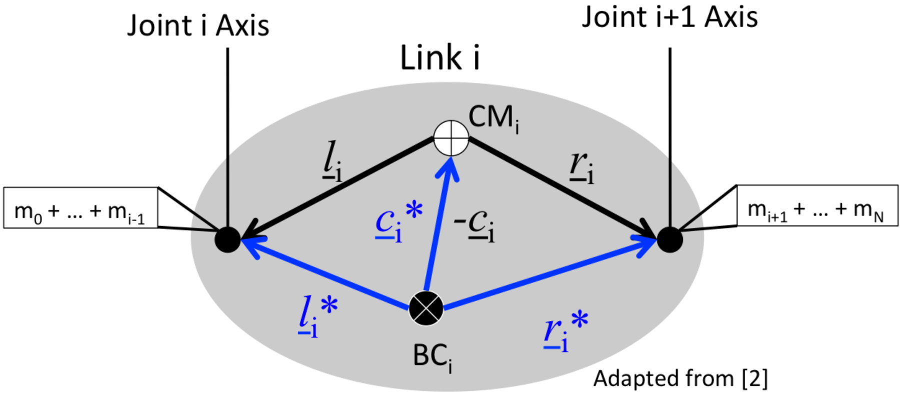


Figure 3.1: Visual representation of characteristics for an N-DOF system [2]



Adapted from [2]

Figure 3.2: Link i Barycenter and Associated Vectors [2]

3.1 Physical Parameters

3.1.1 Centers of Mass

The Link 0 center of mass was measured using a precision scale (Figure 3.3). The manipulator was disconnected from the base, and the weight of the base alone was measured to be 32.245 lb. Next, two of the three base pucks were supported next to the scale such that only Puck 1's load was measured. The scale was adjusted to be level with the support for the other two pucks, however this required some trial and error because the scale's surface depresses when weight is borne. This process was repeated for each base puck. The sum of the loads carried by each puck should equal the total base weight measured, and the sum of the final measurements was within 0.8% of the total measured weight. Results are shown in Table 3.1, where all position measurements are in the Link 0 frame. Since only the planar location of the CM is required for the ABV's EOM, the z-coordinate is assumed to be 0. Recall the Link 0 frame has its origin at the geometric center of the ABV Base and its x-axis pointing toward Joint 1. Note that Puck 2 is misplaced by 0.125 in in the Y_0 direction from its intended position.

Table 3.1: Link 0 Center of Mass Measurements

Puck	**X_0 (m)**	**Y_0 (m)**	**Measured Weight (kg)**
1	-0.121	0.000	3.76
2	0.0508	0.124	5.36
3	0.0508	-0.121	5.39
		Total	14.51

Though the manipulator was disconnected from the base, there were still two brackets, the Joint 1 motor shaft housing, and four associated bolts connected to the base for these measurements. These components move with Link 1, therefore their CM's were removed from the above data to achieve the true Link 0 CM and m_0 shown in Table 3.2.

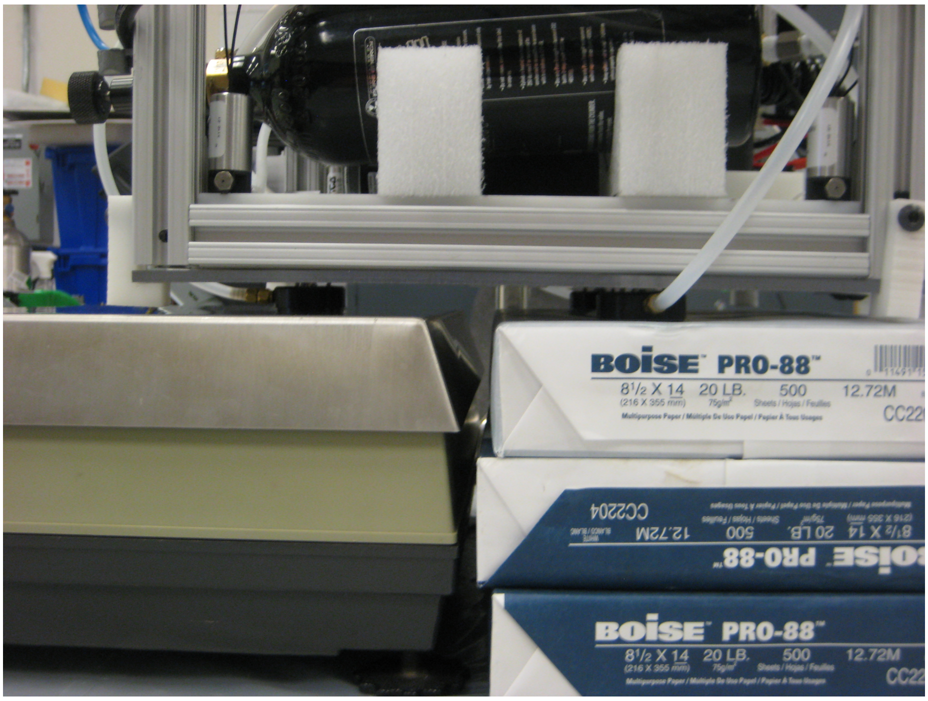


Figure 3.3: Base CM Measurement

The CM's of the manipulator links were calculated using the Pro-Engineer CAD model created before the joints were manufactured. For this thesis, the CAD model was updated to represent the implemented design prior to performing these calculations. The CM information for Links 1 and 2 are given in their respective link frames in Table 3.2. Again, the z-coordinates are assumed to be 0 for the planar system.

Table 3.2: Link k Center of Mass

Link k	$\mathbf{X_k}$ (m)	$\mathbf{Y_k}$ (m)	Total Weight (kg)
0	0.0235	0.0256	14.39
1	0.128	0.0047	4.189
2	0.153	0	3.518

The system CM was calculated in the Link 0 frame for the home position, with Joint 1 and 2 angles equal to 0 (perpendicular to the base frame). Then, all link CM's were transformed into the System frame, resulting in the vectors ${}^{sys}\underline{\rho}_k$. A visual representation is given in Figure 3.4.

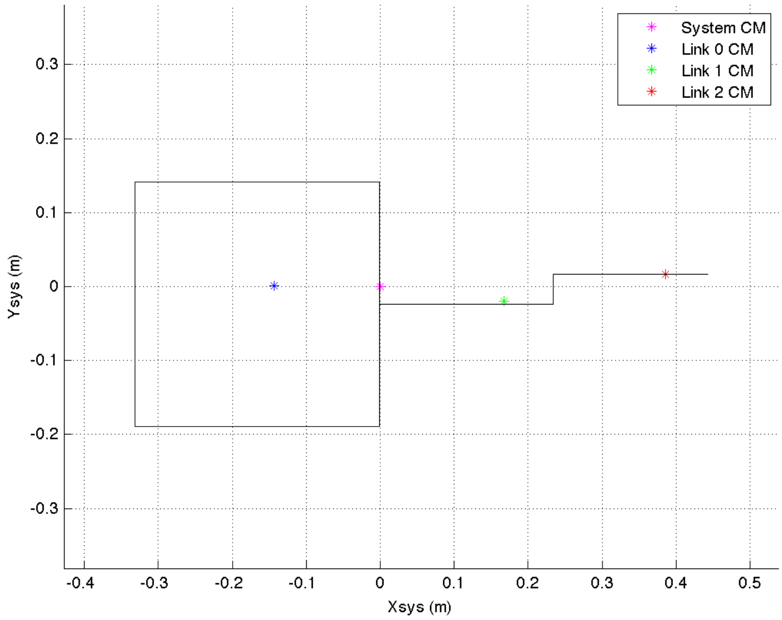


Figure 3.4: Link CM's with respect to the System CM

3.1.1.1 Calculating Joint Locations Relative to Link CM's

The $\underline{l}_k$ and $\underline{r}_k$ vectors are body-fixed vectors defining the location of the link joints in relation to the link CM's (see Figure 3.2). As the link frame origins are located at their respective joint axes, it is easiest to calculate these vectors in their respective link frames where they are constant. The equations defining these vectors are:

$$^{k}\underline{l}_k = {^{k}O_k} - {^{k}\underline{\rho}_k} \qquad\qquad ^{k}\underline{r}_k = {^{k}O_{k+1}} - {^{k}\underline{\rho}_k}$$

Note that $\underline{l}_0$ does not exist, as there is no joint preceding Link 0, and $\underline{r}_2$ considers the end of Link 2 its following joint, as this is where an end-effector would be mounted. O_E is located on the end of Link 2, centered on the 80/20 aluminum bar in the y-direction.

3.1.2 Barycentric Vectors

3.1.2.1 Calculating Link Barycenter Locations

The calculation of a link's barycenter depends on the system's mass distribution μ_i, calculated using the following equation:

$$\mu_i = \begin{cases} 0 & i = 0 \\ \displaystyle\sum_{j=0}^{i-1} \frac{m_j}{M} & i = 1, ..., N \\ 1 & i = N+1 \end{cases}$$

where M is the total system mass, and N is the number of DOF's of the manipulator. For the ABV, $M = 50.360$ lb and $N = 2$. The calculated mass distribution parameters μ_i are presented in Table 3.3 below.

Table 3.3: Mass Distribution μ_i

Link i	0	1	2	3
$\mu_\mathbf{i}$	0.000	0.630	0.790	1.000

The location of a link's barycenter with respect to its CM is calculated by

$${}^{i}\underline{c}_i = {}^{i}\underline{l}_i \mu_i + {}^{i}\underline{r}_i (1 - \mu_{i+1})$$

where $\underline{\mathbf{c}}_\mathbf{i}$ is a body-fixed vector. See Figure 3.5 for a graphical view of the link barycenters.

3.1.2.2 Calculating Barycentric Vectors

The barycentric vectors of a system are expressions of the center of mass, preceding joint, and following joint relative to the link's barycenter as opposed to its CM.[1] They

[1] The $\underline{l}_k$ vectors typically have negative x-values. Papadopoulos represents l_k as a positive scalar in the Appendix E example and includes a negative sign whenever the direction is negative. In this thesis, the negative values are contained in the components of $\underline{l}_k$ for clarity.

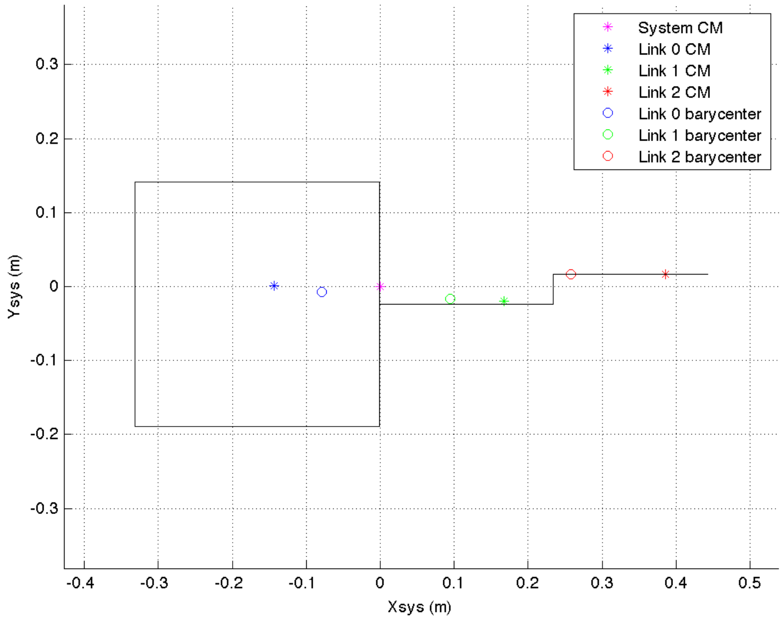


Figure 3.5: ABV Link Barycenters

are indicated by an asterisk and defined by:

$$\underline{l}_i^* = \underline{l}_i - \underline{c}_i \qquad \underline{c}_i^* = -\underline{c}_i \qquad \underline{r}_i^* = \underline{r}_i - \underline{c}_i$$

The barycentric vectors can be compressed into one variable $\underline{v}_{ik}$:

Table 3.4: Barycentric Vectors

	Link 0			Link 1			Link 2		
(m)	$\mathbf{X_0}$	$\mathbf{Y_0}$	$\mathbf{Z_0}$	$\mathbf{X_1}$	$\mathbf{Y_1}$	$\mathbf{Z_1}$	$\mathbf{X_2}$	$\mathbf{Y_2}$	$\mathbf{Z_2}$
${}^k\underline{l}_k^*$	N/A	N/A	N/A	-0.0551	0.0075	0.000	-0.0243	0.000	0.000
${}^k\underline{c}_k^*$	-0.0638	0.0089	0.000	0.0728	-0.0028	0.000	0.128	-0.0001	0.000
${}^k\underline{r}_k^*$	0.119	-0.0167	0.000	0.139	0.0338	0.000	0.185	0.000	0.000

$${}^i\underline{v}_{ik} = \begin{cases} {}^i\underline{r}_i^* & i < k \\ {}^i\underline{c}_i^* & i = k \\ {}^i\underline{l}_i^* & i > k \end{cases}$$

For continuity, all system calculations will be done in the Link 0 frame. Thus, the equations require these barycentric vectors in the same frame, which will introduce joint angles q_1 and q_2 into the equations for ${}^0\underline{v}_{1k}$ and ${}^0\underline{v}_{2k}$. Rotation matrices $\mathbf{{}^0T_k}$ are defined in Section 3.1.3.

$$ {}^0\underline{v}_{ik} = \mathbf{{}^0T_i}\,{}^i\underline{v}_{ik} $$

In line with Virtual Manipulator (VM) dynamics, the k-th body's CM relative to the system CM is compactly represented as:

$$ {}^{sys}\underline{\rho}_k = \sum_{i=0}^{N} {}^{sys}\underline{v}_{ik} \qquad k = 0, ..., N $$

See Figure 3.6 for a graphical representation.

