AN AGING MODEL OF NI-MH BATTERIES FOR USE IN HYBRID-ELECTRIC VEHICLES

A Thesis

Presented in Partial Fulfillment of the Requirements for
the Degree of Master of Science in the
Graduate School of The Ohio State University

By

Ryan H. Somogye, B.S.E.E.

*****

The Ohio State University
2004

Master's Examination Committee:

Dr. Stephen Yurkovich
Dr. Giorgio Rizzoni

Approved by

[Signature]
Advisor
Graduate Program in Electrical Engineering
ABSTRACT

The increasing consumer performance and federal emission demands for automobiles have forced improvements in every technological aspect of the vehicle. Now that hybrid-electric vehicles are in production, they immediately need a battery as advanced as the rest of the power train. Battery technology has lagged behind, so a model is needed that, in addition to dynamic effects, models how the batteries will age with the life of the car.

This document examines the characteristics of Ni-MH cells, the chemistry currently popular in hybrid vehicles, and explains the methods by which they degrade. Electrode deformation and chemical breakdown are most responsible for diminishing cell health. Popular models of batteries, those used for non-hybrid applications, concentrate only on dynamic effects instead of incorporating information about cell life. Even though these models fail in this area, there is a wealth of information available as to how the Ni-MH battery ages. This knowledge is in the form of personal experience with the Buckeye Bullet land speed car, researched data from the tests of others, and experimental data done on the Sanyo 2.6 amp-hour sub-C cell specifically for this topic.

A hypothesis is postulated that connects environmental conditions with the resulting effects in battery performance, weather it be a reduction in capacity, an increased internal resistance, or a change in transient behavior. To prove, or disprove, the
hypothesis, a specially built, computer controlled battery test bench conducts experimental testing over a variety of environmental and usage variables. This test bench implements algorithms to detect a full charge and record data, making the tests exactly repeatable. The results of this experimental data are analyzed and conclusions are drawn that directly support and simultaneous contradict different parts of the supposed hypothesis.

Classical behavior-based modeling techniques are used to construct electrical and thermal model of the Ni-MH cell. Fuzzy logic is then used to implement the connection between environmental conditions and aging effects. The resulting computer model can mimic the experimental tests and well as the expected behavior of the Ni-MH in a hybrid-electric application.

This resulting "aging model" moves toward an implementation of life data into an otherwise all-electrical model to provide a test forum for early evaluations of different power train algorithms. Future uses of the fuzzy-based aging model include reliability studies and future refinement with a greater experimental test base.
Dedicated to all the people that I ignored while getting this thing done.
ACKNOWLEDGMENTS

I wish to thank my advisor, Professor Stephen Yurkovich, for intellectual support in helping me organize and carryout the work presented here. I’d also like to thank Professor Giorgio Rizzoni, Director of the Center of Automotive Research, for supplying equipment and room to conduct the battery testing. Both faculty members of CAR have been great resources for advice during my 5-year participation there, from working on student project teams to a becoming a graduate student. I also was fortunate to receive a graduate fellowship, paid for by the Honda Partnership Program. This relieved me of quite an expense and made the seven quarters of graduate school very enjoyable.

I thank my parents, Ron and Lisa Somogye, for moral support and for taking the time to review my thesis, giving advice on its construction and correcting grammatical errors. I’d also like to thank Shannon, my fiancée, for the greatly needed emotional support.

Finally, I thank the Buckeye Bullet team for providing an educational platform from which my interest in battery technology originated. Working on the team as a young college sophomore was very rewarding, and directly influenced my decision to attain a graduate degree. I hope that this thesis serves as a guide to educate future team members on the care and design considerations of rechargeable batteries.
VITA

December 26, 1979 ........................................ Born — Akron, Ohio

2002 .......................................................... B.S. Electrical Engineering,

Ohio State University

2002 – present ............................................... Graduate Research Fellow,

The Ohio State University

Center for Automotive Research

FIELDS OF STUDY

Major Field: Electrical Engineering
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abstract</td>
<td>ii</td>
</tr>
<tr>
<td>Dedication</td>
<td>iv</td>
</tr>
<tr>
<td>Acknowledgments</td>
<td>v</td>
</tr>
<tr>
<td>Vita</td>
<td>vi</td>
</tr>
<tr>
<td>List of Figures</td>
<td>ix</td>
</tr>
<tr>
<td>List of Tables</td>
<td>xii</td>
</tr>
<tr>
<td>Chapters:</td>
<td></td>
</tr>
<tr>
<td>1 Background Information for Building a Model</td>
<td>1</td>
</tr>
<tr>
<td>1.1 General Battery Overview</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Battery Aging</td>
<td>4</td>
</tr>
<tr>
<td>1.3 Nickel-Metal-Hydride (Ni-MH) Specifics</td>
<td>8</td>
</tr>
<tr>
<td>2 Battery Modeling</td>
<td>14</td>
</tr>
<tr>
<td>2.1 Model History</td>
<td>14</td>
</tr>
<tr>
<td>2.2 Existing Models</td>
<td>15</td>
</tr>
<tr>
<td>2.3 Need for Improvement</td>
<td>26</td>
</tr>
<tr>
<td>3 Testing Strategies and Model Goals</td>
<td>28</td>
</tr>
<tr>
<td>3.1 Why Choose Ni-MH cells?</td>
<td>28</td>
</tr>
<tr>
<td>3.2 Concerns about the Ni-X cell</td>
<td>29</td>
</tr>
<tr>
<td>3.3 Forming a Hypothesis</td>
<td>35</td>
</tr>
<tr>
<td>3.4 Experimental Testing Strategy</td>
<td>37</td>
</tr>
<tr>
<td>4 Experimental Test Bench</td>
<td>40</td>
</tr>
<tr>
<td>4.1 Test Bench Introduction</td>
<td>40</td>
</tr>
</tbody>
</table>
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 1.1</td>
<td>Basic internal structure of a Ni-MH battery.</td>
<td>3</td>
</tr>
<tr>
<td>Figure 1.2</td>
<td>Cut-away Diagram of a Cylindrical Cell. [1]</td>
<td>6</td>
</tr>
<tr>
<td>Figure 1.3</td>
<td>Graph of a Ni-MH cell during discharge. [1]</td>
<td>10</td>
</tr>
<tr>
<td>Figure 1.4</td>
<td>Voltage vs. State of Charge for Ni-MH cells at Various Charging Currents [1]</td>
<td>12</td>
</tr>
<tr>
<td>Figure 1.5</td>
<td>Voltage vs. State of Charge for Ni-MH cells at Various Temperatures. [1]</td>
<td>12</td>
</tr>
<tr>
<td>Figure 2.1</td>
<td>Thévenin Model of a Battery</td>
<td>16</td>
</tr>
<tr>
<td>Figure 2.2</td>
<td>Voltage-Sagging Battery Model</td>
<td>18</td>
</tr>
<tr>
<td>Figure 2.3</td>
<td>Thévenin Model Output to a Step Current Load</td>
<td>19</td>
</tr>
<tr>
<td>Figure 2.4</td>
<td>Voltage-Sagging Model Output to a Step Current Load</td>
<td>20</td>
</tr>
<tr>
<td>Figure 2.5</td>
<td>Lockheed Martin Dynamic Model</td>
<td>22</td>
</tr>
<tr>
<td>Figure 2.6</td>
<td>Lockheed Martin Model 10-msec Time-Constant Step Response</td>
<td>23</td>
</tr>
<tr>
<td>Figure 2.7</td>
<td>Lockheed Martin Model 14-sec Time-Constant Step Response</td>
<td>24</td>
</tr>
<tr>
<td>Figure 2.8</td>
<td>Lockheed Martin Model Charge-Depletion Step Response</td>
<td>25</td>
</tr>
<tr>
<td>Figure 3.1</td>
<td>Ni-Cd battery performance after 2300 cycles [8]</td>
<td>33</td>
</tr>
<tr>
<td>Figure 3.2</td>
<td>Ni-MH performance after 1300 cycles. [8]</td>
<td>34</td>
</tr>
<tr>
<td>Figure 3.3</td>
<td>Dynamic Electrical Model of the Battery</td>
<td>39</td>
</tr>
</tbody>
</table>
Figure 4.1: Entire Battery Test Bench Setup .......................................................... 42
Figure 4.2: Battery Test Bench Main Panel .......................................................... 43
Figure 4.3: Battery-Holding Rig with Peltier Junctions ...................................... 44
Figure 4.4: Close-up of Battery-Holding Rig ...................................................... 45
Figure 4.5: Plot of Battery Voltage, Current, Temperature, and Best-Fit Polynomial .... 51
Figure 4.6: Plot of Standard Deviation Between the Actual Voltage and the Best-Fit Polynomials vs Present Cycle Time .......................................................... 52
Figure 4.7: Plot of the Energy, Amp-hour, and Efficiencies, With Respect to Cycle Number ............................................................................................................ 53
Figure 4.8: Best-Fit Polynomial Standard Error vs Time Shortly Before Charging is Complete ........................................................................................................... 55
Figure 4.9: Acquired Measurements of the Complete Charging Session ............ 56
Figure 5.1: Example Plot of Repeated Charge-Discharge Cycles ...................... 61
Figure 5.2: Example Plot of Energy and Amp-Hour Efficiencies vs. Cycle Number .... 62
Figure 5.3: Raw Recorded Data of an Example Parameterization Test ............... 66
Figure 5.4: Close-up of Battery's Transient Voltage Response During Test ........ 67
Figure 5.5: Resulting Model Output From the Best-Fit Parameters .................... 68
Figure 6.1: Arrangement of the Battery Simulator Subsystems .......................... 78
Figure 6.2: Electrical Subsystem of the Battery Simulator ................................. 80
Figure 6.3: Plot of Voltage and Temperature with dT/dt Indications During Charging [10] ................................................................................................................. 82
Figure 6.4: Thermal Subsystem of the Battery Simulator .................................. 84

x
Figure 6.5: Cycle Life vs. Depth of Discharge [13] .................................................. 90
Figure 6.6: Relationship Between Temperature and Cycle Life [14] ......................... 92
Figure 6.7: Relationship Between Ni-MH cycle age and Storage Capacity. [15] ......... 96
Figure 6.8: Rule Base Surface for Relating Age and Heat Exposure to Cell Capacity ... 97
Figure 6.9: Rule Base Surface for Internal Resistance .............................................. 99
Figure 6.10: Model Parameter Penalizer Subsystem Simulink Diagram ..................... 101
Figure 6.11: Fuzzy Inference System Editor .............................................................. 103
Figure 6.12: Membership Function Editor ................................................................. 104
Figure 6.13: Rule Base Editor ....................................................................................... 105
Figure 6.14: Simulated Base-Line Test Outputs ......................................................... 107
Figure 6.15: Simulated Base-Line Test Outputs ......................................................... 108
Figure 6.16: Simulated Base-Line Electrical Model Parameters ............................... 110
Figure 6.17: Simulated 50°C Test Outputs ................................................................. 112
Figure 6.18: Simulated 50°C Electrical Model Parameters ......................................... 113
Figure 6.19: Simulated 5% Depth of Discharge Test Results .................................... 114
Figure 6.20: Simulated 5% Depth of Discharge Model Parameters ......................... 116
# LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 5.1: Conditions and Criteria of Each Type of Cycle</td>
<td>59</td>
</tr>
<tr>
<td>Table 5.2: List of Benchmarking Results</td>
<td>70</td>
</tr>
<tr>
<td>Table 6.1: Fuzzy Membership Functions and Their Values</td>
<td>94</td>
</tr>
</tbody>
</table>
CHAPTER 1

Background Information for Building a Model

This chapter explains the basic operation of any battery and its role in a larger electrical system. The most important problems or limitations that occur when using batteries are also outlined. The ways in which a rechargeable battery age, via measurable effects on performance, are described. Specific information on Ni-MH cells, including how they function chemically, mechanically, and electrically, is given.

1.1 General Battery Overview

The characteristics of a battery depend widely on the mechanical design, chemical reactions, and materials used in designing the battery. To understand which qualities need to be modeled and how they interact with each other, the purpose and general construction of a battery must be examined.

The sole purpose of a battery is to store electrical energy for later use. Confusion sometimes results when a battery is viewed as a device that generates electricity. A generator is dynamic in its operation, but not so much in its lifetime. For example, as long as fuel is available, the engine will produce heat or mechanical power and a generator will convert it into electricity. A battery, however, changes as energy is drawn
from it. Like all Newtonian-physics-based systems, the battery can only convert between electrical and chemical energy. It is a bucket for electrical energy, not a source.

In addition, batteries are also subject to non-ideal conditions of the real world. This makes the changes experienced by a battery heavily based on the environment and more difficult to model. Even the largest battery under the smallest load experiences irreversible performance changes based on its charge level and age. A battery will never be as good as when it left the factory. System designers must be aware of this fact and should be able to model it. Now, with age, the bucket will develop holes, store less with its decreased volume, and have an opening that slowly shrinks. In this analogy, the holes represent self-discharging, or the slow, steady loss of charge even if the battery is not used. The decreased volume signifies that the battery cannot store as much energy as it could when it was new. The shrinking bucket opening, if the reader imagines storing a fluid, results in spilling more fluid as the bucket is filled and the restricted quick emptying of the bucket. In the same way, the cell's internal resistance causes a more inefficient charge and discharge as the age increases.

Figure 1.1 [1, p43] shows the basic structure of a Ni-MH battery. Since batteries store energy chemically, a basic knowledge of chemistry is required to understand their functions. Without getting into methods for optimizing performance, the components and functions are as follows. The two electrodes are submersed in an electrolyte solution and separated by mechanical mounting or an insulating layer. Each electrode provides the electrical interface from the battery terminal to the electrolyte. The electrolyte is an
ionic solution, usually aqueous, that provides a transportation medium for the material deposited on or removed from the battery's electrodes.

![Diagram of Ni-MH battery](image)

(Negative Electrode = Hydrogen-absorbing Alloy)  
(Positive Electrode = Nickel Hydroxide)

**Figure 1.1:** Basic internal structure of a Ni-MH battery.

Generally, as the battery discharges, the anode, or positive terminal, accepts electrons from the battery's load. To carry away the electrons, the anode, or a coating on it is chemically reduced to an ion, or charged atom or molecule, and strips away from the electrode into the electrolyte. At the same time, ions originating from the electrolyte are oxidized onto the cathode, or negative electrode. To insure that all ions can reach their destination electrode an ion-permeable insulating layer separates the anode and cathode. This layer does not intuitively seem to perform any insulating task since an ion can move.
right past it. However, its purpose is to prevent a direct mechanical connection between the anode and cathode, as this would be equivalent to shorting the battery.

This description assumes that the anode ion is negative, but the process is similar for a positive ion moving from the electrolyte to the anode. More complicated chemistries have several different reactions taking place simultaneously, yet the basic idea is to remove electrons from the anode and deposit them on the cathode. For a secondary battery, the process is reversed as electrons are reduced from the cathode and into the electrolyte to restore the chemical reaction to the original charged state.

1.2 Battery Aging

There are some important implications of this simple electro-chemical process that affect battery mechanics, and therefore, the model. The electrodes are consumed and deposited on during discharging and charging. When the ions from the electrolyte are redeposited onto the electrodes, either during charge or discharge, they tend to do so in a spherical shape. This is due to a sphere having the property of all points on the surface being equidistant from the center. Batteries with a chemical reaction that consumes a coating on the electrodes are more robust to such deterioration. For example, lead acid batteries react with a lead dioxide encapsulation instead of the lead electrodes. However, any battery's electrodes eventually converge toward a spherical shape and away from the square plate or rolled sheet form of which the cell was originally manufactured.

It would take extra energy, or non-random molecular activity, to cause the ions to redeposit in the shape of a plane or rectangle. However, this is exactly what the battery manufactures desire, since a sphere has the least surface area for its volume. Since the
specific power rating of a battery stems from how much surface area of the electrode is exposed to the electrolyte, the electrodes are manufactured with as much surface area as possible. Electrodes usually take the form of repetitive plates separated by insulating sheets or long strips layered between insulators and rolled up into a cylinder.

As the battery is cycled, applying pressure to the plates counteracts the natural convergence of the electrodes from plates or strips into spheres. In a prismatic battery, or rectangular-shaped cells, the ends are squeezed tightly to prevent ions from depositing in the center of the plate. In cylindrical cells the curved wall of the external container keeps the rolled electrode and insulator wound tightly (similar to the curved end of a tank that holds gas under high pressures). Even so, eventually the electrodes will decompose by successive stripping and re-depositing, leaving the battery with reduced specific power and capacity.

Figure 1.2 [1] shows how a cylindrical cell is commonly manufactured, with the electrodes separated by an insulator, rolled into a cylinder, and sealed in a nickel or nickel-steel can. The safety vent, located at the top center of the cell, vents if the internal pressure becomes dangerously high. Stamping the outside wall closed, over the positive cap, seals the cell.
Cylindrical Type

Figure 1.2: Cut-away Diagram of a Cylindrical Cell. [1]

Some types of batteries, depending on chemistry, will generate gas under certain conditions. The most common reason for gas generation is overcharging. When a battery is overcharged, there can be no more electrical energy converted into chemical energy. Instead of reversing its chemical reaction, the battery electrolyte undergoes electrolysis, or the splitting of water into hydrogen and oxygen by an electrical current. The common lead acid battery is a great example of this. When overcharged, the water
portion of the diluted sulfuric acid electrolyte mixture bubbles away. In a capped battery, the electrolyte level can be checked periodically. In a sealed battery, the hydrogen must re-combine with the oxygen over several charge-discharge cycles, or with the help of catalysts used in the tops of some lead-acid batteries, to return the battery to a nearer-perfect state. In each case, the capacity and power available from the battery has been permanently diminished and any battery management unit is helpless to correct it.

An example of permanent damage from gas generation is found in overcharging of a Ni-MH battery. Hydrogen gas is generated under normal operation since the cathode is a hydrogen-absorbing metal. However, during overcharging, the hydrogen is generated by both the cathode and electrolyte electrolysis. If ignored, the pressure inside the sealed cell can build (most Ni-MH and Ni-Cd cells are low-pressure sealed while in use). This inhibits the normal chemical reaction and will eventually trigger a safety vent (about 60 psi in small cylindrical cells). Once the gas is vented, part of the cell’s electrolyte is lost forever, thus permanently degrading performance.

While it is beyond the scope of this thesis to determine a charging algorithm for each type of battery, as good algorithms for each type of cell already exist, let it be noted that each type of battery chemistry has a specific charging algorithm to replenish the most energy in the least time while minimizing inefficiencies and battery damage. This algorithm is based upon numerous advanced cell design and manufacturing techniques and can be extracted from the battery’s datasheet or other technical papers. However, mysterious this algorithm may be, and they do appear overly complicated at first, it is an
essential part of using the battery correctly. Each charging algorithm will be described or referenced under the specific chemistry to which it applies.

