Modeling Human-Spacesuit Interactions

by

Ann L. Frazer

B.S. Aerospace Engineering
University of Virginia, 2001

Submitted to the Department of Aeronautics and Astronautics in partial fulfillment of the requirements for the degree of Master of Science in Aeronautics and Astronautics at the MASSACHUSETTS INSTITUTE OF TECHNOLOGY
June, 2003

© Massachusetts Institute of Technology, 2003. All Rights Reserved.

Author

Department of Aeronautics and Astronautics
May 23, 2003

Certified by

Professor Dava J. Newman
Department of Aeronautics and Astronautics
Thesis Supervisor

Certified by

Professor of the Practice, Jeffrey A. Hoffman
Department of Aeronautics and Astronautics
Thesis Advisor

Accepted by

Edward M. Greitzer
H.N. Slater Professor of Aeronautics and Astronautics
Chair, Committee on Graduate Students
Modeling Human-Spacesuit Interactions
by
Ann L. Frazer
Submitted to the Department of Aeronautics and Astronautics, May 23, 2003, in partial fulfillment of the requirements for the degree of Master of Science in Aeronautics and Astronautics

Abstract
Dynamic simulations can provide an important, cost saving tool for the purpose of Extravehicular Activity (EVA) planning and training. One important shortcoming of current EVA models is that they lack an accurate representation of the significant torques that are required to bend spacesuit joints. The main objective of this thesis is to quantify the interaction between the human and the Extravehicular Mobility Unit (EMU) spacesuit by developing data-driven models of the joint torque characteristics.

An extensive joint torque database was compiled by utilizing an instrumented robot to act as a human surrogate. The EMU spacesuit was installed on the robot and joint torques were measured for a large number of angular trajectories. The measured torque data were then used to derive mathematical hysteresis models of the torque versus angle characteristics of each joint that are appropriate for implementation into dynamic simulations of suited astronauts. A comparison of the model predictions to experimental data showed that the torque models fit the data well, with $r^2$ values greater than 0.6 in most cases.

Thesis Supervisor: Dava J. Newman
Title: Professor of Aeronautics and Astronautics

Thesis Supervisor: Jeffrey A. Hoffman
Title: Professor of the Practice of Aerospace Engineering
Acknowledgements

I'd like to thank a few of the people who have contributed greatly to the completion of this work as well as to the quality of my MIT experience in general.

First and foremost I need to thank my advisors, Dava Newman and Jeff Hoffman: Dava, thanks for giving me the opportunity to come here and work in this field. You are a definite inspiration both as a mentor and, quite frankly, as just a supercool person in general. I really appreciate your guidance, support, and understanding. I owe a huge debt of gratitude to Jeff Hoffman for stepping up to take the reins when Dava set sail. Your direction, wealth of experience, and wisdom are all greatly appreciated- and a huge thanks for all of the thesis edits, especially those last minute ones.

Thanks to all the folks in the Man Vehicle Lab:
To Chuck and Larry for allowing me to be a part of this group.
To Patricia Schmidt for her help during the beginning stages of this research.
To Chris and Phil for being the gurus that you are- thanks for all of your knowledge and willingness to provide help as well as distractions when I needed them.
To Brad for being a great officemate and demonstrating that there can be engineers who actually “see.”

Additionally I’d like to thank all the friends who have kept me distracted and made sure my experience in Boston was a good one. Jennifer, thank you for the constant support and companionship over the past few years; I owe you big time. Neilie and Craig, thanks for keeping me centered and for making me cookies when things got too crazy. Thanks to Travis, Ajit, and Carlos for making sure I got out of my office on occasion and for visiting me when I couldn’t get out of the office. To all the lgbt@mit and Rainbow Coffeehouse folks- it has been a privilege to work with you all during my time here. Thank you for providing me with an outlet through which I could give back to the MIT community.

A big thanks to the Coca-Cola Company, Erasure, Madonna, and Buena Vista Social Club for helping me stay awake for the past few weeks so that I could finish this thing.

And of course, most importantly, I need to thank all of my family back in SC, especially my mom and dad who have continually stood behind me for the past 25 year- I really appreciate all of your support and could not imagine better parents.

This work funded by NASA Grant NAG9-1089
# Table of Contents

1 Introduction................................................................................................................11
   1.1 Motivation................................................................................................... 11
   1.2 Objectives .................................................................................................... 12
   1.3 Road Map...................................................................................................... 12

2 Background................................................................................................................ 15
   2.1 Overview ...................................................................................................... 15
   2.2 Space Suit Basics ....................................................................................... 16
   2.3 Mobility Issues........................................................................................... 19
   2.4 Hysteresis Modeling ................................................................................. 29
   2.5 Summary ..................................................................................................... 31

3 Spacesuit Experiments.......................................................................................... 33
   3.1 Overview ...................................................................................................... 33
   3.2 M. Tallchief .................................................................................................. 33
   3.3 Robot Tests ................................................................................................... 37
   3.4 Human Tests ................................................................................................ 44
   3.5 Results ........................................................................................................ 48

4 Spacesuit Hysteresis Modeling.............................................................................. 57
   4.1 Overview ...................................................................................................... 57
   4.2 Preisach Model Implementation .................................................................. 57
   4.3 Results ......................................................................................................... 74

5 EVA Operations..................................................................................................... 83
   5.1 Operations Overview .................................................................................. 83
   5.2 Relevance ..................................................................................................... 86
   5.3 Human Robotic Synergies ......................................................................... 88
   5.4 Design Recommendations ......................................................................... 92
   5.5 Summary and Conclusions ....................................................................... 93

Bibliography ................................................................................................................. 97

Appendix A Hysteresis Modeling Scripts................................................................. 99

Appendix B Experimental Subject Consent Form .................................................... 113
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Diagram of the PLSS components [Zorpette, 2000]</td>
<td>17</td>
</tr>
<tr>
<td>2.2</td>
<td>Cutaway of spacesuit fabric layers [Zorpette, 2000]</td>
<td>18</td>
</tr>
<tr>
<td>2.3</td>
<td>Diagram of the spacesuit assembly. Courtesy of Hamilton Sundstrand</td>
<td>19</td>
</tr>
<tr>
<td>2.4</td>
<td>Cylindrical spacesuit joint in (a) an upright and (b) bent configuration</td>
<td>21</td>
</tr>
<tr>
<td>2.5</td>
<td>Flat pattern knee joint in a bent and unbent configuration</td>
<td>23</td>
</tr>
<tr>
<td>2.6</td>
<td>Dionne’s EMU torque data</td>
<td>25</td>
</tr>
<tr>
<td>2.7</td>
<td>Menendez’s pressure data for a flat pattern joint [Menendez et al., 1994]</td>
<td>27</td>
</tr>
<tr>
<td>2.8</td>
<td>Hysteresis plots for elbow and shoulder flexion [Schmidt, 2001]</td>
<td>28</td>
</tr>
<tr>
<td>2.9</td>
<td>Simplest hysteresis transducer</td>
<td>30</td>
</tr>
<tr>
<td>3.1</td>
<td>The RSST’s 12 actuated degrees of freedom</td>
<td>34</td>
</tr>
<tr>
<td>3.2</td>
<td>Robot side view with attachment points highlighted</td>
<td>35</td>
</tr>
<tr>
<td>3.3</td>
<td>Robot computer setup</td>
<td>36</td>
</tr>
<tr>
<td>3.4</td>
<td>Pivoted Hard Upper Torso (HUT) used in the experiment</td>
<td>38</td>
</tr>
<tr>
<td>3.5</td>
<td>Robot with and without the spacesuit installed</td>
<td>39</td>
</tr>
<tr>
<td>3.6</td>
<td>Elbow flexion angle trajectory</td>
<td>40</td>
</tr>
<tr>
<td>3.7</td>
<td>Weight induced torque elimination process</td>
<td>43</td>
</tr>
<tr>
<td>3.8</td>
<td>Spacesuit donning process</td>
<td>45</td>
</tr>
<tr>
<td>3.9</td>
<td>Placement of Motionstar sensors</td>
<td>46</td>
</tr>
<tr>
<td>3.10</td>
<td>Tekscan sensor location</td>
<td>47</td>
</tr>
<tr>
<td>3.11</td>
<td>Database coverage</td>
<td>49</td>
</tr>
<tr>
<td>3.12</td>
<td>Hysteresis shapes for humerus rotation and shoulder abduction</td>
<td>50</td>
</tr>
<tr>
<td>3.13</td>
<td>Torque hysteresis shapes for each flexion joint</td>
<td>51</td>
</tr>
<tr>
<td>3.14</td>
<td>Pressure dependence plot for elbow flexion</td>
<td>53</td>
</tr>
<tr>
<td>3.15</td>
<td>Pressure dependence plot for knee flexion</td>
<td>53</td>
</tr>
<tr>
<td>3.16</td>
<td>Walking at two spacesuit pressures</td>
<td>54</td>
</tr>
<tr>
<td>3.17</td>
<td>Illustration of the foot locus task</td>
<td>55</td>
</tr>
<tr>
<td>3.18</td>
<td>Foot locus results both with and without a handhold</td>
<td>56</td>
</tr>
<tr>
<td>4.1</td>
<td>Simplest hysteresis operator</td>
<td>58</td>
</tr>
<tr>
<td>4.2</td>
<td>Summation of three simple hysteresis transducers</td>
<td>59</td>
</tr>
<tr>
<td>4.3</td>
<td>Block diagram of the Preisach model input and output</td>
<td>60</td>
</tr>
<tr>
<td>4.4</td>
<td>Model saturation limits in the alpha/beta plane [Schmidt, 2001]</td>
<td>61</td>
</tr>
<tr>
<td>4.5</td>
<td>Procedure for drawing the S+/S- boundary [Schmidt, 2001]</td>
<td>62</td>
</tr>
<tr>
<td>4.6</td>
<td>First order reversal curve</td>
<td>64</td>
</tr>
<tr>
<td>4.7</td>
<td>Determining alpha and beta values for shoulder flexion data</td>
<td>66</td>
</tr>
<tr>
<td>4.8</td>
<td>Summing and differencing X(a,b) values</td>
<td>67</td>
</tr>
<tr>
<td>4.9</td>
<td>Interpolation geometry for X(a,b)</td>
<td>68</td>
</tr>
<tr>
<td>4.10</td>
<td>Flowchart of hysteresis modeling process</td>
<td>73</td>
</tr>
<tr>
<td>4.11</td>
<td>Plots of a, b, and X for shoulder abduction and humerus rotation</td>
<td>74</td>
</tr>
<tr>
<td>4.12</td>
<td>Plots of a, b, and X for each flexion angle</td>
<td>75</td>
</tr>
<tr>
<td>4.13</td>
<td>Elbow flexion model verification</td>
<td>76</td>
</tr>
<tr>
<td>4.14</td>
<td>Humerus rotation model verification</td>
<td>77</td>
</tr>
<tr>
<td>4.15</td>
<td>Shoulder flexion model verification</td>
<td>77</td>
</tr>
<tr>
<td>4.16</td>
<td>Hip Flexion model verification</td>
<td>78</td>
</tr>
</tbody>
</table>
Figure 4.17: Knee flexion model verification

Figure 4.18: Ankle flexion model verification.

Figure 4.19: Shoulder abduction model verification compared to angle trajectory.

Figure 5.1: Astronauts performing EVA training the NBL. Image courtesy of NASA.

Figure 5.2: Intelsat VI capture simulated using PLAID. Image courtesy of NASA.

Figure 5.3: M. Tallchief command and effector demonstration.
List of Tables

Table 2.1: EMU Materials 17
Table 3.1: Summary of robot test trajectories 42
Table 3.2: Human spacesuit tests performed 48
Table 4.1: \( r^2 \) values for hysteresis model verification 80
Chapter 1

Introduction

1.1 Motivation
Since the first extravehicular activities were performed in 1965, the capability of EVA astronauts to do useful work outside of their spacecraft has steadily progressed. Likewise, our understanding of EVA astronauts’ capabilities and limitations in reduced gravity have also progressed through in-flight experience, experimentation in neutral buoyancy, and tests of spacesuits and EVA tools. Computer models and dynamic simulations are essential tools for analyzing EVA capabilities and have several advantages over physical simulations, including the absence of inherent time and workspace limitations.

One important shortcoming of current EVA models is that they lack an accurate representation of the torques that are required to bend spacesuit joints. Modern spacesuits are designed to move with astronauts, using bearings and constant-volume joints to minimize resistance to motion. However, the suit-induced torques required to perform EVA tasks still have a significant impact on task performance. The torques required to move spacesuit joints are complicated, nonlinear functions of joint position, rate, and motion history. The lack of quantification of these torques can lead to large variations in predicted task performance.

There are several inherent difficulties in trying to measure the torque angle characteristics of spacesuit joints. Although angle measurement is not difficult, it is impossible to directly measure the joint torques in suited human subjects without using invasive instrumentation. The current study avoids this problem by using a 12 degree-of-freedom instrumented robot in conjunction with suited human test subjects to collect a torque database and uses that data to develop predictive models of the spacesuit joint torques.
1.2 Objectives
The focus of this research is to quantify the interaction between the human and the Extra-vehicular Mobility Unit (EMU) spacesuit by developing data-driven models of the human-suit interaction. The specific objectives of this thesis are:

1. Compile a detailed database of the EMU’s joint torque versus angle characteristics using an instrumented robot as a human surrogate.
2. Explore the limitations of the current EMU in terms of locomotion and range of motion.
3. Use hysteresis modeling techniques to develop data driven models of the spacesuit induced joint torques as a function of angular joint position, and verify these models using experimental multi-joint motion data.
4. Explore the relationship between this research and EVA operations, focusing on planning and training techniques.

1.3 Road Map
This introductory chapter provides the motivation and objectives for the work performed in this thesis. The remaining four chapters go on to discuss background, experimental methods and results, spacesuit hysteresis modeling, and applications to EVA operations.

Chapter Two first gives a brief overview of the importance of modeling in EVA analysis. It then describes the EMU spacesuit and the physiological requirements that drive its design. This section is followed by a discussion of the mobility issues associated with working in a pressurized spacesuit, including a review of past studies which have attempted to measure spacesuit induced joint torques. Finally a brief overview of mathematical hysteresis modeling is presented.

Chapter Three gives an overview of the purpose of the experimental phase of this work, followed by an in depth description of NASA’s Robotic Space Suit Tester (RSST). The experimental methodology is then described for both tests involving the RSST as well as human test subjects. The experiments consisted of a robotic phase in which the RSST was used to measure the joint-torque characteristics, as well as a phase which involved
human test subjects to study locomotion and range of motion. The experimental results are then presented and discussed.

Chapter Four deals with the development of mathematical hysteresis models of the joint torque characteristics of the EMU. It starts with a description of the classical Preisach hysteresis model followed by a graphical interpretation of the model which is useful during the implementation process. These sections are followed by a discussion of the actual numerical implementation process used to derive the mathematical models from the joint-torque data. Finally the models are implemented for several joints and compared to experimental data for verification.