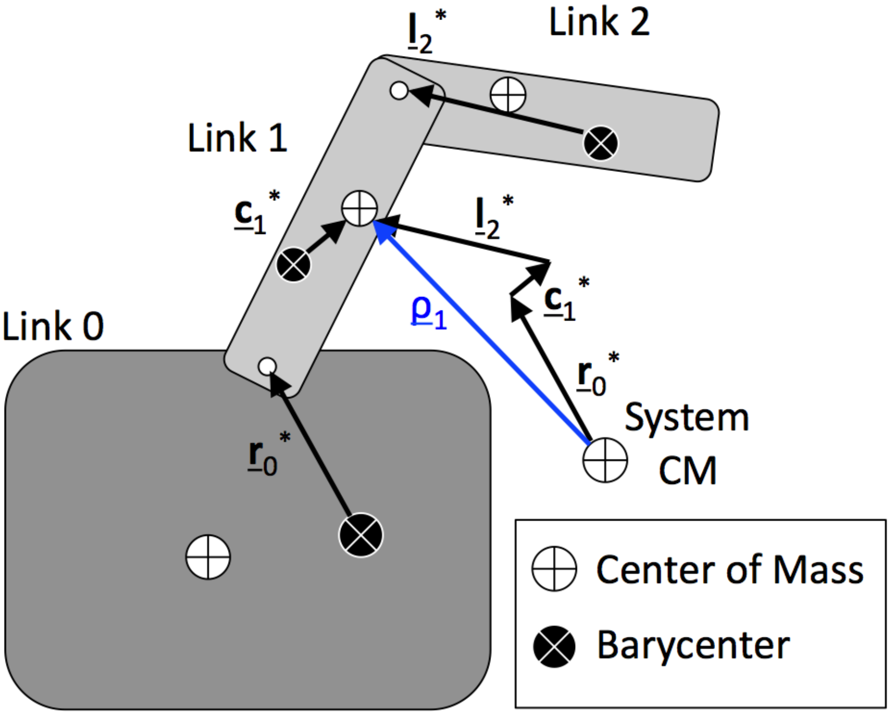


Figure 3.6: Notional Link 1 CM position in System Frame (adapted from [2])

Finally, the inertial position of the k^{th} body's CM represented by $\underline{R}_k$ is calculated by:

$$ \underline{R}_k = \underline{r}_{cm} + \underline{\rho}_k $$

The formulas presented in this section can be extended to find the position of any point p on the vehicle by representing that point in relation to its link's CM. Simply substitute the quantity $\underline{v}_{ik,p}$, given below, for $\underline{v}_{ik}$ in the formulas above. $\underline{r}_{k,p}$ is the vector from the k-th link CM to point m, and δ_{im} is the Kronecker delta.

$$\underline{v}_{ik,p} = \underline{v}_{ik} + \delta_{im}\underline{r}_{k,p}$$

$$\delta_{im} = \begin{cases} 1 & i = k \\ 0 & i \neq k \end{cases}$$

3.1.3 Rotation Matrices

Since both joints are revolute and the system is planar, the rotation matrices $\mathbf{{}^0T_i}$ from the Link 1 and 2 frames to the Link 0 frame are straightforward. Joint angles are represented by q_1 and q_2.

$$\mathbf{{}^0T_1} = \begin{bmatrix} \cos(q_1) & -\sin(q_1) & 0 \\ \sin(q_1) & \cos(q_1) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \mathbf{{}^0T_2} = \mathbf{{}^0T_1} \begin{bmatrix} \cos(q_2) & -\sin(q_2) & 0 \\ \sin(q_2) & \cos(q_2) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

3.1.4 Moments of Inertia

The base vehicle mass was represented as a linear mass distribution function with mass m_0 and CM at the measured CM for the moment of inertia (MOI) calculations. This distribution function was then integrated numerically using the definition of MOI to achieve an estimated MOI about the Link 0 CM. The manipulator link moments of inertia were found using its Pro-Engineer CAD model. The MOI for Link k is represented mathematically as I_k. Typically this is a matrix of inertias, however for the planar system only the I_{zz} value, representing pure rotational inertia about the z-axis, is used. Scalar results are shown in Table 3.5:

Table 3.5: Moments of Inertia

$\mathbf{I_0}$ **(kg-m^2)**	$\mathbf{I_1}$ **(kg-m^2)**	$\mathbf{I_2}$ **(kg-m^2)**
0.251	0.0135	0.0055

3.1.4.1 Calculating Inertia Dyadics

The inertia dyadic D_{ij} is a dyadic representation of the MOI for a body. Since the ABV is a planar system, only the I_{zz} moment of inertia is relevant and the dyadics are scalar values. The simplified equations for the ABV's dyadics are presented below. See [2] for the full derivation[1]. Recall that $\underline{a} \cdot \underline{b} = \|\underline{a}\|\|\underline{b}\| \cos(\theta)$. The units for these parameters are lb-in^2.

$$
\begin{aligned}
{}^0D_{00} &= I_0 + m_0 \left(\underline{c}_0^* \cdot \underline{c}_0^*\right) + \left(m_1 + m_2\right)\left(\underline{r}_0^* \cdot \underline{r}_0^*\right) \\
{}^0D_{11} &= I_1 + m_0 \left(\underline{l}_1^* \cdot {}^0\underline{l}_1^*\right) + m_1 \left({}^0\underline{c}_1^* \cdot {}^0\underline{c}_1^*\right) + m_2 \left({}^0\underline{r}_1^* \cdot {}^0\underline{r}_1^*\right) \\
{}^0D_{22} &= I_2 + \left(m_0 + m_1\right)\left({}^0\underline{l}_2^* \cdot {}^0\underline{l}_2^*\right) + m_2 \left({}^0\underline{c}_2^* \cdot {}^0\underline{c}_2^*\right) \\
{}^0D_{01} &= -M \left({}^0\underline{l}_1^* \cdot \underline{r}_0^*\right) = {}^0D_{10} \\
{}^0D_{02} &= -M \left({}^0\underline{l}_2^* \cdot \underline{r}_0^*\right) = {}^0D_{20} \\
{}^0D_{12} &= -M \left({}^0\underline{l}_2^* \cdot {}^0\underline{r}_1^*\right) = {}^0D_{21}
\end{aligned}
$$

3.1.4.2 Calculating System Inertia Matrices

The equations of motion require several matrices defining system inertia properties. The parameter D_j represents the sum of inertia terms D_{ij} over all i. The parameter D represents the total system inertia about the system CM, expressed here in the

[1] These differ from the equations presented for the planar case in Appendix E of [2] because the ABV's barycentric vectors have non-zero y-components, violating an assumption of that example.

Link 0 frame. These two parameters are scalar for the planar case.

$$\begin{aligned} {}^0D_j &= {}^0D_{0j} + {}^0D_{1j} + {}^0D_{2j} \qquad j = 0, 1, 2 \\ {}^0D &= {}^0D_0 + {}^0D_1 + {}^0D_2 \end{aligned}$$

The projection matrix $\mathbf{{}^0F_k}$ is constructed of the unit vectors of $\mathbf{{}^0T_k}$ parallel to the axis of rotation. This projection matrix represents the effect of the joint rates on the angular momentum of the joint. For the planar case, the $\mathbf{{}^0F_k}$ matrix are 3×2 matrices with non-zero values in the third row only, as rotation only occurs about an axis perpendicular to the plane. $\mathbf{D_q}$ is a 1×2 matrix representing the inertia of the manipulator links, and $\mathbf{D_{qq}}$ is a 2×2 representing the mixed inertias of the manipulator links[1]. Equations for calculating these matrices are shown below.

$$\begin{aligned} \mathbf{{}^0F_1} &= \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \end{bmatrix} \qquad \mathbf{{}^0F_2} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 1 \end{bmatrix} \\ \mathbf{{}^0D_q} &= \begin{bmatrix} {}^0D_1 + {}^0D_2 & {}^0D_2 \end{bmatrix} \\ \mathbf{{}^0D_{qq}} &= \begin{bmatrix} {}^0D_{11} + 2{}^0D_{12} + {}^0D_{22} & {}^0D_{12} + {}^0D_{22} \\ {}^0D_{12} + {}^0D_{22} & {}^0D_{22} \end{bmatrix} \end{aligned}$$

Lastly, the matrices presented in this section are combined into one 5×5 system

[1] Equation (2-50d) of [2] as printed uses the incorrect inertia matrix and is missing the requisite summation over $i = 1, ..., N$.

inertia matrix $\mathbf{H}^{+}$ representing all forms of motion.

$$\mathbf{H}^{+}\left(\underline{q}\right) = \begin{bmatrix} M & 0 & 0 & \underline{0}^{T} \\ 0 & M & 0 & \underline{0}^{T} \\ 0 & 0 & {}^{0}D\left(\underline{q}\right) & {}^{\mathbf{0}}\mathbf{D_{q}}\left(\underline{q}\right) \\ \underline{0} & \underline{0} & {}^{\mathbf{0}}\mathbf{D_{q}^{T}}\left(\underline{q}\right) & {}^{\mathbf{0}}\mathbf{D_{qq}}\left(\underline{q}\right) \end{bmatrix}$$

3.1.5 Thruster Parameters

The ABV thrusters are used to exert forces and torques on the vehicle for this experiment. There are eight thrusters on the vehicle, two on each vertical face as shown in Figure 3.7. Thrusters 1, 4, 5, and 8 are mounted on the bottom level; thrusters 2, 3, 6, and 7 are mounted underneath the middle shelf as shown in Figure 2.3 from Chapter 2. The thrusters are mounted at equivalent distances from the geometric center of the vehicle base such that, if all thrusters exert the same amount of force, pure translation is achieved by commanding thrusters on the same face (e.g. 3 and 7) and pure rotation is achieved by commanding thrusters diagonally opposite from each other (e.g. 4 and 5). Thruster locations and their direction of thrust are provided in Table 3.6, and shown visually in Figure 3.7.

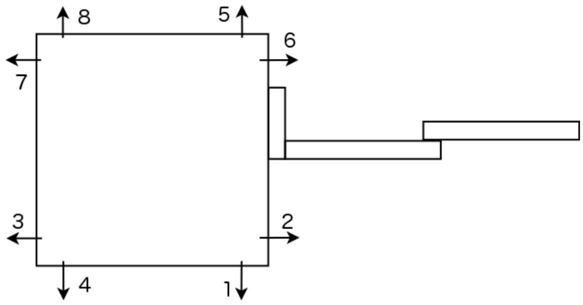


Figure 3.7: Arrangement of ABV Thrusters

The force exerted by a thruster was tested experimentally using a precision scale. The solenoid valve was supported by two pieces of wood secured together to provide a smooth bend in the CO_2 line and ensure steady flow. It was oriented such that it

Table 3.6: Thruster Location and Direction

Thruster	X_0 (m)	Y_0 (m)	Direction
1	0.127	−0.156	$+Y_0$
2	0.152	−0.127	$-X_0$
3	−0.156	−0.127	$+X_0$
4	−0.127	−0.156	$+Y_0$
5	0.127	0.156	$-Y_0$
6	0.152	0.127	$-X_0$
7	−0.156	0.127	$+X_0$
8	−0.127	0.156	$-Y_0$

thrusted directly down onto the scale. See Figure 3.8 for a photograph of the test stand. The scale is calibrated to measure the gravitational force applied to it, so a simple conversion is necessary to translate the displayed mass into applied force:

$$F_{thruster} = m_{measured,kg}g$$

Thruster 3 was tested, as it is easily and safely removable. Other thrusters require detaching the bottom of the frame or moving components to uninstall, putting the hardware integrity at risk during the uninstall and reinstall processes. Thruster 3 was commanded on through the ABV software and remained on until the program was killed. At 80 psi, the scale steadily read 39.0 grams, or 0.039 kg, over multiple periods of up to 5 seconds. This translates into a thrust force of 0.383 N, or 0.086 lbf.

3.2 Kinematics

The kinematics of a system define how its components move in relation to each other without reference to forces and torques. The geometry of each component is used to determine a system's kinematic equations defining its motion, and velocity and acceleration along that path. Typically, kinematic equations are written in terms of end-

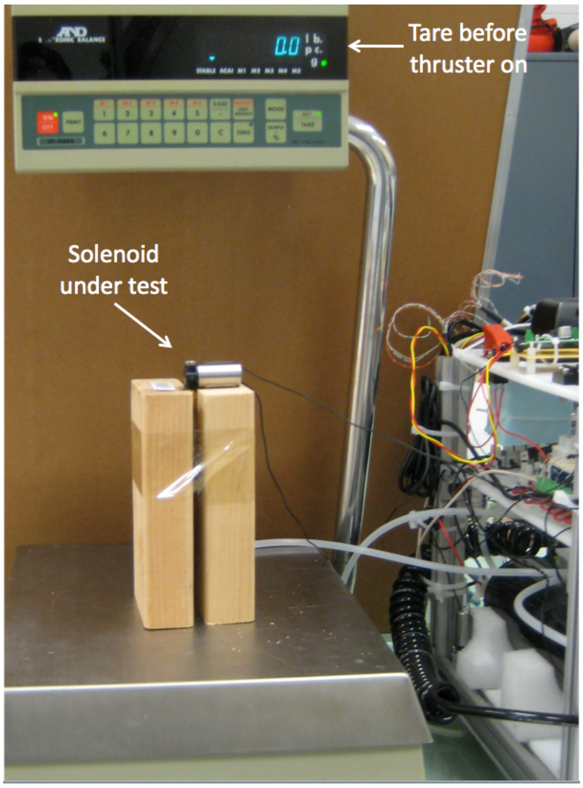


Figure 3.8: Thrusters Test Stand

effector position and velocity in the inertial frame, as the end-effector is what needs directing and requires a fixed reference frame. For the ABV, spacecraft kinematics are of primary interest as it is directly measured with ARToolkit. Link positions can be calculated from the spacecraft state using joint angles and the rotation matrices defined previously. The inertial frame for the ABV is the Table frame, defined by ARToolkit.

3.2.1 Position and Orientation

The position of the Link 0 frame origin O_0 in the Table frame is most important for this model. The Link 1 and 2 frames can easily be found from there using barycentric vectors and joint angles. The rotation matrix $\mathbf{T_0}$ from the Link 0 frame to the Table frame depends on the spacecraft position and orientation as reported by ARToolkit.

Recall that the System frame has the same orientation as the Table frame at all times.

$$\underline{r}_0 = \underline{r}_{cm} + \underline{\rho}_0$$

Orientation of the Link 0 frame is simply the spacecraft attitude θ reported by ARToolkit.