1.3 Nickel-Metal-Hydride (Ni-MH) Specifics

Ni-MH cells are used in many portable high-power products, such as cordless drills and remote controlled toys. They lend themselves well to these applications due to a relatively constant voltage over the discharge range and the ability to be recharged within an hour, sometimes faster. They offer a 2-fold to 3-fold increase [2, p.37] in specific energy over lead-acid cells. Specific power is also increased over the common lead acid cell; however, the factor of increase is not as great and depends heavily on the specific model and manufacturer. The Panasonic Overview of Ni-MH Batteries [1] is a very comprehensive reference. The general ideas in it are applicable to all Ni-MH cells; however, the exact numbers and ratings are manufacturer and model specific.

A basic Ni-MH cell uses nickel metal as the positive electrode and a hydrogen-absorbing metal alloy as the negative electrode. The electrolyte is an aqueous Potassium Hydroxide (KOH) solution, providing mobility to OH- ions. The following chemical equations describe the charge and discharge process of a Ni-MH cell. Charge moves from the left hand side of the equation to the right, and vice versa for discharging. The letter “M” is used to signify the particular metal hydride molecule that absorbs the hydrogen. This is proprietary among different cell manufacturers but all metal hydrides react and bond in the same way.
Positive Electrode: \( \text{Ni(OH)}_2 + \text{OH}^- \leftrightarrow \text{NiOOH} + \text{H}_2\text{O} + e^- \)  

\[ (1.1) \]

Negative Electrode: \( M + \text{H}_2\text{O} + e^- \leftrightarrow M_{\text{H}}_{\text{absorbed}} + \text{OH}^- \)

During discharging, electrons are taken in by the positive electrode, producing free OH- ions at the electrode. These ions flow through the KOH solution, combining with the stored hydrogen from the negative electrode to produce water. Like other rechargeable chemistries, the electrolyte solution increases in molarity the higher the cell’s state of charge.

As a "brother" to the Ni-Cd cell, the two batteries share most of their charge and discharge characteristics. There are a few differences between the two, especially concerning charging. Ni-MH has a charging reaction that is overall chemically exothermic while Ni-Cd cells are overall endothermic [2, p38]. As a result, Ni-Cd cells better realize intense and repeated cycling than Ni-MH as both charging and discharging creates heat in Ni-MH cells. The "tell-tale" voltage signs of overcharging are less detectable in Ni-MH cells than in Ni-Cd cells, as discussed later in this section. Ni-MH cells also suffer from a higher self-discharge rate since the metal alloy has trouble retaining its hydrogen atoms over long periods of time (weeks to months) compared to a more stable bonding between H and Cd in a Ni-Cd cell. Finally, since the Ni-MH and Ni-Cd chemistries both use an aqueous electrolyte solution, this puts a clear lower limit on their temperature range. These cells have sever performance decreases at temperatures below 0 °C since the electrolyte starts to freeze, decreasing ion mobility.
The graph of a common Ni-MH cell under discharge is given in Figure 1.3 [1]. Notice that the voltage "levels off" during the middle of the graph. During this level period, the cell has reached a steady-state reaction, where the cathode can release H atoms quickly enough to satisfy the load’s consumption of OH⁻ ions at the positive electrode. The cathode exhausting its supply of absorbed hydrogen as the cell reaches a fully discharged state causes the decreasing voltage slope. One very important safety design in Ni-based cells requires that the cathode have less surface area and less hydrogen absorbing capacity than the anode. Since hydrogen gas is created at the cathode during discharge, it is important that the anode be able to absorb this gas faster than the cathode can create it. Both the detrimental effect of gas generation, as outlined in section 1.1, and the physical structure of the cell are at stake if the gas were to build up under pressure.

![Graph of a Ni-MH cell during discharge](image)

Figure 1.3: Graph of a Ni-MH cell during discharge. [1]
Alternatively, Figure 1.4 [1] shows the graph of the cell’s voltage under charging. Once again, after a few minutes of transient effects, the voltage reaches a stable level, with a slight positive slope. There are several important details about the charging curve that need to be pointed out. Most Ni-MH cells are charged with a constant current power supply, meaning the voltage steadily increases as the charging nears completion. Ni-MH cells must be monitored so that charging is ceased once certain criteria have been obtained.

As shown in Figure 1.4, the current used to charge the batteries has a significant effect on the terminal voltage (voltage once the cell is charged). The temperature and age of the cell also have a large influence on the terminal voltage. Due to these effects, the charger must monitor other values than just the cell voltage. The better chargers often monitor dV/dt, or the rate of change of the voltage, and cell temperature. Both plots show that the rate of increase of the voltage increases at the time when the cell achieves maximum charge. Cell temperature is also used to determine the cell’s maximum charging voltage, as shown in Figure 1.5 [1].
Figure 1.4: Voltage vs. State of Charge for Ni-MH cells at Various Charging Currents [1]

Figure 1.5: Voltage vs. State of Charge for Ni-MH cells at Various Temperatures. [1]
Since the original release of Ni-MH batteries, there have been improvements in the manufacturing and capacity areas although the general overall chemical reaction remains the same. Some cells marketed to consumers as “increased capacity” batteries have sponge-like or plastic nickel-coated positive electrodes. This increases the surface area, and therefore capacity, but at the price of maximum available power. The metal alloy used in the cathode has also improved. NiFe was one of the first alloys used. Now, for example, Panasonic is concentrating on LaNi5 and other alloys of similar bonding arrangements for their increased hydrogen absorbing ability.

At the time of this writing, most Ni-MH batteries are cylindrical or small prismatic in form and are targeted at consumer products (cordless drills, toys, other high-power portable devices). There are a few Ni-MH cells being constructed specifically for use in HEV automobiles. The chemistry remains the same; however, the ratio of electrolyte to electrodes is lowered providing more room for a stout electrode structure. These cells are mechanically designed for less energy storage but much higher power levels. While these cells are also Ni-MH in nature, their self-discharge rate, internal resistance, etc., are significantly different and need a unique model to accurately describe them.
CHAPTER 2

Battery Modeling

This chapter presents the most common existing battery models and the history behind them. The reasons for their construction, and thus their uses and shortcomings, are described. Lastly, it discussed why a model keeping track of cell aging could be useful in a hybrid-electric system.

2.1 Model History

The models of rechargeable batteries used today in the public domain were developed a number of years ago, mainly for portable electronics and remote telecommunication systems. These models mainly focus on available power, pulsed load behavior, and simulation of internal discrete elements. The designers using these models assumed that when the cell's energy is depleted, the device will stop operating or be plugged into an external power supply for charging. The goal is to use as little energy as possible or to use it most efficiently.

During that usage the device rarely gets a chance to return energy to the batteries. The charge cycle is important, and a set of rules has been developed for each type of chemistry, but every charge resets the device battery life back to 100%. Even if they are only partially recharged, the amount of recharge is not measured and the device only has
voltage to estimate the cell’s state of charge. Cell age is not considered, since the consumer will replace the rechargeable batteries, at a low cost, periodically. As a result, these models lack any aging effects.

Telecommunications installations using existing models are concerned with battery temperature (more so than portable electronics) but few experimental results have been incorporated into making a smart load, or a load that adjusts to battery temperature effects. Usually the maintenance intervals are shortened or the batteries are oversized when the environment is harsh. For HEV applications where battery replacement is costly (and inconvenient for the consumer) and brash over sizing of the battery is counterproductive, a model that takes deterioration into account is highly valuable.

2.2 Existing Models

The Thévenin model is the most basic of all battery models. One can understand why it was developed and why a resistor is “inside” the battery by the battery’s effect of dropping in voltage as more current is drawn from it. The nominal voltage (not defined as but close to the average open circuit voltage through the cells discharge cycle), or the calculated voltage based upon the cell’s chemistry, is set to be the magnitude of the ideal voltage source. Experimental data, taken at different current loads but similar charge levels, provides data to calculate a single lumped resistance. A best-fit line of the terminal voltage vs. current data pairs is calculated and its slope is the cell’s internal resistance. This model is used when the primary concern is maximum available power or how much energy might be lost at certain current levels.
The circuit in Figure 2.1 shows the ideal voltage source and internal resistance (Thévenin model) with a load resistance of 0.12 Ω (ohms) and control switch (the pSpice software [3] used to simulate the model requires two switches to be used - one to close and another to open). The battery model simulates values from a Sanyo Sub-C cell model HR-SC.

Figure 2.1: Thévenin Model of a Battery
An extension of the Thévenin model uses arrangements of resistors, inductors, and capacitors to further model transient effects of a changing load. Multiple stages of resistor-capacitor pairs simulate how the battery voltage stabilizes after a step change in loading current. An inductor can be added in series with the ideal voltage source to model the impedance effect of the cell.

In Figure 2.2 the voltage-sagging model has internal resistance and one resistor-capacitor pair. The steady-state internal resistance is the same as the Thévenin model, but now the voltage sag is modeled. This voltage sag does not reflect the change in battery voltage as the cell is discharged, but rather the slight change in internal resistance due to the transportation delay of electrolyte to and from electrodes. The sag is most prominent when a load is quickly applied or released. From visually interpreting the voltage plots or real batteries, an exponential decay can be used to model this sag, thus the resistor-capacitor pair forming the time-constant of the battery model.
Figure 2.2: Voltage-Sagging Battery Model

The response of the Thévenin model to a step current input can be seen in Figure 2.3. Simply, the output voltage of the battery decreases proportionately to the load current, specifically it drops 4mV for every amp drawn from the cell.
Figure 2.3: Thévenin Model Output to a Step Current Load.

The purple line plots load voltage, the blue line plots battery terminal voltage, and the yellow line plots load current.
The response of the Voltage-Sagging model is shown in Figure 2.4. Here, along with an instantaneous decrease in cell voltage due to internal resistance, the slight exponential decreases, due to ion transport delay, is modeled.

![Figure 2.4: Voltage-Sagging Model Output to a Step Current Load.](image)

The purple line plots battery voltage and the blue line plots load current.
In an effort to further model the dynamic effects of a battery, Lockheed Martin [4] developed a very detailed dynamic model for satellite electronics. The model is based around a 30V Ni-Hydrogen battery of unknown model or manufacturer, but still demonstrates well the particular points of the model.

This model, being a far extension of the Voltage-Sagging model, incorporates two capacitor-resistor pairs to model two time constants, an internal resistance tested under pulse load conditions, and two inductors to model series inductance of the battery under frequency loads of current. It also models the battery's capacitance for energy by replacing the ideal voltage with a 20,000-Farad capacitor. The model even incorporates a non-linear variable resistance in the first resistor-capacitor pair to incorporate the varying internal resistance with state of charge. This proves to be a significant advantaged over the Thévenin model whose values are held constant [5]. Figure 2.5 shows the Lockheed Martin dynamic model.
Figure 2.5: Lockheed Martin Dynamic Model

Figures 2.6 through 2.8 show the step response of the Lockheed Martin dynamic model. Each figure outlines the first RC pair, the second RC pair, and the actual charge depletion of the battery, respectively. This model does a great job of predicting transients when loading or recharging this Nickel-Hydrogen battery. The resulting voltage and current graph for all three models were obtained using the circuit simulation software pSpice by Orcad [3].
Figure 2.6: Lockheed Martin Model 10-msec Time-Constant Step Response.

The purple line plots battery voltage and the blue line plots load current.
Figure 2.7: Lockheed Martin Model 14-sec Time-Constant Step Response

The purple line plots battery voltage and the blue line plots load current.
Figure 2.8: Lockheed Martin Model Charge-Depletion Step Response

The purple line plots battery voltage and the blue line plots load current.
2.3 Need for Improvement

All three of these battery models use experimental data to generate values for the resistors, capacitors, and inductors that mimic the battery's real behavior. The latter even accounts for the discharging of the battery's energy and the non-linear change in internal resistance that accompanies this. However, none of the above models are intended to predict a battery's lifetime.

Another method to construct models of electrochemical systems is to build upon representations of the individual system interactions. The models shown in this chapter, as with the electrical subsystem of the aging model in chapter six, are based upon the behavior and phenomenon recorded from the battery. Behavior-based models are easier to construct and understand be those that are familiar with discrete electrical components. Since the intuition used in the construction of the model, the behavior-interpretation method is used.

Instead, one could develop equations from each chemical interaction and link the equations to form the input and outputs of the system. This model’s origin is not necessarily as evident as the behavior-based models, especially in the conversion of chemical reactions into equations. However, these models are better at predicting the actual voltage output of the cell. The presentation “The Battery as an Electric Circuit Element” [6] presents a model for Ni-H2 (Nickel Hydrogen) batteries and details the voltage plot of the pack during a spacecraft docking. This is a great example of how well a chemical model can match actual performance under seemingly random cycling conditions.
Aside from the space and military industries, few applications have a budget that warrants the complete testing of a battery, over the entire product’s lifetime, several times, to obtain experimental lifecycle data. Just as the general form of the model is known prior to experimental data and simple transient tests are conducted to find values for the components, the battery model of this thesis will attempt to use the data from a few simple tests and information about the cell to build a model predicting transient and lifecycle behavior.
CHAPTER 3

Testing Strategies and Model Goals

This chapter outlines, in much greater detail, the wealth of sometimes-conflicting intuitive data regarding cell use and care. It is then distilled into what is necessary to construct the model. The tests to obtain the desired experimental data are outlined and a hypothesis is formed as to what the experimental results will be.

3.1 Why Choose Ni-MH cells?

The end goal of the fuzzy-logic battery model is to incorporate lifetime prediction into the already proven transient models of Ni-MH batteries. This thesis concentrates only on Ni-MH cells for the following reasons. Lead-acid technology is far too heavy to be seriously considered in production hybrid vehicles. Modern Ni-Cd technology has similar performance as Ni-MH, but the environmental and disposal costs are greater. Ni-MH chemistry is considered the successor to Ni-Cd and has dropped in manufacturing cost to the point of feasibility and performance.

Li-Ion technology, while getting cheaper by the year, is still too expensive for mass production. At the time of this writing, the small number of production hybrid cars by Toyota and Honda has already depleted the battery inventory of their suppliers. Delivery times on new orders placed for these revolutionary vehicles are well over a year.
Both of these cars use cells based on the Ni-MH chemistry, which has been on the market since Sanyo released their first Ni-MH cell named Twicell in 1990 [7]. Li-Ion cells debuted in 1994, also by Sanyo, and have secured an important, high quantity, but small size and cost application in mobile electronic devices. While Li-Ion promises better performance, it will be some time before these cells are plentiful enough to supply the automotive industry with such a large manufacturing demand.

Li-Ion cells also present a unique hazard. Unlike Ni-MH and Ni-Cd cells that are sealed up until a certain pressure (say 60 psi, then a valve opens venting excess gas) Li-Ion cells have the potential to explode if overcharged. Since the testing method used in this thesis are purposely destructive in order to capture lifetime changes in behavior, the demand for a Li-Ion fuzzy logic model does not outweigh the complexity of the necessary testing equipment and the possibility of catastrophic cell failure.

Most importantly, the author has had several years of experience dealing with Ni-MH cells in the Buckeye Bullet. A choice of any other cell would leave the aging model short of intuitive interpretation of the experimental results and researched data.

3.2 Concerns about the Ni-X cell

In order to develop an idea of what tests need to be conducted on the cells, there must be some hypothesis as to what causes premature degradation of the battery. While there is plenty of hearsay advice from battery experts on how to get the most life out of your cell, no one seems to be able to quantify it. The next section concentrates on forming that hypothesis. However, as an example of why it is important to have that
hypothesis, this section familiarizes the reader with the story of the "memory" effect and the opinionated advice that often goes something like this:

1. "Make sure not to overcharge the batteries. Over charging creates gas and needless heat inside the cell."

2. "Avoid discharging the cell too much. The cell will be permanently damaged and cannot be restored by cycling."

3. "Always cycle the batteries periodically. If a battery is left to self-discharge too long, its performance will suffer."

4. "Try not to draw more current from the cell then recommended by the manufacturer. Placing too high of a load on the cell can damage the cell internally."

5. "Never let the cell get too hot or too cold. Temperature extremes can damage the cell's insulation and permanently degrade performance just as any electrical abuse of the cell.

6. "Always discharge the cell fully before recharging. If you don't, the cell will develop a 'memory effect', or only give you that much energy the next time you use it."
The last one is a personal favorite. The Ni-Cd and Ni-MH 'memory' effect is more of an urban legend than greasy food being the cause of acne. The true memory effect only occurs in sintered-plate (small-spaced grid-like sheet of nickel metal used for the positive electrode) Ni-Cd and Ni-H cells [8]. In fact, the exact situation to re-create the true memory effect has only been observed and documented once in the field. All other times have been under laboratory conditions.

The only time that the memory effect was documented in the field was in a satellite orbiting the earth at a constant and periodic rate. Almost all conditions were ideal, including absolutely no overcharging, no vibration, no human factor, etc. As the satellite orbited the earth, part of the time was spent in the sun, using the solar cells to charge the Ni-H batteries. The remainder of the time was spent using the batteries to power the satellite's on-board systems.

A computer was used to control when the systems were shutdown, due to battery depletion, and how the batteries were recharged once sunlight was available. Since the designers oversized the capacity of the batteries, normal satellite operation would drain the batteries to 25% state of charge on each cycle. After an extended time period of experiencing the exact same cycle, the Ni-H batteries started to form crystals on the electrodes. This crystallized electrolyte caused a drop in voltage at 25% state of charge. The computer mis-interpreted this voltage drop as fully depleted batteries and shutdown every other system. Once this pattern emerged, the satellite would stop functioning a few minutes before orbiting back into available sunlight. The term "memory" was born, since
the battery would remember when its discharging would cease (25% state of charge) and act as it had been depleted once that state of charge was reached.

Interestingly, any of the following conditions will completely cure the memory effect: overcharging, discharging to a statistically significantly different state of charge, or discharging below 1.0V/cell. Notice, from above, all of these situations are considered generally bad and the satellite’s computer was programmed to avoid them. The onboard computer was reprogrammed to purposefully overcharge (slightly) the Ni-H battery and the memory effect was slowly cycled away, saving the satellite. Since that famous satellite many years ago, laboratory tests have been able to quantify and qualify the “memory” effect. Reference 8 gives the remaining basic details of the “memory” effect and the reader is encouraged to find more information for better understanding.

It turns out that only large sintered cell Ni-Cd batteries, those used for power backup systems or telecommunication equipment, are susceptible to the “memory” effect. Even those cells have been improved beyond the original sintered sealed design and few memory-susceptible cells remain in use. No Ni-Cd, Ni-MH, or Li-Ion battery available to the consumer market today would be capable of developing memory. Often the “memory” effect is blamed for battery age, misuse, or something called “voltage depression”. Voltage depression is basically the quick dropping of voltage during a cell’s discharge. The total energy available remains the same; it is just available at a lower voltage level. There is too much conflicting information on the cause and remedy of voltage depression to accurately report on it. Most references point to it being cause by
very similar discharge cycles and being cured by one or a few complete discharge cycles. Again, the reader is encouraged to research independently and form an opinion.

Another piece of information that is slowly becoming more popular to see is how cycle depth relates to cell life. When Ni-Cd batteries matured in the consumer market, they were seen as replacement for alkaline batteries (in some applications). These rechargeable batteries had an initial high cost of the cell and charger, but could save on total battery costs if the device or toy was used often enough. Figure 3.1 [9] shows how a Ni-Cd battery fairs after 2300 charge-discharge cycles.