Chapter Five discusses this research from an EVA operations point of view. It starts with an overview of EVA operations, focusing on the EVA planning and training techniques. The relevance of the work presented in Chapters 3 and 4 to EVA operations is then discussed. This is followed by a set of future goals for the RSST and plans for updating the robotic setup. Design recommendations are presented in terms of both suit design and EVA operations, and finally, a summary of the research and results is presented.
Chapter 2

Background

2.1 Overview

Extravehicular activity (EVA), or work done outside of the spacecraft or habitat, is costly in terms of human risk, limited opportunities to perform them, and monetary considerations. Therefore, a significant amount of planning goes into designing EVA’s. The majority of this planning is done in virtual reality trainers or in physical simulations where the astronauts perform the tasks either underwater or on a precision airbearing floor. These physical simulations can be very costly and are not always accurate. Over the past few years studies have begun examining computer dynamic simulations as a tool for EVA planning. Schaffner et al. (1999) performed a study which simulated an astronaut attempting to capture the Intelsat VI satellite using a capture bar that was specifically designed for the EVA task. It was shown that given the dynamics of the system it was impossible to capture the satellite in under 6 seconds, even though the astronauts successfully performed the task numerous times underwater and on a three degree-of-freedom air-bearing simulator.

However, it is not enough just to model the human; a comprehensive analysis must also model the spacesuit. Dave Rahn (1997) performed a study that showed that the inclusion of space suit constraints caused significant differences in results of a simulation of an astronaut performing an EVA large mass handling task. The contribution of this thesis is to provide an accurate model of spacesuit mobility characteristics by measuring the joint torques for realistic motions and using these measurements as the basis for a spacesuit model.


2.2 Space Suit Basics

In order to survive the hostile environment of space, astronauts must be outfitted in spacesuits that provide essential life support during EVA’s. The current US spacesuit, the Extravehicular Mobility Unit (EMU), is manufactured by Hamilton Sundstrand and ILC Dover. It is a complex system that allows the wearer to exist self-sufficiently in space for over 8 hours [Kozsloski, 1994].

The primary purpose of the suit is to provide life support to the wearer in the extreme environment of space. The suit’s requirements are based on the physiological needs of the astronaut. It must provide a pressurized environment, breathing oxygen, radiation protection, CO₂ and contaminant removal, thermal regulation, and protection from micrometeorites. The majority of the life support functions are provided by the primary life support system (PLSS), a backpack-type device that attaches to the EMU. The PLSS includes high pressure oxygen tanks, water tanks, fan/water separator/pump motor assembly, sublimator, contaminant control cartridge, oxygen and water regulators, valves, sensors, and communication equipment (see Figure 2.1). A secondary oxygen tank is attached to the bottom of the PLSS and provides approximately 30 minutes of oxygen in case of emergencies [Kozsloski, 1994].
Figure 2.1: Diagram of the PLSS components [Zorpette, 2000].

The layers of fabric that make up the suit’s soft goods are also essential life support components. These 9 layers of fabric and their functions are listed in Table 2.1 [Kozloski, 1994].

Table 2.1: EMU Materials

The nine layers which make up the EMU fabric components, with layer 1 being the outermost and layer 9 being the innermost. Adapted from Kozloski (1994).

<table>
<thead>
<tr>
<th>Layer</th>
<th>Material</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ortho-fabric: Gore-Tex fibers woven together with Nomex and backed with a network of kevlar</td>
<td>Abrasion/Flame resistance (micrometeoroid protection)</td>
</tr>
<tr>
<td>2-6</td>
<td>Aluminized mylar backed with unwoven Dacron</td>
<td>Thermal insulation</td>
</tr>
<tr>
<td>7</td>
<td>Neoprene-coated nylon ripstop</td>
<td>Liner</td>
</tr>
<tr>
<td>8</td>
<td>Dacron woven with primary and secondary axial lines</td>
<td>Restraint and control of longitudinal extension</td>
</tr>
<tr>
<td>9</td>
<td>Polyurethane-coated nylon</td>
<td>Bladder layer for pressurization</td>
</tr>
</tbody>
</table>
The EMU suit assembly is made up of several components that can best be understood by stepping through the process of donning the suit. When preparing for an EVA the astronaut first pulls on a pair of “space long underwear” called the liquid cooling and ventilation garment (LCVG). The LCVG has tubes distributed throughout the garment through which cooling water runs to provide thermal regulation. At this point the communications carrier assembly, or “Snoopy cap” is donned. Next the astronaut pulls on the lower torso assembly (LTA) like a pair of trousers. The LTA consists of a waist bearing, fabric legs, and built in boots [Kozloski, 1994]. The next step for the astronaut is to crouch under the hard upper torso (HUT), reach into the arms, and pull herself into this upper part of the suit. The HUT serves as the main structural element of the suit assembly. It is made of fiberglass and aluminum, and all of the other suit components, such as the LTA, arms, PLSS, and helmet, attach to this central structure. A display and control module (DCM) is mounted on the front of the HUT and contains all of the external fluid and electrical interfaces, controls and displays. The final step is to don the gloves and helmet. This entire assembly, including the life support system, has a total mass of approximately 118 kg (260
lb_m) when fully charged with oxygen and other consumables for EVA [Newman and Barratt, 1997].

**Figure 2.3**: Diagram of the spacesuit assembly. Courtesy of Hamilton Sundstrand.

### 2.3 Mobility Issues

One of the most critical requirements of the suit is to provide an adequate partial pressure of oxygen required for breathing [Abramov et al., 1994]. The partial pressure of O_2 in the alveolar air of the lungs has to be close to its value when breathing under terrestrial conditions. From a physiological point of view the same level of pressure in the suit as in the space vehicle would represent the best design. However most modern space vehicles such as the space shuttle and ISS operate at sea-level atmospheric pressure. A suit pressure
this high would result in considerable design challenges and major mobility constraints. If the current suit was pressurized to 1 atm (101.3 kPa, 14.7 psi) the joints would be nearly impossible to bend. On the other hand if the pressure is too low, there is a higher risk of the astronaut experiencing decompression sickness, or “the bends” if the transition from spacecraft to suit pressure is made without eliminating nitrogen from the astronaut’s blood and joint cavities. Decompression sickness (DCS) occurs when nitrogen traces in the bloodstream expand and create tiny bubbles in the tissue during a sudden decrease in pressure [Newman and Barratt, 1997]. Therefore, an important trade-off exists between physiological and mobility considerations [Abramov et al., 1994].

The EMU has an operating pressure of 30 kPa (4.3 psi). This fairly low pressure is favorable in terms of astronaut mobility, but it requires an extended prebreathe period (up to 3.5 hours depending on the specific protocol utilized), during which astronauts must breath pure oxygen in order to remove nitrogen from the bloodstream and reduce the chance of experiencing the bends or DCS. A mixed-gas suit pressure of greater than 57 kPa (8.3 psi) requires no prebreathe period for a minimal chance of DCS and allows the astronaut to don the suit and immediately perform useful EVA work.

Abramov et al. (1994) presented a study that explores the effect of suit operating pressure on characteristics such as mobility, amount of oxygen available to compensate for leakage, mass, strength and metabolic rate. The study involved measuring properties of the Russian Orlan suit at several different operating pressures. The nominal pressure of the Orlan suit is 40 kPa (5.8 psi). It was shown to that for a pressure of 50 kPa (7.25 psi) there was a degradation in performance and increase in suit mass. The bending moment of several Orlan joints increased by 10-25%, and the hand mobility decreased significantly.
In order to understand the link between pressure, suit design and mobility the work required to bend a joint can be analyzed. The work done during an arbitrary deformation of a suit segment is

\[ W = W_p + W_e \]  

(2.1)

where \( W_p \) is the work done against pressure and \( W_e \) is the work done against the elastic forces in the garment [Iberal, 1970].

Neglecting the work due to the fabric, which is usually small, and instead focusing on the pressure

\[ W = \int_{V_1}^{V_2} (-p)\,dV = -p(V_2 - V_1) \]  

(2.2)

This is the work required to change a volume of gas in a constant pressure process [Newman and Barratt, 1997]. If Equation 2.2 is applied to the bending of a cylindrical shaped joint such as the elbow of the EMU then the initial volume of the joint is the area of the cross-section times the length of the joint (Figure 2.4a).

\[ V_1 = AL = \frac{\pi}{4}D^2L \]  

(2.3)

![Figure 2.4: Cylindrical spacesuit joint in (a) an upright and (b) bent configuration.](image)
Assuming the cross section stays circular when bent and the inner and outer edges (Figure 2.4b) are approximated by circles then the final volume is the area of the cross section times the centerline length.

\[ V_2 = \frac{\pi D^2}{4} \left( L - \frac{D\theta}{2} \right) = \frac{\pi D^2 L}{4} - \frac{\pi D^3 \theta}{8} \]  
(2.4)

Approximating the distance through which the joint activation force acts to be \( \frac{L\theta}{2} \), the force can be calculated as

\[ F = \frac{W}{d} = \frac{p(V_1 - V_2)}{L} = \frac{p\pi D^3}{4L} \]  
(2.5)

With the parameters from the EMU these forces are almost at the crew member’s maximum strength limits and are outside of the limits for waist bending [Newman and Barratt, 1997]. Therefore, it is essential that suit designers keep the volume changes that occur in the suit as small as possible. This can be accomplished by utilizing certain design and tailoring techniques such as incorporating pleats. For example, an elbow joint is constructed such that as the volume of the fabric component decreases on the inner edge of the joint, it increases on the outer edge, figure 2.5 [Newman and Barratt, 1997]. A restraint cord runs down the centerline of the joint in order carry the axial loads produced by internal pressure. This cord keeps the suit arm from lengthening when it is pressurized. These fabric bending joints are known as flat pattern joints and are located between the scye bearing and upper arm bearing, and at the elbow, wrist, hip, knee, and ankle joints.
Bearings are also used in order to facilitate joint rotation in the suit. The EMU has bearings at the interface between the HUT and the arm, which is called the scye, on the upper arm, at the wrist, and at the waist.

2.3.1 Measuring spacesuit torques

One of the most profound effects on the astronaut’s mobility is that of suit-induced torques. These torques, which are mainly caused by volume changes as discussed in the previous section, cause the springback effect that astronauts have to work against in order to bend a joint. To accurately understand and model the mobility characteristics of the spacesuit, suit-imposed joint torques must be quantified. However, it is inherently difficult to directly measure the torques exerted on a wearer by the suit without using an invasive procedure or restricting the occupant’s motion. This problem is tackled in several ways in the literature, all of which involve indirect measurements and estimates of the torques.

The most common method is to measure the torques required to bend an empty spacesuit. Dionne (1991) examines the characteristics of a Shuttle EMU against those of two advanced high pressure spacesuit designs, the AX-5 and Mark III. Each of the suits was
pressurized and the joints were bent by applying a torque to the exterior of the suit. The angular displacement of the joint was then measured for given torques in order to produce torque vs. angle curves. Figure 2.6 presents these torque curves for the EMU. Torques were measured for the shoulder, knee and elbow of each suit.
Figure 2.6: Dionne's EMU torque data.
A similar technique was used on a Russian Orlan suit in order to study the effects of operating pressure on characteristics such as mobility and metabolic rate [Abramov et al., 1994]. Empty, pressurized suit joints were bent by applying a torque to the exterior of the suit. Both the knee and elbow joints were tested at pressures of 30 kPa (4.3 psi) (EMU operating pressure), 40 kPa (5.8 psi) (Orlan operating pressure), 45 kPa (6.5 psi), and 50 kPa (7.25 psi). Abramov concluded that a pressure greater than 40 kPa imposes a degradation of performance and growth of spacesuit mass. Additionally, the extent of the spacesuit specifications degradation is mainly determined by its design and engineering features. In particular, increasing the operating pressure will cause a more considerable increase in the weight of an open or semi-open life support system (LSS) than a closed-loop type. Therefore, optimal engineering solutions must be found which counterbalance astronaut mobility with LSS selection.

Similarly, Menendez et al. (1994) measured the torques in three different types of soft-suit joints which were considered for the European EVA spacesuit. Isolated elbow joints were bent externally and the torques were recorded for five different operating pressures. Figure 2.7 shows the results for an asymmetrical flat pattern joint. The flat pattern joint tested exhibits hysteretic behavior and produces low torques from 20 to 90 degrees. Additionally there seems to be relatively little dependence on internal pressure over this range. At the two extremes of the hysteresis curve, however, the torque varies significantly with pressure.
Figure 2.7: Menendez’s pressure data for a flat pattern joint [Menendez et al., 1994].

Very few studies have been performed in which the torques of an occupied suit were determined. One study in which human subjects were used is reported in Morgan et al. (1994). However, Morgan’s experiment is concerned with the astronaut’s joint strength rather than the actual torques exerted by the suit. The maximum constant velocity strength that the test subject could exert both with and without a spacesuit were measured using a dynamometer. The difference between the suited and unsuited strength can loosely be interpreted as the force exerted by the suit. However, this interpretation is dependent on the assumption that the subject actually exerted the same amount of force at all times in both cases.

A third and more rigorous technique for measuring joint torques combines both human subjects and an instrumented robot as described by Schmidt (2001). Suited humans were asked to perform representative EVA motions while their joint angles were recorded using a 3D video motion capture system. The suit was then installed on a Robotic Space Suit
Tester, which is also used in the current study, and pressurized to 30 kPa (4.3 psi). The motions and angles from the human subjects were then used to drive the robot, and torques were recorded at 11 joints. Figure 2.8 shows the torques for elbow and shoulder flexion. These torques are significantly higher than the empty suit torques of Figure 2.6, indicating the importance of someone actually wearing the spacesuit when measuring joint torque characteristics. This is due to the fact that having a body in the suit reduces the internal gas volume of the spacesuit. Additionally, the body imposes certain hard stops and limitations on bending due to fabric bunching which are not present in an empty suit.

![Hysteresis plots for elbow and shoulder flexion](image)

**Figure 2.8:** Hysteresis plots for elbow and shoulder flexion [Schmidt, 2001].
2.4 Hysteresis Modeling

By inspecting the torques in Figures 2.6-2.8 it can be seen that hysteresis, or cyclical energy dissipation, is a key property of spacesuit joints. When an astronaut flexes a joint from a fully extended position to a fully flexed one and back again, the suit does not simply "bounce back" and return all the previously applied work. A significant amount of hysteresis is observed in a softsuit such as the EMU, which has to be accounted for when modeling spacesuit induced joint torques.

2.4.1 Previous Work

A fundamental description of hysteresis was developed by Preisach in the 1930’s. In an attempt to describe the hysteresis nonlinearities observed in magnetic systems Preisach developed a model that has since been used in a number of fields including piezoceramics and now, spacesuits. One of the most comprehensive accounts of the Preisach model and its implementation is given by Mayergoyz (1991). In addition to the purely mathematical description, Mayergoyz also lays out a geometric interpretation of the Preisach model that greatly facilitates understanding and implementation of the model.

Ge and Jouaneh (1995) used the Preisach model to describe the hysteresis in piezoceramic actuators for tracking control applications. Using a modified Preisach model they were able to predict the response of actuators within 3%. This paper includes one of the clearest accounts of the numerical implementation of the model based on the graphical interpretation set forth by Mayergoyz.