3.2.2 Velocities

The Cartesian velocity of the Link 0 frame origin can be found by differentiating its position expression with respect to time.

$$\underline{\dot{r}}_0 = \underline{\dot{r}}_{cm} + {}^{sys}\underline{\dot{\rho}}_0$$

Since the Link CM vectors that make up $\underline{\rho}_k$ are body-fixed, the velocity of each link's CM must include the angular velocity of all the links. Angular velocities ω_k for the ABV are scalar quantities as the system only rotates about axes perpendicular to the air-bearing plane. See [2] for the complete derivation.

$$\underline{\dot{\rho}}_0 = \underline{\omega}_0 \times {}^0\underline{c}_0^* + \underline{\omega}_1 \times {}^1\underline{l}_1^* + \underline{\omega}_1 \times {}^2\underline{l}_2^*$$

The angular velocity vectors $\underline{\omega}_k$ have non-zero values in the z-coordinate only. Since all axes of rotation are parallel, the angular velocities are the same in all frames and therefore frame transformations are unnecessary. The spacecraft angular velocity is indicated by $\dot{\theta}$ and the joint rates for Links 1 and 2 are indicated by $\dot{q}_1$ and $\dot{q}_2$ respectively.

$$\|\underline{\omega}_0\| = \dot{\theta} \qquad \|\underline{\omega}_1\| = \dot{\theta} + \dot{q}_1 \qquad \|\underline{\omega}_2\| = \dot{\theta} + \dot{q}_1 + \dot{q}_2$$

3.2.3 Jacobians

The Jacobian is a matrix that relates joint velocities to linear velocities. This dynamics model uses three distinct Jacobians and a fourth Jacobian that is a composition of the other three. All Jacobians are found in the Link 0 frame, as that is were all velocities are calculated, and relative to the spacecraft CM (denoted by the additional subscript S). Since forces and torques will be applied directly to the spacecraft using the ABV's thrusters, the Link 0 Jacobians are most relevant. See Chapter 2 of [2] for a full derivation. The skew-symmetric representation of a vector is:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix}^{\times} = \begin{bmatrix} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{bmatrix}$$

${}^{\mathbf{0}}\mathbf{J_{11}}$ is a 3×3 matrix that maps the spacecraft angular velocity into Cartesian velocity in the Link 0 frame. Because the ABV is a planar system, this Jacobian only has four non-zero scalar elements, indicated by c.

$${}^{\mathbf{0}}\mathbf{J}_{\mathbf{11},\mathbf{S}} = -\left[{}^{0}\underline{c}_{0}^{*}\right]^{\times} - \left[{}^{0}\underline{l}_{1}^{*}\right]^{\times} - \left[{}^{0}\underline{l}_{2}^{*}\right]^{\times} = \begin{bmatrix} 0 & 0 & y \\ 0 & 0 & -x \\ -y & x & 0 \end{bmatrix}$$

${}^{\mathbf{0}}\mathbf{J_{12}}$ is a $3 \times N$ matrix for the ABV that maps the joint velocities into Cartesian velocities. For the planar case, the third row mapping to Cartesian z-velocity is always 0, and other elements are scalar.

$${}^{\mathbf{0}}\mathbf{J}_{\mathbf{12},\mathbf{S}} = -\left[{}^{0}\underline{l}_{1}^{*}\right]^{\times} {}^{\mathbf{0}}\mathbf{F_1} - \left[{}^{0}\underline{l}_{2}^{*}\right] {}^{\mathbf{0}}\mathbf{F_2} = \begin{bmatrix} y & y \\ -x & -x \\ 0 & 0 \end{bmatrix}$$

$^{\mathbf{0}}\mathbf{J}_{\mathbf{22},\mathbf{S}}$ maps the joint velocities into angular velocities, and is equivalent to $^{\mathbf{0}}\mathbf{F}_{\mathbf{0}}$ which is a 3 × 2 matrix of zeros.

$$^{\mathbf{0}}\mathbf{J}_{\mathbf{22},\mathbf{s}} = {}^{\mathbf{0}}\mathbf{F}_{\mathbf{0}} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

The last Jacobian, $^{\mathbf{0}}\mathbf{J}^{+}$, is a 6×8 matrix containing all three of the previous Jacobians in order to provide a complete state mapping from joint space to Cartesian space. For the state vector $\underline{x}$ containing both Cartesian and angular states, where $\mathbb{I}_n$ is a $n \times n$ identity matrix,

$$\dot{\underline{x}} = \begin{bmatrix} \dot{\underline{r}}_0 \\ \underline{\omega}_0 \end{bmatrix} = \mathbf{J}^{+} \begin{bmatrix} {}^0\dot{\underline{r}}_{cm} \\ {}^0\underline{\omega}_0 \\ \dot{\underline{q}} \end{bmatrix}$$

where

$$^{\mathbf{t}}\mathbf{J}^{+} = \begin{bmatrix} \mathbf{T_0} & \mathbf{0} \\ \mathbf{0} & \mathbf{T_0} \end{bmatrix} {}^{\mathbf{0}}\mathbf{J}^{+}\left(\underline{q}\right)$$

$$^{\mathbf{0}}\mathbf{J}^{+}\left(\underline{q}\right) = \begin{bmatrix} \mathbb{I}_3 & {}^{\mathbf{0}}\mathbf{J}_{\mathbf{11}} & {}^{\mathbf{0}}\mathbf{J}_{\mathbf{12}} \\ \mathbf{0} & \mathbb{I}_3 & {}^{\mathbf{0}}\mathbf{J}_{\mathbf{22}} \end{bmatrix}$$

These Jacobians can all be simplified using the planar case assumptions that $\dot{r}_{0,z}$, $\omega_{0,x}$, and $\omega_{0,y}$ are equal to 0. In expanded form, where c is a scalar, the velocity expression above is shown in Figure 3.9. Red boxes mark the parameters that are 0 for the planar case, and red lines indicate the rows and columns that are always equal to $\underline{0}$ or multiplied by 0 for the planar case. This graphic shows that $^{\mathbf{0}}\mathbf{J}_{\mathbf{11},\mathbf{S}}$ is reduced to a 2 × 1 matrix, $^{\mathbf{0}}\mathbf{J}_{\mathbf{11},\mathbf{S}}$ is reduced to a 2 × 2 matrix, and $^{\mathbf{0}}\mathbf{J}_{\mathbf{22},\mathbf{S}}$ is reduced

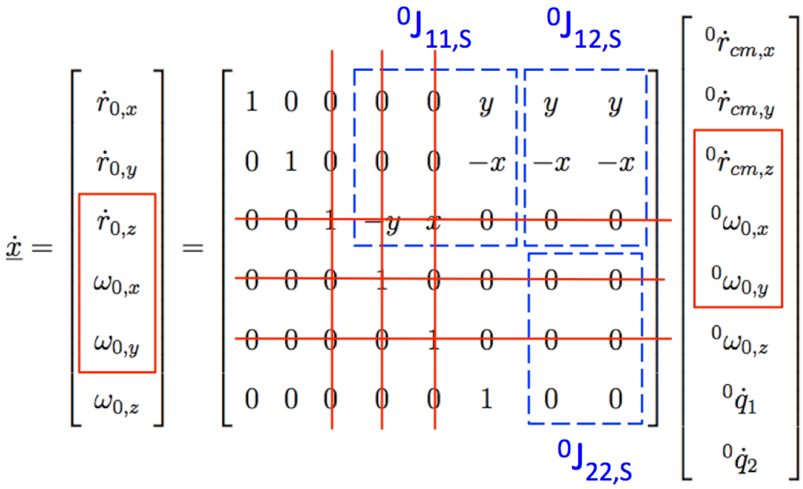


Figure 3.9: Spacecraft Jacobian simplification for the planar case.

to a 1 × 2 matrix. These Jacobians can again be concatenated into a 3 × 5 system Jacobian where:

$$
{}^{\mathbf{0}}\mathbf{J}_{\mathbf{S}}^{+}\left(\underline{q}\right) = \begin{bmatrix} \mathbb{I}_2 & {}^{\mathbf{0}}\mathbf{J}_{\mathbf{11,S}} & {}^{\mathbf{0}}\mathbf{J}_{\mathbf{12,S}} \\ \underline{0}^T & 1 & {}^{\mathbf{0}}\mathbf{J}_{\mathbf{22,S}} \end{bmatrix}
$$

3.3 Dynamics

The dynamics of a system describe how forces and torques applied to the system affect its motion. This thesis is investigating free-motion dynamics, so the only external forces and torques applied to the system are from the ABV thrusters. All end-effector forces and torques are equal to 0, and therefore omitted from the EOM.

3.3.1 Generalized Forces

The generalized forces Q for a system are expressed in terms of generalized coordinates. The number of generalized coordinates is the number of DOF of the system, and all generalized coordinates are independent. In the absence of end-effector forces and torques, the linear generalized forces Q_x and Q_y are:

$$
Q_x = {}^0f_{x,S} \qquad Q_y = {}^0f_{y,S}
$$

where ${}^{0}\underline{f}_{S}$ is a vector of linear forces ${}^{0}f_{x,S}$ and ${}^{0}f_{y,S}$ applied to the spacecraft (Link 0).

The total torque on the spacecraft Q_θ includes the torque ${}^{0}n_S$ applied directly to the spacecraft and the rotational effects of the linear forces applied to the spacecraft. Note that a Jacobian's transpose maps the same parameters in the opposite direction. Here, linear forces are mapped into joint space using the ${}^{\mathbf{0}}\mathbf{J}_{\mathbf{11},\mathbf{S}}{}^{T}$ Jacobian.

$$Q_\theta = {}^{0}n_S + {}^{\mathbf{0}}\mathbf{J}_{\mathbf{11},\mathbf{S}}{}^{T} \begin{bmatrix} {}^{0}f_{x,S} \\ {}^{0}f_{y,S} \end{bmatrix}$$

The forces exerted on the system cause configurations changes, captured in $\underline{Q}_q$. Here, ${}^{\mathbf{0}}\mathbf{J}_{\mathbf{12},\mathbf{S}}{}^{T}$ maps Cartesian velocities into joint space velocities.

$$\underline{Q}_q = \begin{bmatrix} Q_{q1} \\ Q_{q2} \end{bmatrix} = {}^{\mathbf{0}}\mathbf{J}_{\mathbf{12},\mathbf{S}}{}^{T} \begin{bmatrix} {}^{0}f_{x,S} \\ {}^{0}f_{y,S} \end{bmatrix}$$

3.3.2 Non-linear Centripetal and Coriolis Terms

The non-linear terms C_1^+ and $\underline{C}_2^+$ represent the centripetal and Coriolis forces inherent in the manipulator motion. They include the terms $\hat{d}_{ij}$, which are dependent on the cross-product of barycentric vectors and calculated here to provide more compact equations for the $\underline{C}^+$ vector of non-linear terms. $\hat{d}_{ij}$ will always be a vector perpendicular to the air-bearing surface and thus can be considered scalars for the planar case. The equations for the ABV's cross-product terms follow equation(2-48) of [2]:

$$\begin{aligned} \hat{d}_{01} &= -M\left({}^{0}\underline{l}_1^* \times \underline{r}_0^*\right) = -\hat{d}_{10} \\ \hat{d}_{02} &= -M\left({}^{0}\underline{l}_2^* \times \underline{r}_0^*\right) = -\hat{d}_{20} \\ \hat{d}_{12} &= -M\left({}^{0}\underline{l}_2^* \times {}^{0}\underline{r}_1^*\right) = -\hat{d}_{21} \end{aligned}$$

The non-linear terms are defined by standard centripetal and Coriolis cross-product formulas. The generalized formulas, available in [18], are quite complex and therefore are not presented here. Since the ABV is a planar system and all barycentric vector cross-products are perpendicular to the plane, this model can use the same formulas for the nonlinear terms that Papadopoulos presents in Section 2.5 of [2], which discusses the simplifications for the planar model. These equations are reprinted in this thesis's symbolism here:

$$C_1^+ = \left(\hat{d}_{10} + \hat{d}_{20}\right) \|\omega_0\|^2 + \left(\hat{d}_{01} + \hat{d}_{21}\right) \|\omega_1\|^2 + \left(\hat{d}_{02} + \hat{d}_{12}\right) \|\omega_2\|^2$$

$$\underline{C_2^+} = \begin{bmatrix} \left(\hat{d}_{10} + \hat{d}_{20}\right) \|\omega_0\|^2 + \hat{d}_{21}\|\omega_1\|^2 + \hat{d}_{12}\|\omega_2\|^2 \\ \hat{d}_{20}\|\omega_0\|^2 + \hat{d}_{21}\|\omega_1\|^2 \end{bmatrix}$$

These two parameters are combined into a 5 × 1 vector

$$\underline{C}^+ = \begin{bmatrix} \underline{0} \\ C_1^+ \\ \underline{C_2^+} \end{bmatrix}$$

3.3.3 Equations of Motion

The non-linear terms can be used along with the physical parameters described above to create equations of motion for the ABV:

$$\begin{aligned} &Q_x = M^0 \ddot{x}_{cm} \qquad Q_y = M^0 \ddot{y}_{cm} \\ &Q_\theta = {}^0D\left(\underline{q}\right)\ddot{\theta} + \mathbf{{}^0D_q}\left(\underline{q}\right)\underline{\ddot{q}} + C_1^+ \\ &\underline{Q}_q = \mathbf{{}^0D_q^T}\left(\underline{q}\right)\ddot{\theta} + \mathbf{{}^0D_{qq}}\left(\underline{q}\right)\underline{\ddot{q}} + \underline{C_2^+} \end{aligned}$$

Lastly, in order to determine complete equations of motion, set the generalized forces defined above with the behavioral motion defined by $\mathbf{H}^+$ and $\underline{C}^+$. These equations are shown in compact matrix form:

$$\mathbf{H}^{+}\left(\underline{q}\right)\begin{bmatrix}\ddot{x}_{cm}\\ \ddot{y}_{cm}\\ \ddot{\theta}\\ \ddot{\underline{q}}\end{bmatrix}+\underline{C}^{+}\left(\underline{q},\dot{\theta},\dot{\underline{q}}\right)=\begin{bmatrix}\underline{0}\\ \underline{\tau}\end{bmatrix}+{}^{\mathbf{0}}\mathbf{J}_{\mathbf{S}}^{+T}\left(\underline{q}\right)\begin{bmatrix}{}^{0}\underline{f}_{S}\\ {}^{0}n_{S}\end{bmatrix}$$

4

Methodology and Results

Several different experiments were performed; the derived equations of motion (see Chapter 3 were evaluated for the gathered data and compared to the observed accelerations. For initial testing, one joint was moved at a time. Next, both joints were moved for 2-DOF testing. In this “sidearm”motion the manipulator began along the frame of the vehicle and ended wrapped around the frame on the other side (see Figure 4.1). Lastly, the ABV’s thrusters were actuated to provide forces and torques on the base while the 2-DOF manipulator trajectories were occurring. For all experiments, the Black Jaguars were commanded in position control mode. The manipulator was power-cycled in between each run to erase the configuration parameters. Additionally, the pucks were cleaned with ethyl alcohol prior to testing per manufacturer recommendation. The air-bearing surface was covered when not in use to reduce contamination, however it too was cleaned prior to testing if necessary.

