Usability Engineering Applied to an Electromagnetic Modeling Tool

Samuel K. Fortson

Thesis submitted to the faculty of the Virginia Polytechnic Institute and State University in partial fulfillment of the requirements for the degree of

Master of Science
In
Geosciences

Chester J. Weiss, Chair
Scott D. King
Woodrow W. Winchester, III

June 8th, 2012
Blacksburg, VA

Keywords: Geophysics, Magnetotellurics, Graphical User Interface, Usability Engineering
© Samuel K. Fortson, 2012
Usability Engineering Applied to an Electromagnetic Modeling Tool

Samuel K. Fortson

ABSTRACT

There are very few software packages for model-building and visualization in electromagnetic geophysics, particularly when compared to other geophysical disciplines, such as seismology. The purpose of this thesis is to design, develop, and test a geophysical model-building interface that allows users to parameterize the 2D magnetotellurics problem. Through the evaluation of this interface, feedback was collected from a usability specialist and a group of geophysics graduate students to study the steps users take to work through the 2D forward-modeling problem, and to analyze usability errors encountered while working with the interface to gain a better understanding of how to build a more effective interface. Similar work has been conducted on interface design in other fields, such as medicine and consumer websites.

Usability Engineering is the application of a systematic set of methods to the design and development of software with the goal of making the software more learnable, easy to use, and accessible. Two different Usability Engineering techniques – Heuristic Evaluation and Thinking Aloud Protocol – were involved in the evaluation of the interface designed in this study (FEM2DGUI). Heuristic Evaluation is a usability inspection method that employs a usability specialist to detect errors based on a known set of guidelines and personal experience. Thinking Aloud Protocol is a usability evaluation method where potential end-users are observed as they verbalize their every step as they work through specific scenarios with an interface. These Usability Engineering methods were combined in an effort to understand how the first prototype of FEM2DGUI could be refined to make it more usable and to understand how end-users work through the forward-modeling problem.

The Usability Engineering methods employed in this project uncovered multiple usability errors that were corrected through a refinement of the interface. Discovery of these errors helped with refining the system to become more robust and usable, which is believed to aid users in more efficient model-building. Because geophysical model-building is inherently a difficult task, it is possible that other model-building graphical user interfaces could benefit from the application of Usability Engineering methods, such as those presented in this research.
Glossary of Terms

**Graphical User Interface (GUI):** Tool for human interaction with a computer through direct manipulation of graphics on the screen.

**Heuristic Evaluation (HE):** Usability inspection method that utilizes usability specialists to judge whether each aspect of an interface follows a particular set of established usability principles (Nielsen, Usability Inspection Methods, 1994).

**Human Computer Interaction (HCI):** The study of understanding and supporting human interacting with and through technology (Carroll, 1997).

**Iterative Development:** Process of discovering new goals, prototyping, and evaluation while maintaining strong involvement from the users (Carroll, 1997).

**Thinking Aloud Protocol (THA):** Usability test method that involves having a test subject use an interface while continuously verbalizing their thoughts of the system.

**Usability:** The learnability, efficiency, memorability, low error rate, and user satisfaction of a particular software environment (Nielsen, 1993).

**Usability Engineering:** A discipline in which usability is systematically approached, improved, and evaluated (Nielsen, 1993).

**User-Centered Design (UCD):** Design and development process which aims to determine the goals and needs of the users, what tools they need, what tasks they wish to perform with the system, and how they wish to perform these tasks (Draper & Norman, 1986).

# Table of Contents

Glossary of Terms.................................................................................................................. iii

Table of Contents .................................................................................................................... iv

List of Figures ........................................................................................................................ viii

List of Tables ........................................................................................................................ x

Chapter 1: Introduction .......................................................................................................... 1

1.1 Overview ......................................................................................................................... 1

1.2 Literature Review .......................................................................................................... 2

1.2.1 Graphical User Interface (GUI) ................................................................................. 3

1.2.2 Usability Engineering .............................................................................................. 4

1.2.3 User-Centered Design ............................................................................................. 7

1.2.4 Usability Evaluation Methods .................................................................................. 8

1.2.5 Heuristic Evaluation ............................................................................................... 9

1.2.6 Thinking Aloud Protocol .......................................................................................... 10

1.2.7 Combining Usability Engineering Methods ............................................................. 12

1.3 Research Synopsis ........................................................................................................ 12

1.4 Document Organization ............................................................................................... 14

Chapter 2: Interface Design .................................................................................................. 15

2.1 Principles of the Magnetotellurics Forward Problem, FEM2D, and Triangle ............. 15

2.1.1 Magnetotellurics and the Forward Problem ............................................................ 15

2.1.2 FEM2D .................................................................................................................... 18

2.1.3 Triangle .................................................................................................................. 22

2.2 Design and Development Process .............................................................................. 23

2.2.1 Initial Design and Development ............................................................................. 23
2.2.2 Feature 1: Accommodate (tilted) layered Earth ................................................................. 27
2.2.3 Feature 2: Allow for the inclusion of simple isolated bodies .............................................. 28
2.2.4 Feature 3: Allow for the insertion of a background seismic image for reference ............. 28
2.2.5 Feature 4: Allow for mesh generation and construction with Triangle (Shewchuk, 1996) ................................................................................................................................. 28
2.2.6 Feature 5: Generation of input files for FEM2D ................................................................. 29
2.2.7 Feature 6: Apply necessary parameters to the different layers and/or isolated bodies. . 29
2.2.8 Feature 7: Convert FEM2D output into a format for plot generation ......................... 30
2.3 Walkthrough ......................................................................................................................... 31

Chapter 3: Justification and Generalized Methodology for Usability Evaluation of FEM2DGUI ................................................................. 36

3.1 Chapter Overview ................................................................................................................. 36
3.2 Usability Evaluation Methods ............................................................................................. 36
  3.2.1 Usability Inspection Methods .................................................................................... 37
  3.2.2 Usability Test Methods ............................................................................................ 38
  3.2.3 Analysis of Usability Inspection and Usability Test Methods .................................... 39
3.3 Generalized Usability Evaluation of FEM2DGUI ............................................................. 41
  3.3.1 Generalized Heuristic Evaluation Methodology ....................................................... 41
  3.3.2 Generalized Thinking Aloud Protocol Methodology .............................................. 44

Chapter 4: Results ...................................................................................................................... 46

4.1 Chapter Overview ................................................................................................................. 46
4.2 Heuristic Evaluation of Interface ....................................................................................... 47
  4.2.1 Method .................................................................................................................... 47
  4.2.2 Evaluators and Results ............................................................................................ 48
4.3 Thinking Aloud Protocol .................................................................................................... 52
  4.3.1 Participants ............................................................................................................. 52
  4.3.2 Equipment .............................................................................................................. 53
4.3.3 User Tasks .................................................................................................................. 53
4.3.4 Pilot Study and Changes to Evaluation Method .......................................................... 54
4.3.5 Results .......................................................................................................................... 55
4.4 Strengths ......................................................................................................................... 55
4.5 Usability Problems ......................................................................................................... 56
  4.5.1 Issue #1: Unintuitive and ambiguous icons in the toolbar ......................................... 58
  4.5.2 Issue #2: The user’s progression through tasks from start to finish is ambiguous ...... 59
  4.5.3 Issue #3: Placing points on the edges was difficult for users .................................... 61
  4.5.4 Issue #4: Placing attributes is unintuitive and their location can be ambiguous ....... 63
  4.5.5 Issue #5: “Triangle is Running” dialog box does not work correctly ......................... 63
  4.5.6 Issue #6: Scale bar spacing and units are not controlled by the user ....................... 64
  4.5.7 Issue #7: Receiver dialog box is unintuitive ............................................................... 65
  4.5.8 Issue #8: Insertion of the AEI and exactly horizontal lines is difficult ....................... 66
  4.5.9 Issue #9: Users cannot add points to lines ................................................................. 67
  4.5.10 Other Usability Problems ......................................................................................... 67

Chapter 5: Discussions and Conclusions ......................................................................... 69
  5.1 Design and Development ............................................................................................. 69
  5.2 Conclusions from Usability Evaluation ...................................................................... 70
  5.3 Refinements Completed ............................................................................................... 72
    5.3.1 Issue #1: Unintuitive and ambiguous icons in the toolbar ..................................... 72
    5.3.2 Issue #2: The user’s progression through tasks from start to finish is ambiguous ...... 74
    5.3.3 Issue #3: Placing points on the edges was difficult for users ................................. 75
    5.3.4 Issue #4: Placing attributes is unintuitive and their location can be ambiguous ....... 75
    5.3.5 Issue #5: “Triangle is Running” dialog box does not work correctly ......................... 75
    5.3.6 Issue #6: Scale bar spacing and units are not controlled by the user ....................... 76
    5.3.7 Issue #7: Receiver dialog box is unintuitive ............................................................... 76
    5.3.8 Issue #8: Insertion of the AEI and exactly horizontal lines is difficult ....................... 76
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>User interface hierarchy</td>
</tr>
<tr>
<td>1.2</td>
<td>Hierarchy of system usefulness</td>
</tr>
<tr>
<td>2.1</td>
<td>Logical Flow of Model Builder Interface</td>
</tr>
<tr>
<td>2.2</td>
<td>Opening screen of the interface</td>
</tr>
<tr>
<td>2.3</td>
<td>Model with line at the air-earth interface, and anomalous body, and a deeper layer</td>
</tr>
<tr>
<td>2.4</td>
<td>Attributes (conductivity values) added to the model</td>
</tr>
<tr>
<td>2.5</td>
<td>Dialog box for adding receivers</td>
</tr>
<tr>
<td>2.6</td>
<td>Receivers (upside-down triangles) added to the model</td>
</tr>
<tr>
<td>2.7</td>
<td>Dialog box for specifying nodes per skin depth and source frequency</td>
</tr>
<tr>
<td>2.8</td>
<td>Generated mesh where colors denote log10 conductivity</td>
</tr>
<tr>
<td>4.1</td>
<td>Flowchart illustrating the sequence of evaluative events</td>
</tr>
<tr>
<td>4.2</td>
<td>Toolbar with difficult to understand icons. The circled icons are those that gave evaluator and participants the most difficulty</td>
</tr>
<tr>
<td>4.3</td>
<td>Users must be very precise to correctly place points on drawing edge</td>
</tr>
<tr>
<td>4.4</td>
<td>Users are unable to change the default spacing and units for scale bars</td>
</tr>
<tr>
<td>4.5</td>
<td>When the &quot;Add&quot; button is clicked new receiver elements appear at the bottom of the list without the scroll bar adjusting making it difficult to recognize the new receiver elements</td>
</tr>
<tr>
<td>4.6</td>
<td>Solid white background and no reference lines make it difficult for users to add horizontal lines</td>
</tr>
<tr>
<td>5.1</td>
<td>(a) Tool bar before refinement. All icons are enabled. (b) Toolbar after refinement during program startup. Some of the icons are disabled</td>
</tr>
<tr>
<td>5.2</td>
<td>Example of a dialog box that helps users progress through tasks</td>
</tr>
<tr>
<td>5.3</td>
<td>GUI Showing Grid in Background</td>
</tr>
<tr>
<td>5.4</td>
<td>Opening screen for FEM2DGUI</td>
</tr>
<tr>
<td>5.5</td>
<td>Simple resistive body in a half-space</td>
</tr>
<tr>
<td>5.6</td>
<td>Adding receivers to the model</td>
</tr>
<tr>
<td>5.7</td>
<td>Completed model with receiver and conductivity attributes</td>
</tr>
<tr>
<td>5.8</td>
<td>Setting the &quot;nodes-per-skin-depth&quot;, frequency, MT mode</td>
</tr>
</tbody>
</table>
Figure 5.9: Completed mesh ................................................................................. 81
Figure 5.10: Apparent resistivity and phase output .................................................. 82
List of Tables