2.4.2 Incorporating the Preisach Model

The Preisach model represents a hysteretic system as a weighted superposition of simple hysteresis transducers, $\gamma_{\alpha\beta}$ Figure 2.9.
These transducers are defined by their input switching values $\alpha$ and $\beta$. For increasing values of input the output follows the bottom branch and saturates at $+1$ and for decreasing input the output follows the top leg, saturating at $-1$. These simple transducers can be summed and scaled in order to cover an entire range of outputs. The mathematical form of the Preisach hysteresis model output for a given input $u(t)$ is

$$f(t) = \int \int_{a \times b} \mu(\alpha, \beta) \gamma_{\alpha \beta}(u(t)) d\alpha d\beta$$  \hspace{1cm} (2.6)$$

where $\mu(\alpha, \beta)$ is a weighting function of the model and $\alpha$ and $\beta$ are the up and down switching values of the input [Doong and Mayergoyz, 1985]. A more thorough account of the Preisach model and its implementation is given in Chapter 4, Spacesuit Hysteresis Modeling.

2.4.3 Space suit models

The Preisach model was first used to model suit mobility by Rahn (1997). Using torque data from Dione's empty spacesuit experiments, Rahn utilized the Preisach model to describe the hysteretic torque characteristics for several of the EMU joints. These mod-
els were then used as predictive tools that were implemented into a dynamic simulation of a suited astronaut. Simulations of an astronaut performing an EVA large mass handling task were run both with and without the spacesuit constraints. Rahn concluded that the addition of the suit induced joint torques had a significant effect on the simulation results. For certain postures the joint work with the suit was as much as an order of magnitude greater than the unsuited work.

A second hysteresis modeling effort was undertaken for the EMU by Schmidt (2001). Rahn’s models were based on relatively limited joint data which came from Dione’s empty suit measurements. Since it was determined that actually having an occupant in the suit when measuring the torques made a significant impact on the results, Schmidt used data from her human and robot trials in order to develop more accurate joint models. These models were then compared to other experimental data in order to test their validity. It was shown that in most cases the models fit the data well, with $R^2 > 0.6$ for elbow flexion, hip abduction, and knee flexion. The models for hip flexion and ankle flexion did not fit as well because a large portion of the human generated joint angles which were used for the verification exceeded the range of the model coefficients.

2.5 Summary
The EMU spacesuit is a complex system which provides the vital life support necessary to allow astronauts to perform useful work outside of their vehicles. One of the major challenges in designing a pressurized suit such as the EMU is that of balancing the physiological requirements with mobility considerations. The internal gas pressure and bulkiness of the fabric oftentimes make the suit joints extremely difficult to bend. Several studies have been performed which attempt to measure these joint torques, including one by Schmidt (2001) that utilized an instrumented robot in conjunction with suited human subjects in order to produce realistic data. These joint torques can be modeled using special mathe-
matical modeling techniques that account for the hysteresis nonlinearities in the spacesuit data.

The contribution of this thesis is to extend Schmidt’s work by developing a complete joint-torque database that covers a larger range of joint angles. The database is then used to develop more accurate hysteresis models with greater angle ranges than those previously developed. Finally the applications of this research to EVA operations is discussed.
Chapter 3

Spacesuit Experiments

3.1 Overview
The purpose of the experimental phase of this work was to quantify the interaction between the spacesuit and the wearer by analyzing the joint torques required to bend an occupied spacesuit. Since joint torques cannot be measured directly from a human inside the spacesuit, an instrumented robotic space suit tester was utilized. The joint torques that were measured using the robot were then applied to develop data-driven mathematical hysteresis models for each joint. These models can be used as tools to predict the torques exerted by an astronaut when he or she performs activities in the spacesuit.

3.2 M. Tallchief
NASA’s Robotic Space Suit Tester (RSST) is an anthropomorphic robot that was custom built by Sarcos Inc. (Salt Lake City, Utah) for NASA under the Small Business Innovative Research (SBIR) program and is currently on loan from NASA to MIT. The RSST is affectionately known as M. Tallchief because its graceful movements are reminiscent of the famous ballerina Maria Tallchief. The robot’s primary purpose is to serve as a surrogate astronaut in order to measure the joint torques exerted by the spacesuit on the human wearer. There are 12 fully actuated degrees-of-freedom on the robot’s right side, and 12 posable joints on the left side, as shown in Figure 3.1.
1. Shoulder flexion
2. Shoulder Abduction
3. Humerus Rotation
4. Elbow Flexion
5. Wrist Rotation
6. Hip Abduction
7. Hip Flexion
8. Thigh Rotation
9. Knee Flexion
10. Ankle Rotation
11. Ankle Flexion
12. Ankle Inversion

**Figure 3.1:** The RSST's 12 actuated degrees of freedom.

The robot is suspended by a crane and is supported by a bolt at the head and a cable that is attached to the back of the torso segment (Figure 3.2). Both of these supports can be adjusted to change the orientation of the robot.
A hydraulic pump provides the actuation of the joints. Hydraulic fluid circulates from the pump, through each joint, and back to the pump. All of the hydraulic lines and electrical cables exit through a hole in the robot’s head, which allows for a spacesuit to be installed and pressurized. The joint deflections are measured via potentiometers at each joint. Additionally, the joints are equipped with strain gauge load cells that measure the torque for each degree of freedom.

The robot is controlled using two computers and an Advanced Joint Controller (AJC) cage. The computer setup consists of a user interface, run on a RadiSys Corporation (Bea-
EPC-5 486 processor, and a controller, run on an EPC-6 386 processor. The two computers are connected through a VME backplane. The EPC-6 receives commands from the user interface, processes them, and sends corresponding commands to and from a low level robot controller, the AJC. The AJC cage consists of 12 circuit boards, one for each joint, that contain analog control loops for position, velocity, and torque (see Figure 3.3).

Figure 3.3: Robot computer setup.

The robot’s joint positions can be controlled manually using knobs on each of the AJC circuit boards or remotely via the Robotic Space Suit Tester Application (RSSTA) program. RSSTA is a Windows 3.1 based application that is run on the EPC-5 computer. Each of the joints can be moved using a slider in the positioning window. Alternatively, multi-joint trajectories can be loaded from pre-programmed files and executed. These trajectories can be created interactively by using the sliders to position the robot and saving a series of
trajectory points. This list of points can then be run at a later time at varying speeds. Trajectories can also be created outside of the RSSTA application and imported, as was the case in the current study.

3.3 Robot Tests

A series of experiments were conducted that utilized M. Tallchief to gather joint torque data on the Extravehicular Mobility Unit (EMU) spacesuit. These tests were designed around obtaining data that would be suitable for implementation into hysteresis models to quantify joint characteristics.

3.3.1 Experimental setup

The spacesuit used in the experiment was a class III EMU provided by Hamilton Sundstrand (Windsor Locks, CT). Class III hardware is not flight qualified, but rather is approved for demonstrations or non-hazardous testing. There are a few slight differences between the suit used in this experiment and the Class I, flight qualified suits currently used on the Space Station and Shuttle. First, Class I suits are generally more rigid than Class III suits because they are usually newer and have been used less. Additionally, the scye bearing, which connects the arm to the HUT, is slightly different on the suit used in the tests. The Class III suit uses what is called a pivoted HUT. In this design, the scye bearing and arm attachment interface that supports the shoulder joint is joined to the rigid HUT through a bellows section and a pivot. This allows the angle of the scye opening to change in order to make the donning process easier. It also permits slight variation in the plane of rotation of the scye bearing during pressurized use of the suit. The newer suit design in use for ISS has a fixed scye bearing rigidly connected to the HUT at a slightly different angle and is called a planar HUT.
In order to protect the suit from any of the hard metal components, the robot was first dressed in a wet-suit, and a plastic cover was placed over the exposed end of the wrist rotation shaft. This protected the suit’s bladder layer from accidentally being punctured. Also, to facilitate the donning process of the spacesuit and to eliminate the need to accurately size both arms and legs, the nonfunctional left arm and foot of the robot were removed.

Several steps were taken to install the spacesuit. First, the robot was lowered to a sitting position on the floor and the bolt and cable that attach it to the crane were disconnected. All of the electrical and hydraulic lines had to be disengaged to install the Hard Upper Torso (HUT). The shoulder latch in the right arm was released, which allowed the upper portion of the arm to be rotated to a vertical position. The arm could then be brought through the sleeve and the HUT was pulled down over the robot’s head, after which the shoulder latch was engaged again. The HUT was attached to the robot with bolts that went through the neck ring of the suit and into the robot’s neck plate. The robot was then reconnected to the crane, hydraulic lines, and electrical cables and raised to its normal hanging position. The Lower Torso Assemble (LTA), without the right boot, was then pulled up
over the legs and attached to the HUT. The final step was to don the right boot. Figure 3.5 shows the robot before and after the suit was donned. As each step was completed it was essential to make sure that the robot’s joints were properly aligned with the joints of the suit. Misalignment can hinder the mobility of the joints causing higher torques to be exerted in order to bend them.

![Figure 3.5: Robot with and without the spacesuit installed.](image)

The suit was pressurized to 30 kPa (4.3 psi) using four scuba tanks. Because of the high leakage rate of air through the hole in the neck area of the robot that allows the hydraulic lines and electrical connections to pass through, each air tank lasts approximately 30 minutes. Therefore experimental runs lasted approximately 2 hours, after which the suit had to be depressurized and the tanks had to be refilled.
3.3.2 Data Collection

The majority of the tests that were performed consisted of moving a single joint through an increasing and then decreasing oscillatory pattern. These tests were specifically designed to collect data that would be used in developing mathematical models of the joint torques. To identify the Preisach model coefficients it is necessary to vary the input between several distinct minima and maxima. In addition, the maximum input and output of the Preisach model is set when the model coefficients are identified. When the model is later implemented (see 4.2.3 Numerical Implementation), if the input is greater than the previously set maximum, the model output will be invalid. Subsequently, during the experiments the robot was driven with computer generated input that moved the joints through numerous minima and maxima and covered the entire range of motion of the joint.

**Figure 3.6:** Elbow flexion angle trajectory.

Complex, multi-joint motions were also studied as representative of natural EVA tasks. These tests included such motions as stepping, walking, and reaching. Analyzing tasks such as walking is critical for determining the types of changes that should be made to
future suits to enable locomotion for planetary surface exploration. The trajectories for these tests were derived from data collected in a previous experiment in which human subjects were asked to perform EVA related tasks while suited in an EMU. Their joint angles and positions were recorded using a video motion capture system.

As mentioned previously, internal pressure plays an integral role in spacesuit mobility. The higher the pressure, the stiffer the suit is and the harder the joints are to bend. This is especially important when considering efficient surface exploration, such as that of the Moon or Mars. In order to study the effect of the internal pressure on the joint torques the joint was driven using the same trajectory at six different pressures: 0, 6.9, 13.8, 20.7, 26.1, and 29.6 kPa (0, 1, 2, 3, 3.8, and 4.3 psi). It should be noted that 26.1 kPa (3.8 psi) was the pressure used in the Apollo era spacesuit and 26.9 kPa (4.3 psi) is the pressure used in the current EMU. This test was performed for the elbow and knee joints, both of which are critical in terms of mobility.

Table 3.1 shows a summary of all of the tests that were performed using the suited robot. The hysteresis identification trajectories were run for seven joints. Different trials were run for which the adjacent joints were set at various angles. The data from these trajectories were used to produce the mathematical hysteresis models of the joints. Additionally, test trajectories were run which moved the joint through motions other than those of the identification trajectories. These were used to collect data with which to validate the hysteresis models.
Table 3.1: Summary of robot test trajectories.

<table>
<thead>
<tr>
<th>Test Type</th>
<th>Description</th>
<th># Of trials</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hysteresis Identification</td>
<td>elbow flexion</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>humerus rotation</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>shoulder flexion</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>shoulder abduction</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>hip flexion</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>knee flexion</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>ankle flexion</td>
<td>4</td>
</tr>
<tr>
<td>Test trajectories</td>
<td>each joint</td>
<td>1</td>
</tr>
<tr>
<td>Knee pressure effects</td>
<td>Pressure (psi)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>3.8</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>4.3</td>
<td>1</td>
</tr>
<tr>
<td>Elbow pressure effects</td>
<td>Pressure (psi)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>3.8</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>4.3</td>
<td>1</td>
</tr>
<tr>
<td>Complex motions</td>
<td>walking</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>stepping</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>low reach</td>
<td>4</td>
</tr>
</tbody>
</table>

3.3.3 Data Reduction

The torque data recorded from the robot includes not only the torques due to the space-suit, but also components due to the weight of the robot’s limbs and the wet-suit. Because there is no record of the robot’s mass properties, an empirical method was utilized to eliminate the torque due to the weight of the robot’s limbs. First, the torque was measured with the suit installed on the robot. This data is represented by the solid blue line in Figure 3.7a. The next step in the process was to measure the torques with only the wet-suit on the robot. This second set of data is a measure of the torques due to both the wet-suit and the weight of the robot. The wet-suit data is represented by the dashed red line in Figure 3.7a.
The two data sets were then aligned and the second set was subtracted from the first to obtain the torques due to the suit alone, Figure 3.7b.

![Graph showing Elbow Flexion Torque with and without Spacesuit](a)

![Graph showing Elbow Flexion Torque due to Suit](b)

**Figure 3.7**: Weight induced torque elimination process.

### 3.3.4 Error analysis

Errors in the data are the result of two sources. The first source is the error in the trajectory angles between the suited and unsuited conditions. Errors in the trajectory following were caused by the significant loads imposed by the spacesuit. As a result of these loads, the amplitudes of some of the joint angles did not reach their fully commanded positions. This effect was especially pronounced in joints such as shoulder abduction and hip flexion where the spacesuit loads are high. The RMS errors between the suited and unsuited robot joint angles range from about 0.5 to 3 degrees.
The second source of error is that of the torque measurement. These errors result from noise, bias and quantization. The calibration factors utilized in the robot software were previously evaluated by subjecting the joints to known torques and comparing them to the torque output from the software. It was determined that the calibrations factor errors were all less than 4%. Noise and quantization effects result in a random error of approximately 0.113 Nm (1 in-lb). Because the RSSTA software stores the torque values as integers in the units of inch pounds, the resolution is the larger of the torques corresponding to either 1 A/D count or 0.113 Nm (1 in-lb). The torque resolution is 0.226 Nm for hip flexion and 0.113 Nm for all other joints [Schmidt, 2001]

3.4 Human Tests

In addition to the robotic experiments, several tests that utilized suited human subjects were performed. These tests were designed to study suited locomotion and range of mobility limits. As we look forward and consider designs for the next generation of spacesuits, locomotion becomes increasingly important. A versatile suit is necessary so that not only can astronauts perform EVA in microgravity, but also in reduced gravity environments such as the Martian surface where they will be required to traverse rocky terrain.