4.1 Data Measurement and Recording

All necessary experimental data was recorded in a file `Data.txt` during the control loop execution. The file was a simple tab-delimited format with floating-point numbers. Its contents are listed in Table 4.1. The time stamp for each line of data was

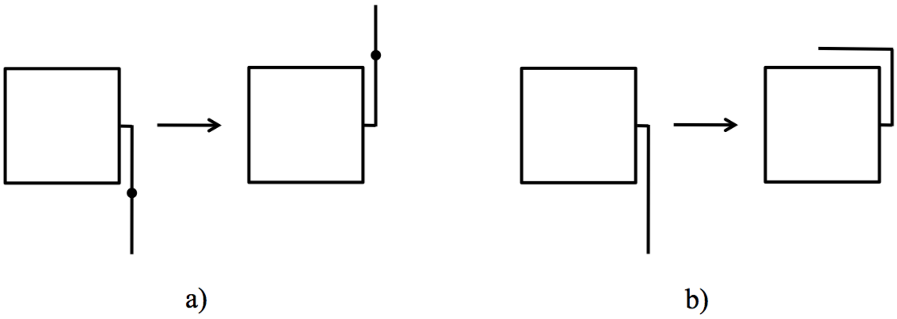


Figure 4.1: a) 1-DOF test motion b) 2-DOF ”sidearm” test motion

provided by the QNX operating system. The inertial frame x-position, y-position, and attitude θ of the base (Link 0) were supplied by ARToolkit through the ground station. Position data is in mm and attitude in radians. The position, attitude, and linear velocity of Link 0 are not used in the model assessment that follows because these are ignorable variables in the system dynamics, however are included in the data file for completeness.

Table 4.1: Recorded Parameters

Column	Data	Units	Source
1	Time	s	Operating System
2	Base X position	mm	ARToolkit
3	Base Y position	mm	ARToolkit
4	Base attitude	rad	ARToolkit
5	Base angular velocity	deg/s	IMU
6	Base X acceleration	g	IMU
7	Base Y acceleration	g	IMU
8	Thruster status	lb_f	`actuator` Object
9	Link 1 angle	rad	Absolute Encoder
10	Link 2 angle	rad	Absolute Encoder
11	Link 1 joint rate	rad/s	Black Jaguar
12	Link 2 joint rate	rad/s	Black Jaguar
13	Link 1 applied current	A	Current shunt
14	Link 2 applied current	A	Current shunt

The IMU’s gyroscope directly provides the angular velocity of Link 0 in degrees/second as measured in the IMU frame. The IMU’s accelerometer provides linear accelerations $\ddot{x}$ and $\ddot{y}$ in g as measured in the IMU frame. The thruster status is a flag set by the `actuator` object of the ABV software that is set to 1 if thruster 3 is on, and 0 if no thrusters are on. It is then multiplied by the force per thruster such that applied force in lb$_f$ is recorded.

The joint angles are reported from the absolute encoders via the Freeduino. The Freeduino reports data in encoder counts which is converted to radians in the `ArduinoMsg` object for recording. The 0-point was calibrated prior to testing using a right-angle edge to accurately set the home position. The Black Jaguars report motor speed in revolutions per minute. The Black Jaguar data is extrapolated across the gearbox and converted to radians per second before recording in the data file to represent the speed of the physical link.

Lastly, the current applied to each joint motor is reported from the current shunts via the Freeduino. The data is reported as an integer value representing the location within the possible voltage range for the corresponding analog pin. In the `ArduinoMsg` object this value is converted to amperes using a bias value and a sensitivity value. The bias value was found experimentally by reading the sensor's output voltage with a multimeter when the current output was 0 A. The sensitivity value was found experimentally by supplying several constant known currents to the motor and reading the sensor output voltage with a multimeter. The current data in amperes is recorded in the data file.

4.2 Data Analysis

All data analysis for this thesis was performed in MATLAB. The raw data is read into a structure; each element of the data structure is converted into the appropriate units for analysis and put in a vector for implementation. The converted units are shown in Table 4.2 and aim for consistency throughout the analysis. The spacecraft angular rate is transformed into the Link 0 frame simply by changing the sign on the data value, as the IMU frame z-axis points into the floor. The y-acceleration is handled in the same way. See Figure 2.8 for a coordinate frame diagram.

Table 4.2: Units for Analysis

Data	**Analysis Units**
Time	s
Base angular velocity	rad/s
Base X acceleration	m/s^2
Base Y acceleration	m/s^2
Base angular acceleration	rad/s^2
Thruster status	N
Link 1 angle	rad
Link 2 angle	rad
Link 1 joint rate	rad/s
Link 2 joint rate	rad/s
Link 1 joint acceleration	rad/s^2
Link 2 joint acceleration	rad/s^2
Link 1 applied current (torque)	N-m
Link 2 applied current (torque)	N-m

The Black Jaguar motor speed reports were not consistent with the observed joint motions as determined by the absolute encoders, so these data points were not used. Three parameters were calculated using a five-point central-differencing method [27] and then processed through a moving average filter using MATLAB's built-in `filter` function to reduce noise. The angular velocity of joints one and two are calculated from their respective position reports, and the joint angular accelerations are calculated from the unfiltered angular velocity data. The base angular acceleration is calculated from the IMU angular rate data. The formula for a five-point central difference first derivative is

$$f'(x) = \frac{-f(x+2h) + 8f(x+h) - 8f(x-h) + f(x-2h)}{12h}$$

This formula assumes that all time steps h are equal. Since the time steps in these data sets are not all equal, the central difference formula uses the average of the time steps in the area of concern. Note that the moving average filter will induce a slight lag into the observed data.

The current data is converted to joint torque using the manufacturer-provided torque constant for the motor [28]. Masses are in kilograms, lengths in meters, and angles in radians. The physical parameters presented in Chapter 3 are reported by the function calculating those parameters, which also use pounds and inches for all units. The analysis script calculates the dynamics parameters for each time step in the Link 0 frame. Rotation matrices for the Link 1 and 2 frames are constructed using the joint angle data for that time step.

The EOM are rearranged for ease of comparison to be:

$$\begin{bmatrix} {}^{0}\ddot{x}_{cm} \\ {}^{0}\ddot{y}_{cm} \\ \ddot{\theta} \\ \ddot{\underline{q}} \end{bmatrix} = \mathbf{H}^{+^{-1}}\left(\underline{q}\right)\left(\begin{bmatrix} \underline{0} \\ \underline{\tau} \end{bmatrix} + {}^{\mathbf{0}}\mathbf{J}_{\mathbf{S}}^{+^{T}}\left(\underline{q}\right)\begin{bmatrix} {}^{0}\underline{f}_{S} \\ {}^{0}n_{S} \end{bmatrix} - \underline{C}^{+}\left(\underline{q}, \dot{\theta}, \dot{\underline{q}}\right)\right)$$

$\mathbf{H}^{+}$ is a square, positive definite matrix by definition, therefore it is always invertible. The left side of this equation represents the observed accelerations, and the right-hand side represents the expected dynamics based on the state of the system at that time step. In order to calculate the acceleration of the system CM in the Link 0 frame, recall these equations from Chapter 3:

$$\underline{r}_0 = \underline{r}_{cm} + {}^{sys}\underline{\rho}_0 \qquad \underline{\rho}_k = \sum_{i=0}^{N} \underline{v}_{ik} \qquad \dot{\underline{\rho}} = \sum_{i=0}^{N} \underline{\omega}_i \times \underline{v}_{ik} \qquad k = 0, ..., N$$

Rearranging the first equation, differentiating twice, and transforming into the Link 0 frame provides:

$${}^{0}\ddot{\underline{r}}_{cm} = {}^{0}\ddot{\underline{r}}_0 - {}^{0}\ddot{\underline{\rho}}_0$$

where

$$^{0}\ddot{\underline{\rho}}_0 = \dot{\underline{\omega}}_0 \times {}^{0}\underline{c}_0 + \underline{\omega}_0 \times \left(\underline{\omega}_0 \times {}^{0}\underline{c}_0^*\right) + \dot{\underline{\omega}}_1 \times {}^{0}\underline{l}_1 + \underline{\omega}_1 \times \left(\underline{\omega}_1 \times {}^{0}\underline{l}_1^*\right) + \dot{\underline{\omega}}_2 \times {}^{0}\underline{l}_2 + \underline{\omega}_2 \times \left(\underline{\omega}_2 \times {}^{0}\underline{l}_2^*\right)$$

Data for the Link 0 linear acceleration $^{0}\ddot{\underline{r}}_0$ are found directly in the data file.

The $\mathbf{H}^+$, $\mathbf{^0J_s^+}^T$, and $\underline{C}^+$ are calculated using physical parameters as described in Chapter 3. As described above, the joint torques $\underline{\tau}$ are calculated from the current data. The forces and torques applied to the base, $^{0}\underline{f}_S$ and $^{0}n_S$ respectively, are found using the thruster status and the measured position of the thrusters provided in Table 3.6. In the graphs throughout this section, the “Observed”data points are the accelerations resulting from sensor acceleration and velocity data (shown in blue), while the predicted (“Calculated”) data points are the accelerations calculated using the dynamics equations above (shown in dashed red), which incorporate the torque and position data to predict accelerations.

4.2.1 Additional Considerations

It is important to synchronize the beginning of each motion in order to observe consistency in the system performance. The time steps are not all equal, and the time stamps of each step vary from run to run due to system execution time. However, joint motion always begins in the 101st data cycle as instructed by the software. Thus, each run’s time vector is recalibrated for plotting such that the 101st data cycle represents 0 seconds in time. This allows for much clearer visual interpretation of the data without loss of fidelity.

There are some aspects of the physical system that are not accounted for here and will contribute to inconsistencies between the model and the experimental data. Propellant is being expelled throughout the experiment which causes the system center of mass to change as the experiments progress. Additionally, carbon dioxide is stored in liquid form in the propellant tanks, and therefore is susceptible to slosh during sudden movements such as the maneuvers performed here. Lastly, there is friction in each joint, in the gearbox of each motor, and with the air-bearing surface at each point of contact that will contribute to delayed or damped responses. The effects of friction are presented in Section 4.7.

4.3 1-DOF Trajectory

For this experiment, the two manipulator links were braced together to act as a single link manipulator (see Figure 4.2). Only Joint 1 was actuated. Since Joint 2 was fixed in place, any current measurements were ignored. It is assumed that any current applied to the Joint 2 motor by the still-active motor controller was ineffective due to the brace. Additionally, the mass and inertia of the brace were added to the barycentric vector calculations. The dynamics equations were converted into 1-DOF form to maintain a valid, invertible $\mathbf{H}^{+}$ matrix.

Ten runs were performed for this experiment. Joint motions are shown in Figure 4.3. Note that the position of the joints is not as important as how it moves, therefore the manipulator does not need to be set to exactly the same position each time.

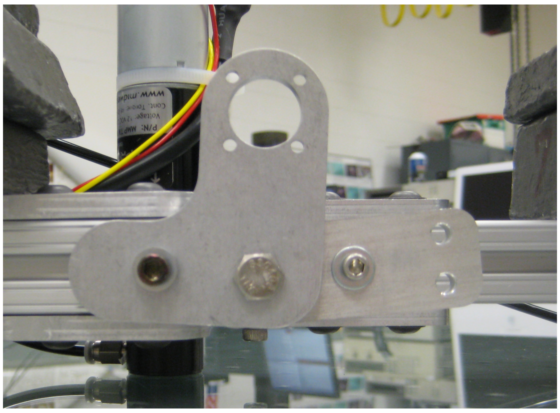


Figure 4.2: Link 2 secured in place for the 1-DOF experiment.