![Capacity, Self-Discharge and Internal Resistance of a 'standard' Nickel Cadmium Battery](image)

Figure 3.1: Ni-Cd battery performance after 2300 cycles [9]

Now that Ni-MH cells have become popular, they are marketed as a replacement for older Ni-Cd technology. Advertisers focus on higher energy densities, and they are
correct, as Ni-MH do store more energy in the same volume or for the same weight (approximately 30% more than modern Ni-Cd cells [2, p37]). However, not all characteristics of the Ni-MH cell are improvements over Ni-Cd. Figure 3.2 shows how the internal resistance, capacity, and self-discharge rate are related to number of cycles.

![Capacity, Self-Discharge and Internal Resistance of a Nickel Metal Hydride Battery](image)

Figure 3.2: Ni-MH performance after 1300 cycles. [9]

From the previous two figures, it is obvious that Ni-MH cells experience a greater decrease in performance as the number of used cycles increases. In fact, although this data should only be taken as qualitative because the exact cell models and conditions are not documented, it could be said the Ni-Cd cell would provide more energy storage over it’s lifetime than the Ni-MH cell. Therefore, the Ni-MH cell offers similar internal
resistance, similar self-discharge rates, and great energy capacity when new, these
advantages over a Ni-Cd cell disappear after approximately 500 cycles.

Basically, the Ni-MH cell is not a complete replacement for the Ni-Cd
technology. Its performance, although better than the Ni-Cd when new, degrades quickly
once the cell is subjected to numerous cycles. It has less than half the cycle life but does
not contain and hazardous materials (such as the cadmium in Ni-Cd cells). Some
interesting discussion can result from these observations. If the Ni-Cd cell is more
resistant to age (and higher temperatures concerning self discharge), then why are most
hybrid automobiles using Ni-MH? Does the disposal cost outweigh the possibility of
having to replace the hybrid pack twice as often? Do the automotive companies test the
pack to complete failure (this would take an extremely long time, considering that Honda
already offers an eight-year warranty on their hybrid batteries and the depth of discharge,
or how far the cell is discharged before recharging, is usually only +5% or -5%)?

What is most likely is that the research on Ni-MH batteries has grown while the
research on Ni-Cd has dropped, it being almost a fully mature and now outdated
technology. Ni-Cd batteries are still used in many manufactured products, such as
cordless razors, telecommunication installations, fighter jets, locomotives, etc. Not
having been part of the design, it is unknown whether this is for their low cost or lifetime
advantages over Ni-MH.

3.3 Forming a Hypothesis

Like the “memory” effect, the behavior of batteries is widely misunderstood and
rarely quantified. The story of the “memory” effect, combined with the hear-say
information and what the reader may have heard about Ni-Cd and Ni-MH cells shows just how important is it to have a hypothesis and provide an organized structure to the experimental data collecting of this thesis.

From numerous papers, personal experience, and primitive experimental data, there are some simple factors that should influence battery life and performance. These factors are rate of charge and discharge (current limits), operating temperatures, and depth of charge and/or discharge. Higher current applications have the reputation of being harder on a cell's expected lifetime. This seems obvious, but the reason for these experiments is to quantify this lifetime decrease. High temperatures and discharging too deeply are also causes of degradation that will be quantified.

Going beyond simple tabulation of the lifetimes of cells in different conditions, one of these attributes probably makes a greater difference that the others. The hypothesis formulated here, and hopefully proven later, is that operating temperatures and depth of discharge effect the cell's lifetime more than do higher currents. As a result, the batteries cycled under higher current loads should show less wear than the batteries cycled under higher temperatures or cells that were overcharged or over-discharged. Personal experience does not give enough information to make a distinction between temperature abuse and charge/discharge abuse at this time, as every situation in which these two factors were present, the cells had significantly diminished performance.

This might be good news. Designers could depend on good battery lifetimes given proper cooling and battery management units, as opposed to needless battery over sizing or significant design changes in fear that high currents mean the death of a cell.
3.4 Experimental Testing Strategy

To verify the hypothesis, experimental data on cell transient performance and energy storage capacity under different conditions will be collected. One of the three variables - temperature, current (rate of charge and discharge), or over discharge - will be changed during a given test. This provides the ability to construct a 3D matrix, where each entry is a set of data points for a battery tested under those certain conditions. The data points can be compared with other data points along a given dimension of the matrix, relating how cell lifetime is effected by that particular variable.

The experimental strategy will use four Sanyo cells; model HRSC-2.6Ah. Since there are three variables to be experimented across, three cells will provide a set of data points, one each, for the progressing values of the same variable. The fourth cell acts as the control cell (cycled for the same amount as the other cells, but at normal conditions). Each cell is benchmarked periodically after every 50 cycles of use. In each case, the manufacturer's data and the control cell will be used to provide additional data points.

Specifically, for example, the temperature and depth of discharge will be held constant for one of the experimental cells. The cycling current, however, will be varied away from the control conditions. The cells will all be subjected to a fixed number of full cycles. After cycling at the specific conditions, the cell's capacity and equivalent series resistance will be measured again at the baseline cycle (manufacturer's recommended use).

Every 50 cycles, two tests will be performed, one on model the battery's transient behavior and one to gauge its energy capacitance. A 4C and 8C (C being the amp-hour
capacity of the cell, xC is a way to normalize the current for comparing tests among other cells. Throughout this document, 1C represents 2.6A, 2C represents 5.2A, etc.) discharge cycle will measure the cell’s equivalent model parameters, as in Figure 3.3, at various states of charge. Those model parameters found by this transient test are internal resistance, first order resistance, and first order capacitance. A 1C charge and 1C discharge cycle, per the manufacturer’s datasheet recommendations, will measure the cell’s capacity, giving the equivalent energy storage capacitance of the cell.

The idea for this electrical model comes from a paper [10] estimating the deterioration of lead-acid cells. It is crucial to understand that the models presented in chapter two, while deemed insufficient in their neglecting of aging affects, do not lack in their transient response. In fact, the Lockheed Martin model of Figure 2.5 goes beyond the requirements of a hybrid-electric power train in predicting transient effects that have little bearing on motor performance. It is assumed that a large energy storage element, an internal resistance, and a single time constant provide sufficient mirroring of the actual battery voltage, as seen in section 5.2 where the benchmarking procedure is described. Any less would fail to capture the discontinuity at the beginning of a current load, the leveling-off curve directly after that, and the subsequent slight negative slope in cell voltage (seen later in Figure 5.4). A more complex of a model would not add significantly to the behavior of the voltage plot and is not warranted.
Figure 3.3: Dynamic Electrical Model of the Battery

The health of the cell will be reflected by the calculated model parameters and not the particular discharge characteristics of the one cycle. Comparing these parameters to those of a new cell and to a control cell will provide a quantifiable effect on cell health of the given cycle conditions. The trends in which these parameters change, for each type of test performed, will be used in constructing the model parameter penalizer subsystem. This data will verify (or disprove) the accuracy of the hypothesis in Section 3.3 and the nickel-based chemistry concerns of Section 3.2.
CHAPTER 4

Experimental Test Bench

This chapter is dedicated to describing the automated test bench built to acquire the experimental data. Along with Appendix B, a collection of MATLAB code relating to the test bench, can function as a user's manual for operating the test bench for future needs. This chapter explains the test bench capabilities, how the hardware of the test bench was designed, and how the MATLAB software was customized to cycle Ni-MH cells.

4.1 Test Bench Introduction

An automated test bench was used to acquire the experimental data. Since the cycling required to derive the experimental data is very tedious and highly affect by human error, a desktop computer running MATLAB will control the test bench. This ensures that all cells are tested equally except for the condition intended to be the variable.

The test bench basic function consists of the ability to cycle up to six Ni-MH or Ni-Cd cells in series at a maximum 40A charge (limited by power supply) and a 100A discharge while controlling the temperature within the range of 0C to 60C. The test bench receives two analog signals from the computer (using a DAQ card as explained
later) specifying the desired current and temperature. The signals are 0V to 10V signals with the scaling given in the schematic. Two other signals from the computer, digital TTL logic, are the current and temperature enable. Since analog circuit voltages vary slightly with noise and re-calibration, the enable signals allow the current and temperature drives to be completely and surely turned off to avoid trickle current or slight temperature deviations. The test bench also generates several analog signals that are sent back to the computer. These signals transmit actual current, actual temperature, and actual cell voltage.

4.2 Test Bench Layout and Hookup

Figures 4.1 and 4.2 are photographs of the test bench. As setup for this thesis, it is entirely mounted on the blue metal panel. The gold-color anodized aluminum heat sink in the center contains the MOSFETs (type of transistor - metal-oxide-semiconductor field effect transistor) to regulate charge and discharge current. There are four additional MOSFETs on the heat sink used to control the Peltier junctions (discussed later). The top binding posts (on the black plastic mount) are used to connect to the test bench power supplies and the cell(s) under test. The gray plastic box in the upper-right hand corner is the control circuitry. The analog signal conditioning and digital logic are both contained within this box. All signals eventually terminate or originate from this box. One of the connectors is used to interface with the DAQ card in the personal computer. The breakout card mounted atop the box allows the selection of DAQ card pins and connects to the DAQ card by a 50-pin cable (a regular SCSI internal hard drive cable will work). Refer to the schematic for complete pinouts and instructions.
Figure 4.1: Entire Battery Test Bench Setup
Figure 4.2: Battery Test Bench Main Panel

Figure 4.3 shows the custom-machined all-aluminum battery holder that incorporates the Peltier junctions. This battery holder can cool or heat the battery, depending on the direction of current applied to the Peltier junctions. It consists of dies that are custom machined for each battery size. This thesis only used Sub-C size cells. If other cells are tested in the future, a new set of dies can be machined to accommodate the size of the new cell. Dimensional drawings for the entire holder are contained in Appendix B.
The die is used to clamp in the Peltier junction by way of the machined and threaded mounting bars. Each side of the Peltier junction and the sidewall of the cell are coated with heat-sink compound (use regular, not silicon based grease). Once the Peltier junction, grease, die, and clamping bars are all secured (tighten screws slowly and in partial turns to avoid deformity, as the Peltier junctions are extremely brittle), the one side of the Peltier junction contacts the die, the other side contacts the aluminum base.
plate. Figure 4.4 shows a close-up of the cell and interchangeable holding die. During cooling, heat is removed from the die and transferred to this plate. Heating does the reverse.

Figure 4.4: Close-up of Battery-Holding Rig
On the opposite side of the base aluminum block mounts the heatsink and fan. Since Peltier junctions have a maximum temperature differential between the surfaces, it is important to keep the surface opposite to the cell close to room temperature. The large heat sink and fan, although not bleeding or absorbing large amounts of heat, keep the base-block side of the Peltier junction at room temperature, allowing the cell-side to achieve the lowest or highest possible temperatures.

The battery holder has several wires and connectors on it. These connect to the 12v supply for the fans, the Peltier controller output (to determine current and direction of Peltier junctions), and the thermistor(s) inputs. All the plugs are either uniquely labeled or polarized to avoid an incorrect connection. Note that the temperature sensors are glued to the cell die and not the cell itself to avoid replacing them each time a new cell is tested. This also gives a more accurate reading, and the sides of a cylindrical cell are more representative of its internal temperature than the ends. This is due to the internal space required for electrical connections and safety vents.

The test bench also has a few other external components that are worth mentioning. It requires a minimum of two power supplies. These supplies must be isolated from the wall to avoid a short circuit and noisy ground loops. Personal computers, including laptops, are not electrically isolated. Isolation is defined as no wire on the input to the power supply can be traced, directly or through non-isolated components, to the power supply output. Isolated power supplies usually have a large transformer, usually step-down, and often use linear regulation. Again, use only isolated power supplies to avoid ground loops through the DAQ card.
The first of the two power supplies is used to provide charging current for the cell under test. Since the MOSFETs operate in the linear region, providing a linearly regulated constant current charging supply, they generate significant heat. A more efficient and more expensive design might utilize a switching supply with improved filtering, but the linear method was chosen since the four small MOSFETs on the board are capable of 100A in either charge or discharge and were less than $10 each. In order to ensure that the MOSFETs have enough voltage difference to provide stable regulation without unnecessary heat, it is recommended that the charging power supply be set to constant voltage mode at 2V higher than the maximum anticipated voltage of the cell under test. The diode brick mounted on the gold heat sink allows two isolated power supplies to be hooked in parallel, tying their grounds to the main ground and their positive terminal to each respective diode, providing additional current for charging. Make sure that the power supplies are the identical make and model and set to the same mode and voltage.

The second power supply feeds the control logic, Peltier junctions, and fans. The control logic has an upper limit of 15V. This is due to the maximum gate voltage allowed by the MOSFETs. The second power supply, however, was set to 12V, as this is what the Peltier junctions and fans require. This power supply must also be isolated and rated at 6A for each Peltier junction used in the battery holder. In this case, the maximum consumption from the second supply was 12A, since two Peltier junctions were used.
The cell under test simply connects to the last two binding posts on the top block of the test bench (refer to schematic). The DAQ card connects to the 50-pin header on the top of the control board box. Since most DAQ cards no longer rely on a 0.1" spacing, 50-pin header, an adapter must be used. The National Instruments PC-ISA card used has a high-density 68-pin plug. An adapter from a PXI assembly was used to convert the plug styles from the high-density 68-pin to the standard 50-pin header. This pinout should remain the same with any 68-pin E-series National Instruments card.

4.3 Test Bench Control Program

The program to control the operation of the test bench runs on a personal computer equipped with MATLAB 6.5.1 and the Data Acquisition Toolbox. There is no reason why the program cannot be self contained, but this would have required programming a user interface, building drivers, an interface for the DAQ card, and compiling it from a suitable language. These tasks were unnecessary to construct a working test bench and would only be done if the test bench were a marketable product. For now, however, the program must be run within MATLAB.

The program is capable of cycling the battery under test while varying (or holding constant) the charge current, discharge current, temperature, depth of charge, and depth of discharge, all at constant values or pre-programmed vectors. All conceivable limits, such as maximum/minimum voltage, temperature, current, etc. are user-settable as constants in the main program file. While charging the cell, the program can detect when the cell is charged and can terminate the charge. During discharge, the limit used to trigger end of cycle is a minimum voltage limit, often set from 0.9V/cell to 1V/cell.
Every ten seconds (also user adjustable), for the duration of the test, the program generates a plot of cell temperature, voltage, current, and the best-fit polynomial (explained later). A second plot shows the error of the best-fit three-degree polynomial during the charging portion of the cycle. A third plot shows the resulting charge/discharge energy (Whrs) efficiency and amp-hour (Ahrs) efficiency for each completed cycle, and is updated at the end of each charge-discharge cycle. The efficiency plot can be used to determine how much more energy and Ahrs are required to charge the cell than is available during discharge. The efficiency of the cell is not simply judged by the energy-efficiency of the cell, since it might result that an aged cell has a higher internal resistance but is still capable of holding close to its original amp-hour rating. This would result in a much greater decrease in energy efficiency than amp-hour efficiency.

The program is divided into two concurrently executing parts, one running "behind the scenes" to sample data, record data, and update the DAQ card outputs (desired current and desired temperature) and the other running in the command prompt to process the sample data. Defining a function called "single sample" that performs all the tasks necessary for one complete sample does this. The MATLAB timer calls this function on a user defined sampling interval (MATLAB term, refer to "help timer", T_sample = 1 second).

The "single sample" function, when called periodically by a timer, first samples all inputs, calculates the measurement values from those inputs, and appends them onto
large matrices in the MATLAB memory. It then writes the sampled values to a text file, storing them on the hard drive for later analysis and in case of a computer crash. Finally, the function updates the DAQ card outputs based on variables determined during live data analysis.

The second concurrent process analyzes the sampled data and makes decisions on detected patterns. This process determines when charging begins and ends, when discharging begins and ends, calculates cycle statistics for later analysis, and generates, every ten seconds, the three plots.

During charging, a specialized detection algorithm, which is explained later, determines when the cell is fully charged. This allows overcharging to be avoided completely, reducing the amount of experimental error from an unknown amount of overcharging, or done purposefully, allowing it to be controlled as an experimental variable. During discharging, the analysis code simply waits for the battery voltage to reach a given threshold, 0.9 volts/cell when held constant, to stop a discharge session. After the discharge has ended, the software then computes and displays the absorbed and returned energy and amp-hours of the cell, along with respective efficiencies.

Example plots gathered from testing used to de-bug the test bench are shown in the following figures. Figure 4.5 shows the plots of voltage, current, temperature, and best-fit third-degree polynomial. Figure 4.56 shows the standard deviation error, between the best-fit polynomial and the actual battery voltage. Figure 4.7 shows the calculated efficiencies from each cycle, plotted with respect to the cycle number.
Figure 4.5: Plot of Battery Voltage, Current, Temperature, and Best-Fit Polynomial

The y-axis applies to all four plots: current is plotted in amp, temperature is plotted in °C, best-fit polynomial is dimensionless.
Figure 4.6: Plot of Standard Deviation Between the Actual Voltage and the Best-Fit Polynomials vs Present Cycle Time

The x-axis is time in seconds, the y-axis is standard deviation.
Figure 4.7: Plot of the Energy, Amp-hour, and Efficiencies, With Respect to Cycle Number

The y-axis is in watt-hours for charge and discharge energy, amp-hours for charge and discharge Ahrs, and dimensionless for efficiencies.
4.4 Charge Detection Algorithm

The most unique feature of the test bench software is its ability to detect a 100% charge. It uses a customized algorithm to detect curvature changes in the graph of the battery voltage to precisely terminate constant current charging with overcharging or excess heating of the cell. Without accurate detection of the complete, yet not excessive, charging of a cell, the experimental results could be skewed by not isolating one variable in each test.

This algorithm generated the coefficients of a third degree polynomial that best fit the voltage of the battery under charge, from the start of a cycle (a complete cycle always begins with charging, followed by discharging of the cell) up to the present time. Since the cell voltage first appears to be an exponential curve during the first few minutes of charging, the best-fit polynomial is able to approximate the voltage curve with less and less error as time passes and the cell charges.

During the middle of a charging session, the battery voltage appears less like an exponential curve and more like a third degree polynomial as the voltage fails to level off and instead increases at a constant slope. This is more like the behavior of a third degree polynomial than the exponential curve, so the error further decreases. The time when the error between the best-fit polynomial and the actual battery voltage is the least is when the cell approaches a full charge and the voltage curves upward, increasing at an increasing rate. Figure 4.8 shows the point where the error is at its lowest.
Figure 4.8: Best-Fit Polynomial Standard Error vs Time Shortly Before Charging is Complete

The x-axis is time in seconds, the y-axis is standard deviation.

Shortly after this, the cell reaches a full charge and the voltage begins to level off. This is the second inflection point of the battery voltage graph and cannot be approximated by a third-degree polynomial. The error between the best-fit polynomial and the actual battery voltage increases sharply. This is very easy to detect, as the algorithm compares the present error with the lowest error achieved throughout the
charging session. Once the present error is 110% of the lowest error (a modifiable threshold), charging stops. Figure 4.9 shows the measurements and calculated error after charging has been detected and terminated.