3.4.1 Experimental setup

The same class III spacesuit was used in both sets of experiments. After the robot trials were finished, the suit was sent to Hamilton Sundstrand where it was resized for the human test subjects. It was equipped with a mock-up of the PLSS backpack, which allowed the HUT to be attached to a stand during the donning/doffing process, Figure 3.8. This offset the weight of the garment. The donning procedure began by pulling on the lower torso assembly with the boots attached. Next the subjects crouched beneath the HUT and pushed up into it. The two pieces were then joined together and the communications cap was donned. Finally the gloves and helmet were slid on and connected to the rest
of the suit. At this point the HUT could be released from the stand and the entire weight of the suit was supported by the test subject.

The experiment was approved by MIT’s Committee on the Use of Humans as Experimental Subjects and an informed consent form was obtained from each test participant. The test group consisted of 5 male subjects each of whom were approximately the same height of $183cm \pm 5cm$ ($6ft \pm 2in$) in order to keep from having to resize the suit. One of the subjects was an astronaut with over 25 hours of on-orbit EVA experience, and two other subjects had previously participated in extensive ground based testing of the EMU. The experimental sessions were run by MIT investigators and Hamilton Sundstrand engineers.
who provided test direction, essential life support in the form of breathing gas, cooling water for thermal control, and two-way communications.

A MotionStar position and orientation measurement system (Ascension Technology Corporation, Burlington, VT) was used to record the positions of body segments during the tests. The system consists of nine six-degree-of-freedom sensors and an Extended Range Transmitter. Position and orientation are determined by transmitting a pulsed DC magnetic field that is measured by all sensors being used. From the magnetic field characteristics, each sensor independently computes its position and orientation and sends this information to a host computer. The nine sensors were placed above and below the ankle, knee, hip, shoulder, elbow and wrist joints and oriented such that the z-axis of the sensor corresponded to the vertical axis of the body, Figure 3.9.

![Figure 3.9: Placement of Motionstar sensors](image-url)
In addition to the position data, heart rate and pressure on the surface of the thigh were also recorded. Heart rate was measured using a (Polar USA, Woodbury, NY) heart rate monitor. An I-scan pressure measurement system (Tekscan Inc., Boston, MA) was used to measure the pressure on the surface of the upper right thigh in order to determine possible contact points with the waist bearing, Figure 3.10. The Tekscan system uses a grid of resistive-based sensors in order to measure the pressure over a given surface area. This data is then presented as a color-coded, real-time display on a PC and can be recorded for later review and analysis.

![Tekscan sensor location](image)

**Figure 3.10:** Tekscan sensor location.

### 3.4.2 Data Collection

Seven different tests were performed in order to gather information about the range of motion limits of suited humans. These tests included both leg and arm motions. Each test
was recorded on video that can be synchronized with the motion data in order to better visualize the results. A description of each test is provided in Table 3.2.

**Table 3.2: Human spacesuit tests performed.**

<table>
<thead>
<tr>
<th>Test</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Foot Locus</td>
<td>The subject moved his right foot in an upward spiral motion starting at the bottom with the largest circle feasible and moving up as high as possible. This was repeated 4 times both with and without a handhold.</td>
</tr>
<tr>
<td>Treadmill</td>
<td>The subject walked on a treadmill at a steady pace of approximately 1.5-2 mph for 30 seconds. This test was performed at two suit pressures, 13.8 kPa (2 psi) and 29.6 kPa (4.3 psi).</td>
</tr>
<tr>
<td>Foot Height</td>
<td>Subject raised his right foot as high as possible. This was repeated 4 times, both with and without a handhold.</td>
</tr>
<tr>
<td>Step</td>
<td>Subject walked onto and over a 6 inch step. This was repeated three times.</td>
</tr>
<tr>
<td>Low Reach</td>
<td>Subject bent forward and made the lowest mark possible on a sheet of grid paper.</td>
</tr>
<tr>
<td>Hand Reach</td>
<td>The Subject used his hand to trace the largest planar envelope possible at three heights: head level, chest level, and waist level.</td>
</tr>
<tr>
<td>Task Board</td>
<td>The subject was timed completing two EVA tasks: untightening and retightening a bulkhead connector and connecting and disconnecting an electrical cable.</td>
</tr>
</tbody>
</table>

**3.5 Results**

**3.5.1 Robot Experimental results**

Figure 3.11 shows the data coverage of the experiments compared to the Spacesuit design specifications. The yellow line is the coverage from the current set of experiments and the other three lines are from previous experiments performed by Schmidt (2001). It should be noted that not all of the angles included in the suit specifications are actually attainable by humans. This coverage represents the most complete published database of joint torque data collected to date.
As seen in Figure 3.11 torques were measured at seven separate joints. Figures 3.12 and 3.13 present examples of the different shapes and magnitudes of the torque versus angle characteristics demonstrated by each joint. Each of the plots represents one of the hysteresis identification trials listed in Table 3.1. These data have been processed in order to remove the torque due to the weight of the robot’s limbs. The input trajectories from which each of these plots was produced consists of an increasing then decreasing oscillatory pattern such as the one shown in Figure 3.6. Therefore each plot is made up of 18 minor hysteresis loops overlaid on each other.
Figure 3.12: Hysteresis shapes for humerus rotation and shoulder abduction.
Figure 3.13: Torque hysteresis shapes for each flexion joint.

The type of joint involved plays an important role in the torque-angle characteristics exhibited. For example, the knee and elbow plots are very similar in shape. Both of these are cylindrical flat-pattern joints which bend in only one degree of freedom. Recall that flat patterns joints have pleats on one side that open as the joint is bent while the pleats on
the other side of the joint collapse in order to maintain a somewhat constant volume. Restraint chords run down the side of the knee and elbow joints to prevent them from lengthening when the suit is pressurized. This type of joint is well suited for single axis hinge joints such as the knee and elbow, but are not well suited for multiple degree of freedom joints such as the shoulder and hip. The sharp peaks in the torque at the angle extremes can be explained by the high degree of fabric bunching at the back of the joint when it is bent as well as the high degree of gas compression.

Likewise, the ankle flexion, hip flexion, shoulder flexion, and shoulder abduction joints exhibit fairly similar torque versus angle shapes. These joints consist of flat pattern pieces as well, however, the shoulder and hip/waist also contain other components such as bearings which facilitate motion in multiple degrees of freedom.

On the contrary the humerus rotation joint exhibits a slightly different shape. The torques are small compared to the other joints and there are not sharp peaks in the torque at the angle extremes as there are with the other joints. This is because simply rotating the arm inside the suit does not produce nearly as much volume change or fabric bunching as bending does.

Figures 3.14 and 3.15 display the pressure dependence of the joint torques for the knee and elbow. As the pressure is increased the joint torques imposed on the subject are higher. This is especially noticeable at the angle extremes where the torque increases quickly for the higher pressure data. Additionally the attainable angle range decreases as the pressure rises. There is not a significant difference between the torques at 3.8 psi (Apollo suit operating pressure) and 4.3 psi (EMU operating pressure).
3.5.2 Human Experimental Results

Unfortunately the interaction of the sensors with the magnetic field of the treadmill motor was too great to deduce significant results from the Motionstar data for the walking trials. However, video footage and subjective comments showed that the participants found it significantly easier to walk at the lower pressure. They noted that they had to take
short quick steps at the higher pressure, whereas they were able to take longer strides at the lower pressure. Subjects also noted the hard stop induced by the waist bearing during the walking, stepping, and foot height tasks. This was apparent in the tekscan data as a line of high pressure against the sensors. Figure 3.16 shows part of the video footage for one walking stride at each of the two pressures. The subjects were able to move more easily and take longer strides at the lower pressure.

13.8 kPa (2 psi)

![Image 1](image1)

29.6 kPa (4.3 psi)

![Image 2](image2)

**Figure 3.16:** Walking at two spacesuit pressures.

Results from the foot locus and foot height tasks showed that the subjects were able to lift their foot significantly higher when they used a handhold. For the foot height test the average maximum height achieved with a handhold was 39.8 cm and 32.5 cm without.
This suggests that the use of a device such as a walking stick could increase mobility during locomotion on the Moon or Mars.

An illustration of the foot locus task is shown in Figure 3.17. The test subjects were asked to make an upward spiral motion starting at the bottom with the largest circle feasible and moving up as high as possible. This was repeated 4 times with and without a handhold. Figure 3.18 shows results from the foot locus task for one of the test subjects, both with and without a handhold. The large loop represents the lowest loop achieved and the small loop is the highest. With a handhold the highest loop achieved was at a height of 17.5 cm and without the handhold it was 12.4 cm. The coordinate frame in these plots is centered on the stationary left foot.

![Figure 3.17: Illustration of the foot locus task.](image)
Figure 3.18: Foot locus results both with and without a handhold.
Chapter 4

Spacesuit Hysteresis Modeling

4.1 Overview

Using the joint torque data collected during the experimental phase of this study, mathematical models were developed that allow prediction of the torques exerted by the spacesuit for any given angle trajectory that is within the bounds of the model. This thesis contributes a model that describes the hysteresis nonlinearities in the data for seven arm and leg joints: shoulder flexion, shoulder abduction, humerus rotation, elbow flexion, hip flexion, knee flexion, and ankle flexion. In general, one of the most widely applied hysteresis models is the Preisach model, which was developed to describe the hysteretic characteristics of magnetism. The model was later expanded to a purely mathematical form that makes it applicable to a large number of hysteretic systems. The following sections give an overview of the Preisach model along with a detailed account of the identification and implementation procedures. Finally, results from the current spacesuit data are fit to an improved hysteresis model.

4.2 Preisach Model Implementation

A mathematically rigorous version of the Preisach model was developed by the Russian mathematician M. Krasnoselskii for numerical implementation [Krasnoselskii and Pokrovskii, 1989]. However, Krasnoselskii’s implementation required the numerical evaluation of double integrals, a time consuming procedure. Doong and Mayergoyz (1985) proposed an implementation that is based on explicit formulas for the integrals and, as such, avoids the actual evaluation of the double integrals. Another advantage of Mayergoyz’s implementation process is that the experimental data used in the identification of the model is directly involved in these explicit formulas. The following sections follow
Mayergoyz’s treatment of the Preisach model for consideration and final implementation into a spacesuit model.

4.2.1 Classical Preisach Model

As stated in the background section the Preisach model can be represented by a weighted superposition of the simple hysteresis operator $\gamma_{\alpha\beta}$ in Figure 4.1. This operator is simply a rectangular loop in the input, output domain and can take on one of two output values, -1 and +1. The values $\alpha$ and $\beta$ represent the “up” and “down” switching values, respectively. To obtain more complicated hysteresis transducers with non-unity outputs, these simple operators can be summed. Figure 4.2 shows the summation of three simple transducers with switching values $(\alpha_1, \beta_1)$, $(\alpha_2, \beta_2)$, and $(\alpha_3, \beta_3)$.

![Figure 4.1: Simplest hysteresis operator.](image-url)
Then the overall output of the Preisach model is

$$f(t) = \int \int \mu(\alpha, \beta) \gamma_{\alpha \beta} u(t) d\alpha d\beta$$

(4.1)

where $\mu(\alpha, \beta)$ is a weight function and is characteristic of the hysteresis transducer. A block diagram of the process by which the simple transducers are weighted and superimposed is shown in Figure 4.3.
4.2.2 Geometric Interpretation

The actual numerical implementation of this model is rather complex and is greatly facilitated by a graphical representation [Mayergoyz, 1991]. This interpretation is based on the fact that there is a one-to-one correspondence between the operators, $\gamma_{\alpha\beta}$, and the points $(\alpha, \beta)$ in the half plane $\alpha > \beta$. That is, each point in the $\alpha\beta$ plane can be identified with only one particular $\gamma$-operator whose "up" and "down" switching values are equal to the coordinates $(\alpha, \beta)$ at the point.

Consider the triangle in Figure 4.4 that is bounded by the lines $\alpha = \beta$, $\alpha = \alpha_0$, $\beta = -\alpha_0$, where $\alpha_0$ is the saturation limit of the output. The weighting function, $\mu(\alpha, \beta)$, is defined as a finite function at every internal point and is equal to zero outside of the triangle.
The output of the model is then Eq. 4.1 integrated over this triangular region. To perform the integration the triangle can be divided into two regions. The area $S^+$ corresponds to the region in which the hysteresis operators, $\gamma_{\alpha\beta}$, are in the "up" or $+1$ position. The area $S^-$ is the region in which the operators are in the "down" or $-1$ position, Figure 4.4. Separating Equation 4.1 into $S^+$ and $S^-$ pieces and substituting $+1$ or $-1$ for the output, $\gamma_{\alpha\beta} u(t)$, the equation becomes

$$f(t) = \int_{S^+} \int \mu(\alpha, \beta) d\alpha d\beta + \int_{S^-} \int \mu(\alpha, \beta) d\alpha d\beta$$

(4.2)

Once the boundary between the two regions is known this equation can be evaluated to determine the output, $f(t)$. The boundary is constructed from the input history, $u(t)$. A simple set of rules can be applied to the input in order to draw the boundary:

1. The boundary starts at $\alpha=\alpha_0$ segment if the initial input is descending and the $\beta=-\alpha_0$ segment if the initial input is ascending
2. Subsequent boundary segments are drawn horizontally or vertically depending on whether $u(t)$ is increasing or decreasing
horizontal line segment at $\alpha=u$ for increasing input
vertical line at $\beta=u$ for decreasing input
3. A line segment becomes obsolete and is removed if its $\alpha$ value is less than the $\alpha$ value of a later segment that has the same $\beta$ value
4. A line segment becomes obsolete if its $\beta$ value is less than the $\alpha$ value of a later segment that has the same value
5. The last line segment ends at $\alpha=\beta$

As this procedure is continued a complex staircase of all of the input extrema is produced in the triangular region. The horizontal and vertical links meet at the vertices with $(\alpha, \beta)$ coordinates that correspond to the past values of local minima and maxima in the input, $u(t)$. Inherent in this method is the "wiping out" property of the model. If at any point $u(t)$ increases above the past upper extrema, several of the vertices are wiped out. Therefore large input swings effectively reset the system. Figure 4.5 shows an example of the procedure for a monotonically decreasing input.

Figure 4.5: Procedure for drawing the $S+/S-$ boundary [Schmidt, 2001].
4.2.3 Numerical Implementation

As noted in section 4.2.1 the evaluation of the double integrals in Equation 4.2 can be circumvented by using explicit formulas based on experimental data. Mayergoyz (1991) showed that the weighting functions could be determined from a set of first order reversal curves. Reversal curves are obtained when the input is cycled through a number of minima and maxima. If the input and output are plotted against each other, the reversal curves attach to the major ascending branch and each is formed when a increase along this branch is followed by an input decrease.