Figure 4.4 shows the observed and calculated accelerations of the system CM. The calculated acceleration of the system CM is always 0 for this experiment set because no external forces or torques were applied to the spacecraft base. However, as discussed previously, friction, pro-

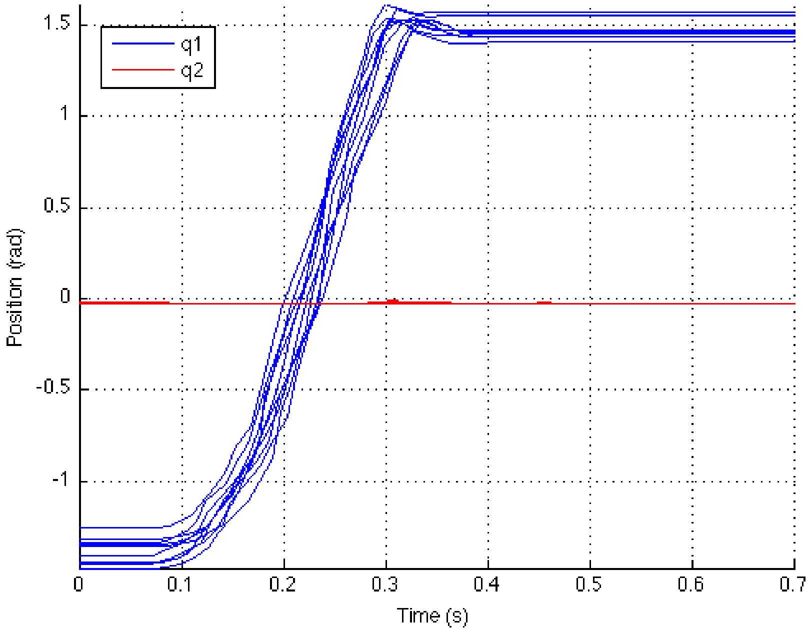
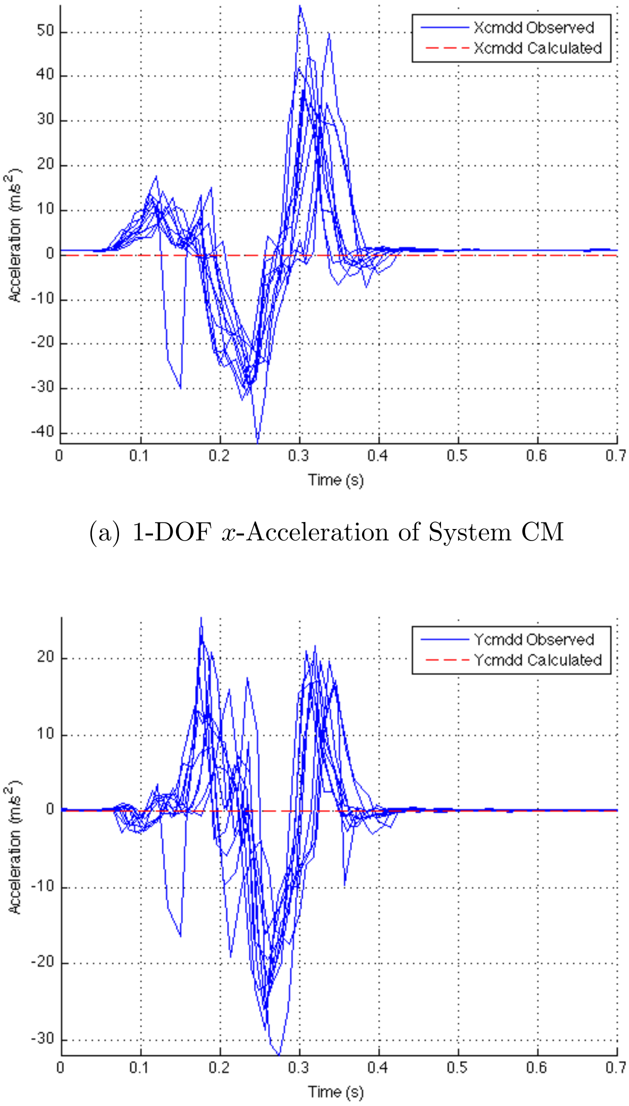


Figure 4.3: 1-DOF Joint Motions

pellant usage and slosh will contribute to changing the system CM throughout the maneuver. The corresponding changes in acceleration are seen in the experimental data shown in blue. The effects of friction on the system CM are discussed in Section 4.7.

The model for $\ddot{\theta}$ includes the angular velocities of the other links in the C_1^+ term, and therefore it should see some variation during this maneuver. Results are shown in Figure 4.5. It is easily seen that observed values follow the expected trend provided by the model, although at smaller magnitude. The maximum expected magnitude seems quite large at 200 rad/s^2, however this acceleration is only expected for about 0.02 seconds. This correlates to a 0.08 rad change in position which is physically reasonable.

The joint torque $\underline{\tau}$ and the nonlinear term C_2^+, which for the 1-DOF case are scalar values, contribute to Joint 1 accelerations. In the 1-DOF case, the C_2^+ term only contains a spacecraft angular velocity term. The results for this parameter are shown in Figure 4.6. The experimental data follows the expected trend provided by



(a) 1-DOF x-Acceleration of System CM

(b) 1-DOF y-Acceleration of System CM

Figure 4.4: 1-DOF System CM Acceleration

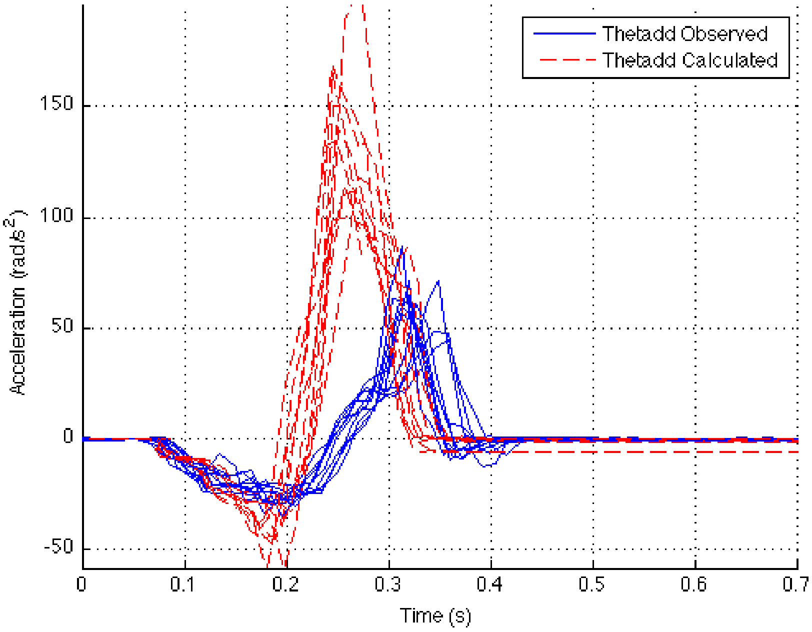


Figure 4.5: 1-DOF Angular Acceleration of Base

the model, however at slightly higher magnitude. The apparent lag in experimental data is due to filtering. Joint 2 acceleration is not observed for this case because it is secured to Joint 1.

The 1-DOF experiment highlights some system traits that will persist throughout the experimental discussion. Linear accelerations of the system CM are not identically zero as expected during the maneuver, suggesting unmodeled external forces. The angular accelerations of the vehicle base and each link follow the expected qualitative behavior, although at different magnitudes. The nonlinear $\underline{C}^{+}$ vector contributes the most to the large peaks in the model's behavior due to centripetal and Coriolis accelerations.

4.4 2-DOF Trajectory

The "sidearm" motion was executed in this experiment to create greater perturbations in the system. Joint motions are shown in Figure 4.7. The range of motion for Joint 2

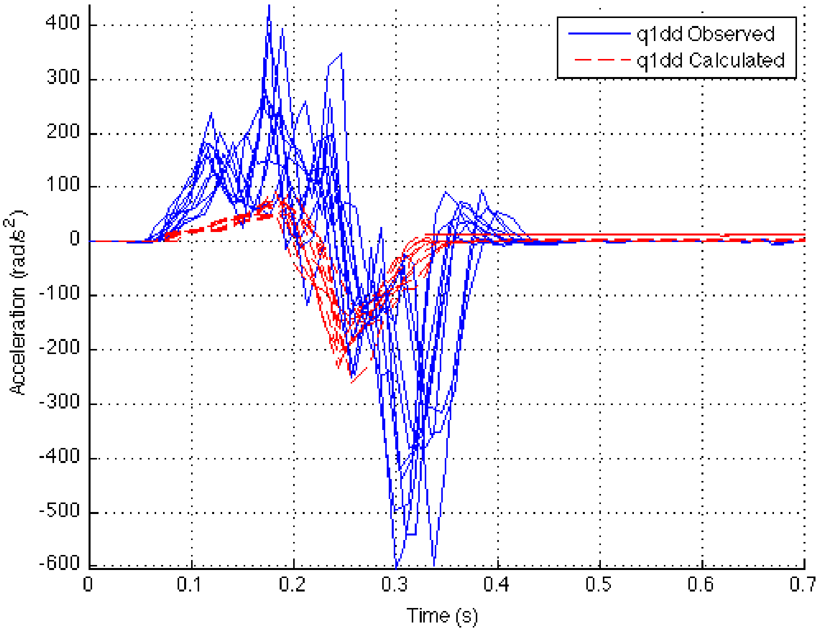


Figure 4.6: 1-DOF Angular Acceleration of Joint 1

is limited in the negative direction by the joint structure and in the positive direction by the vehicle frame and carbon dioxide tubing that resides outside the frame on the $+y_0$ face of the vehicle.

Results for the linear acceleration of the system CM are shown in Figure 4.8. As in the previous test case, no acceleration of the system CM was expected per the model, shown in red. There is again some motion of the system CM due to friction and propellant. The magnitude of the system CM accelerations is generally consistent with those observed in the 1-DOF case, although the second peak of the x-acceleration is about half as large.

The base angular acceleration $\ddot{\theta}$ is shown in Figure 4.9. As in the 1-DOF case, the model expects much higher base acceleration than is observed. This is due to the nonlinear $\underline{C}^+$ term. This feature will be investigated in Section 4.7.

The Joint 1 and 2 accelerations are shown in Figure 4.10. Similar to the 1-DOF case, the Joint 1 acceleration is larger than expected at first, however it does not exhibit the acceleration range predicted by the model. The nonlinear effects of the

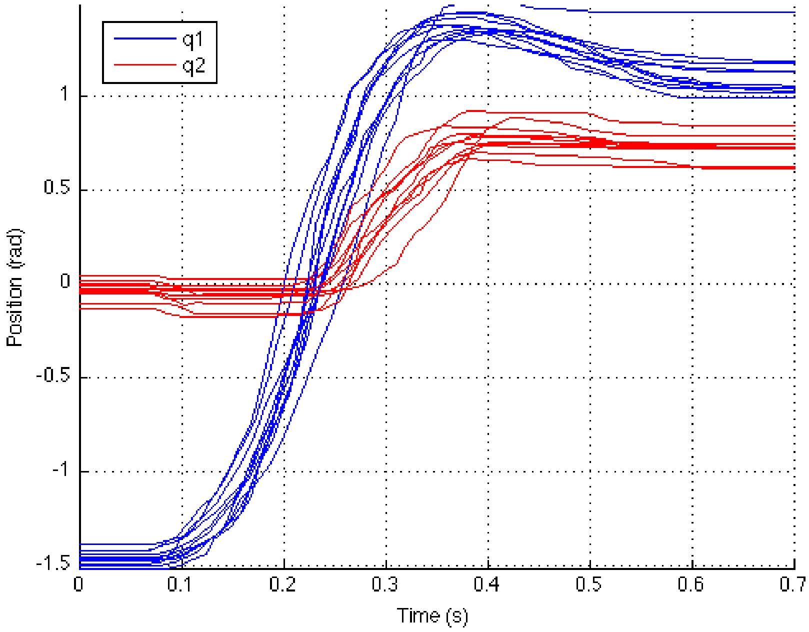
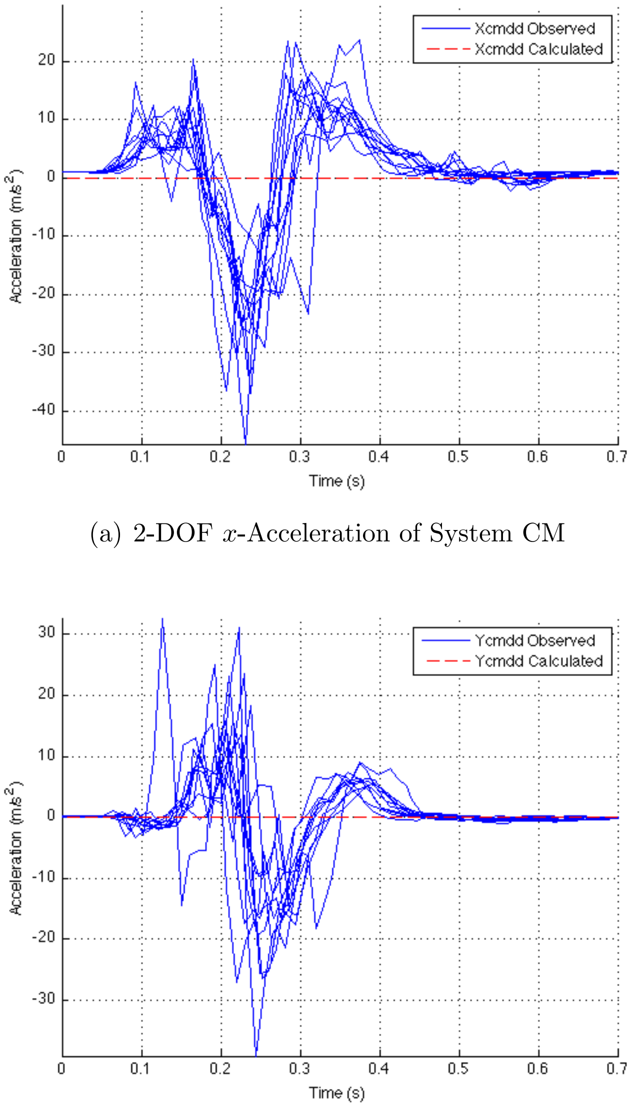


Figure 4.7: 2-DOF Joint Motions

Joint 2 motion on the Joint 1 motion are smaller than predicted. Notice the Joint 2 acceleration is negative at first, due to the motion of Joint 1, and the controller overcompensates trying to get to the commanded position while counteracting the moment caused by the Joint 1 motion that is occurring simultaneously. There is noticeable oscillation at the end of the Joint 2 motion as the controller settles. Additionally, there are two outlying data sets most noticeable in the Joint 2 acceleration plot. Most data sets have a peak $\ddot{q}_2$ around 600 rad/s^2, while two reach about 750 and 950 rad/s^2. A similar outlying data set is noticable in the valley of the $\ddot{q}_1$ data set - most data sets have a minimum of about -425 rad/s^2 but one run reaches about -700 rad/s^2. These statistical anomalies are most likely due to inconsistencies in the performance of the air-bearing pucks.