Table 2.1: User task analysis questions ................................................................. 24
Table 3.1: Nielsen's Ten Usability Principles .......................................................... 43
Table 4.1: Results from Heuristic Evaluation. Usability problems discovered by the usability specialist are shown along with corresponding cells explaining each Nielsen’s Usability Heuristic the problem violates, redesign recommendations, and the severity of the problem determined through the discussion between the usability specialist and the designer of FEM2DGUI. Usability problems marked with an asterisk are problems that were fixed for the next iteration of FEM2DGUI. Problems marked with a plus sign were indirectly corrected during redesign and redevelopment. ........................................................................................................... 52
Table 4.2: List of problems found by usability specialist and end-user participants, as well as list of problems found only by the specialist ........................................................................................................................................ 57
Table 4.3: List of problems found by end-user participants, categorized by how many users found the problems. For the problems only found by one end-user, and only the problems most affecting usability have been listed............................................................................................................ 58
Table 5.1: Answers to post-test questionnaires from Thinking Aloud Protocol and the informal final test. Results from participants B1, B2, and B3 are shown as well as averages and changes from the first questionnaire to the second ........................................................................................................... 85
Table 6.1: Thinking Aloud Protocol and questionnaire results from participant B1 .................. 100
Table 6.2: Thinking Aloud Protocol and questionnaire results from participant B2 .................. 102
Table 6.3: Thinking Aloud Protocol and questionnaire results from participant B3 .................. 104
Chapter 1: Introduction

1.1 Overview

The programmer’s ability to influence how a user of an interface conducts work and interprets results is significant. For example, a simple drawing tool will not provide the same results as a complex illustration environment. These environments themselves can also greatly influence a user of software – a graphic designer in this case – gets work done, depending on how each environment handles different sets of tasks. In other areas, such as consumer websites, interfaces have been shown to directly affect how a consumer decides on purchases (Huang & Fu, 2009). Since 1974 (Myers B. A., 1998) Graphical User Interfaces (GUIs) have become the dominant mechanism of interaction with a computer. GUIs are tools for human interaction with a computer through direct manipulation of graphics on the screen. GUIs replace the older, command-line interfaces that allowed for only text-based data visualization and keyboard input (Figure 1.1). While GUIs have become standard in desktop publishing and office productivity, they are less widely used in scientific computing. It is believed that the field of multi-dimensional geophysical analysis, and more specifically the sub-discipline of computational electromagnetics, would greatly benefit from further incorporation of GUIs. Furthermore, the tools for geophysics utilizing GUIs are often only found in specific industries, as in IGMAS+, the 3D gravity and magnetic modeling tool for oil and gas projects (Schmidt et al., 2007), or are reliant on 3rd-party proprietary software such as MATLAB, used for the 3D resistivity inversion package RESINVM3D (Pidliseck et al., 2007).

Understanding why certain inferences are made while a scientist interacts with a particular interface impacts the design and evaluation of scientific tools. In this study, the evaluation of a software prototype using concepts from User-Centered Design (See Glossary on iii) and Usability Engineering (See Glossary on iii) was conducted, attempting to answer the question: How does the structure of an environment or application affect how the user reasons through the 2D model-building process? An interface can be designed and developed to better suit users’ needs, and contribute to the production of models that can be built efficiently with an appropriate level of complexity, by incorporating known interface design techniques along with
an understanding of what steps a scientist makes while working through a model-building application. In addition to the need for more openly available, non-proprietary data visualization and model construction tools, the application of the Human-Computer Interaction (HCI) techniques of User-Centered Design and Usability Engineering will help to provide the building blocks of the initial interface for EM modeling and provide guidelines for quantitative design and evaluation metrics that could become the standard for EM software.

Figure 1.1: User interface hierarchy

1.2 Literature Review

According to Carroll (1997), Human-Computer Interaction is a science of design that has its roots in psychology but has since morphed into the study of how social and cognitive requirements drive the design process so that computers are designed to facilitate human activity and experience. The literature review consists of a brief history of GUIs, a look into the Usability
Engineering methods of Heuristic Evaluation (See Glossary on iii) and Thinking Aloud Protocol (See Glossary on iii), a few sample case studies using these methods, and a brief look at how Human-Computer Interaction goes beyond Usability Engineering. This review should provide the framework for understanding how human-computer interaction pertains to the design, development, and evaluation of the cognitive processes that will go into a geophysical computer-interface.

1.2.1 Graphical User Interface (GUI)

A GUI is a tool for human interaction with a computer through direct manipulation of graphics on the screen. Today GUIs such as those found in Microsoft Windows and Mac OS are ubiquitous and almost all software written utilizes user interface toolkits like Qt, GTK+, and Swing (Myers B. A., 1998). The most prevalent user interface paradigm today is the “window, icon, menu, pointing device” (WIMP) interface (Chignell & Waterworth, 1991; See Figure 1.1). The origins of the WIMP interface date back to 1974 with the development of the Smalltalk system at Xerox PARC (Myers B. A., 1998). This interface type became generally popular with the introduction of the commercially successful Macintosh in 1984 (Myers et al., 2000). Besides changes in aesthetics due to technological advances, WIMP interfaces have not changed much over the decades since the original GUI created at Xerox PARC, as they still rely on utilization of a pointing device to interact with the computer through widgets such as menus, forms, or toolbars (van Dam, 1997).

Van Dam (1997) states that the main advantages of WIMP-style interfaces and the reasons for their widespread use are that their ease of use, ease of learning, and consistencies in look-and-feel, lead to easy transfer of knowledge from one interface to another. “Point and click” has become the standard way in which to interact with a computer. Van Dam (1997) further demonstrates that interaction with these types of interfaces is not natural because users cannot touch, talk to, or gesture towards them as during communication with humans. Furthermore, the more complex a WIMP style interface becomes, the more layers of “point and click” and clutter of widgets must be added to accommodate the increased functionality. Widget clutter can decrease the usability of software because it can add to user frustration and force users to spend more time interacting with the interface and less time completing a task. Post-WIMP interfaces
(Myers et al., 2000; van Dam, 1997) are those that use at least one method of interaction other than “point and click”. These types of interfaces include smart phones that support voice and/or touch input (i.e. the iPhone), PDAs, and virtual reality systems.

1.2.2 Usability Engineering

Because of the nearly universal adoption of the WIMP-interface model since 1974, there has been much research conducted on the design, development, and evaluation of these interfaces. In fact, over the past 30 years, there have been more than 2,300 papers contributing to the Human-Computer Interaction literature making significant impact in the development of general principles of design and methodologies for evaluating proposed and existing designs for human-computer systems (Boehm-Davis, 2008). Methodologies in interface design and evaluation prove useful for designers and developers when planning and implementing GUIs. While programmers tend to understand hardware and efficiency concerns of the software, they typically are not the primary users of the software and may have difficulty visualizing the common problems that could arise while using the interface they design (Nielsen, 1993). Starting in the 1970s work promoting Iterative Development (See Glossary on iii) began to emerge. Iterative Development came about as an improvement to the waterfall model (Royce, 1987). The waterfall model is a development method where each step in the process is a concrete step that cascades into the next step after completion. Refinements can prove to be costly in this model because users may only suggest new features and change requirements after design at each step has been finalized (Royce, 1987). Iterative Development is the process of discovering new goals, prototyping, and evaluating through small increments, while maintaining strong involvement from the users (Carroll, 1997). However, there were differing opinions as to how people should correctly apply Iterative Development to software and how to avoid a trial-and-error approach. Usability Engineering arose as a way to standardize Iterative Development through the use of usability principles (Discussed in section 1.2.5).

According to Diaper and Sanger (2006), the five goals for the discipline of Human-Computer Interaction, as determined in 1989, were “to develop or improve the safety, utility, effectiveness, efficiency, and usability of systems that include computers”. Although usability was the least important on this list, it has become one of the most important concepts throughout
the discipline of Human-Computer Interaction since this time (Diaper & Sanger, 2006). Despite the functionality of a particular interface, it has been suggested that its usability (how “user-friendly” it is) is the ultimate indicator of its commercial success (van Dam, 1997). Advances in user interface methodology were driven mostly due to the success of desktop productivity tools during this time period, and it was found that the success of an application relied heavily on how usable it was (van Dam, 1997). Thus, the process of improving an application’s usability has pushed much of the Human-Computer Interaction work into design and evaluation methodology research, a goal that has more roots in engineering than in psychology and social sciences (Carroll, 1997). Usability Engineering became its own distinct practice of study during this time period as a way to apply structure to the Iterative Development and improve usability of software (Carroll, 1997).

Some researchers disagree with improving usability only to make software more commercially acceptable. Dix (2010) states that the “goal of an evaluation of a prototype within Human-Computer Interaction research should always be for gathering understanding not improving the prototype as an artifact.” Since the first papers published in Human-Computer Interaction in 1959 (See Shackel, 1997), the practice of studying Human-Computer Interaction has become highly fragmented and disjointed (Boehm-Davis, 2008). This is due mostly to the “great variety of methodologies, theoretical perspectives, driving problems, and people that have become part of Human-Computer Interaction” (Carroll, 1997). Carroll (1997) suggests that as software systems become more sophisticated and domain-specific there will likely be more research directed towards those domains. Thus, according to Dix (2010), as human-computer interaction has grown as a science, it has “encompassed a whole study of human endeavor and activity” that has gone beyond “the practice of creating systems and situations where people could effectively…interact with technology”. Thus, the study of usability and how it relates to Human-Computer Interaction is believed by some to have progressed beyond development of methodologies for making software commercially acceptable and into the of understanding how software affects users and how users affect software.

While usability is an important standalone concept for consideration during the development and implementation of an interface, it is worth discussing how usability fits into the framework of a system’s usefulness. The usefulness of a system is determined by whether it can be used to achieve a desired goal (Nielsen, 1993). Usefulness can then be divided into utility and
usability (Figure 1.2). Utility is defined as whether a system has the functionality required to complete some task, and usability is how easily that functionality can be used (Nielsen, 1993). Most often, these concepts do not exist as distinct entities, but are highly intertwined where each relies on the other. While both utility and usability are necessary components of a useful system, the question of which is more important is dependent on the type of system that is being designed (Grinstein et al., 2003). A system for solving highly technical problems may focus more on utility, and a system designed for entertainment may focus more on usability. While designers of a system may focus on utility over usability, it can be argued that the utility of a system will emerge out of an engagement between users and developers as they pursue usability (McLaughin & Skinner, 2000).

An example of software designed through improving usability is the Generic Mapping Tools (GMT). Before GMT (Wessel & Smith, 1991), earth scientists were often required to
engage in the tedious, sometimes expensive, process of creating publication-ready figures. The designers of GMT have sought to decrease the tedium of mapping and displaying data by combining a variety of data manipulation and display tools into one free, open source software package. As GMT has become more popular throughout the scientific community, the designers and developers have improved usability by incorporating feedback from users into newer versions that satisfy existing users while attracting new users (Wessel & Smith, 1991). Iterative development of GMT has led to improvements in utility, such as portability to hardware platforms other than UNIX and new mapping programs, and usability, such as bug fixes, more tutorials and examples, and full documentation (Wessel & Smith, 1998). While GMT does not employ a GUI for parameter input and requires UNIX shell-scripting, it has improved utility and usability over earlier methods of data display, providing scientists with a wide range of flexible mapping tools in one software package, by focusing effort on improving usability.