![Graph showing acquired measurements of the complete charging session.](image)

**Figure 4.9: Acquired Measurements of the Complete Charging Session.**

The x-axis is time in seconds, the y-axis is standard deviation.
This method allows a charge termination point to be determined based on the
characteristic shape of the voltage graph, and not an arbitrary voltage of \( dv/dt \) limit.
Charge termination also cannot be determined by battery temperature since it is held
constant by the Peltier junctions during the tests. It also allows overcharging to become a
test variable, since it can be controlled and measured. The software allows for a pre-
defined overcharge and a percent of the complete charge session.

Although the MATLAB program is currently designed for Ni-MH cells, future
users can expand it to accommodate other battery chemistries and different tests.
Basically, the hardware amounts to a programmable power supply or load that can be
controlled through a National Instruments DAQ card and a desktop PC. The software is
used to automate any testing, provide safety limits, and record the required data. The test
bench has a life beyond this thesis and will stay part of the Battery/Super Capacitor Lab
at Ohio State’s Center for Automotive Research.
CHAPTER 5

Experimental Results

This chapter explains the cycling conditions of the test cells along with example test bench graphs of these cycling sessions. The exact benchmarking procedures for model dynamic parameters and cell capacity are outlined. The experimental results are presented and interpreted by how they relate to previously known intuitive knowledge and what effect they might have on the construction of the fuzzy-logic model.

5.1 Summary of Cycling Conditions

Each of the four (three experimental and one control) Sanyo HRSC-2.6Ah cylindrical sub-C cells was fully cycled approximately 100 times (the exact number of cycles is tabulated later). Table 5.1 shows the exact conditions of each type of cycle, the commencing and termination conditions, and the values of the control variable during that cycle. The discharge session of each cycle was terminated at a minimum cell voltage, as shown in Table 5.1. The charging portion, however, was terminated using an algorithm to detect the second inflection point of the battery voltage, as graphed since the charging began. The details of this algorithm were explained in Section 4.4.
<table>
<thead>
<tr>
<th>Cycle type</th>
<th>Charge Current</th>
<th>Charge Termination</th>
<th>Discharge Current</th>
<th>Discharge Termination</th>
<th>Temperature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control</td>
<td>5.2A</td>
<td>Algorithm</td>
<td>10A</td>
<td>0.9V at cell</td>
<td>25 °C</td>
</tr>
<tr>
<td>Heat at 50°C</td>
<td>5.2A</td>
<td>Algorithm</td>
<td>10A</td>
<td>0.9V at cell</td>
<td>50 °C</td>
</tr>
<tr>
<td>Discharge at 8C (C=2.6A)</td>
<td>5.2A</td>
<td>Algorithm</td>
<td>20A</td>
<td>0.9V at cell</td>
<td>25 °C</td>
</tr>
<tr>
<td>Discharge to 0.5V/cell</td>
<td>5.2A</td>
<td>Algorithm</td>
<td>10A</td>
<td>0.5V at cell</td>
<td>25 °C</td>
</tr>
</tbody>
</table>

Table 5.1: Conditions and Criteria of Each Type of Cycle
Figures 5.1 and 5.2 show example screen shots of the plots generated by MATLAB during each cycle. Figure 5.1 shows battery voltage, battery current, battery temperature, and the best-fit third-order polynomial used for the charge termination algorithm. Figure 5.1.2 shows the charging energy, charging amp-hours, discharging energy, discharging amp-hours, the energy efficiency, and the amp-hour efficiency of each cycle, plotted against cycle number. Notice that the average charge and discharge amp-hours is significantly lower than the cell's rated 2.6Ah capacity. This is due to the discharge load being a 4C (four times the rated capacity of 2.6Ahrs, or 10.4Amps) constant current. The cell terminals reach the cutoff, 0.9V, before the cell is completely depleted. To avoid possible damage, the test is stopped at 0.9V.

To cycle the batteries quickly enough, and obtain results for this thesis, a constant load is used. If the load was turned off and time was let to pass, the battery voltage would rise as more ions from the electrolyte accumulated at the electrodes (this happens by diffusion, replacing those that were depleted right at the electrode surface). This would allow more energy to be discharged from the cell, but would greatly increases the time required for one cycle. It would also bring the cell too close to over-discharging, and for those two reasons, is not done. As a result, the average cycle capacity is less than the rated 2.6 amp hours.
Figure 5.1: Example Plot of Repeated Charge-Discharge Cycles.

The y-axis applies to all four plots: current is plotted in amp, temperature is plotted in °C, best-fit polynomial is dimensionless.
Figure 5.2: Example Plot of Energy and Amp-Hour Efficiencies vs. Cycle Number.

The y-axis applies to all six plots: energy is plotted in watt-hours, Ahrs in amp-hours, efficiencies are dimensionless.

5.2 Summary of Benchmarking Tests

As stated earlier (Section 3.4), the experimental strategy includes two types of benchmarking tests to be performed at regular 50-cycle intervals. The first test, applying a constant 2.6A load on the cell, determines its capacity, or the duration at which it can
flow its rated IC current. Normally the energy storage capacity of a cell is the most important measurement used in determining battery size for a particular application. However, since this measurement is an integration of the power available over time (power being cell voltage times cell current) the internal resistances affect available energy. The battery model parameters are measured using a separate test, so care must be taken to measure the cell's capacity without the ability of the modeled circuit elements affecting the results. Also, since the energy storage characteristic of the battery is modeled using a very large capacitor, the equation for a capacitor directly relates charge movement (known current during a measured time) to the voltage change and capacity rating. These two reasons result in a charge-integration method of measuring the battery's capacity while ignoring cell voltage.

The protocol for the capacity test requires that the cell be discharged to 0.9V at 2.6A. After the condition is satisfied, it is considered to be a fully discharged cell. The reason for this condition is somewhat technical and somewhat arbitrary. 0.9V/cell is the recommended and documented (battery manufacturer's datasheet, Appendix A) lower limit to avoid damaging the cell. The current of 2.6A is the arbitrary portion, as some fixed current must be chosen to make the test repeatable. If 100mA were chosen, harm might come to the cell from being over discharged. If a higher current were chosen, such as 10A, the cell terminals would reach 0.9V with substantial energy left in it.

After fully discharging the cell, it is charged for exactly 1.1 hours at a fixed current of IC, or 2.6Ahrs, as recommended by the manufacturer. It is then immediately discharged at 2.6A until the voltage reaches 0.9V again. This is timed and is the result of
the test. This test is performed with a laboratory-grade power supply capable of regulating voltage and current and timed with a stopwatch. There is not an example of this test in this section, as it would be redundant. Look to Section 5.3 of this chapter to see the tabulated results of this test on the real test cells.

The second test assigns values to the model parameters of the cell's equivalent circuit model. This is done by applying a step current load to the cell under test and recording the output voltage on a digital storage oscilloscope. The cell under test is first discharged completely, by the same conditions used in the capacity benchmark test, and then charged for 10 minutes at 2.6A. The cell is then subjected to step current loads of 10A and 20A and step current sources of 5A and 10A for a few hundred milliseconds (the time is not exact as it is controlled manually by the user). After the waveforms are downloaded from the scope to a computer and saved, the cell is charged another 10 minutes and the step sources and loads are repeated. This is done until the cell has been charged to full capacity via six 10-minute sessions. This gives six states of charge for the four tests, providing 24 total data sets for each cell per benchmarking session.

A MATLAB program (find_model_parameters.m) processes the voltage and current waveforms (voltage coming from cell terminals, current coming from the voltage across a known resistance) to find the best-fit three model parameters of internal resistance, first order resistance, and first order capacitance. This program uses brute processing force to find the parameters that give the least error between the model and raw data, accounting for a constant voltage slope of charging or discharging the cell, and using a decaying exponential that simulates what the equivalent circuit output would be.
Figure 5.3 and Figure 5.4 show the total recorded raw data and a voltage close-up used for the parameterization test as plotted by MATLAB. Figure 5.5 shows the raw battery voltage and the resulting best-fit model output, via the equation:

\[
V_{\text{best fit}} = V_{\text{open circuit}} + I_{\text{load}} \cdot (R_{\text{internal}} + R_j \cdot (1 - e^{-\frac{t}{R_j C_i}})) + \int_{0}^{t} \frac{I_{\text{load}}}{C_{\text{capacity}}} dt \quad (5.1)
\]

When zoomed in on the raw recorded data (Figure 5.4), notice how the voltage drops instantly (internal resistance), then drops further by a decaying exponential (first time constant generated by the first resistance and first capacitance), until the exponential disappears and the slope is a constant decrease as the cell discharges (modeled by the capacity capacitor, the name given to the very large capacitor in Figure 3.4.1 that represents the cell’s capacity to store energy).
Figure 5.3: Raw Recorded Data of an Example Parameterization Test

The y-axis applies to both plots; battery voltage is plotted in volts, current in amps.
Figure 5.4: Close-up of Battery's Transient Voltage Response During Test

The y-axis applies to both plots. This figure is zoomed-in to the voltage, plotted in volts.
5.3 Experimental Results

The experimental results cannot simply be found, recorded, and entered into the model, especially a model using fuzzy-logic implementation. Interpretation of the results and how to use them is critical in developing a good model - one that can both be usable despite the time and data gathering restrictions of this thesis and could accommodate data acquired later.
First, consider the results in table form, for each benchmarking session. Table 5.2 shows these results from the benchmarking tests previously described. The model parameters that are listed as averages were calculated based on several sets of data taken at each benchmarking session. As discussed above, bad data (values that simply could not be correct based on intuitive knowledge of the cell) was ignored. The remaining valid data was then averaged to determine one of value to represent each model parameter. In most cases, the bad data was found to be in the form of a first capacitance value that was too small, yet gave the least standard deviation as calculated by MATLAB. This is the reason that the parameter test was repeated several times for each cell and benchmarking session. The first resistance and internal resistance values for all tests were coherent with duplicate tests and as expected by intuition.
<table>
<thead>
<tr>
<th>Test cell</th>
<th>Testing Cycles</th>
<th>Capacity Test Result</th>
<th>Average Internal Resistance</th>
<th>Average First Resistance</th>
<th>Average First Capacitance</th>
</tr>
</thead>
<tbody>
<tr>
<td>New Cell</td>
<td>0</td>
<td>2.287 Ahrs</td>
<td>4.78 m Ω</td>
<td>3.03 m Ω</td>
<td>27.4 Farads</td>
</tr>
<tr>
<td>Control Cell</td>
<td>52</td>
<td>2.452 Ahrs</td>
<td>4.94 m Ω</td>
<td>2.54 m Ω</td>
<td>31.1 Farads</td>
</tr>
<tr>
<td>Heat at 50C</td>
<td>65</td>
<td>2.170 Ahrs</td>
<td>23.4 m Ω</td>
<td>3.33 m Ω</td>
<td>19.7 Farads</td>
</tr>
<tr>
<td>Discharge 8C</td>
<td>62</td>
<td>2.156 Ahrs</td>
<td>4.83 m Ω</td>
<td>2.53 m Ω</td>
<td>27.85 Farads</td>
</tr>
<tr>
<td>Discharge to 0.5V</td>
<td>64</td>
<td>2.373 Ahrs</td>
<td>4.48 m Ω</td>
<td>2.50 m Ω</td>
<td>19.8 Farads</td>
</tr>
<tr>
<td>Control Cell</td>
<td>102</td>
<td>2.438 Ahrs</td>
<td>5.05 m Ω</td>
<td>2.77 m Ω</td>
<td>20.9 Farads</td>
</tr>
<tr>
<td>Heat at 50C</td>
<td>116</td>
<td>2.172 Ahrs</td>
<td>33.4 m Ω</td>
<td>3.03 m Ω</td>
<td>24.5 Farads</td>
</tr>
<tr>
<td>Discharge 8C</td>
<td>112</td>
<td>2.297 Ahrs</td>
<td>5.52 m Ω</td>
<td>2.78 m Ω</td>
<td>26.4 Farads</td>
</tr>
<tr>
<td>Discharge to 0.5V</td>
<td>115</td>
<td>2.370 Ahrs</td>
<td>5.45 m Ω</td>
<td>2.74 m Ω</td>
<td>23.8 Farads</td>
</tr>
</tbody>
</table>

Table 5.2: List of Benchmarking Results

Next, examine results qualitatively. The new cell performs close to datasheet specifications (see Appendix A) of capacity, providing 2.287 Ahrs (the minimum is 2.3Ahrs, the marketed 2.6Ahrs is the typical capacity). The other cells have capacities
close to the minimum, but vary between other cells. In fact, the capacity of one single cell varies very little as it is tested. It can be concluded that any given cell is manufactured with a slightly different capacity than any other given cell. The data shows that the average capacity is 2.29 Ahrs with a deviation of about 100 mAhrs.

This is a good time to point out a serious consideration in building battery packs. When most battery packs are assembled the cells are wired in series and they are treated as a single battery of increased voltage. When the pack is drained, one cell is discharged first. At this point, the pack should be recharged. Most of the damage to battery packs occurs when it continues to be used beyond this point. The output voltage of the pack is slightly reduced; say by 10% if there are 10 cells, so the load fails to recognize this and continues to draw current. The other cells that are not fully discharged begin to drive the discharged cell into a reversed voltage state. That cell is permanently damaged and the pack will never perform the same again. In other words, the pack is only as strong as its weakest cell.

Now that hybrid vehicles are becoming popular, technical discussions of their battery packs often focus on equalization, or the effort to bring all cells in the pack to the same state of charge. In systems where the pack is periodically fully charged, equalization can be obtained by slow, purposeful overcharging. Industrial lead-acid-based (a chemistry very tolerant of overcharging) equipment, such as electric forklifts, often has equalization settings on their chargers. This setting simply adds 10% to 20% more time to the charging sequence.
In applications where packs are often completely discharged, and under high currents, such as R/C models, higher quality packs come with matched cells. Each cell is subjected to a one-cycle test that measures some basic performance metrics, like capacity and equivalent internal resistance. The cells are then grouped into good, better, and best categories. Packs are made from only good cells, or only better cells, etc, and sold at increasing prices. The best pack available has only the best cells used in its construction, with all cells giving the best capacity and becoming fully discharged at times closer to all other cells. This makes the drop off in pack voltage very apparent and less effort will be taken to discharge the pack further, preventing harm to the weakest cell. In the same way, although the “good” packs are the least expensive and are guaranteed to have capacities that are sub-average, they will have a greater cycle life since the matching will help prevent the reversing of the weakest cell in the pack.

However, in hybrid applications, the desired change in state of charge is on the order of 5% to 10%, not fully discharged and then fully charged. This greatly extends the life of the pack, but more importantly, makes equalization by slow overcharging impossible. It would also not be feasible to sort through all the batteries and classify them as good, better, or best. In the hobby industry, the quantities are low and this is done by hand at the local hobby shop.

To equalize hybrid packs, several different methods have been devised, but they can all be divided into two categories, passive and active. Passive equalization systems strategically discharge individual cells, usually into a resistive load. The cells that are most charged are purposefully discharged so that they have the same state of charge as
other cells. The entire pack is then charged more often to counteract this wasting of energy. Active systems require much more hardware but do not waste the excess charge on cells. Instead, an active system might locate the cell with the highest state of charge, drain some off into a storage capacitor, and deposit it into the cell that has the lowest state of charge. This cycle then repeats until the cells with the highest and lowest states of charge are sufficiently similar.

The next important pattern to notice from the table is the internal resistance and first resistance of the new cell. The datasheet claims that the internal resistance of the cell is 4 mΩ, which coincides with the reading in the internal resistance column. Note that the datasheet claims that the 4 mΩ internal resistance is measured by impedance testing (not steady state). Feeding a current sine wave into the cell and measuring the RC or RL network behavior is one form of impedance testing. For this thesis, the internal resistance is found using a step current load and voltage sampled quickly by a digital storage oscilloscope. In any case, the steady-state equivalent series resistance would be the internal resistance in series with the first order resistance, or 7.82 mΩ. This illustrates the clear difference between a manufacturer’s claim and what the user might see in a real application.

Continuing to examine the internal resistances and first resistances, one notices that the internal resistance column changes noticeably between the different cells tested. However, the first resistance column changes very little. As discussed earlier, during the battery overview of Chapter 1, some of the battery’s internal equivalent resistance comes from the interaction between the electrode surface and the electrolyte. This resistance is
less as the cell is first put under load since the ions in the electrolyte had have time to migrate to (or from) the electrode surface. As the load continues, this resistance increases as the spent ions need to escape from the electrode surface and new ions move in. This spending of the ions on the immediate surface of the electrode can be modeled as a first order system, represented in the model and acquired data by the first_resistance and first_capacitance values.

None of the harsh-environment tests had any significant effect on the first_resistance value, aside from the distribution around an average that is inherent to experimental data. The first_capacitance, although subject to increased deviation from an average, is also interpreted as being constant within the conducted tests. The deviation of first_capacitance between tested cells, as shown in the results in Table 5.2, is equal to the deviation, of the same variable, encountered when running repeated tests on the same cell. For this reason, it is interpreted that none of the tests had an effect of the time constant of the dynamic electrical battery model.

However, there were some interesting effects on the internal resistance parameter. Obviously, heat adversely affects the cell the most. Internal resistance was increased approximately five times in the cell held at 50 °C than in any other cell. Both the cell discharged at twice the normal rate and the cell discharged to half the recommended termination voltage fained equally with the control cell.

It is assumed here that if the 50 °C cell were to be tested further, that its internal resistance would increase as a leveling-off exponential. In fact, it is assumed that all
changes to the model parameters, whether a worse parameter would mean a larger model value or a smaller model value, would do so in the fashion of the following equation:

\[ a(1 - e^{-by}) \]  

(5.1)

The constants \( a \) and \( b \) are general gains. This is the simplest approximation that the data can support, as the internal resistance value for the 50 °C cell jumped 500 percent during the first 65 cycles, and then jumped another 43 % during the next 54 cycles.

The hypothesis stated in Section 3.3 was partially correct, at least for cells in the earlier stages of use (around 100 cycles). Heat was proven to be very detrimental. Although still not recommended, the lower termination voltage of 0.5V / Cell seemed to play no role in destruction of the model parameters. Obviously more testing time would be ideal, but such an undertaking would be months, if not years long. This is one of the main reasons that the fuzzy-logic model, discussed in the next chapter, is based both on experimental data conducted specifically for this thesis, data obtained through research, and personal intuition of Ni-MH cells.
CHAPTER 6

Building a Fuzzy-Logic Battery Simulator

This chapter outlines the details of the fuzzy logic battery simulator, including the subsystems used to simulate the electrical, thermal, and parameter penalizing aspects of the model. Either interpretations from the experimental data, researched outside information, or cited intuitive knowledge is used to justify the construction of each model part. An example of how a human designer would interpret the experimental results is given, and then fuzzy logic is summarized as a way to "insert" that decision-making information into the simulator. Finally, the fuzzy-logic battery simulator is subjected to some basic tests to verify that, first, it successfully implements the intuitive knowledge used to construct the rule base, and second, that the simulator behaves as a real cell would.