The same phenomenon observed in the 1-DOF experiment are also present during the 2-DOF maneuver. The trends that the model predicts are in fact present, however the magnitudes slightly different. Linear acceleration of the center of mass is not expected, however due to the physical design some acceleration is observed. And, the



(a) 2-DOF x-Acceleration of System CM

(b) 2-DOF y-Acceleration of System CM

Figure 4.8: 2-DOF System CM Acceleration

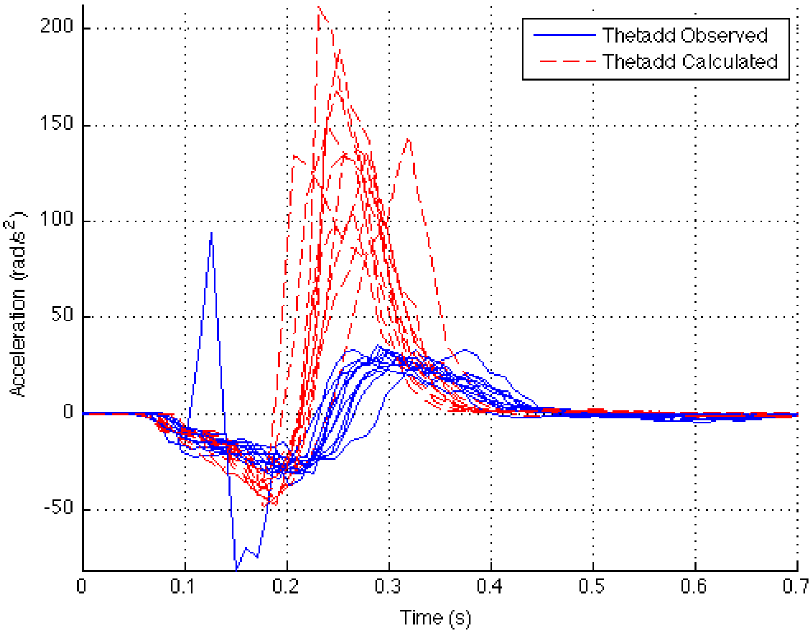


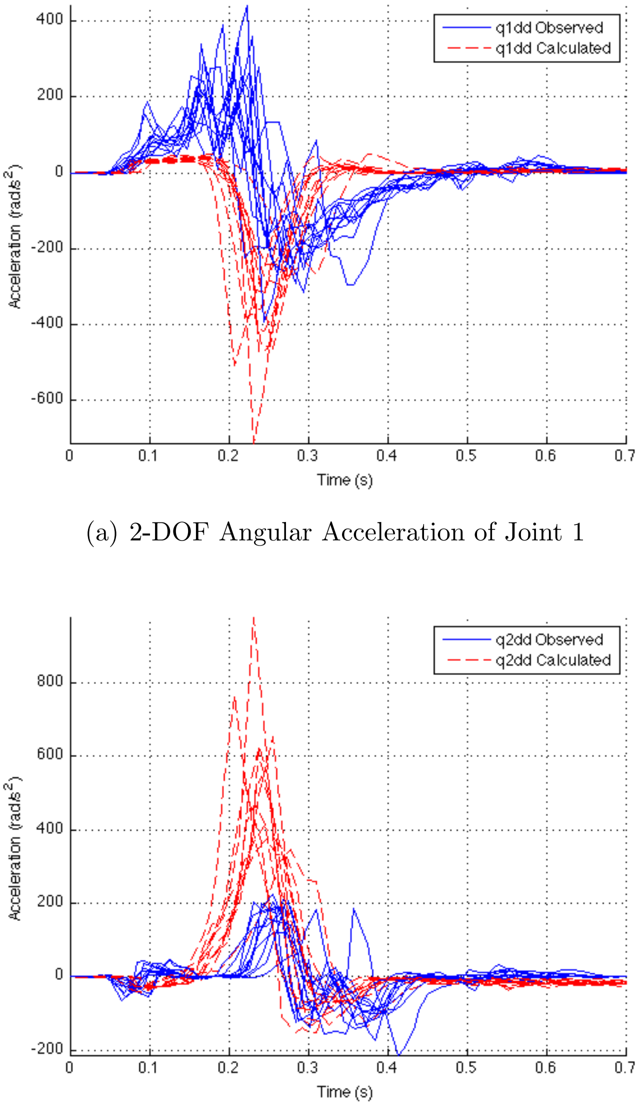
Figure 4.9: 2-DOF Angular Acceleration of Base

nonlinear centripetal and Coriolis forces exhibited by the physical system are more subdued than the model expects.

4.5 2-DOF Trajectory with Thruster Actuation

This experiment performs the same maneuver as the previous 2-DOF experiment, with one thruster exerting force on the spacecraft base to add complexity. In this experiment, the model does expect some acceleration of the system CM since the thruster applies an external force and torque. The thruster was actuated throughout the maneuver. The joint motions for this experiment are shown in Figure 4.11.

A very slight increase in the system CM x-acceleration due to thruster actuation is shown in the close view in Figure 4.12. Figure 4.12(c) shows that no linear y-acceleration is expected. This is because thruster 3 exerts force in the x-direction only. The observed accelerations of the system CM are very close to those exhibited in the previous 2-DOF case. The thruster only produces 0.38 N of force, so it only



(a) 2-DOF Angular Acceleration of Joint 1

(b) 2-DOF Angular Acceleration of Joint 2

Figure 4.10: 2-DOF Angular Acceleration of Joints

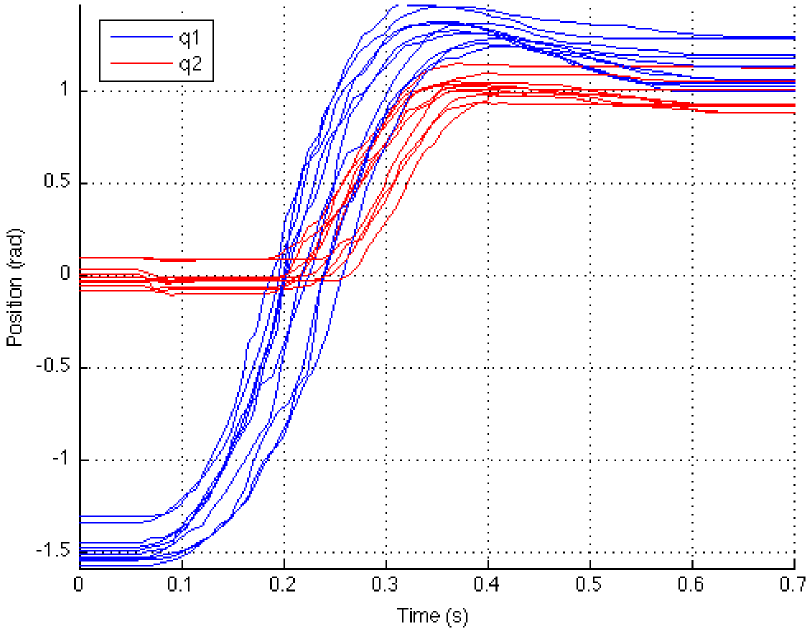
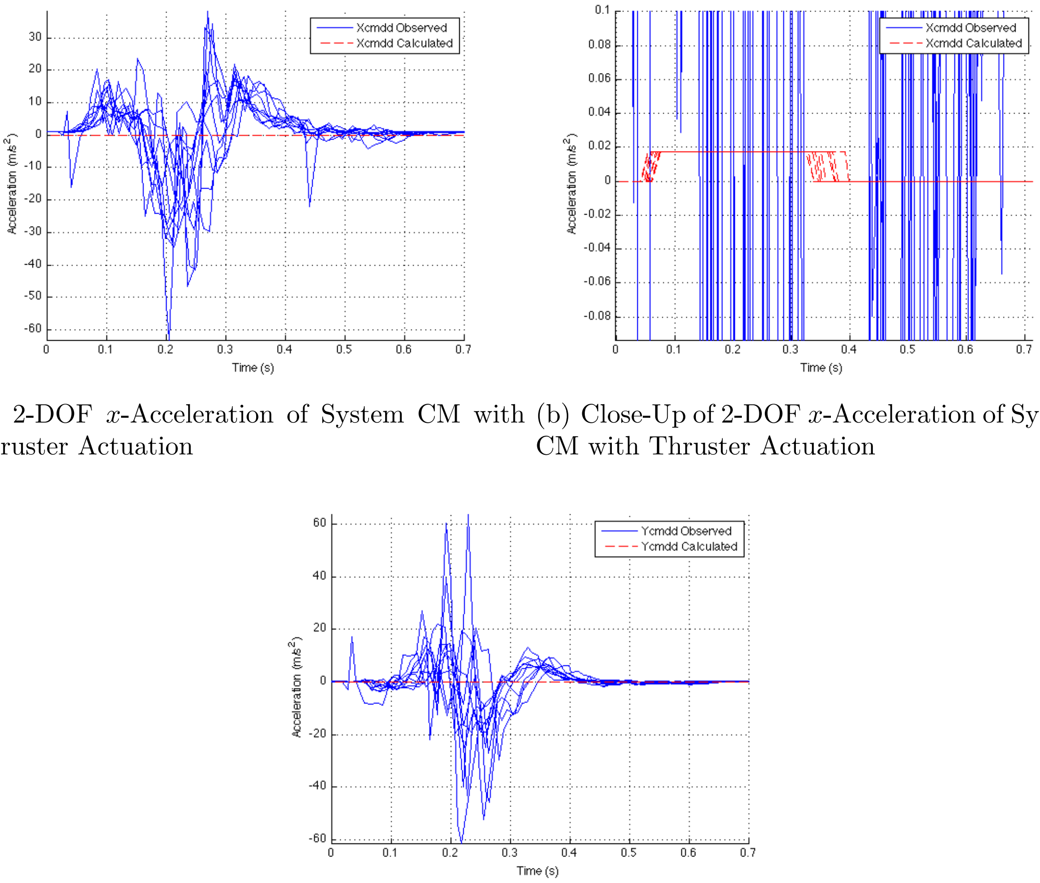


Figure 4.11: 2-DOF Joint Motions with Thruster Actuation

creates 0.017 m/s^2 of acceleration for the 22.09 kg ABV. Additionally, the force it does exhibit is overshadowed by the effects of friction and propellant usage.

The base acceleration results are shown in Figure 4.13. This graph closely resembles the results for base acceleration from the previous 2-DOF experiment. Again the system exhibits less nonlinear acceleration than predicted by the model, as seen by the smaller magnitude of the observed data.

The joint accelerations for this case are shown in Figure 4.14. Again, the experimental data follows the trends established by the model but with different magnitudes. These runs with thruster actuation have similar magnitude ranges for the joint accelerations than the previous 2-DOF experiment, with the exception of one outlying data set.



(a) 2-DOF x-Acceleration of System CM with Thruster Actuation

(b) Close-Up of 2-DOF x-Acceleration of System CM with Thruster Actuation

(c) 2-DOF y-Acceleration of System CM with Thruster Actuation

Figure 4.12: 2-DOF System CM Acceleration with Thruster Actuation

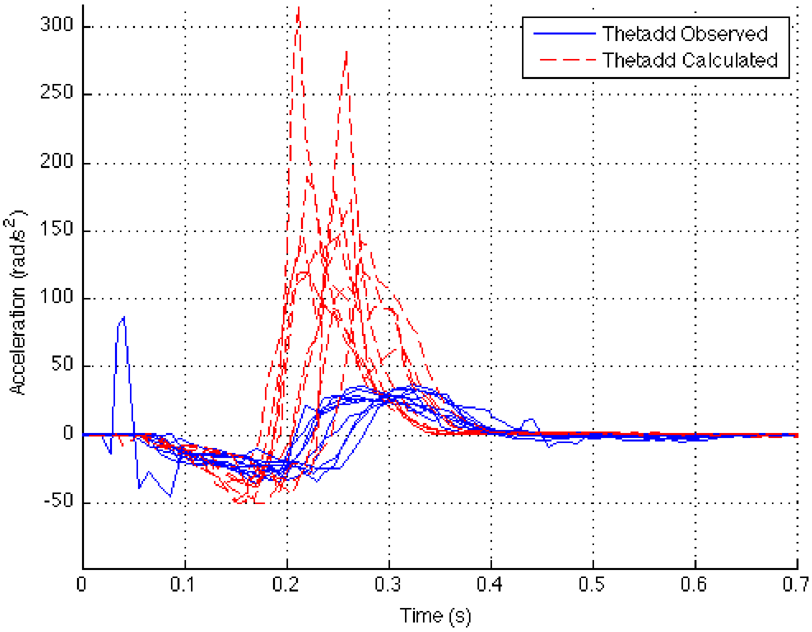
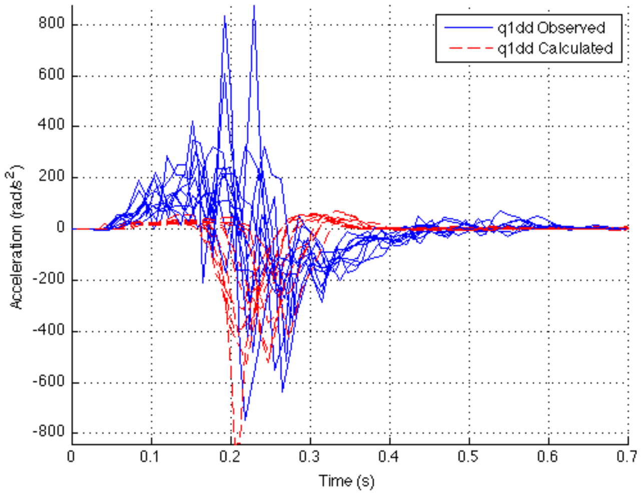


Figure 4.13: 2-DOF Angular Acceleration of Base with Thruster Actuation

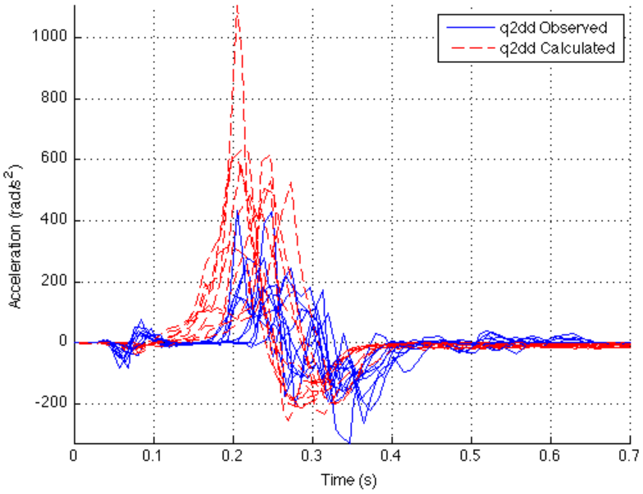
4.6 Alternate Analysis

One disadvantage of the previous analysis is the noise in the differenced acceleration data. An alternate analysis method is to compare the experimental position data for the base frame and each link against an integrated version of the model EOM. Results from this method are shown here for the 1-DOF case. The numerical integration was performed using MATLAB's `ode45` integrator, which uses a variable step Runge-Kutta algorithm. The initial state was the average of the position and velocity over the time period just before the maneuver. Measured torques did not produce the observed accelerations when used as input to the integration, so a sample control law was derived from an experimental case and used as input instead. The inertial position of the base frame was calculated using the integrated state and the known link CM's.