1.2.3 User-Centered Design

Although usability is a goal in most software projects today, this was not always the case. In the mid-1970s, rarely were computer systems designed with the end-user needs in mind. However, by the mid-1980s, the User-Centered Design (UCD) development process came about, in which usability is the primary goal (Carroll, 1997). User-centered design aims to “ask what the goals and needs of the users are, what tools they need, what kind of tasks they wish to perform, and what methods they would prefer to use” (Draper & Norman, 1986). Although specifics defining the key principles of user-centered design have changed slightly since 1986 (Gulliksen et al., 2003), the idea of keeping the users’ goals and needs number one has stayed the same.

An example of User-Centered Design employed on geophysical software is a dynamic picking system for 3D seismic data. Picking is the process by which seismologists localize, segment, and label structures, such as faults and geologic boundaries. This is accomplished by choosing points of interest on a 2D seismic cross-section. Today, picking is done by a human clicking on a computer image of the seismic cross-section with a mouse. For making picks in a 3D volume, a person must choose several of these points of interest on many 2D slices comprising the 3D volume while determining how each slice fits into the volume. This “slice-by-
“slice” picking technique, known as static picking, can be tedious and require a high cognitive load (Salom et al., 2009). Dynamic picking is a new interaction method presented by Salom et al. (2009) that simultaneously displays and animates a user-specified-path-based 2D slice of the volume displayed in a 3D view. An experiment was designed and carried out to test static picking versus their system of dynamic picking. The results of this experiment showed that 3D dynamic picking increases users’ volumetric perception of the data and leads to more rapid treatment of the volume. Salom et al. (2009) found they were able to employ a User-Centered Design approach during the development of their 3D dynamic picking application that helped to develop software that respected the expert user’s workflow while also allowing for the introduction of new concepts into the program, benefitting novice users.

1.2.4 Usability Evaluation Methods

Usability is made up of five attributes: learnability, so the user can rapidly get work started with the system; efficiency, so a high level of productivity can be achieved by the user; memorability, allowing the user to easily return to productive use of the system after a period of non-use; errors, reducing the rate of errors that occur while using the system; and satisfaction, making the system likable and pleasant to use (Figure 1.2). Usability Engineering is a discipline in which usability is “systematically approached, improved, and evaluated” (Nielsen, 1993). Holzinger (2005) states that Usability Evaluation methods are used to ensure that a given user interface exhibits most, if not all, of these usability characteristics. Usability evaluation methods are divided into two categories, usability inspection methods and usability test methods. Inspection methods identify usability problems and improve usability by comparing an interface against established standards. Common inspection methods include Heuristic Evaluation (See Glossary on iii), cognitive walkthroughs, and action analysis. Holzinger (2005) explains that usability test methods are methods that directly involve the user; therefore, they are indispensable tools because they provide information on how a user responds to and behaves when interacting with an interface. Common test methods include Thinking Aloud (See Glossary on iii), field observation, and questionnaires (Holzinger, 2005). In this research, one inspection method – Heuristic Evaluation – and one test method – Thinking Aloud Protocol – were utilized are further discussed in the following sections. These Usability Evaluation methods are integral to the
understanding of how the design of an interface can influence how a scientist works through the 2D EM problem, in an effort to design and create a more effective interface.

1.2.5 Heuristic Evaluation

Heuristic Evaluation (HE) is a common, informal, usability inspection method (Holzinger, 2005). It is a technique where usability specialists evaluate the usability of an interface using established principles, such as Nielsen’s Usability Heuristics (Nielsen, 1994; details of Nielsen’s Usability Heuristics are discussed in Chapter 3). Usability specialists are typically recommended when performing this type of study even if the software is highly domain specific. Heuristic Evaluations have been conducted where the evaluators are made up of domain specialists as opposed to usability specialists; however, these studies have been shown to produce lower quality results than those conducted with usability specialists (Nielsen, 1994). The evaluators use the interface individually, typically several times, and make determinations of usability using aggregated results and already-determined usability principles most appropriate for the particular study. For example, a traditional, WIMP-style GUI would potentially be evaluated against a different set of principles than a web-based service where additional principles become important (Holzinger, 2005). According to Nielsen (1994), three to five evaluators is an acceptable number, because at least three are needed to justify the cost of the study and after five there are diminishing returns concerning the number of usability issues found. Further, when three or more evaluators aggregate their results, there is better performance since different evaluators are able to locate a broader range of usability problems.

A benefit of Heuristic Evaluation is the application of known usability principles during interface development and the rapid and effective identification of usability issues early in development. However, there are a few disadvantages to Heuristic Evaluation, such as the separation between usability specialists and end users, difficulty in accounting for unknown users’ needs, and difficulty in identification of domain-specific problems (Holzinger, 2005). Also, Heuristic Evaluation does not explain to designers and developers how to fix the usability problems that are found. Heuristic Evaluation will only indicate whether usability errors are present (Nielsen, 1994).
These advantages and disadvantages of Heuristic Evaluation were seen in a study where it was applied to a computer-based, patient-education system (Joshi et al., 2009). In this study, three usability experts evaluated the patient-education system using Nielsen’s Usability Heuristics. Severity ratings were given to each of the discovered Nielsen’s Usability Heuristics violations, and the development team subsequently prioritized the changes needed, according to end-user needs, and updated the system. However, some of the changes suggested by the usability experts were not implemented due to the users of this system having a different set of requirements than the developers. The researchers of this study found that Heuristic Evaluation allowed for rapid evaluation of their patient education system. Known principles (Nielsen’s Usability Heuristics) were applied to the evaluation; therefore, both the software developers and domain experts understood the principle violations. However, because usability experts tend not to possess domain-specific knowledge, this study kept the domain experts, usability experts, and designers working together to ensure a system that was viable in practice. At the end of this study, they found that their newer prototype was more closely in tune with Nielsen’s Usability Heuristics, and was generally improved due to Heuristic Evaluation (Joshi et al., 2009).

1.2.6 Thinking Aloud Protocol

The usability test method used in the present study is commonly called Thinking Aloud (THA) (Gould & Lewis, 1985). Nielsen (1993) suggests that Thinking Aloud “may be the single most valuable Usability Engineering method”. Thinking Aloud has its roots in psychological research, but was widely embraced by the Human-Computer Interaction community in the 1980s (Carroll, 1997). A Thinking Aloud test involves having a participant use an interface while continuously verbalizing their thoughts of the system. This method makes it easy to understand how individual users interact with the system and highlights what concerns or misconceptions they may have. Typically, participants are audio and/or video recorded to later perform detailed protocol analysis. There are variations to the single-user test: constructive interaction, where two participants perform the test together; retrospective testing, where a participant watches the video of the test they took to provide more extensive comments; and the coaching method, where the evaluator attempts to steer the participant in the right direction while using the system (Nielsen, 1993).
Advantages of Thinking Aloud include that it can discover why users behave in a certain way and how they use the system, that it can produce a large amount of qualitative data from a limited number of users, performance and preference data can be collected at the same time, and early tests can provide hints into problems that could occur later during development (Nielsen, 1993; Holzinger, 2005). Some of the disadvantages are that it can seem unnatural to some users, users will often work differently when talking aloud than when working silently, and the method is time-consuming due to the necessary user briefings prior to testing (Nielsen, 1993; Holzinger, 2005).

In a study conducted by Kushniruk et al. (1996), Thinking Aloud was used to evaluate how physicians (participants) interacted with an outpatient information system. The focus of this study was to test whether the interface was able to allow clinicians to properly express information. Thinking Aloud was deemed an appropriate method of user testing to understand both problems users come across when using the system and physicians’ difficulties in using the medical vocabulary. All of this work was conducted as a part of the Iterative Development process. The researchers video-recorded participants while they were “thinking aloud” and then analyzed these sessions using methods of analysis from cognitive science. Results of the study included general usability ratings (based on users’ interactions and concerns with the system) and an analysis of how users tried to use medical vocabulary while using the system. Users suggested changes that they felt would benefit them while using the system, and based on these recommendations, short- and long-term goals were prioritized and later implemented into the system. The researchers found that Thinking Aloud provided valuable information about how users used and perceived the outpatient information system using only nine test subjects (Kushniruk et al., 1996).

Usability Engineering and the collection of usability metrics provide both quantitative and qualitative results. The quantitative metrics include task-completion time using the system and the accuracy of the results. Qualitative metrics include the perceived ease-of-use of the system and user satisfaction. The five aforementioned usability attributes (learnability, efficiency, memorability, low rate of errors, and satisfaction) can typically be measured by applying a rating score to them and comparing them to another system that attempts to achieve the same goals (Nielsen, 1996). The collection and use of quantitative and qualitative results have been shown to increase user productivity and decrease user errors, training costs, and
support. Also, changes are easier to implement, because Usability Engineering leads to usability errors being found early in development when they are easier to correct (Mayhew & Mantei, 1994). Usability Engineering not only has been employed to great success with GUIs, but Huang and Fu (2009) have shown that usability is a key factor in website usage and online service success as well.

1.2.7 Combining Usability Engineering Methods

While Usability Evaluation methods (inspection and test methods) are informative and useful individually, combining different methods during different stages of software design and development provides for a more complete usability analysis (Nielsen, 1993; Carroll, 1997; Holzinger, 2005). Gabbard, Hix, and Swan (1999) use both Usability Engineering and User-Centered Design to develop a methodology for the creation of the user interaction components of virtual environment applications. Their methodology consists of conducting a user task analysis, expert guidelines-based evaluation (Heuristic Evaluation), formative user-centered evaluation (end-user testing, such as Thinking Aloud Protocol), and summative comparative evaluations (aggregated analysis of guidelines-based evaluation and end-user testing). Although at the time of publication they had completed only the first three steps of their methodology, the results were positive leading up to the summative comparative evaluations, and were expected to continue to be positive (Gabbard et al., 1999). Further analysis of different Usability Evaluation methods, how they compare to each other, and how they are used to complement one another is provided in Chapter 3.