Consider the reversal curve pictured in Figure 4.6a. As the input to the hysteresis transducer begins to increase from $u_0$ to $u_1$ a horizontal line segment moves up the triangle in the $\alpha\beta$ plane. At the point where the input is equal to $u_1$ the $S^+/S^-$ boundary in the $\alpha\beta$ plane is a horizontal line at $\alpha=u_1$, Figure 4.6b. The input then decreases from $u_1$ to $u_2$ and a vertical segment is added to the boundary at $\beta=u_2$, Figure 4.6c.
According to Eq. 4.2 the output $f(u)$ is the double integral of $\mu$ taken over the $S^+$ region minus the double integral of $\mu(\alpha, \beta)$ taken over the $S^-$ region. Using this result the difference between the output at $u=u_1$ and $u=u_2$ is

$$f(u_2) - f(u_1) = F_{\text{max}} - 2 \int_{S_i^-} \mu(\alpha, \beta) d\alpha d\beta - \left[ F_{\text{max}} - 2 \int_{S_i^+} \mu(\alpha, \beta) d\alpha d\beta \right]$$

(4.3)

$$f(u_2) - f(u_1) = 2 \int_{S_i^-} \int_{S_i^+} \mu(\alpha, \beta) d\alpha d\beta$$

where $F_{\text{max}}$ is the double integral of $\mu$ over the entire triangle. Therefore the difference between the two output values is equal to the shaded triangle shown in Figure 4.6 with vertices $(u_1, u_2), (u_1, u_1), (u_2, u_2)$. If a function $X(\alpha_i, \beta_j)$ is defined as
\[ X(\alpha_i, \beta_j) = (f(u_i) - f(u_j)) \tag{4.4} \]

then by comparing the incremental changes taking place in the \( \alpha \beta \) plane to those on the first order reversal curves, it can be shown that the double integral of the \( \mu(\alpha, \beta) \) over an arbitrary triangular region is simply 1/2 times the value of the function \( X \) evaluated at the point corresponding to the vertex \( (\alpha_i, \beta_j) \).

The numerical implementation proposed by Doong and Mayergoyz (1985) and refined by Ge and Jouanneh (1995), utilizes the function \( X(\alpha_i, \beta_i) \) directly to obtain the numerical expression for the output \( f(t) \) instead of solving for the weighting function \( \mu(\alpha, \beta) \). The method avoids differentiation and integration by calculating the integrals of \( \mu(\alpha, \beta) \) over a mesh of triangles in the \( \alpha, \beta \) plane from output differences as in Eq. 4.3. Sums and differences of the triangular integrals are then used to determine the output for any input history.

The actual identification of the model coefficients was performed using the Matlab script idx.m. As an example, consider the shoulder flexion hysteresis data shown in Figure 4.7. The data has been preprocessed in order to remove the torque contribution due to the weight of the robot. The script first determines the location of each of the maximums. These are designated as the alpha values. The angles of the branches attached to that maximum are the beta values corresponding to that alpha value (Figure 4.7). The quantity \( X(\alpha, \beta) \) is then the torque at the current \( \alpha \) minus the torque at the current angle value, \( \beta \). A vector of \( \alpha \) values and matrices containing the \( \beta \) and \( X \) values are saved in a file which is used in the implementation of the model.
Figure 4.7: Determining alpha and beta values for shoulder flexion data.

If X(α,β) were known at every point in the triangular region then the output to the Preisach hysteresis model could be constructed by simply summing and differencing the X values. Recall that the S+\/S- boundary is drawn according to the rules in Section 4.2.2, Geometric Interpretation. Then, as illustrated in Figure, if the S+\/S- boundary has vertices (α_1, β_1), (α_2, β_2), ..., (α_n, β_n), then the integral of μ(α,β) over S+ is

\[
\int \int_{S^+} \mu(\alpha, \beta) \, d\alpha \, d\beta = \frac{1}{2} \sum_{i=1}^{n} (-1)^{i+1} X(\alpha_i, \beta_i)
\]  

(4.5)
Figure 4.8: Summing and differencing $X(\alpha, \beta)$ values

For points other than those which come from the experimental data, $X(\alpha, \beta)$ can be interpolated from the grid with a nearest-neighbors interpolation method. $X(\alpha, \beta)$ is interpolated from the four experimentally determined points around the desired $\alpha, \beta$ location. The interpolation uses a weighted average of the points based on the distance to the point of interest, Figure 4.9. $X(\alpha, \beta)$ is given by

$$
X(\alpha, \beta) = \frac{(d_2 d_3 d_4) X_1 + (d_1 d_3 d_4) X_2 + (d_1 d_2 d_4) X_3 + (d_1 d_2 d_3) X_4}{d_2 d_3 d_4 + d_1 d_3 d_4 + d_1 d_2 d_4 + d_1 d_2 d_3}
$$

(4.6)

In some cases, where the interpolation point is near the $\alpha=\beta$ line, there are only three surrounding experimental data points. In this case the $X(\alpha, \beta)$ is

$$
X(\alpha, \beta) = \frac{(d_2 d_3) X_1 + (d_1 d_4) X_2 + (d_1 d_2) X_3}{d_2 d_3 + d_1 d_4 + d_1 d_2}
$$

(4.7)
The function \texttt{xmodel.m} reads in the model coefficients that were saved from \texttt{idx.m}. The input to the function is an angular trajectory and the output is the torque predicted by the model. For each time step in input the script \texttt{ab.m} is called in order to determine the $S^+/S^-$ boundary according to the boundary drawing rules in section 4.2.2. Once the boundary is known the values of $X(\alpha,\beta)$ at the vertices are interpolated from the data points via the function \texttt{interpx.m}. The $X(\alpha,\beta)$ values are then added and subtracted according to Eq. 4.5 in order to determine the torque output for the current time step.

### 4.2.4 Error Analysis

Since the model is determined by experimental data, any random errors in the data propagate to the model. Errors in the model coefficients $X(\alpha,\beta)$ come from two sources. The random errors in the measured angles contribute to errors in $\alpha$ and $\beta$, and errors in the measured torques lead to errors in $X$. Because the $X(\alpha,\beta)$ values are calculated from differences in the measured torques, the variance in $X$ due to torque errors is

$$\text{var}(X_r) = \text{var}(\text{torque}_1 - \text{torque}_2) = 2\text{var}(\text{torque})$$

(4.8)

The deviations in $X$ due to changes in $\alpha$ and $\beta$ can be approximated by

$$\Delta X = \Delta \alpha \frac{\partial X}{\partial \alpha} + \Delta \beta \frac{\partial X}{\partial \beta}$$

(4.9)
where $\Delta \alpha$ and $\Delta \beta$ are equal to the standard deviation of the angle measurement errors. Therefore the variance in $X$ due to angle measurement errors is

$$
var(X_\alpha) = var(angle) \left( \frac{\partial X}{\partial \alpha} + \frac{\partial X}{\partial \beta} \right)^2
$$

These two error contributions can be combined such that the total variance in $X(\alpha, \beta)$ is

$$
var(X(\alpha, \beta)) = 2var(torque) + var(angle) \left( \frac{\partial X}{\partial \alpha} + \frac{\partial X}{\partial \beta} \right)^2
$$

If all of the errors in the angle and torque values are assumed to be random, then the error in $X(\alpha, \beta)$ at one point would be uncorrelated with errors in $X(\alpha, \beta)$ at each successive point [Schmidt, 2001]. The error in the output of the model is then equal to the sum of the variance in $X(\alpha, \beta)$ over all of the $(\alpha_i, \beta_i)$ points

$$
var(T) = \sum_{i=1}^{n} var(X(\alpha, \beta)) = (2n)var(torque) + var(angle) \sum_{i=1}^{n} \left( \frac{\partial X}{\partial \alpha_i} + \frac{\partial X}{\partial \beta_i} \right)^2
$$

The variance in the output of the model is calculated in the matlab script errx.m which implements Eq. 4.12. The error function uses a torque error variance of $1 \text{ Nm}^2$ and an angle error variance of $4 \text{ deg}^2$ which are based on the error estimates of Chapter 3.

4.2.5 Hysteresis Model Example
The flowchart on the following pages illustrates the identification and implementation process for elbow flexion. The process starts with the torque and angle data from the hysteresis identification trials discussed in section 3.3.2, Data Collection. This data is read into the Matlab script idx.m which determines the $\alpha$, $\beta$, and $X$ values and outputs them in a vector and two matrices, respectively. Recall from section 4.2.3, Numerical Implementation, that $\alpha$ and $\beta$ map out the angle values over which the model is defined. Likewise, $X_{ij}$ corresponds to the integral of the weighting function $\mu$ over the triangular region with vertex $(\alpha_i, \beta_j)$ and is calculated by subtracting the torque value at $\beta_j$ from the torque value at...
\( \alpha \). These matrices are then stored in a data file called Elbowmodel.mat, which constitutes the Preisach hysteresis model for elbow flexion.

In order to implement and verify the model we start with an arbitrary set of elbow flexion angle and torque data. The angle data is the input for the model, and the torque data serves as a reference that can be compared to the model output for verification purposes. The particular elbow flexion data shown in the flowchart of Figure 4.10 was obtained by gathering joint angle data from a suited human test subject. The angle trajectory was then used to drive M. Tallchief from which the torques were determined. To implement the model the angle data as well as the model data file (Elbowmodel.mat) are read into the Matlab script, xmodel.m. This program contains three internal scripts, ab.m, interpx.m, and errorx.m whose function are described in section 4.2.3, Numerical Implementation. Each of these scripts can be found in Appendix A. Their combined effect is to determine the predicted torque output corresponding to the angular input. The output of the model consists of these predicted torques plus error estimates at each point. This output is then compared to the experimental torque data to determine the model’s accuracy.
Model Identification

Elbow Flexion hysteresis identification data

![Graphs showing hysteresis data with time and angle on the x-axis and torque on the y-axis.]

Elbowmodel.mat (binary data file)

\[
\alpha = \begin{bmatrix}
\alpha_1 \\
\vdots
\end{bmatrix}
\]

\[
\beta = \begin{bmatrix}
\beta_{11} \\
\vdots
\end{bmatrix}
\]

\[
X = \begin{bmatrix}
X_{11} \\
\vdots
\end{bmatrix}
\]

idx.m Matlab script
Model Implementation

- Experimental Elbow Flexion data from M. Tallchief based on angles from human subjects

- Model Output

- Comparison

Matlab Scripts:
- xmodel.m
- ab.m
- interpx.m
- errorx.m

Figure 4.10: Flowchart of hysteresis modeling process.
4.3 Results

The experimental data were used to determine the mesh of alpha and beta values and the Preisach model coefficients $X(\alpha, \beta)$. Figures 4.11 and 4.12 show the triangular mesh for each of the joints plotted in the $\alpha\beta$ plane. The magnitude of the model coefficients over each incremental area of the triangle is represented by the its color with dark blue representing the smallest magnitude and red representing the highest.

Figure 4.11: Plots of $\alpha$, $\beta$, and X for shoulder abduction and humerus rotation.
Figure 4.12: Plots of $\alpha$, $\beta$, and $X$ for each flexion angle.

These hysteresis models were then verified by experimental data. The data were collected during a previous experiment in which human subjects were asked to perform EVA
related tasks while suited in an EMU. Their joint angles and positions were recorded using a video motion capture system. The angle trajectories were then used to drive the M. Tallchief from which joint torques could be measured and recorded for model verification purposes. Figures 4.13-4.18 show the model torque predictions as well as the experimental data for each joint. A 95% confidence interval is also provided in each plot. Each plot gives the model predictions for data from three different test subjects. Because the data came from human subjects, each angle trajectory was different. It should be noted that the test data all come from multi-joint human motions. This ensured that the model could predict for realistic movements and not simply single joint motions.

![Figure 4.13: Elbow flexion model verification.](image-url)
Figure 4.14: Humerus rotation model verification.

Figure 4.15: Shoulder flexion model verification.
Figure 4.16: Hip Flexion model verification.

Figure 4.17: Knee flexion model verification.
Figure 4.18: Ankle flexion model verification.

The models fit very well for most of the joints with $r^2$ values greater than 0.6 for shoulder flexion, elbow flexion, humerus rotation, knee flexion, ankle flexion, and hip flexion. However shoulder abduction did not predict well because many of the angles in the experimental data were outside of the range of the model.
Table 4.1: $r^2$ values for hysteresis model verification

<table>
<thead>
<tr>
<th></th>
<th>Subject 1</th>
<th>Subject 2</th>
<th>Subject 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Elbow Flexion</td>
<td>0.5135</td>
<td>0.5185</td>
<td>0.8252</td>
</tr>
<tr>
<td>Humerus Rotation</td>
<td>0.7353</td>
<td>0.562</td>
<td>0.6007</td>
</tr>
<tr>
<td>Shoulder Flexion</td>
<td>0.8731</td>
<td>0.8363</td>
<td>0.3878</td>
</tr>
<tr>
<td>Knee Flexion</td>
<td>0.7389</td>
<td>0.763</td>
<td>0.852</td>
</tr>
<tr>
<td>Ankle Flexion</td>
<td>0.5285</td>
<td>0.8214</td>
<td>0.8634</td>
</tr>
<tr>
<td>Hip Flexion</td>
<td></td>
<td>0.618</td>
<td>0.6018</td>
</tr>
<tr>
<td>Shoulder Abduction</td>
<td>0.0018</td>
<td>0.00012</td>
<td>0.0015</td>
</tr>
</tbody>
</table>

The error estimates in the model output seem quite conservative in most cases. According to the error analysis that was implemented (Eq. 4.12), the model output error depends on the number of vertices in the boundary. Since there are more $X(a,b)$ values to be summed to obtain the output, it follows that the error will increase as the number of vertices increases. Therefore, sudden directional changes in the data can cause the error to increase. In addition, the angle variance of 4 deg$^2$ and torque variance of 1Nm$^2$ which were used for all of the joints are probably slightly conservative depending on the joint. Future work should involve reexamining the error estimate analysis to be sure that it is accurate. Also, the angle and torque variances used in the model should be determined on a joint by joint basis instead of using the same values for all joints.

In some cases the model did not predict well due to the fact that the human generated joint angles were outside of the model’s saturation limits. That is, the angles did not lie in the triangle over which $\mu(\alpha,\beta)$ was defined in the $\alpha\beta$ plane. Consider the data for shoulder abduction in Figure 4.19. Because of torque limits that were set to protect the spacesuit, the robot only achieved an angle range of approximately 33-53 degrees when suited,
which is significantly less than the human range of motion. Figure 4.19b shows the angular trajectory of the joint. All of the angles above the solid blue line lie outside of the model's defined boundaries. It can be seen from the figure that each time the angular input goes outside of these bounds the output of the model exhibits a discontinuity then goes to zero.

Figure 4.19: Shoulder abduction model verification compared to angle trajectory.
Chapter 5

EVA Operations

5.1 Operations Overview

Because of problems that arose during early EVAs, mission planners and training personnel began searching for better ways of preparing astronauts for what they would face in the unfamiliar environment of space. One way of confronting these issues was to attempt to replicate the space environment by building physical simulators which would allow the astronauts to get a feel for how objects might react in microgravity. The vast majority of NASA’s current EVA planning and training is still done through physical simulation.

A single facility does not currently exist in which all of the conditions that are encountered during an EVA can be simulated. However, physical and computer simulations are utilized to train astronauts for various aspects of their mission. EVA conditions are simulated through the use of neutral buoyancy water tanks, air-bearing floors, human thermal vacuum chambers, reduced gravity aircraft, 1-g mock-ups, and virtual reality.