6.1 A Global View of the Model

The first step in forming a battery simulator is determining which parts of it will be "fuzzy" and which parts will be traditional. Often a model is divided into separate subsystems, each one modeling a certain aspect of the entire system and relaying its particular outputs on to the other subsystems. This battery simulator will have three parts: an electrical model, a thermal model, and model parameter penalizer. Figure 6.1
shows how the three subsystems interact and share information. The electrical model is used to predict the output voltage based on the cell's current state of charge and current input. The thermal model predicts the cell's temperature based upon the heat generated by the internal resistances, thermal capacitance, and thermal resistance to an ambient temperature. Fuzzy logic is used in the model parameter penalizer to determine how and when model parameters are adjusted.
Figure 6.1: Arrangement of the Battery Simulator Subsystems
The electrical model is shown in Figure 6.2. It can also be found in Appendix C with the complete Simulink diagrams and m-file code. The output voltage is simply the sum of three sections of the electrical model; the voltage of the capacity capacitor, the voltage drop over the internal resistance, and the voltage drop over the RC time constant network. The capacity capacitor, although redundant, is named for its representation of the capacity of the battery. It has an initialization value of 9360 Farads and has a voltage ranging from 0V to 1V, representing a 0% to 100% cell state of charge. Since a Ni-MH cell does not have a linear voltage to state of charge relationship, a 1-D lookup table is used to represent the cell's open circuit voltage at a given state of charge.

The first order resistance and capacitor are intertwined with each other in such a way that the cell's current is split between the two. In this way, the model can simulate the voltage drop across the RC network. It was decided to only approximate the first-order behavior of the cell since that is what is intuitively seen on a voltage-versus-time response to a step input. Any higher order approximation, although more accurate, would have only a slight impact on the design of a system that uses such batteries. The time required to separate those parameters would more than double due to greater testing and analysis of the results.
The electrical model also models the generated heat due to voltage drops across internal resistances. The power dissipated by R_internal and R_first are summed and fed to the Heat Power Out terminal. A great deal of heat can be generated as the cell approaches a charged state, or if the cell is already charged and given a positive current. Figure 6.3 [11] shows how the slope of a plot of the cell’s temperature can increase sharply at the exact moment the cell reaches full charge. To account for this behavior, the electrical model also monitors the state of charge, and through a look-up table derived from Figure 6.3, generates additional heat power output when appropriate. Essentially, the entire electrical model becomes a resistor when overcharging occurs, as an actual cell does in reality.
Figure 6.3: Plot of Voltage and Temperature with \( \frac{dT}{dt} \) Indications During Charging

[11]

The thermal subsystem, shown in Figure 6.4, is a straightforward integrator of electrically wasted power into battery temperature. It simply calculates the difference between the generated heat (from the cell’s internal resistance) and the heat dissipated into the ambient surroundings and adjusts, through an integrator, the battery temperature accordingly. The output of this subsystem, cell temperature, is used both as a simulation result and as an input to the model parameter penalizer.

Since the thermal subsystem accounts for the heat lost to the cell’s surroundings, the thermal model requires that ambient temperature and the thermal resistance to that ambient environment as inputs. This way, a designer might mimic the presence of a fan
by switching between two thermal resistances, the higher one when the fan is off, and the
lower one when the fan is on. For this thesis, all simulations will be done assuming one
cell is in free air and that it cools by convection only, and not through its terminals. The
corresponding ambient thermal resistance is given by the formula:

\[ Q = h \cdot A \cdot (T_{batt} - T_{amb}) \]  \hspace{1cm} (6.1)

where \( Q \) is the heat flow in watts, \( h \) is the convection coefficient (W / m\(^2\)-C), \( A \) is the area
of the cell’s surface, \( T_{batt} \) is the battery temperature, and \( T_{amb} \) is the ambient sink
temperature. Since the actual calculation for \( h \) varies with material type, solid
orientation, and relies on other numbers that can only be found precisely by
experimentation, it is assumed that \( h = 15 \), a reasonable value from general published
information about heat transfer by convection [12]. The heat transfer coefficient of the
cell is then found to be:

\[ C_{o e f \_ c o n v e n t i o n} = h \cdot A = 15 \cdot (\pi \cdot d \cdot l + 2 \cdot r^2) = 0.064 \frac{W}{^\circ C} \]  \hspace{1cm} (6.2)

The thermal mass of the cell is held constant at 2.32 J / (g-C), derived from the
mechanical mass of the cell, given in Appendix A, and an assumed composition of half
nickel metal 0.46 J / (g-C) and half water 4.18 J / (g-C).
Figure 6.4: Thermal Subsystem of the Battery Simulator
The third subsystem determines how the electrical model’s parameters are altered in response to harsh conditions. This is the subsystem in which fuzzy logic is used, so the relation of parameter deterioration to cell conditions is developed by building a rule base. The model parameter penalizing subsystem uses the experimental results, data collected from other sources performing their own experiments, and intuitive knowledge from experience with the Backeye Bullet Ni-MH pack. The following sections describe the type of qualitative observances that need to be incorporated into the model and how fuzzy logic functions to achieve this.

Fortunately, Simulink (MATLAB) 6.5.1 comes with a fuzzy logic toolbox, making the construction of membership functions and rules much less complicated. However, just by looking at the model layout, presented later in this chapter, it is impossible to know the contents of each fuzzy rule set. For a better understanding of how the fuzzy blocks combine with the rest of the classic model (sums, integrators, interpolation, etc) the reader should open the model on a computer and examine the fuzzy blocks individually.

6.2 How to Make Use of the Test Results

If an engineer were to be presented with the data table constructed in Chapter 5, he or she would have a pretty good idea of how to design around the Ni-MH chemistry’s limitations. For example, if a mobile device was being designed around the battery, it could very well implement a high-speed digital wireless transceiver that would operate in bursts, as high currents on the battery are not too harmful of battery life. Making the tradeoff between device size and expected battery life depends on the minimum

85
acceptable discharge voltage. Being familiar with the cell's discharge voltage curve, the engineer might chose the 800mAh cylindrical cell over the 750mAh prismatic cell while raising the shutdown voltage of the device from 0.9V/cell to 1V/cell. This increase in battery capacity probably offsets the increases in terminal voltage, but costs more to purchase and install in the device. The engineer justifies this cost by the cycle-depth versus cell life intuitive data presented in Chapter 3. Without the time to truly test the change, their intuition tells them this is a good bet to take, estimating that the longer replacement times should more than offset the larger battery's expense.

Finally, the designer warns the warranty department that the greatest enemy of the device, beside physical damage, is heat. He or she instructs them to be very cautious about replacing devices left in automobiles on hot days. He or she also instructs the literature department to add many warnings about proper environmental care of the device. From the results in results in Chapter 5, heat would cause an increase in internal resistance that would undersupply the transceiver of power during burst operation.

All of the decisions presented in this fictitious scenario are based upon the designer's intuitive knowledge. This knowledge, in turn, is based upon data that he or she has seen or collected from many different sources, including personal experience. The problem in keeping this information from wider and quicker access is that this experience needs to be captured and stored in some form that can be easily processed by computers. This is the job of a fuzzy inference system.
6.3 Summary of Fuzzy Logic

Fuzzy logic, or rule-based logic, is a method of capturing intuitive data for use in computer programs, such as live or simulated decision making. The term “fuzzy" describes the inability, or inaccuracy, of classifying items or situations into discrete groups.

A fuzzy inference system, or rule-based decision-making system, is built around a set of rules, such as the rules that the facetious engineer followed in specifying the battery and behavior of the mobile device. These rules are written directly by the designer or can describe recognized patterns in gathered data. In general, a rule might be, "If this then that" or "If input x is A and input y is B then the output is C." However, since fuzzy logic is designed to model the human knack for indecisiveness, an input could be partially A or mostly B.

Membership functions are used to determine the likelihood that a given input can be classified into a certain group. These are simple, one-dimensional, functions that relate the input’s value to the probability (of the range 0 to 1, membership function output) of the input being classified in that particular group. For example, one possible membership function is a Gaussian distribution. If a public survey were taken for the question, “What is the height of an average human male?” then most would reply 6ft. Some might reply 5ft 10 in and some might reply 6ft 2in. Very few, but more than zero, would mostly likely reply 5ft 8in or 6ft 4in, and so on. A Gaussian distribution could be used as the membership function to the group “male of average height.”
Once the values of the inputs have been assigned probabilities of being in the distinct groups used in the “if” portions of the rules, also called antecedents, they are now ready to be processed by the rules. Most rules use an OR or AND operator to link two or more inputs to one output. A simple way to mathematically implement an OR is to take the maximum value of the membership values. For example, if “x is 0.5 A” and “y is 0.2 B”, the result of “IF x is A or y is B” is 0.5. Conversely, an AND operator can be implemented by taking the minimum of the membership values.

Finally, the single resulting values given by each rule are summed and defuzzified. Defuzzification, as the name implies, results in a real number as the systems output, which can be assigned a physical unit. There are several defuzzification methods, the most popular being center of gravity. This method first applies an output membership function to each rule output. Such a function determines the fixed range of the output and what real number might represent half of the output, two-thirds of the output, etc. However, the result of this membership function is a region on a graph, not a number. Each region, generated by the same output membership function and the unique result of each rule, is then overlaid to form an irregular shape. The center of gravity of this shape (or center of area for two dimensions) is then found, its x-axis coordinate being the crisp result of defuzzification.

Since the connection of environment to cell age is based on intuition and indirect interpretation of data, the application of the fuzzy process is very appropriate. It has been used successfully to facilitate other internal-state based estimations, such as state of charge [13, 14, 15]. It provides the simulation with a powerful tool to define situations
that are "between the rules", just as a designer has to read between the lines of the collected data. In fact, a great deal of the existing information about how batteries age comes in the form of "If this test, then that aging". A more detailed description and several good examples of fuzzy inference systems can be found in Fuzzy Control [16], a comprehensive look at everything fuzzy, or in the MATLAB Fuzzy Toolbox User Guide (quick summary).

6.4 From Test Results to Fuzzy Rule Base

The next step in building the model, once the electrical and thermal subsystems have been designed, is to determine a way to quantify the cell's age and exposure to heat. As stated in Chapter 5, these characteristics are the significant contributors to cell degradation, a reduction capacity and an increase in internal resistance.

To quantify the cell's age, its relationship to depth of discharge must be understood. The depth of discharge is directly related to the cell's age, as seen from Figure 6.5 [17, p.41]. Any secondary battery type has structural changes in the electrodes with every cycle. Cells that are cycled with a high depth of discharge deteriorate quickly due to constant stripping and re-depositing of the solid electrodes. Batteries that are only subjected to small changes in their state of charge can last much longer than those that are fully cycled, well past the point of providing a larger gross energy storage vessel over its lifetime. Hybrid vehicles take advantage of this property by using only a small percent of the battery pack's stored energy. As shown by the graph of data taken from Ni-MH cells, their packs can last an order of magnitude or longer than the datasheet specifies.
Figure 6.5: Cycle Life vs. Depth of Discharge [17]

To keep track of how deeply the cell is discharged, part of the model parameter penalizer determines how much the state of charge of the cell changes before current is reversed and the cell is charged. This is done within a custom s-function, a way for an m-file written in MATLAB to be used as a block in Simulink. The s-function records the previous ten (adjustable) state of charge values, or the voltage on the capacity capacitor, when the current polarity changes. Once ten values are accumulated, the standard deviation is computed, along with the number of samples, and given as function output.
During the accumulation of samples, the function outputs the value "-1", an imposable value of standard deviation for any set of real numbers.

The "dod_fun.m" output is then fed through a 1-D lookup table representing the information from Figure 6.5. The output is inversed and multiplied by the number of samples, thus representing the fraction of the cell's age lost to the previous ten cycles. A counter, representing the life of the cell and initialized at a value of 1, is then reduced by this output fraction. Thus, the cell loses a greater fraction of its life due to heavy cycling.

Heat exposure must also be tallied. It is known from personal experience that heat has an adverse effect on cell performance, even as the cell is simply stored in a warm environment. The model currently does not account for the effects of heat while the cell is not being cycled. A look-up table or fuzzy inference system could easily be added later to transform temperature and idle time into decreased state of charge. During each time step of the simulation, the change in state of charge of the cell is fed into two running counters, one of which is multiplied by the output of Figure 6.6 [18]. The ratio between the two counters will be unity if the cell has had no exposure to heat and 0.4 if it has been exposed to 50 °C from the beginning of the simulation.

Figure 6.6 [18], from testing results presented in a Duracell technical bulletin, outlines the relationship between cell temperature and reduced capacity. The graph shows that ambient temperatures above 20C (68F, close to room temperature) start to affect the cell. At temperatures beyond this, the cycle life decreases linearly with temperature. Although this graph only relates cycle life to temperature, it shows that below a certain temperature, the cell health is relatively unaffected.
Once the age and heat exposure have been quantified, the penalizer subsystem can use fuzzy logic to determine how the cell’s capacity and internal resistance are affected. Two fuzzy inference systems are used within the model penalizer, one for the cell’s capacity and one for internal resistance. Both use five membership functions for input and output. Their values, both qualitative and linguistic, are given in Table 6.1. The input and output value ranges for the membership should be self-evident, coming from
the graphs presented on the previous pages. The internal resistance output values increase by a power of 2. This was done since most human perceptions of intensities, or system properties are exponential in nature and most of the information behind the internal resistance rule base comes from intuition.
<table>
<thead>
<tr>
<th>Input Value</th>
<th>Membership Function Number</th>
<th>Linguistic Variable Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5 (Gaussian)</td>
<td>New</td>
</tr>
<tr>
<td>0.75</td>
<td>4 (Gaussian)</td>
<td>Slightly Used</td>
</tr>
<tr>
<td>0.5</td>
<td>3 (Gaussian)</td>
<td>Moderately Used</td>
</tr>
<tr>
<td>0.25</td>
<td>2 (Gaussian)</td>
<td>Heavily Used</td>
</tr>
<tr>
<td>0</td>
<td>1 (Gaussian)</td>
<td>Completely Used</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Heat Exposure (Input)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
<tr>
<td>0.85</td>
</tr>
<tr>
<td>0.7</td>
</tr>
<tr>
<td>0.55</td>
</tr>
<tr>
<td>0.4</td>
</tr>
</tbody>
</table>

Table 6.1: Fuzzy Membership Functions and Their Values
<table>
<thead>
<tr>
<th>Output Value</th>
<th>Membership Function Number</th>
<th>Linguistic Variable Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>100%</td>
<td>5 (triangular)</td>
<td>Full Capacity</td>
</tr>
<tr>
<td>75%</td>
<td>4 (triangular)</td>
<td>Slightly Reduced</td>
</tr>
<tr>
<td>50%</td>
<td>3 (triangular)</td>
<td>Moderately Reduced</td>
</tr>
<tr>
<td>25%</td>
<td>2 (triangular)</td>
<td>Heavily Reduced</td>
</tr>
<tr>
<td>0%</td>
<td>1 (triangular)</td>
<td>Fully Reduced</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Internal Resistance (output)</th>
</tr>
</thead>
<tbody>
<tr>
<td>100%</td>
</tr>
<tr>
<td>200%</td>
</tr>
<tr>
<td>400%</td>
</tr>
<tr>
<td>800%</td>
</tr>
<tr>
<td>1600%</td>
</tr>
</tbody>
</table>

Table 6.1: Fuzzy Membership Functions and Their Values (continued)

Next, a rule base must be constructed to link the input and output membership functions. Figure 6.7 [19, p2] shows how the Ni-MH cell’s capacity decreases with cycle life. A standard measurement for all batteries is expected cycle life. This is defined as the number of cycles which the cell can endure before the capacity falls to 80% of its
original value. Figure 6.7 confirms the information found in the Panasonic Ni-MH overview datasheet, claiming that Ni-MH cells under the “right conditions” can last more than 500 cycles.

![Graph showing the relationship between Ni-MH cycle age and storage capacity.]

Figure 6.7: Relationship Between Ni-MH cycle age and Storage Capacity. [19]

More importantly, notice that the graph has an increasingly negative slope. As the cell ages, more capacity is lost with each progressive cycle. By examining Figure 6.6 again, the results of the two could be combined (multiplied) to describe how the cell’s capacity decreases with age and certain amounts of exposure to heat. In other words, at average temperatures above 20C, the exponential decrease in the cell’s capacity increases, causing a roll-off in capacity sooner with higher exposure to heat. The rule
base governing the reduction in capacity is given in Appendix C, as it is lengthy. The
same information can be shown in a 3D mesh of the fuzzy logic surface, as in Figure 6.8.

![3D mesh of fuzzy logic surface]

Figure 6.8: Rule Base Surface for Relating Age and Heat Exposure to Cell Capacity.

Finally, the rule base for the cell’s internal resistance must be determined. This
is based more on intuitive knowledge and test results than the rule base for capacity, and
less on available outside testing information. The test results show that the cell exposed
to heat during cycling had an internal resistance that soared, relative to the other cells.
Even after only 100 cycles, 20% of the expected lifetime, the internal resistance was 5
times that of other cells. That particular cell also experienced a mechanical change. After cycling, the cell bottom became concave (opposite a soda can), as if pressure had built up and deformed the cylinder wall. This all suggests that some irreversible damaged was caused quickly (within the first part of its life) by the exposure to heat.

Also, while working on the Buckeye Bullet’s battery pack, experience has suggested that heating a cell, no matter its age, will cause an increase in internal resistance and a decrease in available power. This is especially harmful in a large battery pack, as the hotter cells develop a greater equivalent resistance, leading to more heating and a self-serving cause of failure. It seems that a cell can be only slightly cycled, but if exposed to heat, greatly deteriorated. Internal resistance increases slightly as the cell ages and capacity diminishes, but it can also be pushed along very quickly by heat. Figure 6.9 shows the rule base surface used to penalize the cell’s internal resistance.
Figure 6.9: Rule Base Surface for Internal Resistance.

6.5 Applying the Rule Bases to the Battery Simulator

Figure 6.10 shows the final resulting Simulink diagram of the parameter penalizer subsystem. Both current and state of charge voltage are fed into the “dod_fun.m” s-block, which, as discussed earlier, computes an average change in state of charge, or depth of discharge of the cell. This information is then fed, at periodic intervals, through switching logic and the depth-of-discharge to life-cycle-decrease table, to the cell health counter. The state of charge information and temperature are fed simultaneously to two
running counters, whose ratio represents the amount of heat the cell has been exposed to. This ratio becomes the exposed heat input to the fuzzy logic system.
These two generated inputs go on to drive the fuzzy inference systems that control the "aging" of the cell. Figures 6.11 through 6.13 give screen shots of the fuzzy inference system construction tool of MATLAB. Figure 6.11 shows the main option window where inference system inputs and outputs are determined. The interface also offers options in defuzzification, and/or methods, etc. Figure 6.12 demonstrates how membership functions can be manipulated, assigned shapes, and given linguistic values. Finally, Figure 6.13 displays the rule base window, where input and output membership functions are linked, with various options, to complete the inference system.
Figure 6.11: Fuzzy Inference System Editor
Figure 6.12: Membership Function Editor
The entire model parameter penalizer subsystem combines lookup tables, record keeping, and fuzzy logic to determine how the parameters of the electrical subsystem worsen with use. By dividing the entire model into three sections, changes to any property of the battery simulator can be made easily and without the worry of altering the other subsections.
6.6 Verifying the Fuzzy-Logic Battery Simulator

To verify the fuzzy logic battery simulator this section compares the simulator results with the acquired experimental data from chapter 5. The simulator is also put through a shallow depth of discharge test, similar to those used to benchmark Ni-MH cells intended for use in hybrid power trains.