1.3 Research Synopsis

As previously mentioned, Salom et al. (2009) conducted a comparison study of different seismic picking systems for 3D seismic data using end-user testing. The goal of that study was to determine which system could offer the highest level of precision with the smallest amount of workload (Salom et al., 2009). This study conducted end-user testing and questionnaires to collect data from participants. Although the Salom et al. (2009) study deals with end-user testing
and geophysical software, the focus of their study was to compare two different picking systems and determine the utility of their newly designed dynamic picking system. However, for the present research, because the goal is to determine how well users can work with this interface while building 2D models, the focus is on the usability of a single interface with an assumption that utility will be adequately handled along throughout the project. To understand usability in this context and how it can improve this new model-building interface, a Usability Engineering experiment was designed and conducted. The following steps were taken in this study:

1. Interface design and development
2. Heuristic Evaluation
3. Thinking Aloud test
4. Analysis of Heuristic Evaluation and Thinking Aloud results
5. Refinement of interface
6. Formative Evaluation of the Redesigned Interface

Interface design and development was completed using a user task analysis to inform design on a fully functional interface (FEM2DGUI) for the 2D Finite Element Forward Module (FEM2D) (Weiss, 2010), the EM 2D forward modeling code of Weiss (unpublished). A Heuristic Evaluation was conducted on that interface to find most of the usability problems and general errors. A Thinking Aloud experiment was conducted to discover problems found by a representative group of users. During this experiment, the evaluator prompted and asked questions of the user attempting to understand why they were performing particular actions, how these actions related to the task, how they perceived the data, and what aspects of the user interface influenced their decisions while engaging in the forward-modeling task. An analysis of Heuristic Evaluation and Thinking Aloud was completed in order to understand how the interface and its usability affected the users’ objective reasoning. The results of this analysis were used to refine the interface into a second prototype correcting usability issues and incorporating the understanding of users’ objective reasoning in relation to the task of 2D, magnetotelluric forward-modeling. While the Usability Engineering and User-Centered Design methods introduced above certainly provide this interface with an increase in usability during the iterative design and development process, these methods, particularly Thinking Aloud, provide insight into the way in which an environment or application’s structure affects the user’s reasoning throughout the 2D EM model-building process.
1.4 Document Organization

The remainder of the document is organized as follows. Chapter 2 describes FEM2D (unpublished) and Triangle (Shewchuk, 1996), as well as the process and procedure that went into designing and developing FEM2DGUI. There is a walkthrough of the interface at the end of the chapter. Chapter 3 describes Usability Evaluation (inspection and test methods) methods other than Heuristic Evaluation and Thinking Aloud and justifies why these particular methods were used in this research. This chapter also outlines the method considered for usability assessment. Chapter 4 describes the modifications made to the Usability Evaluation methodology and provides the results of both the Heuristic Evaluation and Thinking Aloud Protocol. Finally, Chapter 5 draws conclusions and discusses the implications of this research on future geophysical interface design and development.
2.1 Principles of the Magnetotellurics Forward Problem, FEM2D, and Triangle

2.1.1 Magnetotellurics and the Forward Problem

Electromagnetic (EM) methods in geophysics have been in use since at least the first half of the 20th century. Rust (1938) used EM methods in hydrocarbon exploration, which demonstrates that some methods were well-established at that time. Today, EM is used for the detection of bodies of high electrical conductivity, such as metallic ores, underground pipes, and cables, as well as for hydrocarbon exploration. EM is also used for delineating faults, shears, thin conducting veins, and in hydrogeophysical studies (Sharma, 1997). These conductive bodies are mapped with EM because the spectrum of electrical resistivity values of materials in the Earth spans many orders of magnitude (Swift, 2006). EM field methods have as their source natural or artificial, electric or magnetic currents. These primary currents induce a secondary current that decays at a particular rate (Sharma, 1997). An EM receiver measures both the primary and secondary fields so that either the secondary or the total (primary plus secondary) field response is interpreted to give resistivity information of the subsurface (Swift, 2006).

The behavior of electromagnetic fields and how electric charges and currents create electric and magnetic fields are described by the Maxwell Equations ((2.1) – (2.4)). In these equations, \( B \) is magnetic induction (Wb/m^2 or Tesla), \( D \) is the dielectric displacement (C/m^2), \( E \) is the electric field intensity (V/m), \( H \) is the magnetic field intensity (A/m), \( \rho \) is the electric charge density (c/m^3), and \( J_{tot} \) is the total electric current density (A/m^3) and the summation of the source currents \( J_s \) and induced currents \( J_{ind} \).

\[
\nabla \cdot B = 0 \quad (2.1)
\]

\[
\nabla \cdot D = \rho \quad (2.2)
\]
Equations (2.5) - (2.7) are three constitutive laws describing the relationships between dielectric displacement and electric field intensity, magnetic induction and magnetic field intensity, as well as electric current density and electric field intensity. In these equations, the value \( \varepsilon \) is dielectric permittivity (F/m), \( \mu \) is magnetic permeability (H/m), and \( \sigma \) is conductivity (S/m). In most electromagnetic investigations of the Earth, we can assume that the magnetic permeability is the same as the magnetic permeability of free space or: \( \mu = \mu_0 = 4.7 \times 10^{-7} \text{H/m} \) (Ward & Hohmann, 2006).

\[
\begin{align*}
D &= \varepsilon E \quad (2.5) \\
B &= \mu H \quad (2.6) \\
J_{\text{ind}} &= \sigma E \quad (2.7)
\end{align*}
\]

EM fields are time harmonic and have an \( e^{i\omega t} \) time dependence (Ward & Hohmann, 2006). Therefore, the application of a Fourier transform to equations (2.3) and (2.4), along with the utilization of the constitutive laws (2.5), (2.6), and (2.7), yields Maxwell’s equations in the frequency domain (Ward & Hohmann, 2006):

\[
\begin{align*}
\nabla \times E &= -i\mu \omega H \quad (2.8) \\
\nabla \times H &= (\sigma + i\varepsilon \omega)E + J_s \quad (2.9)
\end{align*}
\]
We are able to safely ignore the displacement currents, because we are mostly dealing with low frequencies where displacement currents are much smaller than conduction currents, or $\varepsilon \omega \ll \sigma$ (Ward & Hohmann, 2006). Taking the curl of the equations (2.8) and (2.9) yields:

$$\nabla \times \nabla \times \mathbf{E} + i\omega \mu_0 \sigma \mathbf{E} = -i\omega \mu_0 \mathbf{J}_s$$  \hspace{1cm} (2.10)

Magnetotellurics (from magneto meaning “dealing with magnetic fields”, and telluric meaning “relating to electric currents in the Earth”) is an EM exploration technique using natural, broadband variations in the Earth’s magnetic field as a power source for EM induction in the subsurface (Simpson & Bahr, 2005). The first work into the study of magnetotellurics (MT) was conducted by Tikhonov (1950) and Cagniard (1953). Tikhonov (1950) briefly noted the mathematical relationship (impedance) between the magnetic and electric fields in the Earth. Cagniard (1953) discussed a combination of telluric and magnetic methods, coining the term magnetotellurics; furthermore, he described how one using the MT method could derive apparent resistivity from impedance, and how the MT method could be used to probe geologic layers in the subsurface.

The variations in the Earth’s magnetic field used for MT mostly come from two different sources: low frequency energy sources (0.001 – 1 Hz) originate from magnetic storms produced by the deflection of charged particles from the solar wind, by the magnetosphere and ionosphere which generates an electric field (Simpson & Bahr, 2005; Sharma, 1997), and high frequency energy sources (1 Hz – 20 KHz) originating from distant lightning events (Sharma, 1997). The high energy and low frequency of magnetic storms make MT sounding ideal for probing the Earth at great depths, sometimes up to hundreds of kilometers at the lowest frequencies (Simpson & Bahr, 2005); whereas, the low energy and high frequency of lightning events are better suited for more shallow investigations of the Earth. MT receivers measure a time series of the electric and magnetic fields of the Earth to a depth equivalent to the skin depth. EM signals from lightning, the ionosphere, and the magnetosphere quickly attenuate as they diffuse into the Earth. The distance over which the electric and magnetic fields attenuate by $1/e$ is known as the “skin depth” (West & Macnae, 2008), shown in (2.11).
Since skin depth is a function of frequency and increases as source frequency decreases, recording a range of frequencies can yield a depth sounding of resistivity (Unsworth, 2005). Two-dimensional MT is analyzed in two different modes: transverse electric (TE) mode, where the electric field is parallel to strike, and transverse magnetic (TM) mode, where the magnetic field is parallel to strike.

Forward modeling is a common technique for the interpretation of geophysical data (Sharma, 1997), including, but not limited to seismic methods, gravity methods, and EM. Forward modeling is used for numerical hypothesis testing and as an iterative method for determining a subsurface model based on observational data. Numerical hypothesis testing is used as a learning tool to study how different parameters affect the observations and as a way to determine how one will set up an experiment. Forward modeling for EM is the process of determining what an EM sensor would measure by creating a geological model and then applying Maxwell’s Equations to compute its electromagnetic response (Gillis, 2010). The iterative method for forward modeling as described by Sharma (1997) is one where a skilled guess for the initial model is made based on known geology of the study area. This is followed by computing the MT response of receivers placed in the model, and a comparison of the simulated receiver responses with observed receiver response. The initial model is then iteratively updated until the simulated receiver response reasonably matches the observed receiver response.

2.1.2 FEM2D

The purpose of FEM2DGUI is to allow users to quickly parameterize a representation of the subsurface, run a simulated MT experiment using FEM2D, and produce an output graph of apparent resistivity and phase along the air-earth interface. FEM2DGUI only works with the 2D
MT case; therefore, conductivity only varies in the horizontal (x) and vertical (z) directions, and there is a plane-wave source assumption. Furthermore, the following boundary conditions must be applied to derive the governing PDEs (Equations (2.13a) - (2.14b)) for MT:

1. Tangential $E$ is continuous across the air-earth interface
2. $E$ goes to zero as the depth below the air-earth interface increases to infinity
3. $E$ goes to zero as the height above the air-earth interface increases to infinity
4. Normal $B$ is continuous across the air-earth interface

As mentioned above, TE and TM are the two modes for MT data interpretation. In the TE mode, $E$ is parallel to conductivity strike and the source currents are only directed in the $y$-plane. In TM mode, $H$ is parallel to conductivity strike and the source currents are directed only in the $x$-plane. That is,

\[ E = E_y(x,z)\hat{y} = u(x,z)\hat{y} \]  

\[ H = H_y(x,z)\hat{y} = u(x,z)\hat{y} \]  

Where $u$ is strike-directed electric field (2.12a) in the TE mode, and $u$ is the strike-directed magnetic field (2.12b) in the TM mode. In both modes, $I$ is current in amperes and $\delta_d$ is the Dirac delta function. The right-hand-side of each AIR equation states that there exists a source current at some height, $h$, above the air-earth interface. Equations (2.13a) - (2.14b) are the governing PDEs for TE and TM modes in two dimensions.

**TE:**

AIR: \[- \nabla^2 u + i \omega \mu_0 \sigma u = I \delta_d(z + h)\]  

EARTH: \[- \nabla^2 u + i \omega \mu_0 \sigma u = 0\]  

**TM:**

AIR: \[- \nabla \left( \frac{1}{\sigma} \nabla u \right) + i \omega \mu_0 u = I \delta_d(z + h)\]  

EARTH: \[- \nabla \left( \frac{1}{\sigma} \nabla u \right) + i \omega \mu_0 u = 0\]
Because (2.13a) - (2.14b) share a similar structure, FEM2D solves the generalized equation (2.15), where $k$ is 1 in TE mode and $1/\sigma$ in TM mode, $u$ is the strike-directed electric field in TE mode and the strike-directed magnetic field in TM mode, $q$ is $i\omega\mu_0\sigma$ in TE mode and $i\omega\mu_0$ in TM mode, and $s$ is $\Im(\delta z + h)$ in the AIR regions and 0 in the EARTH regions. Distinguishing between TE and TM modes or air and Earth regions is accomplished through the appropriate choice of $k$, $q$, and $s$.

$$-
abla \cdot (k \nabla u) + qu = s \quad (2.15)$$

From (2.15) one can solve for $u$ and in the TE mode, apply Faraday’s Law (2.8) to determine the x and z components of $\mathbf{H}$, and in the TM mode, apply Ampere’s Law (2.9) to determine the x and z components of $\mathbf{E}$. With these components of $\mathbf{E}$ and $\mathbf{B}$, one can find the MT impedances for the TE and TM modes, (2.16a) and (2.16b).