The control law for joint 1 was derived using classical control principles for a



(a) 2-DOF Angular Acceleration of Joint 1 with Thruster Actuation



(b) 2-DOF Angular Acceleration of Joint 2 with Thruster Actuation

Figure 4.14: 2-DOF Angular Acceleration of Joints with Thruster Actuation

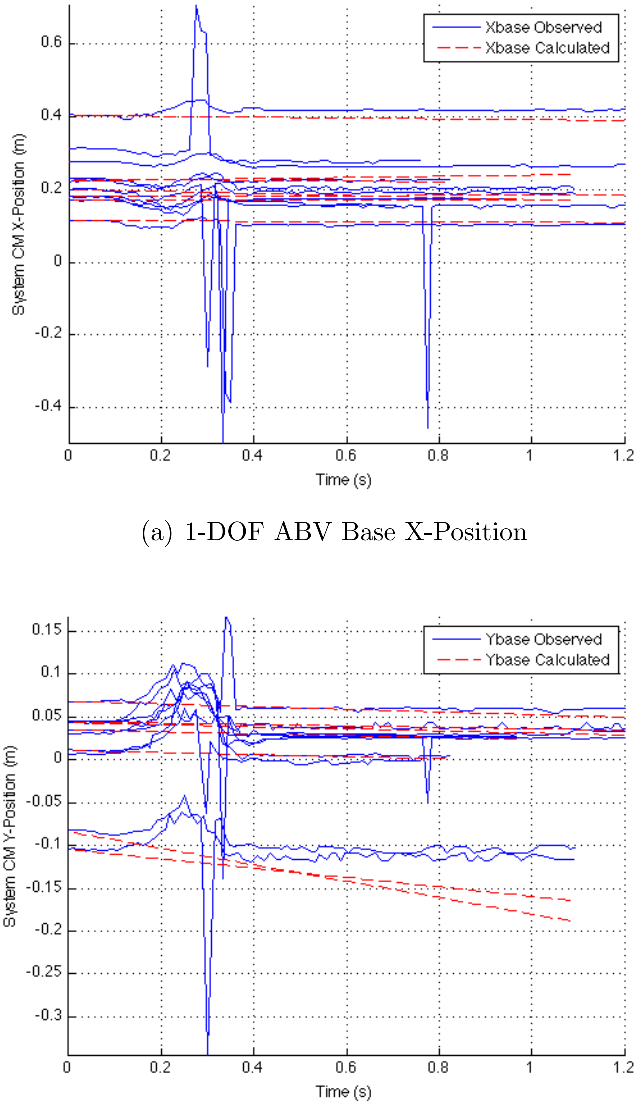
second-order system. The derived control laws are:

$$\tau_1(t) = -7\dot{q}_1 - 125(q_1 - q_{1ss})$$

$$\tau_2(t) = -8\dot{q}_1 - 80(q_2 - q_{2ss})$$

where q_{1ss} and q_{2ss} are the steady-state values for q_1 and q_2.

The results for the 1-DOF (Figures 4.15-4.17) and 2-DOF (Figures 4.18-4.20) experiments are much better when these control laws are implemented. The directly measured data is plotted here: position and orientation of the ABV base in the Table frame as reported by ARToolkit and joint angles as reported by the encoders. There were several data dropouts in the ARToolkit data which were filled in using linear interpolation. The variations in the base orientation are much closer, although not as quite as large in magnitude as predicted by the integrated model. The rise time and steady state values of the joint motions are very much in line with the experimental data. This is partly by design as the steady state values were fed into the controller, but the most important feature is the velocity of the joint motion, as seen by the nearly equivalent slope of the experimental vs. integrated joint motions. These graphs show that the observed data is in fact reasonable with the appropriate torques applied. The main revelation from the exercise is that the proportional-derivative controller being implemented inside the Black Jaguar control modules is not behaving as expected. The logical conclusion is that this controller is saturated due to excessively large commanded changes in position, in these cases about 180 degrees at once. This saturation causes unpredictable motion as seen in many of the graphs in this chapter.



(a) 1-DOF ABV Base X-Position

(b) 1-DOF ABV Base X-Position

Figure 4.15: 1-DOF Inertial Position of ABV Base

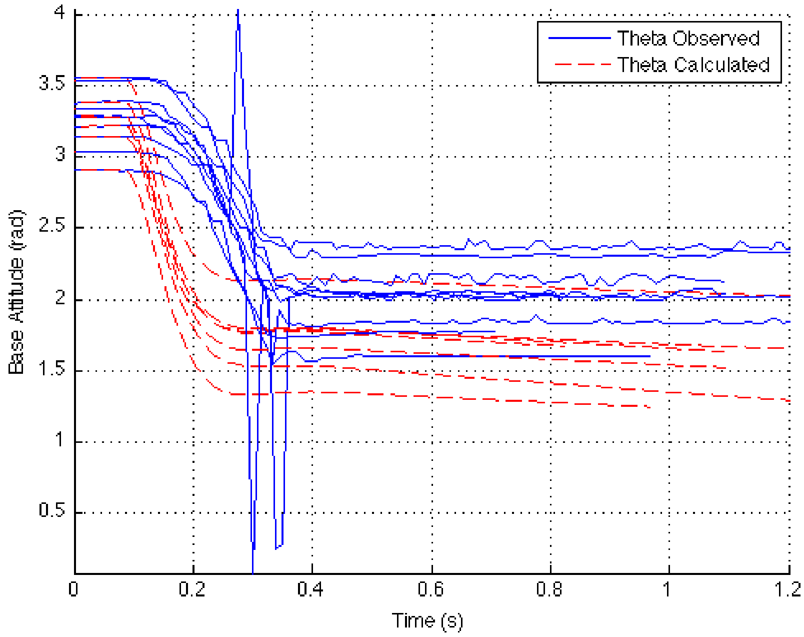


Figure 4.16: 1-DOF Orientation of Base

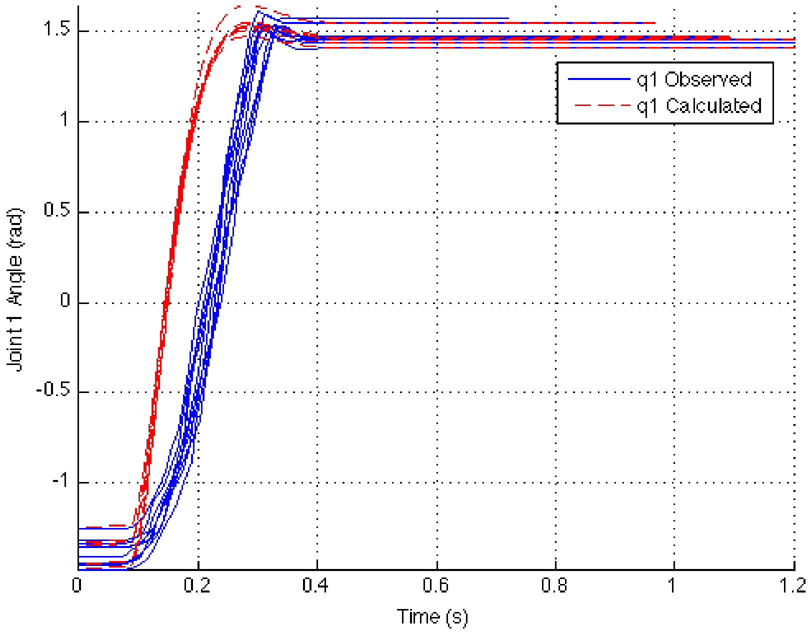
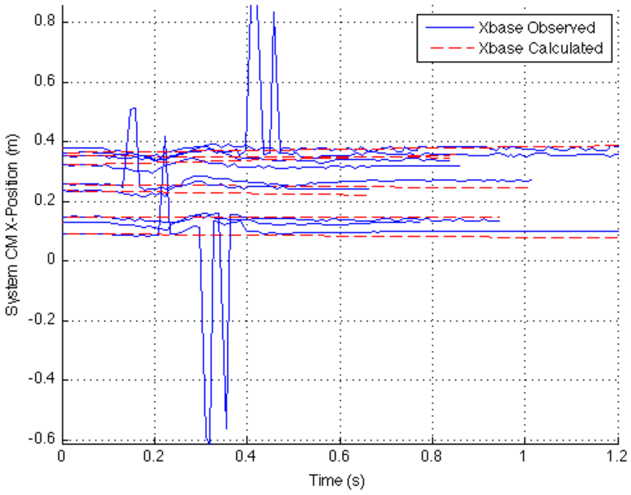
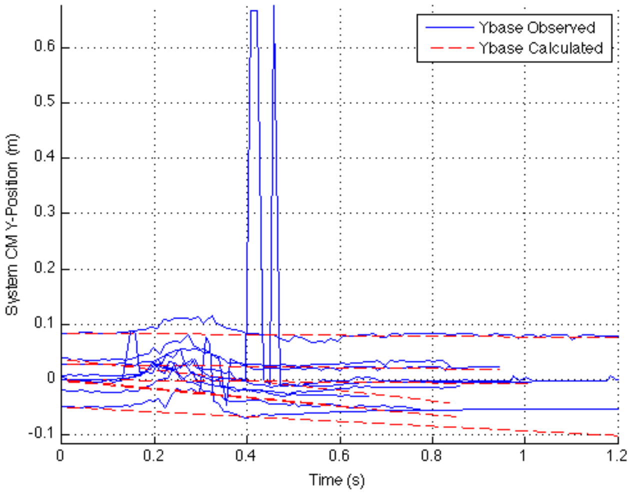


Figure 4.17: 1-DOF Joint 1 Position



(a) 2-DOF ABV Base X-Position



(b) 2-DOF ABV Base X-Position

Figure 4.18: 2-DOF Inertial Position of ABV Base

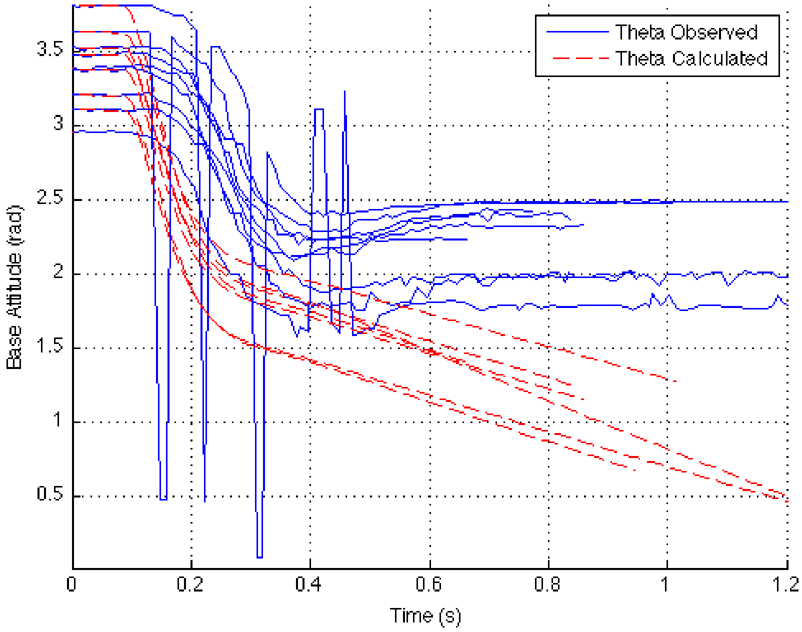
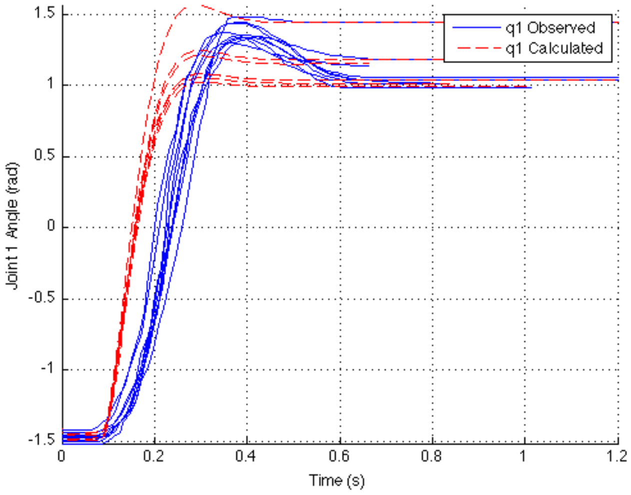


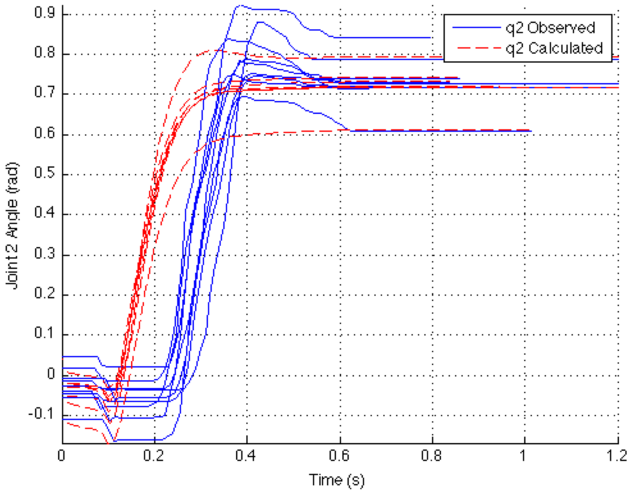
Figure 4.19: 2-DOF Orientation of Base

4.7 Effects of Friction

Friction is inherent in any mechanical system, and this section discusses the effects of two types of friction on the vehicle. First, there is friction in both joints induced by joint motion. This friction includes mechanical friction of the joint components and friction in the motors. Joint friction exerts equal and opposite forces on the adjoining links and therefore induces no net force or torque on the system. The second primary source of friction is the table friction experienced by the five air-bearing pucks on the vehicle. As mentioned before, puck performance is inconsistent and therefore the friction they experience with the table is also inconsistent. The effects of table friction are most clearly seen in the system CM acceleration plots shown in the preceding sections. In each plot, there is a large oscillation in the observed data where the model expects no acceleration. This motion can only be caused by an external force, and friction is the logical source. Improving the puck performance would reduce the effects of table friction.