Neutral buoyancy water tanks provide a helpful practice facility for repeated EVA training. The underwater training allows the astronaut to learn to work efficiently with the pressurized spacesuit. Therefore the vast majority of EVA training takes place in neutral buoyancy. However, there are two major limitations to working in water tanks. The water induces significant drag and the astronaut is only neutrally buoyant in two specific attitudes unless much care is taken to counterbalance the weight. These limitations make such actions as translating in the tanks much different than translating in space [Thuot and Harbaugh, 1995]. Additionally, neutral buoyancy simulations are very expensive, so other means have to be used to make astronauts as well prepared as possible before going into
the water. Figure 5.1 shows astronauts training for EVA in the Neutral Buoyancy Laboratory (NBL).

![Image of astronauts training for EVA](image)

**Figure 5.1:** Astronauts performing EVA training the NBL. Image courtesy of NASA.

Air-bearing floors are used to simulate large mass-handling tasks. The object that has to be manipulated by an EVA astronaut is placed on air-bearing pads that are connected to a high pressure air source that lifts the object approximately 0.025 m above the floor. However, in most cases only three degrees of freedom can be simulated on the air-bearing floor. Additionally, the astronauts can now simulate mass handling in the “Charlotte” virtual reality facility. This unique haptic simulator consists of a very light 2 foot box attached to motors on an 8 foot cubical frame. The box can be programmed to simulate arbitrary mass and inertia properties. Astronauts use this mass simulator while immersed in a virtual reality world via a head mounted display. As the astronauts move the object by its
handles, the system simulates the dynamics and drives the motors appropriately [Brooks, 1999].

The KC-135 parabolic flight aircraft can provide the astronauts with a limited duration exposure to microgravity. However this training only provides approximately 25-30 seconds of microgravity which permits only parts of tasks or very short tasks to be practiced.

In addition to physical simulations, computer simulation can serve as an important EVA planning tool. The state of the art in EVA computer simulation consists of high-performance computer graphics software known as PLAID. This software allows NASA to address human engineering issues in spacecraft design and analysis and enables human modeling, viewing analysis, animation development, lighting evaluation, crew operations and maintenance analysis, and design concepts visualization. PLAID contains the ability to create and manipulate human computer models in a 3D environment. It allows for the evaluation of the human-machine interface of designs and operations. Additionally, specific human models can be created from individual measurements both with and without the EMU. This software is used during EVA planning to determine reach and visibility of the astronauts. However, while PLAID is able to determine the geometries of a task, there are no dynamics involved.

5.1.1 Mission planning

Once EVA requirements are specified, the planning process begins. The conceptualization of how the EVA will work is well thought out in terms of which tools will be used, how the astronauts will be positioned, and what types of torques they will have to apply. After conceptualization one of the first steps is to take the planned EVA and lay out the choreography in PLAID. The shortcoming here is that PLAID only gives information about reach and visibility. There are no dynamics involved, so the software can determine whether an astronaut can reach a certain worksite configuration but not whether he or she
can actually apply a large enough torque to complete a task in that configuration. Therefore, once it has been determined that worksites can be reached, the next step is to begin physical simulations in the Neutral Buoyancy Lab (NBL). Here the astronauts are able to perform the tasks and determine whether or not they are actually feasible. It should be noted however, that it is extremely costly to perform simulations in the NBL. Therefore, early attention given to potentially questionable tasks using a tool such as a dynamic simulation could save water time and money [Hoffman, 2003].

5.2 Relevance

A more quantitative approach to the analysis of EVA is needed due to the increasing complexity of EVA tasks and the limitations of physical simulations. Several studies have shown that dynamic simulation can be an invaluable tool for EVA planning. Grant Schaffner (1999) performed a study in which he simulated an astronaut attempting to catch the Intelsat VI satellite using a capture bar which was designed specifically for that task. It was shown that given the dynamics of the system it was impossible to capture the satellite in under 6 seconds, even though the astronauts successfully performed the task numerous times underwater and on a 3 degree-of-freedom air-bearing simulator. Because this type of dynamic simulation was not performed before the mission, numerous attempts to capture the satellite on orbit failed before the crew eventually abandoned the originally planned procedures and worked with engineers on the ground to develop and emergency procedure. Repeated capture attempts by the EVA astronaut were unsuccessful because the capture bar could not be held in contact long enough for the latches to activate and seize the satellite. None of the physical simulators available had been able to accurately simulate the dynamics of all five of the bodies (orbiter, RMS, EVA crewmember, capture bar, satellite) that were in motion during the attempts. As a result the astronaut was unable to capture the satellite using the procedures that had been practiced on the Earth.
Figure 5.2 shows a PLAID simulation of the Intelsat VI capture which was performed before the mission. It was incorrectly assumed that the capture would be successful because the simulation did not take into account the dynamics of the system.

![Figure 5.2: Intelsat VI capture simulated using PLAID. Image courtesy of NASA.](image)

While computer simulation does not directly help with astronaut training, it does supply an efficient and inexpensive means of mission planning and determining the feasibility of EVA tasks. Therefore, physical and computational simulations should be used in conjunction during EVA planning and training so that each may compensate for the limitations of the other.

Compiling a detailed database of the joint torque characteristics of the EMU will allow a better understanding of the effects of the spacesuit on the crew members, which must be accounted for in computer simulations. Dave Rahn (1997) performed a study in which he showed that the inclusion of space suit constraints caused significant differences in results.
of a simulation of an astronaut performing a large mass handling task. Therefore, an accurate model of the spacesuit joint torque characteristics is needed. The current research has accomplished this task by developing data driven mathematical hysteresis models of the spacesuit joint characteristics. These models can be included in a dynamic simulation of a suited astronaut in order to account for the hysteretic torques experienced by the astronauts as they work against the suit’s gas pressure and material properties.

5.3 Human Robotic Synergies

As plans arise for missions to the Moon, Mars, and beyond, robots will become increasingly important as supporters of EVA. Humans will operate together with robotic vehicles and devices which act as assistants and exploration partners. Working side by side with humans, or going where the risks are too great for people, machines will expand our ability for construction and exploration. Likewise, as robots become increasingly more capable, spacesuits must also evolve in order to make exploration more productive.

Several projects are currently underway which aim to revolutionize the interaction between humans and robots and display the utility of robots in the area of human spaceflight. One such undertaking is NASA's Robonaut project, which seeks to develop and demonstrate a robotic system that can function as an EVA astronaut equivalent. It keeps the human operator in the control loop through its telepresence control system. Robonaut is designed to be used for EVA tasks, i.e., those which were not specifically designed for robots. In order to control robonaut's 43 degrees of freedom a command/effector relationship is utilized where the operator's motion is followed by the robot. Robonaut's telepresence system includes a Helmet Mounted Display (HMD), force and tactile feedback gloves and posture trackers [Necessary, 2001].
5.3.1 M Tallchief Case Study

In this same light several improvements are currently being considered and implemented in order to make M Tallchief more useful both as a research and learning tool for human spaceflight. Similar to Robonaut’s telepresence system one of our goals is to implement a command and effector system by which the robot could be controlled. The suited robot would be able to follow the motions of an unsuited human performing tasks in real-time. This is a useful quality for several reasons. Having human subjects wear a spacesuit is very time consuming and costly. The suit has to be sized for each individual and life support has to be provided during each test. By reading the joint angles from an unsuited human subject the need for life support is removed and the suit only has to be sized and installed on the robot once. Additionally by measuring the human joint angles and directly reading them to the robot you eliminate the need for time consuming video motion capture analysis and can immediately receive the torque data from the robot. Finally, this method eliminates the need to program complicated robot motions ahead of time, which can be a challenging task due to the complicated nature of inverse kinematics.

A simple 1 degree of freedom prototype was built for the elbow in order to serve as a proof of concept. The device incorporates a potentiometer at the joint and outputs a value between +10 volts and -10 volts depending on the joint angle. Figure 5.3 shows the prototype being used to command the robot’s elbow.
Figure 5.3: M. Tallchief command and effector demonstration.

While this one degree of freedom joint is relatively simple to design for, more complex joints, such as the shoulder, pose a much greater challenge. One potential solution that is currently being investigated is the use of a Motionstar position tracking system. The motionstar system allows real-time motion capture with instantaneous playback of the movements. By using this off-the-shelf system we avoid having to build the complex system of joint sensors that would be required to determine all twelve joint angles. The Motionstar system, with it's small tethered sensors, provides a low bulk solution to the motion capture. In addition, a prewritten Windows-based user interface exists that will
make calibration of the system easier, and the sensor output is designed to be read into a number of commercially available software packages.

In addition to the command/effector improvements, the robot is currently being upgraded such that both position and torque at each joint can be controlled through a Malab/Simulink interface. This provides a much more versatile and familiar user interface for both robot control as well as data acquisition. It also allows various controllers to be created and tested in a very short amount of time, including controllers such as those thought to be used by humans for multi-joint motions. This technology has been demonstrated on the robot's arm by implementing both an impedance controller and an adaptive controller.

As a result of these improvements four areas stand out in which M. Tallchief can now be more useful as a research and learning tool. These areas include spacesuit evaluation and design, planning astronaut motions, theories on multi-joint movement, and robotic controllers.

In addition to the applications to multi-joint movement and robotic controllers mentioned above, the robot would also be better equipped to study spacesuit analysis and design. The current usage of the robot could be expanded such that new suit designs can be analyzed in terms of their joint torque characteristics. The updates to the robot/computer system will provide an easier interface so that other suit designers can come in and try out their suits in a systematic fashion. Furthermore, the robot could be used in order to plan astronaut EVA motions. Similar to the dynamic simulations discussed in the previous section, by using the command and effector robotic control, different ways of completing an EVA could be analyzed. Real-time joint torque data could be obtained in order to determine whether a particular task is feasible.
5.4 Design Recommendations

5.4.1 Spacesuit Design

As we look forward to possible manned missions to Mars, the requirements for spacesuit mobility increase dramatically. Astronauts will have to traverse rocky terrain and cover large distances while doing useful work in a gravity field that is over twice that of the Moon. As seen from the human experiments reported in Chapter 3, the current EMU holds little hope of meeting these requirements. These planetary missions will necessitate a suit that is highly mobile and will require entirely new suit designs.

Advanced gas pressure suits, such as the ILC Dover I-suit and Hamilton Sundstrand H-suit, which are currently being designed, are based on the same principles as the EMU but exhibit much higher mobility. This mobility is achieved using components such as gored joints and multiple rotary bearings, including hip bearings that have been optimized for walking and sitting [Graziosi and Feri, 1999].

Perhaps the best design would be to do away with gas pressure suits all together. This could be accomplished by using mechanical counterpressure (MCP) which works on the principle that it does not matter whether the internal pressure causes wall tension in a vessel or whether tension created in the walls produces internal pressure. Therefore, instead of using gas to provide the pressure necessary at the surface of the body, a tight-fitting garment which squeezes the skin could be used, with breathing gas provided by a pressurized helmet. Since there is no internal gas pressure, there is no work due to volume changes in the gas. Therefore the work that must be done is due to the elasticity of the fabric. Additionally, since the suit is tight-fitting there is no excess bulk to inhibit mobility. Feasibility studies of mechanical counterpressure are currently being carried out to determine the most efficient methods of providing skin surface pressure [Frazer et al, 2002].
As evidenced by the Intelsat VI capture attempt, dynamic simulation can play an important role in EVA planning. This is especially important in instances where potentially questionable tasks are suggested for EVA. This early attention to potential problems would save valuable NBL time. Consequently, multi-body dynamics as well as spacesuit joint torque characteristics should be incorporated into PLAID simulations.

Furthermore, the RSST should be utilized more frequently to analyze new suit designs. The upgrades and improvements to the robot will make it much easier for new spacesuits to be tested and compared. Eventually a standard set of tests could be developed which would run a new suit through a comprehensive analysis and present suit designers with information on joint torques, range of motion, and even workloads and metabolic cost estimates.

There are currently plans underway to test prototype mechanical counterpressure suit components on isolated joints of the robot in order to compare the joint characteristics to those of the EMU. Additionally, a simple spacesuit analog which can be used for studying metabolic cost is being developed by using M Tallchief and data that has been collected from the current EMU.

5.5 Summary and Conclusions

A joint torque and angle database was collected for the EMU spacesuit which covers a large range of EMU joint angles. The data collection utilized an instrumented robot to act as a human surrogate. This methodology allows for more accurate measurements than simply measuring the joint torques of an empty spacesuit. The database represents the most complete set of EMU joint torque data that has been published to date. Both single joint and multi-joint motions were performed. Tests were also performed which quantify the effect of the spacesuit operating pressure on joint torques. As expected these results
showed that as the operating pressure is increased, the joint torques were higher and the range of motion decreased. Suited human experiments were also performed which highlighted the shortcomings of the current EMU in terms of locomotion and range of mobility.

The spacesuit joint torque database is archived in the MIT Man Vehicle Lab and is available on CD-ROM. The contents include the joint versus angle data for each of the robot trials listed in Table 3.1 as well as an Excel spreadsheet explaining the contents of each file. The data are stored in Matlab data structures which contain the raw data (both with and without the spacesuit) as well as the processed, weight compensated data. Additionally, hysteresis model files for each of the joints is included along with the Matlab scripts which are used to implement the model.

The spacesuit joint torque database was then utilized to produce predictive models of the suit mobility. Because of the inherent nonlinearities in the torque versus angle data, special hysteresis modeling techniques had to be utilized. Preisach hysteresis model coefficients were determined for each of the seven joints for which hysteresis identification data were collected. The models of each joint were then verified against multi-joint experimental data that was collected during a previous set of experiments. The models fit very well for most of the joints with $r^2$ values greater than 0.6 for shoulder flexion, elbow flexion, humerus rotation, knee flexion, ankle flexion, and hip flexion.

The model for the shoulder abduction did not fit the data well because of its limited angle range. These model saturation limits are set by the range of the data used in the identification process. In the future care should be taken to collect shoulder abduction data for a larger range of angles from which a new hysteresis model can be determined.