Figures 6.14 and 6.15 (divided into two figures, the upper graph showing the cell early in the simulation, and the lower graph showing the cell near the end of the simulation, when the capacity has been reduced and internal resistance increased) show the simulator outputs for a +5A, 1500 second duration, -10A, 750 second duration square wave (blue line). The test mimics the conditions of the control cell test, with a temperature constant at 25 °C and repetitive constant current cycling. The test subjects the cell to an 80% depth of discharge, as outlined by the \( V_{\text{cap}} \) (voltage of the capacity capacitor) line in green. The red, cyan (light blue), and yellow lines, respectively, show cell output terminal voltage, temperature, and generated heat in watts.
Figure 6.14: Simulated Base-Line Test Outputs

The y-axis applies to all five plots; current is plotted in amps, terminal voltage and capacity-cap voltage in volts, temperature in °C, Pheat in watts
Figure 6.15: Simulated Base-Line Test Outputs

The y-axis applies to all five plots; current is plotted in amps, terminal voltage and capacity-cap voltage in volts, temperature in °C, Pheat in watts

The simulated output terminal voltage of Figure 6.14 resembles that shown by the real battery, as in Figure 4.3.1, when exposed to the same current waveform. Since the temperature was held constant in both cases, experimentally by the Peltier junctions and virtually by a greatly decreased thermal resistance to ambient, the temperature plots in each case hold at 25 °C.
Figure 6.16 shows the plot of the cell capacity and internal resistance, both normalized to one. It also shows the heat ratio and cell health counter. The cell health counter keeps track of the effective cell age from the “dod_fun.m” function’s calculation of depth of discharge. The heat ratio, ranging from 1 to 0.4, represents the cell’s exposure to heat, from none to full, respectively. The simulation agrees with intuitive knowledge that the cell should last several hundred cycles and its internal resistance should raise steadily until that point. In the simulation, the end of life is seen at 550 cycles when the cell capacity takes a sharp dive and internal resistance climbs. As for the experimental data, it verifies the result, both at 50 and 100 cycles, that capacity remains relatively the same, while internal resistance is slightly greater. Refer to Table 5.3.1 for the exact experimental values.

The effects of the reduced capacity can be seen in Figure 6.15 in the form of overcharging and over discharging. Overcharging causes the cell to produce a large amount of wasted heat, as it can no longer absorb and convert electrical energy in chemical. The peak charging voltage of the cell cycled 50 times is 1.4V, correlating with its datasheet. Peak charging voltage of the used cell, around 550 cycles, hits 1.5V, displaying the effect that increased internal resistance has on charging efficiency.
Figure 6.16: Simulated Base-Line Electrical Model Parameters

The y-axis applies to all four dimensionless plots.

Next, the simulation is run at 50 °C, again with a greatly decreased resistance to ambient to mimic the conditions of the experimental cell held at that temperature. Figure 6.17 shows the model’s output after 100 cycles at 50 °C. Notice that the charging voltage is higher and the discharging voltage is lower, meaning that the internal resistance (seen
from Figure 6.18) and generated wasted heat are up significantly compared to the previous simulation.

Capacity, while decreased to 80% after 100 cycles in the simulation, was hardly affected in the experimental data from the same test. Even though intuition and personal experience drove the model's rule base to reduce capacity in this manner, it certainly conflicts with the gathered data. Hopefully, if more testing were to be done, the capacity of the cell under a real test would fall off soon, as the model predicts. Whatever the case, the model successfully predicts that cell performance is greatly affected by slight (compared to the other temperature ranges that occur in automobiles) changes in temperature. Even if the cell retained most of its capacity, it would be useless as a source of specific power due to the increased internal resistance.
Figure 6.17: Simulated 50°C Test Outputs

The y-axis applies to all five plots; current is plotted in amps, terminal voltage and capacity-cap voltage in volts, temperature in °C, Pheat in watts.
Figure 6.18: Simulated 50°C Electrical Model Parameters

The y-axis applies to all four dimensionless plots.

Finally, a shallow depth of discharge test is conducted to ensure that the simulator gives reasonable results. To estimate what an implemented vehicle control strategy would demand of the battery pack, a current waveform of 10A for 5 seconds, -10A for 5 seconds, and a 25 second rest period, repeated 130,000 times, was applied to the simulator. Figure 6.19 shows the simulator outputs. The temperature of the battery holds
at a steady state value of 30 °C, 5 °C higher than ambient. This shows that even a small
duty cycle load on the battery pack can necessitate a good cooling system.

![Plot of Electrical and Thermal Model Outputs](image)

Figure 6.19: Simulated 5% Depth of Discharge Test Results

The y-axis applies to all five plots; current is plotted in amps, terminal voltage and
capacity-cap voltage in volts, temperature in °C, Pheat in watts.
Figure 6.20 plots the model parameter, demonstrating the huge impact depth of discharge has on expected cell life. Even though the 5% depth of discharge was 1/16th the amount the cell was subjected to in the two prior tests, it retained 80% of its capacity to the 110,000th cycle. This is 200 times the life expectancy of the 80% depth of discharge cycle tests, meaning that the cell lasts much longer for each percent discharged at a shallow, rather than a deep, depth of discharge. The internal resistance is larger than the base-line test of Figures 6.14 to 6.16, by about 30%, during the middle of the cell’s life. This is due to the realistic thermal resistance to ambient. Overall, this test confirms what hybrid-electrical battery pack manufacturers have proven [20]; with proper cooling and a very shallow depth of discharge, a battery can withstand several hundred thousand cycles.
Figure 6.20: Simulated 5% Depth of Discharge Model Parameters

The y-axis applies to all four dimensionless plots.

From this point, the fuzzy logic battery simulator could be easily inserted into a total vehicle simulator, such as VP-Sim, to see how an increased internal resistance and reduced capacity affects the vehicle performance and economy. Conversely, the battery management unit could be examined by measuring how well it sacrifices battery health to achieve performance and economy. Be aware that the 100,000-cycle simulation required several hours of computational time on a 2.1Ghz Athlon XP desktop machine. Expect
simulation to increase significantly when the aging model is used inside a total vehicle simulator, especially for life-cycle tests of the battery pack.
CHAPTER 7

Summary and Conclusions

This chapter summarizes the work presented in this thesis and comments on the important contributions. Chapter 7 then discusses possible improvements to the simulator and other ways to solve the problem in using lagging battery technology in hybrid applications.

7.1 Summary

In chapter one the electrical, chemical, and mechanical constructions of common Ni-MH cells are examined. The chemical reactions within the Ni-MH cell are centered on the cathode’s unique ability to store hydrogen in a lattice. This differs from older technologies, where hydrogen is stored as part of a repeatedly formed and deconstructed molecule. The electrical characteristics are illustrated in Figures 1.3 and 1.4, showing the discharge and charge curves, respectively. Mechanically, the cylindrical cell consists of an anode, insulator soak with electrolyte, and a cathode rolled up and packaged into a steel cylinder. These interactions between internal components, such as electrodes, electrolyte, and external casing provide a basis to understand how a battery “ages”.

An analogy, comparing a cell to a bucket that stores particles of energy, gives the reader who might not have a great deal of experience with batteries an idea of how
dynamic they are in operation. Aging effects like reduced cell capacity, raised internal resistance, and increased self-discharge translate directly into shrinking of the analogous bucket, both in volume and opening size, and the creation of holes and a leaky bucket, respectively. The information presented in chapter one, except for the analogy, is simply a summary of the working of a rechargeable battery. It may not along contribute much new information, but it is essential in understanding the rest of the thesis.

The second chapter presents a history of existing models and the results of their simulation. Most of the models available concentrate, in progressively more detail, on transient effect of quick changes in the load. These models can easily be augmented to be as accurate as the application requires by simply adding a time constant, cell inductance, or more dimensions across which to vary the parameters. A very well designed model [2], using several dozen discrete circuit components to model the chemical reactions within the cell, mimics the output voltage perfectly. It still has no provisions for cell age.

The most important contribution that this thesis makes is the connection between cell simulation and “virtual age” of the model. In the past, a dynamic model was created per the demands of the specific application. To insure that the cell would last the desired lifecycle, it was tested under the exact conditions of that application. When a new application came along, a new dynamic model, if needed, was designed and a new round of testing was conducted. In lieu of the hundreds of possible battery management algorithms that a hybrid-electric drive train is capable of supporting, a model that also simulates age could provide a first-stage filter for those algorithms that show promising
results. The fuzzy-logic based battery simulator is not meant to replace real-world
testing, but could greatly reduce development and service time.

Chapter three outlines the specific technical goals of the fuzzy simulator by
analyzing the wealth of intuitive knowledge from both the writer and research. It first
outlines that the simulator is based on Ni-MH cells because they are the medium-term
(next several years) choice for most hybrid vehicles, including those already on the
market. A collection of commonly communicated battery care advice is presented, then
proven or rejected by personal experience or deeper research. Based off of the remaining
acceptable evidence, a hypothesis is formed as to what environmental factors will
influence battery performance and longevity.

The most readily realizable contribution of this thesis is the automatic test bench
built specifically for the experimental testing. Chapter four outlines how it was
constructed, how it behaves, and how the MATLAB-based control program makes the all
the testing conditions, including temperature and charging, repeatable. The test bench is
expandable to higher voltages and current, as well as larger cells by the machining of a
new aluminum die. The test bench and a copy of the software remain at the Center for
Automotive Research for future battery cycling needs.

The fifth chapter describes how the health of the tested cells is judged. Each cell
is benchmarked twice, once during the middle of testing and once at the end. Each
benchmarking test consists of a timed discharge at a regulated 2.6A (from the cell’s
capacity rating of 2.6 Ahrs) to measure capacity. The benchmarking procedure also
subjects the cell to a step current load and records, via a digital storage oscilloscope, the
voltage and current waveforms. Another MATLAB program determines the model parameter that best fit the recorded data by minimization routines.

The resulting data basically shows that heat adversely affected the cell’s health the most, by increasing internal resistance. Interestingly, the first time constant, consisting of a separate RC network, remained unchanged throughout any of the tests. This information, coupled with the intuitive knowledge and additional research in chapter six, gives rise to the fuzzy rule base.

The beginning of chapter six describes how the battery simulator is divided into three subsystems, electrical, thermal, and model parameter penalizer. The electrical subsystem is a Simulink implementation of the circuit model shown in chapter three. The thermal subsystem simply integrates heat generation, less which lost to ambient air through cooling, into temperature. The parameter penalizer uses numerical methods to tabulate the cell’s effective depth of discharge and exposure to heat. It then uses the fuzzy rule surfaces presented in the previous chapter to decide how capacity is decreased and internal resistance is increased.

The results at the end of chapter six show that the model not only corresponds to the experimental and intuitive data, but can also mimic the result found in real hybrid applications. As most automakers use computer simulations during the design process to predict everything from fuel economy to emissions, this fuzzy logic simulator will be a great addition to their toolboxes. It can give a first-look, with very little time investment, as to what type of battery management algorithm and hardware provides the best performance/economy-reliability tradeoff.
7.2 Improvements and Further Work

Obviously this fuzzy-based (and experimental results-based) simulator could be improved with more experimentation, but the improvements suggested here go farther than that and in different directions. The best way to improve the simulator is not necessarily to increase testing, but take a step back and look at the larger problem that this simulator attempts to solve.

The easiest way to increase the accuracy of the simulator is to greatly expand the number of cells tested and types of tests run on those cells. Every situation cannot be tested, but if the cells were tested entirely (to the end of their lives), and several tests for each environmental direction (several temperatures, for example), a full condition matrix could be built. This would provide a plentiful basis for interpretation and rule base construction.

Testing that many cells to their deaths might take a year or more to complete, but not necessarily that many more man-hours. With a sophisticated automatic test bench in a safe, protected location (if the test-bench were to catastrophically fail, only it would be damaged), one person might need to tend to it an hour a day. This could be done as a side project and, once the data was acquired and compiled, would provide plentifully for the construction of a new model.

One assumption that was made during this thesis was that the effects of one environmental variable would be independent from another. The effects could then be combined in the simulation, to estimate what would have happened if two or more environmental variables were combined. This made the construction of the model and
the interpretation of the experimental results more convenient, and although true from personal experience, may not be true when critically examined.

The most beneficial improvement to this simulator might come from taking a step back and wondering why the cell degrades in the first place. It is often said that electrodes erode, gas forms, and insulation breaks down. The questions are, why do these things happen and how can they be modeled separately. Concerning only an electrical model, one might think that the Lockheed Martin model of Figure 2.25 is sufficiently complicated and accurately describes how a battery behaves. For that particular project, the modeling of satellite batteries, that is probably true. However, remember that a battery is mainly a chemical device. Just as entire vehicles are modeled by determining how the smallest parts (torque lost to a bearing, change in speed through a gear, rolling resistance, etc.) interact with each other, the most accurate way to model a battery is to determine how each chemical part interacts with the others.

One of the books used as a reference for this thesis (reference 2, Battery Management Systems: Design by Modeling) supplied little information about how batteries age. It did, however, supply a great deal of information on the electrochemical basics and what types of load batteries are used to drive. To this end, page 104 contains a very complicated model of the Ni-Cd cell, based entirely of discrete devices, such as transformers, diodes, and current sources. One might never expect these to be in a cell model, but the model does a wonderful job of following the real Ni-Cd cell’s voltage output. The approach was to break the cell down into its basic interactions, model each one separately, and then combine to create the entire cell. In fact, the model is broken
down into the elements that represent the nickel to potassium-hydroxide electrolyte, the ionic transfer, and the electrolyte-to-cadmium reaction.

An aging model could be constructed in the same way, by determining how and why each part of the cell ages. The interesting model responses will come from how aged parts of the cell, say a deteriorated metal hydride not capable of absorbing hydrogen as easily, reacts with un-aged parts, such as the electrolyte. Such a model would be the ultimate predictor of battery behavior under any combination of circumstances, but at the same time would take the ultimate amount of time to develop.

One very important field of study that this model hopefully sparks more research in is battery reliability. The qualitative reasoning implemented in the fuzzy logic portion of the aging model makes an interesting base on which to perform classic reliability analysis techniques.

Overall, the main problem might have been missed entirely. Instead of developing a better model to help understand and design around the Ni-MH chemistry’s shortcomings, maybe the solution is to just be aware that those shortcomings exist. The one, single, most important factor in long-term care for Ni-MH cells is temperature. Extreme temperatures affect cell performance much more than current draw, overcharge or over discharge, and even repeated, deep cycling. Unfortunately, the automotive industry is known for its temperature extremes. Without having worked on hybrid applications for long-term reliability, it is hard to say what the biggest design hurdle is and what information would be most useful in passing it. The main problem might be cooling, it might be weight vs. power availability, it might be transient effects during the
first hundred milliseconds of current draw, or it might be cost. Whatever that problem is, be aware that this, or any, model is not an end-all solution, only a way to make the designer more aware of the design constraints.

This fuzzy logic model is recognized to be more of a thesis topic than a widely usable battery simulator. It should not be used to make numerically close decisions in the performance of other hybrid components. However, it makes a great learning tool, especially for the writer, to understand how a Ni-MH battery would age in a stand-alone or hybrid situation, and what might be the next step in designing a robust hybrid power train.
APPENDIX A: Manufacturer's Datasheets for Tested Batteries
# Cell Type HR-SC

## Specifications

<table>
<thead>
<tr>
<th>Type</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nickel-Metal Hydride Battery</td>
<td>SC</td>
</tr>
</tbody>
</table>

### Capacity
- Typical: 2600mAh
- Minimum: 2300mAh

### Nominal Voltage
1.2V

### Charging Current x Time
- Fast Charge: 6A x about 1.1h

<table>
<thead>
<tr>
<th>Charge Condition</th>
<th>Temperature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fast Charge</td>
<td>0°C - 40°C</td>
</tr>
<tr>
<td>Discharge</td>
<td>0°C - 50°C</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Ambient Temp. Condition</th>
<th>Temperature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Less than 30days</td>
<td>-20°C - 50°C</td>
</tr>
<tr>
<td>Less than 90days</td>
<td>-20°C - 40°C</td>
</tr>
<tr>
<td>Less than 1year</td>
<td>-20°C - 30°C</td>
</tr>
</tbody>
</table>

### Internal Impedance
- (after discharge to E.V.=1.0V): About 4mΩ (at 1000Hz)

### Weight
- About 62 g

### Size (Diameter) x Height
- 23.0(D) x 43.5(H) mm

---

1) Single cell capacity under the following condition.
2) Use recommended charging system.
3) After a few charge and discharge cycles under the above 1) condition.
4) With tube.