(a) 2-DOF Joint 1 Position



(b) 2-DOF Joint 2 Position

Figure 4.20: 2-DOF Joint Positions

4.8 Summary of Results

This thesis evaluated three experiments performed with the ABV. The same traits were present in each of the three test cases. The linear acceleration of the system CM was not in line with the model's prediction due to external friction forces from the interaction between the pucks and the air-bearing surface. These external forces overshadowed any system CM motion due to the thruster actuation in the third experiment, so the expected phenomenon was not clearly observed. The joint behavior predicted by the model was present and occurred at the expected time, however the experimental data exhibited different magnitudes of motion than expected. The spacecraft and Joint 2 angular accelerations were smaller than the model expected, suggesting some damping due to environmental forces such as table and joint friction. Conversely, the Joint 1 angular acceleration generally had higher accelerations than the model predicted. This is likely attributable to the model's expected contributions of the nonlinear term $\underline{C}^{+}$.

Additionally, an evaluation of the integrated equations of motion against the observed position behavior showed that the behavior of the Black Jaguar control modules is causing differences between the measured behavior and the model prediction. Implementation of a simulated control algorithm using derived control laws matches behavioral trends much better and more cleanly. The Black Jaguar control implementation deserves further investigation.

5

Conclusions and Future Work

5.1 Conclusions

The research presented in this thesis built from the space manipulator dynamics model developed by Papadopoulos [2], with the focus of experimentally investigating the dynamics behavior of an air-bearing vehicle with a high-inertia, high-rate manipulator attached. A low-cost air-bearing testbed was constructed and used for the experimental evaluation. Papadopoulos's model was tailored for the ABV to determine the expected behavior. Lastly, data was collected from three experiments to compare the actual and expected system dynamics.

The manipulator used for testing was designed for and installed on an existing air-bearing vehicle developed by the 2011 ENAE 484 class. Mechanical and electrical design for the manipulator, as well as implementation of additional sensors and extensive software functionality, were all executed in order to construct a testbed suitable for testing the presented dynamics model. The propellant distribution system for the thrusters was replaced with a more reliable design. Lastly, two air-bearing pucks were added to the puck system to support the manipulator links.

The physical parameters for the ABV were determined experimentally, in the case of the base CM, and mathematically, in the case of the manipulator links. Papadopoulos's model was symbolically tailored for the ABV and implemented in the analysis code based on the calculated physical parameters and the measured motion of each link. The model prediction was then compared to the measured behavior of the system in three separate experiments. The 1-DOF maneuver, 2-DOF maneuver, and 2-DOF maneuver with thruster acceleration all exhibited the same trends in observed behavior as compared to model predictions. The system CM moved much more than the model expected, primarily due to friction between the air-bearing pucks and the table. The angular accelerations of each link (including the spacecraft base as Link 0) followed the behavioral trends expected by the model, however the magnitudes did not align. Again, this can be attributed to unmodeled environmental effects such as friction with the air-bearing surface and within each joint. Lastly, uncertainty in the physical parameters of the system may contribute to some of these differences.

A second analysis method was used to utilize the directly measured data in comparison with the numerical integration of the system EOM. This method showed that the torques derived from the current sensor measurements were not nearly high enough to produce the observed joint motions. A representative control law was derived from the exhibited behavior, and implemented in the integrated simulation. This simulated control law produced expected behavior much closer to what was actually seen in the experiments. This conclusion indicates that much better understanding of the Black Jaguar control modules is required before the system can be used for reliable controls testing.

5.2 Future Work

5.2.1 Testbed Improvements

As the manipulator was not part of the original ABV design, there are a number of possible improvements to the testbed that would make it a more robust, leaner system. Firstly, the puck system will have better performance if the vehicle is supported by only three pucks. As it is, the five puck system requires delicate balancing for optimum performance, and this is not only hard to achieve but also changes as carbon dioxide is consumed and the fuel mass changes. One option is to have the base supported by one 50 lb puck and the elbow and wrist each supported by a 20 lb puck as they are now. However, it could be difficult from a balancing standpoint to support the 13 in × 13 in base levelly on one puck since the center of mass changes as operation time increases. Additionally, stability is compromised if the three pucks become colinear. The other option is to redesign the manipulator joints so that no manipulator pucks are necessary. With the current design, an unsupported link carries all of its forces and moments on the gearmotor output shaft which is deleterious for the motor. This too could be difficult to design considering the amount of manipulator mass and inertia required for the DYMAFLEX experiments. Link 2 exerts a moment on Link 1 about an axis parallel to the air-bearing surface, in addition to the moment exerted by the Link 1 mass. It is crucial that the cantilevered manipulator remain horizontal to avoid damaging the air-bearing surface. Conversely, a larger regulator would allow more propellant to flow to the pucks. While this would decrease the amount of time the system could be used at one time, it would allow the pucks to operate at their nominal pressure of 60 psi.

The Black Jaguar control modules currently use a PD controller, which was saturated during the experiments performed for this thesis. In order to understand the limitations of the Black Jaguar, it would be useful to execute interval position commands (e.g. every 5 or 10 degrees). This process would reveal the size of position command the user can execute without saturating the internal control laws, and therefore determine the range of motions that can be expected to respect the configured control laws.

The ABV's electrical system needs to be streamlined. The manipulator and its sensors were added in piece-by piece to accommodate purposes not envisioned during the base vehicle's original design process, and therefore are not fully integrated with the original ABV electrical system. While this does have its advantages, including a power switch for solely the manipulator, there are many exposed wires that make it a delicate system. The vehicle would be much more robust with a more efficient and secure wiring arrangement.

As discussed in Section 4.5, the thrusters do not exert enough force to perform any meaningful experiments with a planned external force on the base. The solenoids are designed for operating pressures up to 100 psi, so if the robustness of the thruster system was improved to accommodate the higher pressure they could exert more force. It is unlikely, however, that this will be enough of a thrust increase to create noticeable acceleration of the system CM. The alternative is upgrading to more powerful solenoids.

The current sensors on the vehicle for this thesis have a fairly small range. More versatile sensors would allow for better data collection and more aggressive manipulator motions. The Black Jaguar motion controllers have very poor internal current sensing and therefore are not suitable for precision controls testing, which requires high fidelity control of the motor current. However, better controllers are often much higher cost. One option is to install the Elmo controllers currently in other SSL manipulators to replace the Black Jaguars.

Since C++ is an object-oriented programming language, there was little impact to the existing software in order to integrate the manipulator functionality. The experimental usefulness of the ABV could be enhanced by implementing a more precise time-keeping and data recording approach, for example by utilizing the real-time functionality of the QNX OS. Creating a separate thread for the `RobotMsg` processing may increase control loop frequency and simplify the existing software implementation.

5.2.2 Testbed Applications

The ABV is intended as a controls testbed for the SSL's DYMAFLEX project. The $80/20^{TM}$ aluminum used for the manipulator links allows the user to easily change the ratio of the link inertias by adding and removing weights. To a control algorithm, this seems like multiple different vehicles and therefore allows for an added dimension to testing. Manipulator links can be added or removed to change the system configuration. The SSL's current areas of interest in controls include adaptive and coordinated control. The ABV is representative of a system that would use each of these control strategies.

References

[1] J. L. Schwarts, M. A. Peck, and C. D. Hall, "Historical Review of Air-Bearing Spacecraft Simulators," *Journal of Guidance, Control, and Dynamics*, vol. 26, no. 4, pp. 513–522, July-August 2003. vii, 6, 7, 8

[2] E. G. Papadopoulos, "On the Dynamics and Control of Space Manipulators," Ph.D. dissertation, Massachusetts Institute of Technology, 1990. vii, viii, 9, 10, 27, 29, 35, 37, 38, 42, 43, 46, 47, 75

[3] J. J. Craig, *Introduction to Robotics: Mechanics and Control*, 3rd ed. Addison-Wesley Publishing Company, Inc., 2005. 4, 20

[4] B. Wie, *Space Vehicle Dynamics and Control*, 2nd ed. American Institute of Aeronautics and Astronautics, 2008. 4

[5] B. Masselink, M. Mulder, M. M. van Paassen, and A. C. In 't Veld, "Design and Evaluation of a Flight Director for Zero and Partial Gravity Flight," in *AIAA Guidance, Navigation, and Control Conference*, no. AIAA 2009-5987, Chicago, IL, August 2009. 5

[6] *Free-Flying Robot Tested on Parabolic Flights: Kinematic Control*, vol. 28, 2005. 5, 6, 7, 9

[7] M. Nohmi, T. Yamamoto, and Y. Takagi, "Microgravity Experiment for Attitude Control of a Tethered body by Arm Link Motion," in *Proceedings of the 2007 IEEE International Conference on Mechatronics and Automation.* Harbin, China: IEEE, August 2007, pp. 3519–3524. 6

[8] M. Nohmi, "Attitude Control of A Tethered Space Robot by Link Motion Under Microgravity," in *Proceedings of the 2004 IEEE International Conference on Control Applications.* Taipei, Taiwan: IEEE, September 2004, pp. 424–429. 6

[9] G. W. Hawkins, "Marshall Space Flight Center's Role in EASE/ACCESS Mission Management," in *Space Construction Conference.* Hampton, VA: NASA/Langley Research Center, August 1986. 6

[10] (2077, April). [Online]. Available: http://dx12.jsc.nasa.gov/site/index.shtml 6

[11] K. Yoshida, "Space Robotics Research Activity with Experimental Free-Floating Robot Satellite (EFFORTS) Simulators," *Lecture Notes in Control and Information Sciences*, vol. 200, pp. 561–578, 1994. 7, 10

[12] K. Yoshida and Y. Umetani, "Control of Space Free-Flying Robot," in *Proceedings of the 29th Conference on Decision and Control.* Honolulu, HI: IEEE, December 1990, pp. 97–102. 7, 9

[13] W. Soh, N. Hamzah, and W. Steyn, "Attitude Determination and Control Subsystem Hardware in the Loop Test on Air Bearing Trolley," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, no. AIAA-2006-6448, Keystone, CO, August 2006. 7, 9

[14] T. Ustrzycki, D. R. Lee, and H. Chesser, "Spherical Air Bearing Attitude Control Simulator for Nanosatellites," in *AIAA Modeling and Simulation Technologies Conference*, no. AIAA-2011-6272, Portland, OR, August 2011. 8

[15] X. Juan, B. Gang, Y. QinJun, and L. Jun, "Design and Development of a 5-DOF Air-Bearing Spacecraft Simulator," in *International Asia Conference on Informatics in Control, Automation, and Robotics.* IEEE, February 2009, pp. 126–130. 8, 9

[16] H. C. Schubert and J. P. How, "Space Construction: an Experimental Testbed to Develop Enabling Technologies," in *Telemanipulator and Telepresence Technologies IV Conference*, Pittsburg, PA, October 1997. 9

[17] Z. Vafa and S. Dubowsky, "On the Dynamics of Manipulators in Space Using the Virtual Manipulator Approach," in *International Conference on Robotics and Automation*, vol. 4. Raleigh, NC: IEEE, March 1987, pp. 579–585. 9

[18] E. E. Vance, "Adaptive Control of Free-Floating and Free-Flying Robotic Manipulators," Ph.D. dissertation, University of Maryland, 1998. 10, 47

[19] B. Liang, Y. Xu, and M. Bergerman, "Dynamically Equivalent Manipulator for Space Manipulator System: Part 1," in *International Conference on Robotics and Automation*, vol. 4. Albuquerque, NM: IEEE, April 1997, pp. 2765–2770. 10

[20] B. Liang, Y. Xu, M. Bergerman, and G. Li, "Dynamically Equivalent Manipulator for Space Manipulator System: Part 2," in *International Conference on Intelligent Robots and Systems*, vol. 3. Grenoble, France: IEEE/RSJ, September 1997, pp. 1493–1499. 10

[21] *Nelson Air 50-lb Flat Pad Air Bearing Drawing*, Nelson Air. [Online]. Available: http://www.nelsonair.com/assets/FP-C-015-DRAWING.pdf 13

[22] *Nelson Air 20-lb Flat Pad Air Bearing Drawing*, Nelson Air. [Online]. Available: http://www.nelsonair.com/assets/FP-C-010-DRAWING.pdf 13

[23] "MAnned Vehicle for Exploration and Research into the Cosmos," May 2011, ENAE 484 Spring 2011 Term Project Final Report. 16

[24] *RDK-BDC24 Firmware Development Package User's Guide*, Texas Instruments, Software/StellarisWare/docs/SW-RDK-BDC24-UG-8555.pdf. 23

[25] H. Kato, M. Billinghurst, I. Poupyrev, K. Imamoto, and K. Tachibana, "Virtual Object Manipulation on a Table-Top AR Environment," in *Proceedings of the International Symposium on Augmented Reality (2000)*, Munich, Germany, 2000, pp. 111–119. 25

[26] *Relative Equilibrium Positions and Their Stability for a Multi-Body Satellite in a Circular Orbit*, vol. 44, 1975. 28

[27] M. Abramowitz and I. A. Stegun, Eds., *Handbook of Mathematical Functions.* New York, NY: Dover Publications, Inc., December 1972. 52

[28] *MMP TM30-12V GP-051 Data Sheet*, Midwest Motion Products. [Online]. Available: http://www.midwestmotion.com/excessinventory/MMP%20TM30-12V%20GP-051.pdf 53