$$Z_{yx}^{TE} = \frac{E_y}{H_x} \quad (2.16a)$$

$$Z_{xy}^{TM} = \frac{E_x}{H_y} \quad (2.16b)$$

Impedance is useful because one can find resistivity without having to know source strength. To extract true resistivity from the electric and magnetic fields, one must assume electric and magnetic fields are plane waves and start with the boundary-value problem (2.17) describing the relationship between $\mathbf{E}$, $\mathbf{H}$, and resistivity at each angular frequency $\omega$, at the surface of a uniform half-space of conductivity $\sigma$ (Stratton, 2007). In equation (2.17), $k = (1 - i)\alpha$, $\alpha = \sqrt{\frac{\omega\mu_0\sigma}{2}} = 1/\delta$, where $\delta$ is the skin depth, and $\mathbf{n}$ is a downward-pointing unit vector.
\[ H = \frac{k}{\mu_0 \omega} n \times E \]  

(2.17)

Vozoff (2008) explains that (2.17) and, particularly the ratio \( E_x/H_y \) (the following equations are shown for impedance in the TM mode, but they similarly work for the TE mode), is important because

\[ E_x/H_y = (1 + i) \sqrt{\frac{\omega \mu_0}{2\sigma}} \]  

(2.18)

Furthermore, Vozoff (2008) states that (2.18) shows the relationship between the measured electric and magnetic fields and conductivity, or resistivity \( \rho \), where \( \rho = 1/\sigma \). Rewriting (2.18) using the impedance tensor (2.16b) and resistivity shows

\[ Z_{xy} = (1 + i) \sqrt{\frac{\rho \omega \mu_0}{2}} \]  

(2.19)

and, solving for \( \rho \),

\[ \rho_{xy} = \frac{|Z_{xy}|^2}{\omega \mu_0} \]  

(2.20)

Equation (2.20) is the resistivity at the surface of a uniform half-space. This calculation can still be used at a surface in a two-dimensional model, but now the result depends on frequency and is an apparent resistivity, as opposed to the true resistivity (Vozoff, 2008). Cagniard (1953) describes apparent resistivity to be a loose average of the resistivities found in a thickness of ground, and that apparent resistivity can be smaller than the smallest or larger than
the largest of resistivities found in that thickness of ground. Therefore, apparent resistivity is only an approximation of the resistivity structure of the subsurface.

For simple Earth models, (2.15) can be solved analytically. However, when Earth models are more complex, analytic solutions are not practical, and numerical modeling is required. The 2D Finite Element Forward Module (FEM2D) (Weiss, 2010) is a 2D magnetotellurics forward-simulator that solves (2.15) for either the TE or TM mode using the finite element method. The finite element method is a technique used to solve partial differential equations numerically by breaking up the model space into elements with known, finite dimensions. These elements are fully represented by a finite-dimensional set of basis functions, such as the tent function. FEM2D then calculates the response at each node of the elements making up the mesh that defines the model space.

2.1.3 Triangle

As discussed above, FEM2D and the finite-element method require a discretization of a model. Triangle (Shewchuk, 1996) is an open-source C program for two-dimensional mesh generation and construction of Delaunay triangulations, constrained Delaunay triangulations, and Voronoi diagrams. A Delaunay triangulation attempts to maximize the minimum angle of each of the triangles of an area (Bern & Eppstein, 1992). Input into Triangle is a list of points in model space and the segments that connect these points (also known as a planar straight line graph), along with attributes (in this case, conductivity/resistivity), holes (to prevent triangles from being produced within a particular segment), and maximum triangle area (effectively, how many triangles are in a region or segment). The output of Triangle is a file containing the nodes (vertices) of each triangle and another file containing information for each element. These files are read by FEM2D and comprise the elements that into which the model space has been discretized.
2.2 Design and Development Process

2.2.1 Initial Design and Development

The purpose of FEM2DGUI is to provide a simple tool with which to conduct 2D model building, triangular mesh generation, and forward model simulation. The existing methods of creating models and running FEM2D is text-based and the openly available, non-proprietary tools are poorly developed in comparison to tools found in other geophysical disciplines (i.e. seismology); therefore, a new interface (FEM2DGUI) has been designed and implemented. One of the goals of the project is to produce a fully functional Graphical User Interface (GUI) for evaluation; therefore, care was taken in the initial design process to facilitate both the development and testing of FEM2DGUI.

User-task analysis, understanding end-user goals, and application of usability principles were all applied in the design and implementation phases of this project. Furthermore, a “less-is-more” approach, similar to that used in at least one geovisualization tool (Jones et al., 2009), where simple and clear design can lead to easy-to-use, clear applications, was used to keep both the design and implementation of the interface simple and focused. Jones et al. (2009) maintained a user-oriented and minimalistic design approach to maximize users’ understanding of the problem space. These researchers found that the “less-is-more” approach provided a system that provided for clearer presentation of the data. Similarly to Jones et al. (2009), the designer of FEM2DGUI believed that maintaining end-user goals, complying with known usability principles, and placing less emphasis on technological advances, throughout the design and development of FEM2DGUI, would provide for an interface with both high utility and usability.
<table>
<thead>
<tr>
<th>Who will be using this interface?</th>
<th>Electromagnetic methods practitioners</th>
</tr>
</thead>
<tbody>
<tr>
<td>What will they be doing with this interface?</td>
<td>Model building and visualization for the 2.5-D forward modeling problem</td>
</tr>
<tr>
<td>Why are they doing it?</td>
<td>Numerical hypothesis testing</td>
</tr>
<tr>
<td>Where will they be doing it?</td>
<td>On a desktop system</td>
</tr>
<tr>
<td>How will they be doing it?</td>
<td>Individually or in groups</td>
</tr>
</tbody>
</table>

Table 2.1: User task analysis questions

A face-to-face user-task analysis of the designer and expert user of the FEM2D forward-modeling code, Chester Weiss, was conducted as the first step of the interface design in order to understand the particular tasks and the flow a user follows while completing these tasks. A set of questions (Table 2.1), adopted from Jones et al. (2009), were then addressed based on the results of this conversation. From this analysis, it was found that the users for which this system will be developed are electromagnetic-methods practitioners with intermediate to expert knowledge of computer use and the 2D magnetotellurics forward-modeling problem. If, after this tool has passed evaluation processes, it is deemed robust enough for distribution, there will exist a secondary user base consisting of a broader range of academic electromagnetic practitioners. Therefore, the need for a simple, easy-to-use interface design is necessary, to facilitate EM practitioners to rapidly produce MT models, and for a potentially broader, less knowledgeable, user-base.

The user-task analysis also discovered the necessary steps for a successful iteration of a 2D magnetotellurics forward model simulation. The structure of this interface is represented visually in Figure 2.1 and is described here in terms of that figure. Paths from each task in Figure 2.1 are bi-directional indicating that users should have had the ability to seamlessly move from one task to another in any order desired. The users of this system need to be able to define subsurface conducting bodies in real-world coordinates (i.e. draw geologies). These conducting bodies could theoretically be geologic layers, anomalous bodies (i.e. metallic ores, underground pipes and cables, etc.), faults, shears, thin conducting veins, or a variety of other bodies of high conductivity (Sharma, 1997). Defining these bodies consists of creating points (vertices) in the horizontal (x-axis) and vertical (z-axis) directions and defining the segments that connect these points. Users need the ability to apply parameters, such as lithology, conductivity (or resistivity), to the different geologic layers and anomalous bodies (apply lithology/apply conductivity). Users need to be able to create input files for Triangle (Shewchuk, 1996) to generate meshes. Along
with generation and construction, users need to be able to view these meshes (create mesh). Users need to interact with the FEM2D software by creating the necessary input file, running the forward simulator, and retrieving the output from FEM2D (create output). A graph of the data produced by FEM2D must be created (to plotter). Due to the high variability of these subsurface conducting bodies, the need to design and implement a functional interface, and time constraints, the above user needs were broken down into the seven most important features for FEM2DGUI:

1. Accommodate (tilted) layered Earth
2. Allow for the inclusion of simple isolated bodies
3. Allow for the insertion of a background seismic image for reference
4. Allow for mesh generation and construction with Triangle (Shewchuk, 1996)
5. Generation of input file for FEM2D forward modeling simulator
6. Apply necessary parameters to the different layers and/or isolated bodies
7. Convert FEM2D output into a format for plot generation

Figure 2.1: Logical Flow of Model Builder Interface
The design concepts of metaphor, mental models, and style guides were used to develop FEM2DGUI. Interface metaphors use concepts of already defined visuals, communication techniques, and procedures with which users of a system are already familiar (Marcus, 2002). The metaphor of FEM2DGUI is the desktop metaphor employing the features defined in Chapter I during the discussion of the WIMP interface. This was chosen because FEM2DGUI was expected to be run in a desktop computing environment by users familiar with the WIMP-style interface. A mental model is the flow in which a person structures or organizes data in order to accomplish a task (Marcus, 2002). FEM2DGUI design incorporates methods of User-Centered Design and User Task Analysis to understand how a user of this interface steps through the system to accomplish the forward-modeling task. The mental model of this interface is similar to the existing text-based system in the way that the user proceeds through the modeling task (parameterizing models, generating meshes, running FEM2D, and viewing output); however, the forward-modeling process is presented graphically as opposed to text-based. Style guides provide a set of standards that can be applied to an interface to keep the interface consistent. Examples of style guides include Mac, Windows, and KDE. The style guide chosen for this project was the KDE platform style guide: (http://developer.kde.org/documentation/standards/kde/style/basics/index.html), because the code for FEM2DGUI is written in C++ using the Qt framework (http://qt.nokia.com/) and KDE is developed using the Qt framework. Because Qt is so closely tied to KDE, the pre-defined widgets and buttons lend themselves to this particular style. Some aspects of the KDE style include single document interface (windows used in the system are not contained within a larger parent window), the use of already defined accelerators used for that task alone (Ctrl+C will always be a copy command and nothing else), and the consistent use of domain-specific terminology and capitalization techniques.

FEM2DGUI was written in C++ using the Qt framework to provide the widgets (menus, toolbars, etc) and for Qt’s relative ease in creating cross-platform GUI applications. The Qt framework has been developed since 1994 with a “write once, compile anywhere” approach (Blanchette & Summerfield, 2008). The Qt framework employs a powerful 2D graphics canvas allowing many custom-made 2D graphic items to be displayed on the screen, which is necessary for a user interacting with FEM2DGUI. Further, users can interact with these items to
manipulate, click, and drag these items to position them as needed. Also, the Qt framework allows for GUI applications to be run on many different platforms including Windows, Mac, and X11 with a single set of source codes.

This interface is designed to be a “single document interface” relying heavily on dialog boxes to collect information from the users. It is necessary to have a 2D drawing area for users to create models that accepts input from the mouse (as opposed to a stylus or other non-mouse method of input), a toolbar with icons with actions that correspond to actions found in file menus, horizontal and vertical scale bars, a method for users to locate the mouse in real-world coordinates, a method for saving a model, and the ability to load that and other previously saved models at a later time. All design considerations concerning the look-and-feel of FEM2DGUI were made to facilitate users’ interaction with the interface and model building.