These mathematical models are appropriate for implementation into dynamic simulations of suited astronauts for the purpose of EVA training and planning. Dynamic simula-
tion could greatly benefit EVA planning by determining the feasibility of potentially questionable EVA tasks and reducing costly water time in the NBL.
Bibliography


Appendix A

Hysteresis Modeling Scripts

function [alpha, beta, x]=idx(tor, ang)
% reads in torque and angle data and outputs the model coefficients
% does hys model id according to Ge and Jouaneh
% modified version of code written by Schmidt
manualflag=1;
% force inputs to be columns
tor=tor(:);
ang=ang(:);
if manualflag
    clg
    plot(tor)
    title('click on the maxes and mins')
    peakinds=round(ginput(14)); % this must be an even number
    peakinds=peakinds(:,1);
else
    % find peaks of input--these are alpha values
    sd=sign(diff(ang));
    % peak is where deriv changes sign
    peakinds=find(sd.*[0;0;0;0; sd(1:length(sd)-4)]<0)-4;
    peakinds=[setdiff(peakinds,peakinds+2);
      length(ang)];
end %if

% prevent running over array ends at beginning and end
peakinds(find(peakinds<10))=10;
peakinds(find(peakinds>length(tor)-10))=length(tor)-10;
% assuming positive initial slope and even number of peakinds , maxes are odd values of peakinds
% keyboard
maxes=peakinds(2*(0:round(length(peakinds)/2-1)+1));
mins=peakinds(2:2:length(peakinds));
% search nearest 10 points for max
for i=1:length(maxes)
    maxadd=find(tor(maxes(i)+(-5:1:5))==max((tor(maxes(i)+(-5:1:5)))));
    maxadd=maxadd(1);
    maxes(i)=maxes(i)+maxadd-6;
end %for i

% search nearest 10 points for min
for i=1:length(mins)
    minadd=find(tor(mins(i)+(-5:1:5))==min((tor(mins(i)+(-5:1:5)))));
    minadd=minadd(1);
    mins(i)=mins(i)+minadd-6;
end %for i

alpha=tor(maxes);
for i=1:length(maxes)
    peakinds=[peakinds,maxes(i);mins(i)];
end %for
peakinds(1:2*length(maxes))=[];

d=diff(peakinds);
ncols=max(d(1:2:length(d)))+1;
% allocate x, beta
x=-999*ones(length(maxes), ncols);
beta=x;

% % do id
% if maxes(1)<mins(1)
  for i=1:length(maxes)
    inds=peakinds(2*i-1):peakinds(2*i);
    x(i,1:length(inds))=ang(maxes(i))-
    ang(inds)';
    beta(i,1:length(inds))=tor(inds)';
  end % for
% else
  for i=1:length(maxes)-1
    inds=peakinds(2*i):peakinds(2*i+1);
    x(i,1:length(inds))=ang(maxes(i))-
    ang(inds)';
    beta(i,1:length(inds))=tor(inds)';
  end % for
end

x=[ang(maxes) x];
beta=[min(tor)*ones(size(alpha)) beta];

%keyboard

function [u, sigma2u]=xmodel(t)
% implements x model
% reads in a vector of angles and predicts the torque output
% modified version of code written by Schmidt
% force t to be a column
t=t(:);

% load model file
eval('load ''Macintosh HD:Users:Annie:Data:hysmodels:efmodel4.mat'''); % elbow
% s indicates if t goes up or down
s=sign([0;diff(t)]);

% initialize vars
aindices=1;
bindices=1;
% set initial vertex
firsts=s(min(find(s~=0)));
if firsts>0
  alpha=t(1);
  lowt=min([1.01*min(t) .99*min(t)]);
  firstbeta=min([min(min(allbeta(find(allbeta~=999)))) lowt]);
  beta=firstbeta;
else
  hight=max([1.01*max(t) .99*max(t)]);
  firstalpha=max([hight max(allalpha)]);
  beta=t(1);
  alpha=firstalpha;
end %if

u=zeros(size(t));
sigma2u=u;

for index=1:length(t)
  % find alpha, beta for current t
  if s(index)~=0
    % keyboard
  end

end %for
lastsindex = max(find(s(1:index-1)~=0));
if isempty(lastsindex)
    lastsindex = 1;
end %if
if isempty(bindices) & ~isempty(beta)
    disp('empty bindices in xmodel')
    keyboard
end %if
[alpha,aindices,beta,bindices] = ab(t(index),s(index),alpha,aindices,beta,bindices,firsts,s(lastsindex),index);
if firsts < 0 & alpha(1) ~= firstalpha
    disp('***firsts<0 & alpha(1)~=firstalpha')
    alpha = [firstalpha; alpha];
    %aindices = [1; aindices];
    beta = [beta(1); beta];
    %bindices = [1; bindices];
    %keyboard
end %if
if firsts > 0 & beta(1) ~= firstbeta
    disp('***firsts>0 and beta(1)~=firstbeta')
    alpha = [alpha(1); alpha];
    %aindices = [1; aindices];
    beta = [firstbeta; beta];
    %bindices = [1; bindices];
end %if
end %if
if length(alpha) ~= length(beta)
    alpha
end
nverts = length(alpha);
if firsts < 0
    u(index) = max(max(x));
else
    u(index) = 0;
end %if
for j = 1:floor(nverts/2)
    uinc = (interpx(alpha(2*j-1),beta(2*j-1),allalpha,allbeta,x) -
            interpx(alpha(2*j),beta(2*j),allalpha,allbeta,x));
    if firsts > 0
        u(index) = u(index) + uinc;
    else
        u(index) = u(index) - uinc;
    end %if
end %for
sigma2u(index) = errorx(alpha,beta,allalpha,allbeta,x);
if s(min(index+1,length(s))) ~= 0 &
    length(beta) > 1 & length(alpha) > 1
    beta(length(beta)) = [];
    alpha(length(alpha)) = [];
end %if
end %for index
u(1) = u(2);
function [alpha,maxinds,beta,mininds] = ab(t,s,maxes,maxinds,mins,mininds,s1,slast,index)
% modified version of code written by Schmidt
% This function is called in the script xmodel.m in order to
determine the S+/S- boundary at each time step

[junk1,indmx,junk2]=unique(maxes);
inmaxes=maxes(sort(indmx));
[junk1,indmn,junk2]=unique (mins);
inmins=mins(sort(indmn));

maxes=inmaxes;
mins=inmins;
inmaxinds=maxinds;
inmininds=mininds;

if s~=0
    if s>0
        %disp('s>0')
        maxes=[maxes;t];
        if exist('maxinds')
            maxinds=[maxinds;index];
        else
            maxinds=t;
        end %if
    end %if
    if s<0
        %disp('s<0')
        mins=[mins;t];
        if exist('mininds')
            mininds=[mininds;index];
        else
            mininds=t;
        end %if
    end %if
else
    % disp('setting obsmins, s<0')
    obsmins=find(mins(1:length(mins)-1)<=t);
    maxreplace=abs(mins(length(mins))-
inmins(length(inmins)))<10*eps & mininds(min(obsmaxes))<10*eps &
    mininds(length(mininds))<inmininds(length(inmininds));
    if sum(maxreplaces)
        %disp('setting obsmins, s<0')
        obsmins=length(mins);
        obsmins=find(mininds>maxinds(min(obsmaxes)));
    end %if
    else
        obsmins=find(mins(1:length(mins)-1)>=t);
        minreplacene=abs(mins(length(mins))-
inmins(length(inmins))<10*eps & mininds(length(mininds))<inmininds(length(inmininds));
end %if
end %if
% wipe out obsolete vertices
warning off
if sl>O
    if s>O
        obsmaxes=find(maxes(l:length(maxes)-l)<=t);
        maxreplaces=abs(maxes(length(maxes))-
inmaxes(length(inmaxes)))<10*eps & maxinds(length(maxinds))~=inmaxinds(length(inmaxinds));
        if sum(obsmaxes &
            (obsmaxes~=length(maxes)-1 | s*slast<O)) |
            sum(maxreplaces)
                %disp('setting obsmins, s>O')
                obsmins=length(maxs);
                obsmins=find(mininds>maxinds(min(obsmaxes)));
        end %if
    end %if
    if s<O
        %disp('s<O')
        obsmaxes=find(maxes(length(maxes))<=t);
        maxreplaces=abs(maxes(length(maxes))-
inmaxes(length(inmaxes)))<10*eps & maxinds(length(maxinds))~=inmaxinds(length(inmaxinds));
        if sum(obsmaxes &
            (obsmaxes~=length(maxes)-1 | s*slast>O)) |
            sum(maxreplaces)
                %disp('setting obsmins, s<0')
                obsmins=length(maxs);
                obsmins=find(mininds>maxinds(min(obsmaxes)));
        end %if
    end %if
end %if
end %if

102
if sum(obsmins &
  (obsmins~length(mins)-1 | s*slast<0)) | sum(minreplace)
  obsmaxes=find(maxinds>mininds(min(obsmins)));
  %obsmaxes=length(maxes);
end %if
  end %if
else
  if s>0 % this logic doesn't seem to work
  for obs'mins 6/19/01
    obsmaxes=find(maxes(1:length(maxes))-1<=t);
    maxreplace=abs(maxes(length(maxes))-
      inmaxes(length(inmaxes))<10*eps & max-
      inds(length(maxinds))~=inmaxinds(length(inmaxinds)));
      if sum(obsmaxes &
      (obsmaxes~length(maxes)-1 | s*slast<0)) | sum(maxreplace)
        obsmins=find(mininds>max-
        inds(min(obsmins)));
        %obsmins=length(mins);
        end %if
      else
        obsmins=find(mins(1:length(mins)-
        1)>=t);
        minreplace=abs(mins(length(mins))-
        inmins(length(inmins))<10*eps & mininds(length(min-
        inds))~=inmininds(length(inmininds)));
        if sum(obsmins &
        (obsmins~length(mins)-1 | s*slast<0)) | sum(minre-
        place)
% put maxes and mins into alpha and beta
nverts=length(mins)+length(maxes);
if s1>0
   as=0;
   bs=1;
   if nverts/2==round(nverts/2)
      ea=nverts-1;
      eb=nverts-1;
   else
      ea=nverts-2;
      eb=nverts;
   end %if
else % probably problem here for s1<0 case
   as=1;
   bs=0;
   if nverts/2==round(nverts/2)
      ea=nverts-1;
      eb=nverts-1;
   else
      ea=nverts;
      eb=nverts-2;
   end %if
end %if
aindices=floor((as:ea)/2+1)';
bindices=floor((bs:eb)/2+1)';
if aindices>length(maxes)
   aindices=maxes
   error('In ab: aindices>length(maxes)')
end %if
if bindices>length(mins)
   bindices=mins
   error('In ab: bindices>length(mins)')
end %if
alpha=maxes(aindices);
beta=mins(bindices);
if length(alpha)<length(beta)
   alpha=[alpha; t];
   %maxinds=[maxinds; index];
   %beta(length(beta))=[];
   end %if
if length(beta)<length(alpha)
   beta=[beta; t]; % this is where beta
   %mininds=[mininds; index];
   %alpha(length(alpha))=[];
   end %if
else
   alpha=maxes;
   beta=mins;
end %if s~=0

% check results before returning
if (length(alpha)==length(beta))
   error('alpha and beta different length')
end % if
if (sum(alpha<beta))
   alpha
   beta
   error('In ab.m alpha must be > beta')
end %if
if isempty(mininds) & ~isempty(beta)
    disp('in ab, empty bindices')
    keyboard
end %if

function xinterp=interpx(alpha,beta,allalpha,allbeta,x)
%modified version of code written by Schmidt
%This function is called in the script xmodel.m. It interpolates the
%X value at a given point from the stored model coefficients that
%were determined using idx.m

% saturate if necessary
if alpha>max(allalpha)
    alpha=max(allalpha);
end %if
if beta>max(max(allbeta))
    beta=max(max(allbeta));
end %if

% trap "direct hit" case, force single output
if sum(alpha==allalpha)
    aindex=find(alpha==allalpha);
    if sum(beta==allbeta(aindex(1),:))
        bindex=find(beta==allbeta(aindex(1),:));
        xinterp=x(aindex(1),bindex(1));
    end %if
end %if

if ~exist('bindx')
    % find alphas to bracket point
    alphal=max(allalpha(find(allalpha<=alpha)));
    alpha2=min(allalpha(find(allalpha>=alpha)));
    warning off
    % find betas
    betalrow=allbeta(find(allalpha==alphal),:);
    betalrow=betalrow(find(betalrow~=-999));
    beta2row=allbeta(find(allalpha==alpha2),:);
    beta2row=beta2row(find(beta2row~=-999));
    betall=max(betalrow(find(betalrow<=beta)));
    betalr=min(betalrow(find(betalrow>=beta)));
    beta21=max(beta2row(find(beta2row<=beta)));
    beta2r=min(beta2row(find(beta2row>=beta)));
    % this will be empty if triangle
    if isempty(alphal) & ~isempty(beta2l)
        alphal=beta2r;
        betalr=beta2r;
        %disp('alphal=beta2l case')
    end %if
    if alpha<beta
        alpha
        beta
        error('in interpx.m must be > beta')
    elseif alpha==beta % then check for alpha=beta (on line)
xinterp=0;

elseif ~isempty(betall) & ~isempty(betalr) & ~isempty(beta2r) & ~isempty(beta2l)
  % 4 corners

  x4=mean(x(alpha2i,beta2ri));
  else
    x4=0;
  end

  % find distance from alpha,beta to corners of square
  d1=sqrt((alpha-alphal)^2+(beta-betall)^2);
  d2=sqrt((alpha-alphal)^2+(beta-betalr)^2);
  d3=sqrt((alpha-alpha2)^2+(beta-beta2l)^2);
  d4=sqrt((alpha-alpha2)^2+(beta-beta2r)^2);

  % take weighted avg to get x
  xinterp=(1/
    (d2*d3*d4+d1*d3*d4+d1*d2*d4+d1*d2*d3))*
    (d2*d3*d4*x1+d1*d3*d4*x2+
     d1*d2*d4*x3+d1*d2*d3*x4);

  elseif isempty(betall) & ~isempty(betalr) & ~isempty(beta2r) & ~isempty(beta2l)
    % 3 corners, not 1
    alphali=find(allalpha==alphal);
    alpha2i=find(allalpha==alpha2);
    [junk,betalri]=find(all-
     beta(alphali,:)==betalr);
    [junk,beta2li]=find(all-
     beta(alpha2i,:)==beta2l);
    [junk,beta2ri]=find(all-
     beta(alpha2i,:)==beta2r);

    if alphali ~= betalri
      x2=mean(x(alphali,betalri));
    else
      x2=0;
    end

    if alpha2i ~= beta2li
      x3=mean(x(alpha2i,beta2li));
    else
      x3=0;
    end

    if alpha2i ~= beta2ri
      x2=mean(x(alpha2i,beta2ri));
    else
      x2=0;
    end

    elseif isempty(betalr) & ~isempty(beta2l) & ~isempty(beta2r) & ~isempty(beta2l)

      alphali=find(allalpha==alphal);
      alpha2i=find(allalpha==alpha2);
      [junk,betall]=find(all-
       beta(alphali,:)==betall);
      [junk,beta2li]=find(all-
       beta(alpha2i,:)==beta2l);
      [junk,beta2ri]=find(all-
       beta(alpha2i,:)==beta2r);

      if alphali ~= betall
        x1=mean(x(alphali,betall));
      else
        x1=0;
      end

      if alphali ~= beta2li
        x3=mean(x(alphali,beta2li));
      else
        x3=0;
      end

      if alphali ~= beta2ri
        x1=mean(x(alphali,beta2ri));
      else
        x1=0;
      end

      elseif isempty(beta2l) & ~isempty(beta2r) & ~isempty(betall) & ~isempty(betalr)

        alphali=find(allalpha==alphal);
        alpha2i=find(allalpha==alpha2);
        [junk,beta2ri]=find(all-
         beta(alpha2i,:)==beta2ri);
        [junk,betall]=find(all-
         beta(alphali,:)==betall);
        [junk,betalri]=find(all-
         beta(alphali,:)==betalri);

        if alphali ~= betall
          x3=mean(x(alphali,betall));
        else
          x3=0;
        end

        if alphali ~= betalri
          x4=mean(x(alphali,betalri));
        else
          x4=0;
        end

        elseif isempty(beta2l) & ~isempty(betall) & ~isempty(betalr) & ~isempty(beta2r)

          alphali=find(allalpha==alphal);
          alpha2i=find(allalpha==alpha2);
          [junk,betalri]=find(all-
           beta(alphali,:)==betalri);
          [junk,beta2li]=find(all-
           beta(alpha2i,:)==beta2l);
          [junk,beta2ri]=find(all-
           beta(alpha2i,:)==beta2r);

          if alphali ~= betalri
            x2=mean(x(alphali,betalri));
          else
            x2=0;
          end

          if alphali ~= beta2li
            x3=mean(x(alphali,beta2li));
          else
            x3=0;
          end

          if alphali ~= beta2ri
            x4=mean(x(alphali,beta2ri));
          else
            x4=0;
          end
if alpha2i != beta2li
    x3 = mean(x(alpha2i, beta2li));
else
    x3 = 0;
end
if alpha2i != beta2ri
    x4 = mean(x(alpha2i, beta2ri));
else
    x4 = 0;
end