## Typical Characteristics

### Charge

![Charge Graph](image1)

### Discharge

![Discharge Graph](image2)
APPENDIX B: Test Bench Control Program
%Ryan Somogye Thesis testing program.
%main program, define program aprameters below, then hit F5 to run.
%make sure all the functions call are in the same (working) directory
%written for MATLAB 6.5.1 with DAQ toolbox

clear all

global ai ao dio   %global declarations for daq update function
%global declaration for record-keeping txt file name
global target_charge_current target_discharge_current target_temp
global battery_model record_txt_file_name fid total_test_sample_vector
global cycle_loop_enable cycle_loop_counter sample_period cycle_sample_vector
global approx_poly3 approx_poly4 approx_poly4_y approx_poly4_2y
%global approx_poly3_y approx_poly3_2y approx_poly3_3y approx_poly4_3y
%error_poly4 error_poly3
%global overcharge_index charge_stopped_index
%global target_discharge_power power_current_discharge

%initialize daq system, using AT-MIO-16E form National Instruments.
%DAA ID number might have to be changed if running on a different machine.
init_DAQ

%define constants
V_batt_max = 1.8;   %maximum battery voltage limit during charging
V_batt_min = .9;   %discharging stops at this voltage
I_batt_max = 20;
I_batt_min = -20;
T_batt_min = 0;
T_batt_max = 50;

%define testing parameters
target_charge_current = 6;
target_discharge_current = 10;
target_discharge_power = 10; %watts
target_temp = 25;
power_current_discharge = 0;
current_drive_on_off = 1;
temp_drive_on_off = 1;
shutoff_energy_fraction = .8;
battery_model = 'HRSC 2.6Ah';
battery_capacity = 2.6; %capacity in amp-hours

%define charge-termination parameters
%for qualitative explanation of these parameters,
%refer to the thesis for a graph of cell charging V vs time.
min_voltage_for_triggers = 1.3; %minimum voltage of cell before triggers

129
% are used to stop the charging algorithm
min_seconds_at_min_voltage = 30; % minimum number of samples is
% cycle_sample_period at or above min_voltage_for_triggers
% before charging can be stopped by a dv/dt trigger
percent_overcharge = 1; % defines how much more, relative to total
% charge cycle length, the charge continues after
% charge completion is detected
poly3_increase_percent = 10; % defines, by percent of the min poly3 value,
% how much the error_poly3 must be to start overcharging
safety_charge_limit_percent = 100; % defines, by percent of the charge
% integration, how long the charging can continue before,
% overcharging is triggered
minimum_charge_limit_percent = 80; % defines, by percent of the charge
% integration, how long the charging must go before
% it can be stopped by dv/dt detection
actual_power_supply_current = 5.2; % used for determining max charging time
% if the power supply is set lower than desired current to
% reduce regulation ripple
auto_charge_on_off = 1; % turns on and off the autodetection of a charge
% the time-limit setting remains on

% define sampling parameters
plot_update_interval = 10; % update interval for plot in seconds
sample_period = 1;
sample_number = 0;

% create txt file
create_file

% initialize outputs to off
update_outputs(0,25,0,0);

timer_charge = timer('TimerFcn', ...
    'sample_vector = single_sample(target_charge_current, target_temp,
current_drive_on_off, temp_drive_on_off);'...
    ,'Period', sample_period,'ExecutionMode','FixedRate');
timer_discharge = timer('TimerFcn', ...
    'sample_vector = single_sample(-target_discharge_current, target_temp,
current_drive_on_off, temp_drive_on_off);'...
    ,'Period', sample_period,'ExecutionMode','FixedRate');

cycle_loop_counter = 0; cycle_loop_enable = 1;
total_test_sample_vector = [];
last_plot_index = 0;
best_discharge_energy = 0; best_charge_energy = 0;
energy_vector = [];

% one-cycle loop, runs once for each charge-discharge cycle
while cycle_loop_enable == 1

% init sample storage vectors
cycle_sample_vector = [];

charging_loop_enable = 1; discharging_loop_enable = 0;
error_poly4 = []; error_poly3 = []; min_index_for_trigger = 0;
overcharge_index = 0; charge_stopped_index = 0;

start(timer_charge);
% charging loop, runs once per sample period while charging battery
while charging_loop_enable == 1
    pause(1)
    % sample_vector = sample_inputs;
    % put charging stopping criteria here
    % output is the clearing of flag charging_loop_enable
    length_cycle_sample_vector = size(cycle_sample_vector,1);
    comparing_vector_good = 1;
    % check multiple conditions for allowing dvdt triggers
    % check that cycle_sample_vector has been at minimum trigger
    % voltage for at least minimum number of seconds
    if not((cycle_sample_vector(length_cycle_sample_vector) >
    min_voltage_for_triggers)...& (length_cycle_sample_vector >
    (min_seconds_at_min_voltage/sample_period))]
        comparing_vector_good = 0;
    end
    if comparing_vector_good == 1
        comparing_vector = cycle_sample_vector((length_cycle_sample_vector-...
    (min_seconds_at_min_voltage/sample_period)+1):length_cycle_sample_vector,1)...>
    min_voltage_for_triggers*ones((min_seconds_at_min_voltage/sample_period),1);
    for k = length(comparing_vector)
        if not(comparing_vector(k) == 1)
            comparing_vector_good = 0;
        end
    end
    end
    % proceed with dvdt and polyfit stuff
    if comparing_vector_good == 1

131
%store min voltage trigger index
if min_index_for_trigger == 0
    min_index_for_trigger = length_cycle_sample_vector;
end
%generate x index and approximate polynomial to batt voltage
x_fit_index = 1:length_cycle_sample_vector;
warning off MATLAB:polyfit:RepeatedPointsOrRescale;
%compute approx_poly_4
approx_poly4 = polyfit(x_fit_index,cycle_sample_vector(1:... length_cycle_sample_vector,1)',4);
approx_poly4_y = polyval(approx_poly4,x_fit_index);
%compute approx_poly_3
approx_poly3 = polyfit(x_fit_index,cycle_sample_vector(1:... length_cycle_sample_vector,1)',3);
approx_poly3_y = polyval(approx_poly3,x_fit_index);
%generate total std of (sampled_voltage-poly) for current
%index and store in error_poly arrays
error_poly4 = [error_poly4 std(cycle_sample_vector(1:... length_cycle_sample_vector,1)'-approx_poly4_y)];
error_poly3 = [error_poly3 std(cycle_sample_vector(1:... length_cycle_sample_vector,1)'-approx_poly3_y)];
%search and locate global minimum of error poly arrays
[global_min_error_poly3, global_min_error_poly3_index] = min(error_poly3);
%offset global_min_error_poly3_index since it starts after
%cycle_sample_vector, at min_index_for_trigger
if error_poly3(length(error_poly3)) >= (((poly3_increase_percent... + 100)/100)*error_poly3(global_min_error_poly3_index)) &...
    (auto_charge_on_off == 1) & (length_cycle_sample_vector >...
    (minimum_charge_limit_percent/100)*...
    (3600*(battery_capacity/actual_power_supply_current)));
    overcharge_index == 0
    overcharge_index = length_cycle_sample_vector;
    ans = ['Entering Overcharging Stage due to dVdt Detection at cycle_sample_vector('...
        num2str(overcharge_index) ')]
    end
end
if length_cycle_sample_vector > ((safety_charge_limit_percent/100)*...
    (3600*(battery_capacity/actual_power_supply_current)));
    overcharge_index == 0
    overcharge_index = length_cycle_sample_vector;
    ans = ['Entering Overcharging Stage due to max time at cycle_sample_vector('...
    end
132
end

if not(overcharge_index == 0)
    if length_cycle_sample_vector > overcharge_index...
        *(100+percent_overcharge)/100)
        charge_stopped_index = length_cycle_sample_vector;
        ans = ['Charge Stopped Due at cycle_sample_vector('...
            num2str(charge_stopped_index) ')]
        charging_loop_enable = 0;
    end
end

% check max battery voltage
if sample_vector(1) > V_batt_max
    charging_loop_enable = 0;
    ans = 'Charge Stopped Due to Max Voltage'
end

% check for flag and enable accordingly
if charging_loop_enable == 0
    stop(timer_charge);
    update_outputs(0,25,0,0);
end

% plot battery data periodically during test
if rem(size(total_test_sample_vector,1),(plot_update_interval/sample_period)) == 0
    if not(last_plot_index == size(total_test_sample_vector,1))
        plot_data(total_test_sample_vector,cycle_sample_vector);
        last_plot_index = size(total_test_sample_vector,1);
    end
end

pause(2)

discharging_loop_enable = 1;
error_poly4 = []; error_poly3 = [];

start(timer_discharge);
% discharging loop, runs once per sample period while charging battery
while discharging_loop_enable == 1
pause(1)
sample_vector = sample_inputs;

%constant-power discharge
if power_current_discharge == 1
    target_discharge_current = target_discharge_power/sample_vector(1);
end

%check max battery voltage to allow charging
if sample_vector(1) < V_batt_min
    discharging_loop_enable = 0;
    stop(timer_discharge);
    update_outputs(0,25,0,0);
end

%plot battery data periodically during test
if rem(size(total_test_sample_vector,1),(plot_update_interval/sample_period)) == 0
    if not(last_plot_index == size(total_test_sample_vector,1))
        plot_data(total_test_sample_vector, cycle_sample_vector);
        last_plot_index = size(total_test_sample_vector,1);
    end
end

cycle_loop_counter = cycle_loop_counter + 1;
size_cycle_sample_vector = size(cycle_sample_vector);

begin_last_discharge_index = 0;
k_loop = size(cycle_sample_vector,1);
while (begin_last_discharge_index == 0) & not(k_loop == 0)
    if not( cycle_sample_vector(k_loop,4)) < 0 )
        begin_last_discharge_index = k_loop + 1;
    end
    k_loop = k_loop-1;
end

begin_last_charge_index = 0;
k_loop = begin_last_discharge_index - 1;
while (begin_last_charge_index == 0) & (k_loop >= 1)
    if not((cycle_sample_vector(k_loop,4)) > 0)
        begin_last_charge_index = k_loop + 1;
    end
    if k_loop == 1
        begin_last_charge_index = 1;
    end
k_loop = k_loop-1;
end

present_charge_energy =
calculate_energy(cycle_sample_vector(:,1),cycle_sample_vector(:,2),begin_last_charge_index,begin_last_discharge_index-1)
present_discharge_energy = calculate_energy(cycle_sample_vector(:,1),-
cycle_sample_vector(:,2),begin_last_discharge_index,size(cycle_sample_vector,1))
present_charge_Ahrs =
calculate_energy(ones(1,size(cycle_sample_vector,1)),cycle_sample_vector(:,2),begin_last_charge_index,begin_last_discharge_index-1)
present_discharge_Ahrs = calculate_energy(ones(1,size(cycle_sample_vector,1)),-
cycle_sample_vector(:,2),begin_last_discharge_index,size(cycle_sample_vector,1))
energy_vector = [energy_vector; present_charge_energy present_discharge_energy
present_charge_Ahrs present_discharge_Ahrs];

% plot past energies
figure(3)
clf
plot(energy_vector(:,1),'b')
hold on; grid;
plot(energy_vector(:,2),'r')
plot(energy_vector(:,3),'g')
plot(energy_vector(:,4),'c')
plot(energy_vector(:,2)./energy_vector(:,1),'m')
plot(energy_vector(:,4)./energy_vector(:,3),'y')
title('Discharge Energy vs Cycle Number')
xlabel('Cycle Number')
ylabel('Whrs')
legend('Charge Energy','Discharge Energy','Charge Ahrs',...
    'Discharge Ahrs','Energy Efficiency/100','Ahrs Efficiency/100')

if present_discharge_energy > best_discharge_energy
    best_discharge_energy = present_discharge_energy;
end

% if present_discharge_energy <= shutoff_energy_fraction*best_discharge_energy
%    cycle_loop_enable = 0;
% end

pause(8)

end
clear all

global start_exp_fit end_exp_fit open_circuit_voltage_before_charge
global vbat current time_index testing_current vbat_smoothed
global current_smoothed_poly

%set constants
samples_per_second = 100000;
sample_period = 1 / samples_per_second;
current_sense_resistance = 0.4;
testing_current = -10;
nominal_voltage = 1.3; %1.3695;

%load battery voltage from the text file containing time index and ch1
record_txt_file_name = 'HRSC 2.6Ah 2C charge 8C discharge 25C 100-hour benchmark -20A trans 60min1C ch1.txt';
fid = fopen(record_txt_file_name,'r');
header_line1 = fgetl(fid);
[time_index vbat] = textread(record_txt_file_name,'%f %f',headerlines',4);
fclose(fid);
length_data = length(vbat);

%load battery voltage from the text file containing time index and ch1
record_txt_file_name = 'HRSC 2.6Ah 2C charge 8C discharge 25C 100-hour benchmark -20A trans 60min1C ch2.txt';
fid = fopen(record_txt_file_name,'r');
header_line1 = fgetl(fid);
[current] = textread(record_txt_file_name,'%f',headerlines',4);
fclose(fid);

%remove channel 1 additive from channel 2 (current)
current = (current - vbat)/current_sense_resistance;
%store raw values of sampled data
vbat_raw = vbat;
current_raw = current;
%smooth data for analysis
vbat = smooth(vbat,32);
current = smooth(current,32);

%number of samples to activate threshold
threshold_samples = 100;
%find index where charging begins
begin_discharge_index = 0;
k = 1;
while begin_discharge_index == 0
    if abs(current(k:k+threshold_samples)) > .2
        begin_discharge_index = k-1;
    end
    k = k+1;
end

%find index where charge recovery begins
begin_discharge_recovery_index = 0;
k = begin_discharge_index + 1;
while begin_discharge_recovery_index == 0
    if abs(current(k:k+threshold_samples)) < .75*abs(testing_current)
        begin_discharge_recovery_index = k;
    end
    k = k+1;
end

%plot data and indecies
figure(1)
clf
plot(time_index, vbat, 'b')
hold on; grid;
plot(time_index, current, 'r')
line([time_index(begin_discharge_index) time_index(begin_discharge_index)], [0 2])
line([time_index(begin_discharge_recovery_index) time_index(begin_discharge_recovery_index)], [0 2])
title('Actual Battery Voltage and Current')
xlabel('time (seconds)')
legend('Battery Voltage','Battery Current')

figure(2)
clf
plot(vbat, 'b')
hold on; grid;
plot(current, 'r')
line([begin_discharge_index begin_discharge_index], [-2 2])
line([begin_discharge_recovery_index begin_discharge_recovery_index], [-2 2])
title('Actual Battery Voltage and Current')
xlabel('time (seconds)')
%set begin_charge and begin_charge_recovery indicies manually
% begin_discharge_index = 14416;
% begin_discharge_recovery_index = 184000;

%find testing current
test_length = begin_discharge_recovery_index - begin_discharge_index;
test_center_index = (begin_discharge_index + begin_discharge_recovery_index) / 2;
testing_current = mean(current(test_center_index-
test_length/4:test_center_index+test_length/4))

%find the open circuit voltage before the test begins
open_circuit_voltage_before_charge = mean(vbatt(1:begin_discharge_index/2));
open_circuit_voltage_after_charge = mean(vbatt((begin_discharge_recovery_index+length_data)/2:length_data));

%loop for fitting multiple times to find least error
%error increases as exponential disappears and volttae rises
%due to main capacitycapvoltage increasing
%enter guess for time constant and voltage increase
internal_resistance_range = [0.0001 0.04]; %set from 1 mohm to 10 mohms
first_resistance_range = [0.0001 0.01];
first_capacitance_range = [.1 60];
%set brute force precision
internal_resistance_precision = .005;
first_resistance_precision = .005;
first_capacitance_precision = .025;
%set number of elements incremented for comparing the best-fit to the
%actual, used to save computing time
compare_element_increment = 2000;
%set number of samples to be used in fitting
start_exp_fit = begin_discharge_index;
%20000 elements per 100ms of time, set to 2*20000 for 200ms
end_exp_fit = 4*20000+begin_discharge_index;
%round((test_length*.5)+begin_discharge_index);

%calculate a best-fit line from 200ms to 300ms
begin_line_fit = end_exp_fit;
end_line_fit = end_exp_fit+20000; %begin_discharge_recovery_index;
[poly, S] = polyfit(time_index(begin_line_fit:end_line_fit),vbatt(begin_line_fit:end_line_fit),1)

%loops for fitting exp
std_error = 10000000000;
%use brute force method to find best-fit coefs
for h = interenal_resistance_range(1):(interenal_resistance_range(2) -
  interenal_resistance_range(1)))*interenal_resistance_precision:interenal_resistance_range
(2)

    for j = first_resistance_range(1):(first_resistance_range(2) -
    first_resistance_range(1)))*first_resistance_precision:first_resistance_range(2)

    for k = first_capacitance_range(1):(first_capacitance_range(2) -
    first_capacitance_range(1)))*first_capacitance_precision:first_capacitance_range(2)

        output_vector = fit_exp_discharge_begin([h j k compare_element_increment]);
        std_error_temp = output_vector(1);
        if std_error_temp < std_error
            std_error = std_error_temp;
            internal_resistance = h;
            first_resistance = j;
            first_capacitance = k;
        end
    end
end

internal_resistance
first_resistance
first_capacitance

output_vector = fit_exp_discharge_begin([internal_resistance first_resistance
  first_capacitance 1]);
std_error = output_vector(1)
fitted_exp = output_vector(2:length(output_vector));

figure(3)
clf
plot(time_index(start_exp_fit-100:end_exp_fit), vbatt(start_exp_fit-100:end_exp_fit), 'b')
hold on; grid;
plot(time_index(start_exp_fit:end_exp_fit), fitted_exp, 'r')
title('Fitted Model Output')
xlabel('time (seconds)')
ylabel('Volts')
legend('Sampled Voltage','Best-Fit Parameter Output')

output_vector = fit_exp_discharge_begin([internal_resistance first_resistance
  first_capacitance 1]);
std_error = output_vector(1)
fitted_exp = output_vector(2:length(output_vector));

figure(4)
clf
plot(start_exp_fit-100:begin_discharge_recovery_index, vbatt(start_exp_fit-100:begin_discharge_recovery_index), 'b')
hold on; grid;
plot(start_exp_fit:end_exp_fit, fitted_exp, 'r')
clear all

global start_exp_fit end_exp_fit open_circuit_voltage before_charge
global vbatt current time_index testing_current vbatt_smoothed
global current_smoothed poly

%set constants
samples_per_second = 100000;
sample_period = 1 / samples_per_second;
current_sense_resistance = 0.4;
testing_current = -10;
nominal_voltage = 1.3; %1.3695;

%load battery voltage from the text file containing time index and ch1
record_txt_file_name = 'HRSC 2.6Ah 2C charge 8C discharge 25C 100-hour benchmark -20A trans 60min1C ch1.txt';
fid = fopen(record_txt_file_name,'r');
header_line1 = fgetl(fid);
[time_index vbatt] = textread(record_txt_file_name,'%f %f',headerlines',4);
close(fid);
length_data = length(vbatt);

%load battery voltage from the text file containing time index and ch1
record_txt_file_name = 'HRSC 2.6Ah 2C charge 8C discharge 25C 100-hour benchmark -20A trans 60min1C ch2.txt';
fid = fopen(record_txt_file_name,'r');
header_line1 = fgetl(fid);
[current] = textread(record_txt_file_name,'%f',headerlines',4);
close(fid);

%remove channel 1 additive from channel 2 (current)
current = (current - vbatt)/current_sense_resistance;
%store raw values of sampled data
vbatt_raw = vbatt;
current_raw = current;
%smooth data for analysis
vbatt = smooth(vbatt,32);
current = smooth(current,32);

%number of samples to activate threshold
threshold_samples = 100;

% find index where charging begins
begin_discharge_index = 0;
k = 1;
while begin_discharge_index == 0
    if abs(current(k:k+threshold_samples)) > .2
        begin_discharge_index = k-1;
    end
    k = k+1;
end

% find index where charge recovery begins
begin_discharge_recovery_index = 0;
k = begin_discharge_index + 1;
while begin_discharge_recovery_index == 0
    if abs(current(k:k+threshold_samples)) < .75*abs(testing_current)
        begin_discharge_recovery_index = k;
    end
    k = k+1;
end

% plot data and indices
figure(1)
clf
plot(time_index, vbatt, 'b')
hold on; grid;
plot(time_index, current, 'r')
line([time_index(begin_discharge_index) time_index(begin_discharge_index)], [0 2])
line([time_index(begin_discharge_recovery_index) time_index(begin_discharge_recovery_index)], [0 2])
title('Actual Battery Voltage and Current')
xlabel('time (seconds)')
legend('Battery Voltage', 'Battery Current')

figure(2)
cif
plot(vbatt_raw, 'b')
hold on; grid;
plot(current_raw, 'r')
line([begin_discharge_index begin_discharge_index], [-2 2])
line([begin_discharge_recovery_index begin_discharge_recovery_index], [-2 2])
title('Actual Battery Voltage and Current')
xlabel('time (seconds)')