The following sections describe how FEM2DGUI accomplishes each of the seven features outlined earlier in this section:

2.2.2 Feature 1: Accommodate (tilted) layered Earth

Users must be able to add horizontal and dipping layers to the model view. Not only do these layers need to be inserted at different angles, users must also be able to attach other points, layers, and bodies to the layers. These layers can be any shape provided they start on one side of the model view and end on the other, and do not pass through another layer, anomalous body, or themselves, as these would be geologically unreasonable. FEM2DGUI is designed and developed to accommodate a layered Earth where users add points to the screen with the mouse. These points are connected by lines to ensure that users recognize which points are connected and where segments are made. The points float under the mouse until the user places them on the drawing screen. Points become red when they have been placed near the screen edge, where they “snap” into place when clicked. The user loses control of drawing points when they have connected one edge of the screen to the other or to some other point already placed on the screen.
2.2.3 Feature 2: Allow for the inclusion of simple isolated bodies

For anomalous bodies to be drawn, points with connecting lines must be able to be connected with one another or to the points of another layer, shape, or model-view edge, to create a closed shape. As with the layers, lines connecting points cannot intersect any other lines. Because these bodies can be any shape, not restricted to regular geometric shapes such as rectangles and circles, the drawing of these bodies is similar to that of drawing layers: a point is added to the screen followed by more points until the user clicks on the original point or a point on another layer, shape, or model-view edge, to close off the shape. All of these points are designed to have lines connecting them.

2.2.4 Feature 3: Allow for the insertion of a background seismic image for reference

Users must be given an option so they can open a file containing a seismic image for placement into the background of the drawing screen. FEM2DGUI was designed so that users can click a menu option to bring up a dialog box to place an image file into the background of the drawing screen. This image then covers the entire background of the draw-view screen. Image files do not contain metadata such as position and depth coordinates; therefore, users must be careful to set model parameters and coordinates that coincide with the seismic image they have selected.

2.2.5 Feature 4: Allow for mesh generation and construction with Triangle (Shewchuk, 1996)

The design of this feature requires the creation of a text file for input to Triangle that contains certain parameters: vertices, segments connecting vertices, attribute locations and values, and the maximum area of each triangle in each region. For proper Triangle input file creation: points that make up the boundaries must be exactly on the boundaries to ensure that Triangle does not interpret two different regions as one, users must be able to add conductivity values to each of the regions they draw, users must be able to define how dense or coarse the
mesh is in a particular region, and users must be able to specify the input file to Triangle. The points the users add are the vertices that are loaded into the Triangle input file, and the lines connecting these points are the segments Triangle uses to define mesh regions.

Triangle creates two output files. One of these contains node information and the other contains the coordinates of each of the elements of the mesh. These two files are needed by FEM2DGUI to be read so the interface can display this mesh on the screen for users to review. Therefore, a mesh-viewing screen must be available where the triangular mesh will be viewable along with the conducting bodies. The conducting bodies are colored according to their conductivity, where warm colors are more resistive and cool colors are more conductive.

2.2.6 Feature 5: Generation of input files for FEM2D

FEM2D requires four input files. Two of these files, the node and element files, are created by Triangle; however, this interface changes their names to suit the naming scheme required by FEM2D. The third file lists the location of the receivers in x and z coordinates. This input file requires that the users be able to insert receivers into the model view at specific x and z locations. Because FEM2D expects a particular name for the receiver file, FEM2DGUI, not the user, names this file. The fourth file needed for FEM2D contains the source frequency or frequencies. The initial design of this interface only allowed for one frequency to be specified by the user, so this input file was only one line; however, this could be altered in the future to allow for more frequencies per forward simulation. As with the receiver location file, the frequency file was not named by the user, but is named by FEM2DGUI.

2.2.7 Feature 6: Apply necessary parameters to the different layers and/or isolated bodies

The user needs to apply conductivity/resistivity values to different regions of the model space. To insert conductivities the model space is designed so that users can click anywhere in a given region to apply an attribute to that region. Then, a marker shows on the screen to alert the user that an attribute is located there. The user must be able to specify how coarse or fine the
triangular, finite-elements mesh should be for each region. The coarseness and fineness of the different regions is defined by the maximum area of each triangular element in a region. Because mesh generation is a difficult task, FEM2DGUI was designed to make this task as simple as possible. A dialog box prompts the users to define a number representing “nodes per skin depth” when users click the “Generate Mesh” button. The nodes are the vertices of the triangles in the mesh; therefore, the maximum area of the equilateral triangles that comprise a region in the mesh is determined by dividing the number of “nodes per skin depth” declared by the user by skin the skin depth of that region. This value is an approximation of the length of the sides of each equilateral triangle. From the length, the area of each triangle is easily determined (2.21). The “nodes per skin depth” function ensures that more conductive regions have a finer mesh in a region (smaller area per triangle), and less conductive regions have a coarser mesh in a region (larger area per triangle).

\[ A = \frac{1}{2} l^2 \sqrt{3} \]  

(2.21)

2.2.8 Feature 7: Convert FEM2D output into a format for plot generation

Ultimately, the purpose of building these models and generating these meshes is to run the forward model simulations. For these simulations to be useful, an output graph must be produced by the interface for the user to interpret. For this, a third party plotting library, QWT, is used because the Qt toolkit does not have any predefined graphing or charting libraries. The charts show apparent resistivity vs. position as well as a graph of phase angle vs. position. Slight modification is needed in the FEM2D code to output apparent resistivity and phase angle into a text file at each receiver location. The FEM2D code needs modification to compute the impedance tensor. From this impedance tensor the apparent resistivity and phase can be easily calculated. These values are written to a file to be read by the interface for graphical output. After the forward model simulation, the graph shows the results, unless no viable output is produced and then the user is alerted to the fact that the forward model did not produce any output.
2.3 Walkthrough

The initial screen the user interacts with represents a cross-section of the sub-surface where the user can draw conducting bodies by making points by clicking the mouse. This screen also opens with a dialog box so the user can select the coordinates of the model immediately (Figure 2.2). When these points are drawn on the screen, lines connecting the points are also drawn to represent boundaries (Figure 2.3). For simplicity, there is one tool for drawing these points as well as layers and anomalous bodies, connection of points to one another, point deletion, and movement of points after they have been inserted into the drawing area. Also, the dialog box allows the user to insert a seismic stacked section to the background, and the user can define the x and z coordinates of the stacked section. As the user moves around the drawing area, the status bar of the main window displays the mouse location in real-world coordinates. Also in this screen, the user can assign attributes, such as conductivity values, to the conducting bodies that will be used in Triangle’s mesh generation and construction algorithms (Figure 2.4). Receivers can also be added to the model (Figure 2.5 and Figure 2.6). The user then clicks a menu item to create the necessary files and define parameters for Triangle to generate the mesh (Figure 2.7). Along with these menus, there are toolbar icons at the top of the screen to perform the most likely to be repeated tasks.

After Triangle has generated a mesh, that mesh can be viewed by clicking a menu item that changes the state of the interface to mesh-viewing mode. This mode has an almost identical look-and-feel to the original state, except now the mesh is displayed and certain items, such as the point creator are disabled because they would have no use here. The mouse location is still given in real-world coordinates in the status bar and the user can click a menu item to return to the drawing mode to change the layout of the conducting bodies. From this mesh drawing state the user can click another menu item to generate the necessary input files for FEM2D and run the forward model simulator. Figure 2.8 shows a generated mesh and how it looks in the mesh display mode. The user now has the ability to run a forward-model simulation. After a simulation is run the output is displayed in a new window created by FEM2DGUI.
Figure 2.2: Opening screen of the interface

Figure 2.3: Model with line at the air-earth interface, and anomalous body, and a deeper layer
Figure 2.4: Attributes (conductivity values) added to the model

Figure 2.5: Dialog box for adding receivers
Figure 2.6: Receivers (upside-down triangles) added to the model

Figure 2.7: Dialog box for specifying nodes per skin depth and source frequency
Figure 2.8: Generated mesh where colors denote log10 conductivity
Chapter 3: Justification and Generalized Methodology for Usability Evaluation of FEM2DGUI

3.1 Chapter Overview

The purpose of Chapter 3 is to explain, analyze, compare, and contrast different methods for a usability analysis, as well as provide a generalized methodology for the usability analysis of FEM2DGUI. The analysis of these methods is more detailed than Chapter 1 to provide the reader with more background information into methods other than Heuristic Evaluation and Thinking Aloud Protocol. This is followed by an explanation of why Heuristic Evaluation and Thinking Aloud Protocol are the most proper methods for the analysis of FEM2DGUI. Sections 3.3 and 3.4 provide detailed methodology for the Heuristic Evaluation and Thinking Aloud Protocol conducted in this study. Results of the Heuristic Evaluation and Thinking Aloud Protocol are found in Chapter 4. This is because there were slight deviations to the methodology presented in this chapter that will be discussed in Chapter 4.

3.2 Usability Evaluation Methods

In Chapter 1, Heuristic Evaluation and Thinking Aloud Protocol are introduced as Usability Evaluation Methods. However, there exist many other ways to measure the usability of software, and these Usability Evaluation methods are presented in the next three sections to provide background and justification as to why Heuristic Evaluation and Thinking Aloud Protocol were used for this study. There are four common evaluation methods: automatic, using software that measures usability; empirical, using usability test methods; formal, using exact models and formulas to calculate usability; and informal, usability inspections made using rules of thumb and experienced evaluators (Nielsen, 1994). Automatic methods, although desired because they allow for the rapid inspection of projects such as web projects that move through a quick iterative process, are still lacking in how well they improve the usability of interfaces.
Ivory et al. (2003) showed that automatic evaluation methods for the evaluation of web sites show potential, but are lacking, and the intuition of an experienced web site developer provides the best design considerations and evaluations. Empirical methods (usability test methods) are the most common Usability Evaluation methods, but are often costly and time consuming (Nielsen, 1993). Formal methods are difficult to conduct and do not scale well for large projects (Nielsen, 1994). Informal methods are most commonly known as Usability Inspection methods. These employ usability experts as evaluators to examine and evaluate the user-related aspects of a user interface. Usability Inspection methods are fast to implement and yield satisfactory results with few evaluators (Mack & Nielsen, 1994; Nielsen, 1994). Both informal (Usability Inspection) and empirical (Usability Test) methods were used to evaluate the usability of the software in this project.

3.2.1 Usability Inspection Methods

Heuristic Evaluation was the Usability Inspection methods utilized for this project. Listed here are the most common Usability Inspection methods according to Nielsen (1994). These methods are presented here to provide background and insight into why Heuristic Evaluation was chosen over the various other Usability Inspection methods:

- **Heuristic Evaluation** – informal, usability specialists judge whether each dialogue element follows established usability principles,
- **Cognitive Walkthroughs** – designers simulate a user’s problem solving process at each step through the dialogue, checking if the simulated user’s goals and memory content can be assumed to lead to the next correct action,
- **Formal usability inspections** – combine heuristic evaluation and cognitive walkthroughs using exact models and formulas to calculate usability measures,
- **Pluralistic walkthroughs** – similar to cognitive walkthrough, but many people are involved in the design process including users, developers, and usability specialists,
- **Feature inspection** – list sequence of features used to accomplish typical tasks, checks for cumbersome or unnatural steps, and steps that require extensive knowledge/experience in order to assess a proposed feature set,
- Consistency inspection – designers representing multiple projects inspect an interface to see whether it operates in the same way as their own designs,
- Standards inspection – an expert on some interface standards inspect the interface for compliance.