% find distance from alpha, beta to corners of square
    d2 = sqrt((alpha - alpha1)^2 + (beta - beta1)^2);
    d3 = sqrt((alpha - alpha2)^2 + (beta - beta21)^2);
    d4 = sqrt((alpha - alpha2)^2 + (beta - beta2r)^2);

    xinterp = (1/(d3*d4 + d2*d3 + d2*d4 + d2*d3))*
        (d3*d4*xl + d2*d4*x3 + d2*d3*x4);

elseif ~isempty(betali) & ~isempty(betalr) & ~isempty(beta2r) & ~isempty(beta21)
    % no betali -> no point 2
    % find corresponding x's
    alphali = find(allalpha == alphali);
    alpha2i = find(allalpha == alpha2);
    [junk, betali] = find(all-beta(alphali, :) == betali);
    [junk, beta2li] = find(all-beta(alpha2i, :) == beta2li);
    [junk, beta2ri] = find(all-beta(alpha2i, :) == beta2ri);

    if alphali != betali
        xl = mean(x(alphali, betali));
    else
        xl = 0;
    end
    if alpha2i != beta2li
        x3 = mean(x(alpha2i, beta2li));
    else
        x3 = 0;
    end
    if alpha2i != beta2ri
        x4 = mean(x(alpha2i, beta2ri));
    else
        x4 = 0;
    end

    % find distance from alpha, beta to corners of square
    d2 = sqrt((alpha - alpha1)^2 + (beta - betali)^2);
    d3 = sqrt((alpha - alpha2)^2 + (beta - beta21)^2);
    d4 = sqrt((alpha - alpha2)^2 + (beta - beta2r)^2);

    xinterp = (1/(d3*d4 + d2*d3 + d2*d4 + d2*d3))*
        (d3*d4*xl + d2*d4*x3 + d2*d3*x4);

    if alphali != betali
        xl = mean(x(alphali, betali));
    else
        xl = 0;
    end
    if alpha2i != beta2li
        x3 = mean(x(alpha2i, beta2li));
    else
        x3 = 0;
    end
    if alpha2i != beta2ri
        x4 = mean(x(alpha2i, beta2ri));
    else
        x4 = 0;
    end
[junk, beta1ri] = find(all-beta(alphali,:) == betalr);
[junk, beta2ri] = find(all-beta(alpha2i,:) == beta2r);

if alphali ~= betalri
    x1 = mean(x(alphali, betalri));
else
    x1 = 0;
end

if alphali ~= beta1ri
    x2 = mean(x(alphali, beta1ri));
else
    x2 = 0;
end

if alpha2i ~= beta2ri
    x4 = mean(x(alpha2i, beta2ri));
else
    x4 = 0;
end

% find distance from alpha, beta to corners of square
d1 = sqrt((alpha-alphali)^2 + (beta-betall)^2);
d2 = sqrt((alpha-alphali)^2 + (beta-betalr)^2);
d4 = sqrt((alpha-alpha2i)^2 + (beta-beta2r)^2);

xinterp = (1/(d2*d4 + d1*d4 + d1*d2 + d1*2 + d2*4)) * (d2*d4 * x1 + d1*d4 * x2 + d1*d2 * x4);

elseif isempty(betalr) & isempty(betall) & ~isempty(beta2r) & ~isempty(beta2l)
% 2 corners at alphal
disp('2 corners at alphal')
alphali = find(allalpha == alphal);
[junk, betalli] = find(all-beta(alphali,:) == betall);
[junk, beta1ri] = find(all-beta(alphali,:) == beta1r);
if alphali ~= betalli
    x1 = mean(x(alphali, betalli));
else
    x1 = 0;
end

elseif ~isempty(betalr) & ~isempty(betall) & isempty(beta2r) & isempty(beta2l)
% 2 corners at alphal
disp('2 corners at alphal')
alphali = find(allalpha == alphal);
[junk, betalli] = find(all-beta(alphali,:) == betall);
[junk, beta1ri] = find(all-beta(alphali,:) == beta1r);
if alphali ~= betalli
    x1 = mean(x(alphali, betalli));
else
    x1 = 0;
end

elseif ~isempty(betall) & ~isempty(beta1r) & ~isempty(beta2l) & ~isempty(beta2r)
% 2 corners at beta2
disp('2 corners at beta2')
alpha2i = find(allalpha == alpha2);
[junk, beta2li] = find(all-beta(alpha2i,:) == beta2l);
[junk, beta2ri] = find(all-beta(alpha2i,:) == beta2r);
if alpha2i ~= beta2li
    x3 = mean(x(alpha2i, beta2li));
else
    x3 = 0;
end

if alpha2i ~= beta2ri
    x4 = mean(x(alpha2i, beta2ri));
else
    x4 = 0;
end

d3 = sqrt((alpha-alpha2)^2 + (beta-beta2l)^2);
d4 = sqrt((alpha-alpha2)^2 + (beta-beta2r)^2);
xinterp = (1/(d4+d3)) * (d4*x3 + d3*x4);
end
if alphali ~= betali
    x2 = mean(x(alphali, betali));
else
    x2 = 0;
end

d1 = sqrt((alpha - alphal)^2 + (beta - betal)^2);
d2 = sqrt((alpha - alphali)^2 + (beta - betali)^2);
xinterp = (1/(d2+d1))*(d2*x1+d1*x2);
elseif isempty(beta2r) & isempty(betali)
    elseif isempty(beta2l) & isempty(betali)
    elseif ~isempty(beta2r) & ~isempty(beta2l) & ~isempty(betali)
        % 2 corners at r
        alphali = find(allalpha == alphal);
        alpha2i = find(allalpha == alpha2);
        [junk, betali] = find(allbeta(alphali, :) == betali);
        [junk, beta2li] = find(allbeta(alpha2i, :) == beta2li);
        if alphali ~= betali
            x2 = mean(x(alphali, betali));
        else
            x2 = 0;
        end
        if alpha2i ~= beta2li
            x3 = mean(x(alpha2i, beta2li));
        else
            x3 = 0;
        end
        d2 = sqrt((alpha - alphali)^2 + (beta - betali)^2);
        d3 = sqrt((alpha - alphal)^2 + (beta - betali)^2);
        xinterp = (1/(d3+d2))*(d3*x1+d2*x2);
elseif isempty(beta2r) & ~isempty(beta2l) & ~isempty(betali)
    elseif ~isempty(beta2r) & ~isempty(betali) & ~isempty(beta2l)
        % 2 corners at l
        alphali = find(allalpha == alphal);
        alpha2i = find(allalpha == alpha2);
        [junk, betali] = find(allbeta(alphali, :) == betali);
        [junk, beta2li] = find(allbeta(alpha2i, :) == beta2li);
        if alphali ~= betali
            x1 = mean(x(alphali, betali));
        else
            x1 = 0;
        end
        if alpha2i ~= beta2li
            x3 = mean(x(alpha2i, beta2li));
        else
            x3 = 0;
        end
        d1 = sqrt((alpha - alphal)^2 + (beta - betal)^2);
        d3 = sqrt((alpha - alphal)^2 + (beta - beta2li)^2);
        xinterp = (1/(d3+d1))*(d3*x1+d1*x3);
elseif ~isempty(beta2r) & ~isempty(betali) & ~isempty(beta2l)
        elseif ~isempty(beta2l)
            % 1 point case--have point 1
            alphali = find(allalpha == alphal);
            [junk, betali] = find(allbeta(alphali, :) == betali);
            if alphali ~= betali
                x1 = mean(x(alphali, betali));
            else
                x1 = 0;
            end
            xinterp = x1;
elseif ~isempty(betalr) & isempty(betall) & isempty(beta2r) & isempty(beta2l)
    % 1 point case--have point 2
    alphali=find(allalpha==alphal);
    [junk,betalri]=find(allbeta(alphali,:)==betalr);
    if alphali~=betalri
        x2=mean(x(alphali,betalri));
    else
        x2=0;
    end
    xinterp=x2;
end

elseif isempty(betalr) & isempty(betall) & isempty(beta2r) & ~isempty(beta2l)
    % 1 point case--have point 3
    alpha2i=find(allalpha==alpha2);
    [junk,beta2li]=find(allbeta(alpha2i,:)==beta2l);
    if alpha2i~=beta2li
        x3=mean(x(alpha2i,beta2li));
    else
        x3=0;
    end
    xinterp=x3;
end

elseif isempty(betalr) & isempty(betall) & ~isempty(beta2r) & isempty(beta2l)
    % 1 point case--have point 4
    alpha2i=find(allalpha==alpha2);
    [junk,beta2ri]=find(allbeta(alpha2i,:)==beta2r);
    if alpha2i~=beta2ri
        x4=mean(x(alpha2i,beta2ri));
    else
        x4=0;
    end
    xinterp=x4;
end

% big if

if isempty(xinterp)
    disp(['betalr= ',num2str(betalr)])
    disp(['betall= ',num2str(betall)])
    disp(['beta2r= ',num2str(beta2r)])
    disp(['beta2l= ',num2str(beta2l)])
    %error('xinterp not assigned')
    xinterp=-999;
end

%if

function sigma2u=errorx(alpha,beta,allalpha,allbeta,x)
%This function is called by xmodel.m and calculated the error
%in the model according to Schmidt (2001)
if length(alpha)~=length(beta)
    error('In errorx, alpha and beta must be same length')
end

len=length(alpha)-1;
sigma2angle=2^2;
sigma2torque=1^2;

for index=1:len
    alphal=alpha(index)+1;
end
alpha2=alpha(index)-1;
betal=beta(index)+1;
beta2=beta(index)-1;

if alpha2<beta(index)
    alpha2=alpha(index);
end %if

if betal>alpha(index)
    betal=beta(index);
end %if

xal=interpx(alphal,beta(index),allalpha,allbeta,x);
xa2=interpx(alpha2,beta(index),allalpha,allbeta,x);
xb1=interpx(alpha(index),betal,allalpha,allbeta,x);
xb2=interpx(alpha(index),beta2,allalpha,allbeta,x);

dxda=(xal-xa2)/(alphal-alpha2);
dxdb=(xb1-xb2)/(betal-beta2);

sigma2u=2*len*sigma2torque+sigma2angle*sum(sigma2term);
if sigma2u>1000
    keyboard
% end %if
Appendix B

Experimental Subject Consent Form

Quantifying Astronaut Performance for Future Space Suit Development

Principal Investigators:
Prof. Dava J. Newman

Co-Investigators:
Patricia Schmidt
Annie Frazer
Massachusetts Institute of Technology

I. VOLUNTARY PARTICIPATION, RIGHT TO WITHDRAW
Participation in this experiment is voluntary and the subject may withdraw consent and discontinue participation in this experiment at any time without prejudice.

II. PURPOSE AND OBJECTIVE OF EXPERIMENT
The purpose of this study is to quantify joint motions and torques required for humans to perform tasks in a space suit. This study’s aim is to understand, simulate, and predict the capabilities of suited astronauts in a variety of scenarios. Subjects wearing a space suit will perform predetermined arm and leg motions, while the positions of body segments are recorded using an external video motion capture system. The recorded body motions will then be used to command a suited anthropomorphic robot while torques at robot joints are measured. 2.A fair explanation of the procedures to be followed and their purposes, including identification of any procedures which are experimental.

III. EXPERIMENTAL PROTOCOL
As a subject, I will perform pre-determined arm and leg motion tasks while either wearing the NASA Extravehicular Mobility Unit (EMU) (suited) or not wearing the space suit (unsuited). Motion tasks include arm and leg swing, arm reaching tasks, locomotion over even ground, stepping up onto a 30 cm step and down, and choreographed motion involving multiple joints. In addition, a hand positioning task will be performed with a two-link manipulandum, in which hand motions will be disturbed by forces provided by the manipulandum. Each task will be repeated 6 times. I understand that three-dimensional kinematics data of my body segment positions will be recorded as well as pressure between my body and the spacesuit.

IV. FORESEEABLE INCONVENIENCE, DISCOMFORT, AND RISKS TO THE SUBJECT
As a spacesuited subject, I will come into contact with the pressure suit and understand that my arms and legs may rub up against the suit. This is standard and causes no severe pain or stress. If I notice any persistent joint or muscular pain, I should disqualify myself.
from further testing. Delayed-onset muscle soreness may occur due to the motions performed as part of the experiment or from wearing the spacesuit.

V. RISK MINIMIZATION
Care has been taken in the experiment design to prevent injury in all phases of testing. Treatment for sore muscles or other injuries incurred from participation in this experiment will be available through the M.I.T. Medical Department, at the expense of the my insurance carrier where applicable.

VI. REMEDY IN THE EVENT OF INJURY
In the unlikely event of physical injury resulting from participation in this research, I understand that medical treatment will be available from the MIT Medical Department, including first aid emergency treatment and follow-up care as needed, and that my insurance carrier may be billed for the cost of such treatment. However, no compensation can be provided for medical care apart from the foregoing. I further understand that making such medical treatment available, or providing it, does not imply that such injury is the investigator's fault. I also understand that by my participation in this study I am not waiving any of my legal rights.

VII. COMPENSATION
I will receive no compensation for participating in this experiment.

VIII. ANSWERS TO QUESTIONS
I may receive answers to any questions related to this experiment by asking the test conductor or contacting the Principal Investigator at (617) 258-8799

IX. IN THE EVENT OF UNFAIR TREATMENT
I understand that I may also contact the Chairman of the Committee on the Use of Humans as Experimental Subjects, MIT 253-6787, if I feel I have been treated unfairly as a subject.

X. SIGNATURE
I, ________________, have read and understand the information (Subject's Printed Name)
contained in this consent form and agree to participate as a subject in this experiment.

(Parent/Subject’s Signature) (Date)

(Witness)