142
%set begin_charge and begin_charge_recovery indicies manually
% begin_discharge_index = 14416;
% begin_discharge_recovery_index = 184000;

%find testing current
test_length = begin_discharge_recovery_index - begin_discharge_index;
peak_test_current = mean(current(test_center_index-
test_length/4:begin_discharge_index/2);mean(vbatt1:begin_discharge_index/2));

%find open circuit voltage before the test begins
open_circuit_voltage_before_charge = mean(vbatt1:begin_discharge_index/2));
open_circuit_voltage_after_charge = mean(vbatt1:(begin_discharge_recovery_index+length_data/2):length_data));

%loop for fitting multiple times to find least error
%error increases as exponential disappears and voltages rises
%due to main capacity voltage increasing
%enter guess for time constant and voltage increase
internal_resistance_range = [0.0001 0.04]; %set from 1 mohm to 10 mohms
first_resistance_range = [0.0001 0.01];
first_resistance_precision = .005;
first_resistance_precision = .005;
first_resistance_precision = .025;
%set number of elements incremented for comparing the best-fit to the
%actual, used to save computing time
compare_element_increment = 2000;
%set number of samples to be used in fitting
start_exp_fit = begin_discharge_index;
%20000 elements per 100ms of time, set to 2*20000 for 200ms
end_exp_fit = 4*20000+begin_discharge_index;
%round(test_length*.5)+begin_discharge_index;

%calculate a best-fit line from 200ms to 300ms
begin_line_fit = end_exp_fit;
end_line_fit = end_line_fit+20000; %begin_discharge_recovery_index;
[poly, S] = polyfit(time_index(begin_line_fit:end_line_fit),vbatt(begin_line_fit:end_line_fit),1)

%loops for fitting exp
std_error = 10000000000;
%use brute force method to find best-fit coefs
for h = internal_resistance_range(1):(internal_resistance_range(2) -
internal_resistance_range(1))*internal_resistance_precision:internal_resistance_range
(2)

for j = first_resistance_range(1):(first_resistance_range(2) -
first_resistance_range(1))*first_resistance_precision:first_resistance_range(2)

for k = first_capacitance_range(1):(first_capacitance_range(2) -
first_capacitance_range(1))*first_capacitance_precision:first_capacitance_range(2)

output_vector = fit_exp_discharge_begin([h j k compare_element_increment]);
std_error_temp = output_vector(1);
if std_error_temp < std_error
    std_error = std_error_temp;
    internal_resistance = h;
    first_resistance = j;
    first_capacitance = k;
end
end
end

internal_resistance
first_resistance
first_capacitance

output_vector = fit_exp_discharge_begin([internal_resistance first_resistance
first_capacitance 1]);
std_error = output_vector(1)
fitted_exp = output_vector(2:length(output_vector));

figure(3)
clf
plot(time_index(start_exp_fit-100:end_exp_fit), vbatt(start_exp_fit-100:end_exp_fit), 'b')
hold on; grid;
plot(time_index(start_exp_fit:end_exp_fit), fitted_exp, 'r')
title('Fitted Model Output')
xlabel('time (seconds)')
ylabel('Volts')
legend('Sampled Voltage', 'Best-Fit Parameter Output')

output_vector = fit_exp_discharge_begin([internal_resistance first_resistance
first_capacitance 1]);
std_error = output_vector(1)
fitted_exp = output_vector(2:length(output_vector));

figure(4)
clf
plot(start_exp_fit-100:begin_discharge_recovery_index, vbatt(start_exp_fit-100:begin_discharge_recovery_index), 'b')
hold on; grid;
plot(start_exp_fit:end_exp_fit, fitted_exp, 'r')
APPENDIX C: Battery Model and Accompanying Code
%Ryan Somogye Thesis
%MATLAB code for battery simulator

clear all
global SOC_history I_history I_polarity_switch update_life_counter

%define simulation constants
T_sample = .1; %1ms
sim_speed_boost = 10;

repeat_number = 600;
pos_amplitude = 5;
neg_amplitude = -10;
pos_period = 15000/sim_speed_boost;
pos_rest_period = 0;
neg_period = 7500/sim_speed_boost;
neg_rest_period = 0/sim_speed_boost;

current_in_base = [pos_amplitude*ones(1,pos_period)...
                  zeros(1,pos_rest_period) neg_amplitude*ones(1,neg_period) ... 
                  zeros(1,neg_rest_period)];
current_in = []; 

for k = 1:100
    current_in = [current_in current_in_base];
end

current_in_base_2 = current_in;
current_in = []; 

for k = 1:repeat_number/100
    current_in = [current_in current_in_base_2];
end

T_end = length(current_in)*T_sample;
time_index = [T_sample:T_sample:T_end];
current_in = [time_index current_in];

%define constants and init values for electrical model
%define SOC to open circuit voltage vectors
SOC_to_Vopen_input = [0.001 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1];
SOC_to_Vopen_output = [1.1 1.16 1.225 1.27 1.275 1.275 1.28 1.3 1.325 1.4];
%set initial value for electrical model parameters
cell_capacity_ahrs = 2.6;
C_capacity_init = cell_capacity_ahrs*3600/sim_speed_boost;
R_internal_init = .004;
R_first_init = .003;
C_first_init = 25;
%set initial state of charge (range 0 to 1)
SOC_init = 0.05;
%set SOC to heat wasted vectors
SOC_to_charge_heat_wasted_in = ...
   [0 0.92 0.93 0.94 0.95 0.96 0.97 0.98 0.99 1];
SOC_to_charge_heat_wasted_out = ...
   [0 0.0078125 0.015625 0.03125 0.0625 0.125 0.25 0.5 1];

%define constants and init values for thermal model
%define min, max, and starting temperatures
Temp_min = 0;
Temp_max = 60;
Temp_init = 50;
%define thermal properties of cell and surroundings
thermal_mass = 2.32/sim_speed_boost; %J/degC
temp_ambient = 50;
R_thermal_to_ambient = .064*100; %W/degC

%define constant used in model parameter penalizer
%define depth of discharge to expected cycle life
%programmed from Life Cycle Assessment for Five Batteries
%for Electric Vehicles under Different Charging, reference 8
dod_to_life_dock_input = [5 10 20 30 40 50 60 70 80 90 100];
dod_to_life_dock_input = dod_to_life_dock_input/200;
dod_to_life_dock_output = ...
   [1/10000 1/5000 1/2800 1/1750 1/1400 1/1000 1/850 1/650 1/500 1/400 1/350]*3/5;
%define temperature to cell life arrays
%programmed from graph from Duracell Tech Bulliten, ch 7
temp_to_cell_life_input = [0 10 18 20 30 40 50];
temp_to_cell_life_output = [1 1 1 0.98 0.78 0.60 0.42];
%load fuzzy inference system structures into memory
test_5mf_cap = readfis('test 5mf cap 4');
test_5mf_rint = readfis('test 5mf rint 3');

%define constant for dod_fun used in simulation
SOC_history = SOC_init;
I_history = 0;
I_polarity_switch = 0;
update_life_counter = 0;
sim('elec_test_3')

time_plot = time;

figure(1)
clf
plot(time_plot,current, 'b')
grid on; hold;
plot(time_plot,V_term_out, 'r')
plot(time_plot,V_cap_out, 'g')
plot(time_plot,Temp_cell_out, 'c')
plot(time_plot,heat_power_out, 'y')
xlabel('time (seconds)')
legend('Current', 'V term', 'V cap', 'Temp', 'P heat')

figure(2)
clf
plot(time_plot,cell_health_output, 'b')
hold on; grid;
plot(time_plot,heat_ratio_output, 'r')
plot(time_plot,std_output, 'g')
plot(time_plot,capacity_factor,'c')
plot(time_plot,rint_factor, 'y')
xlabel('time (seconds)')
legend('Cell Health','Heat Ratio','std out','cap factor','rint factor')

cycle_length_sec = length(current_in_base)*T_sample;
%plot with respect to cycle length
figure(3)
clf
plot(time_plot/cycle_length_sec,current, 'b')
grid on; hold;
plot(time_plot/cycle_length_sec,V_term_out*10, 'r')
plot(time_plot/cycle_length_sec,V_cap_out*10, 'g')
plot(time_plot/cycle_length_sec,Temp_cell_out, 'c')
plot(time_plot/cycle_length_sec,heat_power_out*10, 'y')
xlabel('Cycle Number')
title('Plot of Electrical and Thermal Model Outputs')
legend('Current', 'V term*10', 'V cap*10', 'Temp', 'P heat*10')
axis([98 101 -10.5 52])

figure(4)
clf

149
```
plot(time_plot/cycle_length_sec,cell_health_output, 'b')
hold on; grid;
plot(time_plot/cycle_length_sec,heat_ratio_output, 'r')
plot(time_plot/cycle_length_sec,capacity_factor,'g')
plot(time_plot/cycle_length_sec,rint_factor, 'y')
%plot(time_plot/cycle_length_sec,std_output, 'c')
xlabel('Cycle Number')
legend('Cell Health','Heat Ratio','Cap factor','Rint factor')
axis([0 575 -0.5 7])

%use this segment of code to create the fuzzy surface
heat_vector = [];
output_vector = [];

for k = 0.4:0.01:1
    heat_vector = [heat_vector k];
    temp_heat_value = k;
    sim('fuzzy_block_rint')
    output_vector = [output_vector rint];
end

figure(4)
c1f
mesh(heat_vector,age,output_vector)
xlabel('heat')
ylabel('age')
zlabel('R internal')
```
%Ryan Somogye Thesis
%function to determine average depth of discharge and number of cycles that
%the average was taken from
%records the V_cap values when current changes polarity.
%after max_length V_cap values, computes std

function dod_fun_output = dod_fun(dod_fun_input)

global SOC_history I_history I_polarity_switch update_life_counter

I_value = dod_fun_input(1);
V_cap_value = dod_fun_input(2);

SOC_history_length = 10;

if (I_history*I_value) < 0
    %keep track of number of times I switched polarity
    I_polarity_switch = I_polarity_switch + 1;
    %save SOC at the switch to V_cap_history
    SOC_history = [SOC_history V_cap_value];
end

if not((I_history*I_value) < 0) & (I_polarity_switch >= SOC_history_length)
    I_polarity_switch = 0;
end

%save present current value for next time
if not(I_value == 0)
    I_history = I_value;
end

%executed only when life counter will be docked
if I_polarity_switch >= SOC_history_length

%compute std of first SOC_history_length elements
%the last one is not used since it will be saved while the
%first SOC_history_length are deleted and the last element
%is counted next time
SOC_std = std(SOC_history);
%save the last element as the first next time
SOC_history_temp = SOC_history(SOC_history_length + 1);
SOC_history = [];
SOC_history = SOC_history_temp;
%output number of cycles used in determining ave SOC_std;
SOC_cycles_count = SOC_history_length;
update_life_counter = 1;
else
    update_life_counter = 0;
end

switch update_life_counter
    case 1
        dod_fun_output = [SOC_std SOC_cycles_count/2];
case 0
    dod_fun_output = [-1 0];
otherwise
    error('This is impossible')
end
<table>
<thead>
<tr>
<th>Age Linguistic Value</th>
<th>Connection</th>
<th>Heat Exposure Linguistic Value</th>
<th>Capacity Linguistic Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>New</td>
<td>and</td>
<td>Not Exposed</td>
<td>Full Capacity</td>
</tr>
<tr>
<td>New</td>
<td>and</td>
<td>Slightly Exposed</td>
<td>Full Capacity</td>
</tr>
<tr>
<td>New</td>
<td>and</td>
<td>Moderately Exposed</td>
<td>Full Capacity</td>
</tr>
<tr>
<td>New</td>
<td>and</td>
<td>Heavily Exposed</td>
<td>Full Capacity</td>
</tr>
<tr>
<td>New</td>
<td>and</td>
<td>Completely Exposed</td>
<td>Full Capacity</td>
</tr>
<tr>
<td>Slightly Used</td>
<td>and</td>
<td>Not Exposed</td>
<td>Full Capacity</td>
</tr>
<tr>
<td>Slightly Used</td>
<td>and</td>
<td>Slightly Exposed</td>
<td>Full Capacity</td>
</tr>
<tr>
<td>Slightly Used</td>
<td>and</td>
<td>Moderately Exposed</td>
<td>Full Capacity</td>
</tr>
<tr>
<td>Slightly Used</td>
<td>and</td>
<td>Heavily Exposed</td>
<td>Full Capacity</td>
</tr>
<tr>
<td>Slightly Used</td>
<td>and</td>
<td>Completely Exposed</td>
<td>Slightly Reduced</td>
</tr>
<tr>
<td>Moderately Used</td>
<td>and</td>
<td>Not Exposed</td>
<td>Full Capacity</td>
</tr>
<tr>
<td>Moderately Used</td>
<td>and</td>
<td>Slightly Exposed</td>
<td>Full Capacity</td>
</tr>
<tr>
<td>Moderately Used</td>
<td>and</td>
<td>Moderately Exposed</td>
<td>Full Capacity</td>
</tr>
<tr>
<td>Moderately Used</td>
<td>and</td>
<td>Heavily Exposed</td>
<td>Slightly Reduced</td>
</tr>
<tr>
<td>Moderately Used</td>
<td>and</td>
<td>Completely Exposed</td>
<td>Moderately Reduced</td>
</tr>
<tr>
<td>Heavily Used</td>
<td>and</td>
<td>Not Exposed</td>
<td>Full Capacity</td>
</tr>
<tr>
<td>Heavily Used</td>
<td>and</td>
<td>Slightly Exposed</td>
<td>Full Capacity</td>
</tr>
<tr>
<td>Heavily Used</td>
<td>and</td>
<td>Moderately Exposed</td>
<td>Slightly Reduced</td>
</tr>
<tr>
<td>Heavily Used</td>
<td>and</td>
<td>Heavily Exposed</td>
<td>Moderately Reduced</td>
</tr>
<tr>
<td>Heavily Used</td>
<td>and</td>
<td>Completely Exposed</td>
<td>Severely Reduced</td>
</tr>
<tr>
<td>Completely Used</td>
<td>and</td>
<td>Not Exposed</td>
<td>Severely Reduced</td>
</tr>
<tr>
<td>Completely Used</td>
<td>and</td>
<td>Slightly Exposed</td>
<td>Completely Reduced</td>
</tr>
<tr>
<td>Completely Used</td>
<td>and</td>
<td>Moderately Exposed</td>
<td>Completely Reduced</td>
</tr>
<tr>
<td>Completely Used</td>
<td>and</td>
<td>Heavily Exposed</td>
<td>Completely Reduced</td>
</tr>
<tr>
<td>Completely Used</td>
<td>and</td>
<td>Completely Exposed</td>
<td>Completely Reduced</td>
</tr>
<tr>
<td>Age Linguistic Value</td>
<td>Connection</td>
<td>Heat Exposure Linguistic Value</td>
<td>Capacity Linguistic Value</td>
</tr>
<tr>
<td>----------------------</td>
<td>------------</td>
<td>-----------------------------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>New</td>
<td>and</td>
<td>Not Exposed</td>
<td>Baseline</td>
</tr>
<tr>
<td>New</td>
<td>and</td>
<td>Slightly Exposed</td>
<td>Baseline</td>
</tr>
<tr>
<td>New</td>
<td>and</td>
<td>Moderately Exposed</td>
<td>Baseline</td>
</tr>
<tr>
<td>New</td>
<td>and</td>
<td>Heavily Exposed</td>
<td>Baseline</td>
</tr>
<tr>
<td>New</td>
<td>and</td>
<td>Completely Exposed</td>
<td>Baseline</td>
</tr>
<tr>
<td>Slightly Used</td>
<td>and</td>
<td>Not Exposed</td>
<td>Baseline</td>
</tr>
<tr>
<td>Slightly Used</td>
<td>and</td>
<td>Slightly Exposed</td>
<td>Baseline</td>
</tr>
<tr>
<td>Slightly Used</td>
<td>and</td>
<td>Moderately Exposed</td>
<td>Slightly Increased</td>
</tr>
<tr>
<td>Slightly Used</td>
<td>and</td>
<td>Heavily Exposed</td>
<td>Moderately Increased</td>
</tr>
<tr>
<td>Slightly Used</td>
<td>and</td>
<td>Completely Exposed</td>
<td>Heavily Increased</td>
</tr>
<tr>
<td>Moderately Used</td>
<td>and</td>
<td>Not Exposed</td>
<td>Baseline</td>
</tr>
<tr>
<td>Moderately Used</td>
<td>and</td>
<td>Slightly Exposed</td>
<td>Slightly Increased</td>
</tr>
<tr>
<td>Moderately Used</td>
<td>and</td>
<td>Moderately Exposed</td>
<td>Moderately Increased</td>
</tr>
<tr>
<td>Moderately Used</td>
<td>and</td>
<td>Heavily Exposed</td>
<td>Heavily Increased</td>
</tr>
<tr>
<td>Moderately Used</td>
<td>and</td>
<td>Completely Exposed</td>
<td>Severely Increased</td>
</tr>
<tr>
<td>Heavily Used</td>
<td>and</td>
<td>Not Exposed</td>
<td>Slightly Increased</td>
</tr>
<tr>
<td>Heavily Used</td>
<td>and</td>
<td>Slightly Exposed</td>
<td>Slightly Increased</td>
</tr>
<tr>
<td>Heavily Used</td>
<td>and</td>
<td>Moderately Exposed</td>
<td>Moderately Increased</td>
</tr>
<tr>
<td>Heavily Used</td>
<td>and</td>
<td>Heavily Exposed</td>
<td>Heavily Increased</td>
</tr>
<tr>
<td>Heavily Used</td>
<td>and</td>
<td>Completely Exposed</td>
<td>Severely Increased</td>
</tr>
<tr>
<td>Completely Used</td>
<td>and</td>
<td>Not Exposed</td>
<td>Slightly Increased</td>
</tr>
<tr>
<td>Completely Used</td>
<td>and</td>
<td>Slightly Exposed</td>
<td>Moderately Increased</td>
</tr>
<tr>
<td>Completely Used</td>
<td>and</td>
<td>Moderately Exposed</td>
<td>Heavily Increased</td>
</tr>
<tr>
<td>Completely Used</td>
<td>and</td>
<td>Heavily Exposed</td>
<td>Severely Increased</td>
</tr>
<tr>
<td>Completely Used</td>
<td>and</td>
<td>Completely Exposed</td>
<td></td>
</tr>
</tbody>
</table>
BIBLIOGRAPHY

[1] Panasonic Ni-MH Overview, Panasonic OEM battery division, February 2002
http://www.panasonic.com/industrial/battery/


Lockheed Martin Missiles and Space, IEEE, 1996


Electrochemical Systems for Circuit Applications”, University of Akron
Electrical and Computer Engineering Department, March 2003


“A Deterioration Estimation System for 200-Ah Sealed Lead-Acid Batteries”
NTT Interdisciplinary Research Labs, Tokyo, 180 Japan


155