3.2.2 Usability Test Methods

Usability Test (empirical) methods, as mentioned in Chapter 1, are the direct ways in which to gather information about how an end user interacts with a particular system (Holzinger, 2005). These methods are most commonly Thinking Aloud Protocol, field observation, and questionnaires. Thinking Aloud Protocol is described in detail in Chapter 1 and justification for its use in this study is provided later in this chapter. Field observation involves an observer taking notes while users work with software under normal conditions. These types of tests provide good results for obvious usability problems provided the observer is able to remain as unobtrusive as possible to the user (Holzinger, 2005). Questionnaires are used to gather information directly from users that would be difficult or impossible for a test administrator or observer to ascertain, such as their backgrounds, opinions about a particular task, and opinions about the software they are testing (Dumas & Redish, 1993). Often, these usability test methods are combined as a way to gather more information from users. For example, questionnaires can be given to users before and after Thinking Aloud Protocol to collect more information about the user. Pre-test questionnaires are primarily used to gather information about the user, such as understanding of the domain, knowledge of the software being tested, computer experience, etc. Post-test questionnaires are used to collect information about users’ experience and satisfaction with the software during a test. These questionnaires allow for users to provide further feedback that they were unable to give during the test, or that they wish to reiterate. In order to take full advantage of both usability inspection and usability test methods, they should be combined (Holzinger, 2005). There have been several studies showing that combining inspection and test methods leads to discovery of usability problems that these methods may independently overlook (Nielsen, 1994).
3.2.3 Analysis of Usability Inspection and Usability Test Methods

The following are studies that have been conducted comparing the advantages and disadvantages of several different Usability Evaluation Methods, and have influenced the decisions about which methods are best for this study.

Jeffries et al. (1991) compared four evaluation methods—heuristic evaluation, software guidelines, cognitive walkthroughs, and usability testing—to examine their relative effectiveness in finding usability problems. The Heuristic Evaluation of Jeffries et al. (1991) differs slightly from that described in Chapter 1 in that they did not use a set of defined guidelines or principles such as Nielsen’s Usability Heuristics. They attempted to determine how many and what kind of usability problems each of these methods was able to uncover, as well as the relative cost-effectiveness of each method. Each evaluation was performed using typical, real-world scenarios. The results of the study show that heuristic evaluation found the most recurring and general usability problems and also had the greatest cost-effectiveness. However, the findings suggested that Heuristic Evaluation focused too much on specific, one-time, and low-priority problems. Usability Testing was found to be the second most effective method at finding problems and the problems discovered were mainly high-level, high-priority problems; however, this was the most expensive evaluation to conduct. The authors concluded that cognitive walkthroughs and guidelines-based evaluations are useful when applied by software designers in lieu of usability specialists, but Heuristic Evaluation and Usability Testing were the most effective evaluation methods—most likely due to the evaluators’ knowledge in usability studies (Jeffries et al., 1991).

Karat et al. (1992) performed another study to examine how Usability Testing methods and Usability Inspection methods compare to one another. The informal methods were individual walkthrough and group walkthrough methods. The term “walkthrough” in this study is more similar to Heuristic Evaluation and less similar to Cognitive Walkthrough. Evaluators completed self-guided exploration and scenario-based tasks to develop a list of usability problems that conflicted with a set of usability principles. Like Jeffries et al. (1991), the study aimed to answer how the results of Usability Testing differed from walkthrough methods concerning the number of usability problems discovered, the reliability of these differences, the cost-effectiveness of
these methods, and the human factors involvement. The results of this study show that the Usability Testing of users uncovered more usability problems than both the individual and group walkthrough methods, as well as discovering many more severe problems. This study also found that Usability Testing was the most expensive test option but it was considered the most cost-effective due to the greater number of uncovered usability problems (Karat et al., 1992).

Desurvie et al. (1992) compared Heuristic Evaluation with Cognitive Walkthroughs. As well as studying how these methods compared to one another in how well they were able to find usability problems, each method was performed with three different groups of evaluators: experts, system designers, and non-experts. Their findings showed that experts performing the Heuristic Evaluation were best at predicting the most serious problems. Using both cognitive walkthrough and heuristic evaluation with system designers and non-experts yielded almost identical results. All groups except the expert groups found a very small number of problems.

The discrepancies between the Jeffries et al. (1991), Karat et al. (1992) and Desurvie et al. (1992) studies are probably due to the expertise of the evaluators conducting the different sets of evaluations. However, despite the discrepancies between the three studies, there are many similarities. They all show that of the different Usability Inspection methods Heuristic Evaluation uncovers the most usability problems—provided that the evaluators are usability specialists. According to Jeffries et al. (1991) and Desurvire (1992), this is not surprising because Heuristic Evaluation is essentially using the intuitions of experienced usability specialists. These studies also described how Usability Inspection methods are unable to completely replace Usability Testing methods. This is why, as described in Chapter 1, this project will conduct both a Heuristic Evaluation with usability specialists and a Usability Test of end members.

Heuristic Evaluation is a good choice for the present research because, as mentioned above, it is cost-effective and good for finding usability problems. It is also a good method for Usability Inspection of FEM2DGUI because Heuristic Evaluation is good for testing prototypes and domain specific software. The evaluation conducted here is similar to the case study of the Heuristic Evaluation presented by Nielsen (1994). They found that performing Heuristic Evaluation on a highly domain specific problem—in their case, telephone company software—is an effective method for uncovering usability problems. Here, not only was Heuristic Evaluation used to find usability problems and to understand the steps users take to accomplish tasks, but it
also provided a framework to better understand how usability problems are uncovered through the observation of usability experts and what types of problems to expect when conducting the Thinking Aloud Protocol.

As discussed above, end-user testing provides the proper determination of usability problems and how they affect the end user; therefore, a usability test, as introduced in Chapter 1, is conducted in this study. The goal of the end-user testing is to understand how this interface’s usability affects how well the end-user can accomplish their task; more specifically, to understand how users respond to the way this interface allows for the drawing of conducting bodies, applying attributes to these bodies, generating a triangular mesh that correctly discretizes the model-space, and how they are able to run the forward-modeling software and view the output. Determining how end-users perform these tasks with this interface provides valuable information on improving the interface in the next iteration of design and development, as well as insight into how the scientist using this interface understands how models are built. There have been no previous usability tests conducted on this interface; therefore, there are no established concerns other than speculation made by the designer. However, Heuristic Evaluation performed before the empirical testing highlights usability concerns to be properly addressed later during the empirical testing.

3.3 Generalized Usability Evaluation of FEM2DGUI

3.3.1 Generalized Heuristic Evaluation Methodology

For the Heuristic Evaluation of FEM2DGUI, the evaluators are usability specialists and are expected to have little or no understanding of the domain because it is highly specific to a particular geophysical problem. However, the evaluators could have some varying degree of knowledge about the domain so a pre-test questionnaire is given to them in order to assess their knowledge of terminology and the domain in general, as well as some generic information about the evaluators themselves, such as their level of Usability Evaluation experience. The pre-test questionnaire is used for assessing the evaluators’ general knowledge of the domain. The
Heuristic Evaluation is broken up into four phases: pre-evaluation training session, actual evaluation, debriefing session, and a severity rating phase. For the pre-evaluation training session a short lecture is given to the evaluators on electromagnetic geophysics and the 2D forward modeling problem to provide context, followed by a presentation on the specific scenario in which they are engaging. The evaluators are not introduced to FEM2DGUI to ensure that they have an unbiased perspective during testing. Although the evaluators will certainly be familiar with Nielsen’s Heuristics (1994), there will also be a brief review of the principles in Table 3.1 to ensure that all of the evaluators understand the guidelines used in this study. During the evaluation the evaluators make two passes through the system focusing on the usability problems that arise. The first pass is an exploratory phase where evaluators are not given any tasks to complete. Instead, they interact with the interface at their own pace to learn the different aspects of the system. In the second pass through the system, evaluators are asked to work through three particular scenarios:

- Drawing a 2D model that represents a cross-section of the Earth and applying conductivity values to each of these layers and/or anomalous bodies
- Generating a 2D triangular mesh of the model
- Running the forward-model simulation and generating necessary output

Each of these tasks are given to the evaluators without informing them on how to complete the task so that they have a chance to navigate the menus and dialogs finding usability problems as they interact with the software. As the evaluators make each pass through the interface they are asked to find as many usability problems as possible, identify what principles they violate and to rate the problem as major or minor. Throughout the evaluations the FEM2DGUI system designer acts as an observer and notes the usability problems identified by the evaluators. This keeps the evaluators from having to write reports on the usability problems they find. After the evaluations have been completed, a short debriefing session is conducted to allow the evaluators and observer to discuss the results of the individual evaluations and speculate on potential improvements to the interface. This phase of the evaluation provides not only potential usability improvements, but insight into the types of problems that are likely to be uncovered by the end users. The severity rating of each of the usability problems helps to determine where to allocate the most resources in the refinement of the interface. The severity of a usability problem is a combination of occurrence frequency of problem, the impact of the problem, and the persistence
of the problem (Nielsen, 1994). These severity ratings are determined through questionnaires given to the evaluators after the inspections. The questionnaires list all of the usability problems found and these problems are rated on a 1 (no problem) to 5 (usability catastrophe) scale for severity. This type of questionnaire was employed by Nielsen (1994) with good results.

<table>
<thead>
<tr>
<th>Visibility of system status:</th>
<th>The system should always keep users informed about what is going on, through appropriate feedback within reasonable time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Match between system and the real world:</td>
<td>The system should speak the users’ language, with words, phrases, and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.</td>
</tr>
<tr>
<td>User control and freedom:</td>
<td>Users often choose system functions by mistake and will need a clearly marked “emergency exit” to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.</td>
</tr>
<tr>
<td>Consistency and standards:</td>
<td>Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow pattern conventions.</td>
</tr>
<tr>
<td>Error prevention:</td>
<td>Even better than good error messages is a careful design which prevents a problem from occurring in the first place.</td>
</tr>
<tr>
<td>Recognition rather than recall:</td>
<td>Make objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for user of the system should be visible or easily retrievable whenever appropriate</td>
</tr>
<tr>
<td>Flexibility and efficiency of use:</td>
<td>Accelerators—unseen by the novice user—may often speed up the interaction for the expert user to such an extent that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.</td>
</tr>
<tr>
<td>Aesthetic and minimalist design:</td>
<td>Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.</td>
</tr>
<tr>
<td>Help users recognize, diagnose, and recover from errors:</td>
<td>Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.</td>
</tr>
<tr>
<td>Help and documentation:</td>
<td>Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user’s task, list concrete steps to be carried out, and not be too large.</td>
</tr>
</tbody>
</table>

Table 3.1: Nielsen's Ten Usability Principles
3.3.2 Generalized Thinking Aloud Protocol Methodology

The processes for these tests are based on ideas presented by Dumas and Redish (1993). The Thinking Aloud Protocol is broken up into four phases: pre-test questionnaire, pre-evaluation training session, actual evaluation, and debriefing questionnaire. The end-users should be three to five experienced geophysicists that have intermediate to expert computer experience. A small group of non-novice users is appropriate for this study, because this is the first iteration of user testing and the designer has little experience in producing electromagnetic geophysics GUIs and believes that the opinions of better informed users provides better insight into advancements in future iterations. Before the testing, users are asked to fill out a pretest questionnaire to gather information concerning general user information, such as job title, length of time doing this work, computer experience, and whether or not they have prior experience with any other geophysical model builders. This pre-test questionnaire is used to assess what types of prior experiences participants have had with geophysical modeling software and determine how this prior knowledge may influence their responses and feedback during the Thinking Aloud Protocol. Before testing, users are given a brief presentation outlining the 2D magnetotellurics forward-modeling problem and a brief demonstration on how to use the FEM2DGUI interface. Although users are expected to have at least some knowledge of the domain, this presentation serves as a refresher. Also, this presentation introduces the users to Nielsen’s Usability Heuristics (Nielsen, 1994) so they have a general idea of the types of usability problems they should be looking for during the testing. Unlike the Heuristic Evaluation, where the evaluators are supposed to move through the interface with no prior knowledge of how it works, user testing produces the best results when users are told how the interface works first (Nielsen, 1993). This tutorial steps the user through a complete task. The user is shown how to draw a layer over a half-space with a small anomalous body. Attributes are then added to each region of the model. These attributes correspond to physically reasonable conductivities. A mesh is then generated followed by the running of the forward-model simulation. This demonstration gives the users the proper understanding of FEM2DGUI interface before they try to complete a scenario alone. Users are also informed on the two scenarios in which they are engaging. The outline for each scenario is the same as described above for Heuristic Evaluation. The first scenario is rather simple, in order to ease the user into using the software. The second scenario is
more complicated and contains multiple, dipping layers with anomalous bodies at different locations in the model.

As users work through the scenarios they are asked to “think-out-loud”, describing their experience and verbalizing their thoughts, as the observer records their reactions, opinions, and judgments. It is sometimes necessary to prompt the users with questions to probe a little deeper for information. For example, the observer may find it useful to ask the user to verbalize what they are thinking, or ask why they performed a certain keystroke or pushed a particular button. These questions need to be neutral to protect the integrity of the test: “What are you thinking right now?” is a neutral question, whereas “Are you thinking _____?” is not (Dumas & Redish, 1993). It has been shown that observers to these tests are not able to record every user response and they might miss crucial details (Nielsen, 1993); therefore, throughout the testing, users are video and audio recorded so that the tests can be reviewed at a later date. Also, because only one observer is used during this testing, videotaping is necessary for capturing review actions that occur too quickly for a single person to write down, as well as for timing crucial points during each user’s interaction with the interface, such as how long they spend recovering from errors or how long it takes each user to work through each scenario from start to finish (Dumas & Redish, 1993). The test is considered completed when the user has successfully completed the scenarios. After the testing has been completed, the users are given a debriefing questionnaire asking them to rate their satisfaction with different aspects of the interface, such as overall satisfaction, satisfaction with the menus and prompts, ease of recovery from errors, ease of drawing, ease of moving through the processing steps, and any further comments they would like to provide.
Chapter 4: Results

4.1 Chapter Overview

As discussed in 1.2.1, a graphical user interface, such as FEM2DGUI, is designed to aid users in accomplishing particular tasks through direct manipulation of graphics on the screen. How well an interface can be used to achieve a desired goal is referred to as usefulness, and usefulness is subdivided into utility and usability. Utility is the ability of an interface to function in a user-expected manner, and usability is how easy it is for users to obtain a high level of utility from an interface (Nielsen, 1993). Often usability and utility are closely entwined, which makes discerning a usability problem from a utility problem difficult, among all of the different problems uncovered during a Usability Evaluation (Grinstein et al., 2003). Heuristic Evaluation (a usability inspection method) and Thinking Aloud Protocol (a usability test method) are both utilized here and were conducted almost simultaneously in order to distinguish between usability and utility problems, as well as to discover as many usability problems as possible in a short time. Following the completion of these tests, the results from both tests were synthesized to determine which problems should be addressed for interface redesign (Figure 4.1 depicts the evaluation steps taken in this chapter). Finally, a formative evaluation of the redesigned interface involving the same end-users from the Thinking Aloud Protocol was utilized to check the validity and effectiveness of the methodology employed here to determine whether changes made to FEM2DGUI based on the results of the Usability Evaluation helped to improve usability. The results of this test are presented in 5.5.

The purpose of the Heuristic Evaluation is to uncover usability errors in FEM2DGUI with the help of a usability specialist. A usability specialist is able to recognize and diagnose usability problems as well as provide redesign recommendations for increasing overall usability. A Heuristic Evaluation utilizing the methodology outlined in Chapter 3 and Nielsen’s Usability Heuristics (Nielsen, 1994) as the guidelines for evaluation was conducted as a part of the
Usability Evaluation of FEM2DGUI. A brief review of the method, an analysis of the evaluator, and the results of the Heuristic Evaluation is presented in 4.2.

The Heuristic Evaluation and Thinking Aloud Protocol studies were conducted simultaneously (Figure 4.1). These tests were intended to gather feedback from potential end-users of the interface to assess usability problems as they are interpreted by the user. Many of the usability problems found by the usability specialist were also found by the users, but interpretations of these problems and redesign recommendations often differed from those of the usability specialist. Also, these participants were more familiar with the types of 2D modeling problems FEM2DGUI aids users in solving, and these participants were able to provide a usability analysis while working through a more task-driven environment. Results of the Thinking Aloud Protocol are presented in 4.3.

![Flowchart illustrating the sequence of evaluative events](image)

**Figure 4.1: Flowchart illustrating the sequence of evaluative events**

### 4.2 Heuristic Evaluation of Interface

#### 4.2.1 Method

As discussed in 3.3.1, the method for the Heuristic Evaluation involved using a guidelines-based inspection with usability specialists performing free exploration of
FEM2DGUI, and walking through a specifically designed scenario to examine usability with and without following specific tasks. The choice to use both free exploration and a scenario-based task walkthrough was made because each of these methods provides benefits during Heuristic Evaluation of an interface. With free exploration, usability specialists can focus on understanding the usability of the interface because they are not forced to concentrate on a specific task (Nielsen, 1993). The scenarios were designed to ensure that the specialist interacted with all of the dialog elements that occur within FEM2DGUI. While the usability specialist interacted with FEM2DGUI during both the free exploration and scenario-based task walkthrough, they were asked to make notes of usability problems. The observer wrote down the usability problems as the specialist pointed them out verbally. After the completion of the free exploration and scenario-based walkthrough, the observer and specialist discussed each of the problems. The discussion between the observer and specialist was to determine which of Nielsen’s Usability Heuristics each problem violated, recommendations of how the problems could be resolved, and to assign a severity rating to the usability problems. Severity ratings are used to determine which usability problems require the most redesign and development resources.

4.2.2 Evaluators and Results

The Heuristic Evaluation of FEM2DGUI was conducted at Virginia Tech and involved a collaboration of usability specialists and the interface designer. Although Nielsen (1994) recommends 3 – 5 usability specialists for a guidelines-based evaluation of software, due to scheduling conflicts and a strict testing timeline, there was only one usability specialist available to conduct the testing. The usability specialist used was a human factors expert and Ph.D. candidate at Virginia Tech. Using only one usability specialist instead of 3 - 5 has limitations. Single evaluators find on average only 35 percent of usability problems while performing Heuristic Evaluations (Nielsen, 1994). However, because this is the first usability test conducted on FEM2DGUI there are some advantages to having only one evaluator. Most of the glaring usability errors can be diagnosed by only one usability specialist; therefore, later tests have greater potential for the discovery of usability problems overshadowed by first-order problems. These first-order problems would likely be repeated by multiple usability specialists. The follow-
up evaluation found in 5.5 discusses how having only one usability specialist has affected the overall Usability Evaluation of FEM2DGUI.

Table 4.1 shows the results from the Heuristic Evaluation conducted on FEM2DGUI. This table contains a description of each of the problems discovered, along with which of Nielsen’s Usability Heuristics was violated, a redesign recommendation, and a severity rating. According to Nielsen (1994), the severity rating of a particular usability problem is a combination of: the frequency of problem occurrence, the impact of the problem when it occurs, and the persistence of the problem. Typically, evaluators will be asked to rate the severity of all of the usability problems found after all of the evaluators have completed the Heuristic Evaluation (Nielsen, 1994). Due to there being only one evaluator during this evaluation, severity ratings were determined by the usability specialist immediately after the evaluation during the debriefing session. Redesign recommendations were also determined during the debriefing session through collaboration between the usability specialist and interface designer.

Eight of the thirteen usability problems found during the Heuristic Evaluation were flagged for redesign and redevelopment because they were considered severe by the usability specialist or they were similarly discovered by end-users during the Thinking Aloud Protocol. Most of these problems are the result of ambiguities in how icons and dialog boxes are designed or difficulties with drawing on the screen. For example, the usability specialist found it unclear what steps to make and in what order to make them. This is a severe problem, because one goal of FEM2DGUI is to streamline the MT forward-model process. Three problems were resolved through more indirect methods while redesigning and refining the interface following both the Heuristic Evaluation and Thinking Aloud Protocol. One example of a problem being resolved after the resolution of another problem was the issue with a progress bar not being displayed during the deletion of a mesh. At the time of Heuristic Evaluation, the method in which FEM2DGUI deleted meshes was a long process; however, refinement of mesh generation has led to instantaneous mesh deletion. A more detailed description of which usability problems were flagged for redesign following Heuristic Evaluation and Thinking Aloud Protocol is found in 4.5 and a discussion on how interface refinement was conducted is found in 5.3.
<table>
<thead>
<tr>
<th>Usability Problem</th>
<th>Usability Heuristics(s) Violated</th>
<th>Redesign Recommendation</th>
<th>Severity of Problem</th>
</tr>
</thead>
</table>
| * It is unclear what steps to make and in what order to make them throughout use of the interface. | - Recognition rather than recall  
- Visibility of system status  
- Match between system and the real world  
- User control and freedom  
- Error prevention  
- Help and documentation | Allow the user to click “Generate Mesh” and “Run Forward Model” buttons only when the steps that are needed before these steps are completed. Have documentation explaining the steps needed to take throughout use of the interface. | 4                 |
| * Icons are confusing.                                                           | - Match between system and the real world  
- Consistency and standards  
- Error prevention | Redesign icons so their meaning is more apparent (i.e. change colors of the “Generate Mesh” and “Run Forward Model” buttons). Place the icons in the drop-down menus as well as the on the toolbar. | 4                 |
| * When the “Add Attribute” button is clicked it is unclear what the user is expected to do or where the attribute will be located. | - Recognition rather than recall  
- Visibility of system status  
- Error prevention | Right-click to add attributes, alert the user that the attribute button has been clicked, and/or Add Attribute dialog box should popup where the attribute will be placed. | 4                 |
| “Nodes per skin depth” is ambiguous.                                              | - Match between system and the real world  
- Recognition rather than recall  
- Help and documentation | No redesign recommendations given. | 2                 |
| + Relative location of “OK” and “Cancel” buttons are used in dialog boxes seem backwards. The user is more likely to click “OK” rather than “Cancel”. | - Consistency and standards | Switch the positions of the “OK” and “Cancel” buttons. | 2 |
| + When drawing dense meshes the user has to wait or cancel and start over. | - User control and freedom | Add a pause button so the user can leave the drawing and come back where it left off. | 2 |
| Drawing models is hard for left-handed people because the interface is designed for right-handed mouse input. | - User control and freedom | Implement a way for the drawing to be accomplished by a stylus. | 5 |
| + Progress bars are available for running Triangle and FEM2D but not when deleting the meshes. | - Consistency and standards | Add a progress bar for deleting the mesh lines. | 3 |
| * Drawing points is unintuitive and difficult, especially placing the points on the side of the model. | - Flexibility and efficiency of use | Line drawing should be a dragging motion as opposed to automatically following the mouse, text that hovers near the pointer to alert the user as to what is going on, and/or a magnifying glass to zoom in on where the pointer is for more accurate placement of points. | 4 |
| * It is difficult to make straight lines. | - Flexibility and efficiency of use | Add grid lines in the drawing screen so the user can easily tell the straightness of a line. | 4 |
**Table 4.1: Results from Heuristic Evaluation.** Usability problems discovered by the usability specialist are shown along with corresponding cells explaining each Nielsen’s Usability Heuristic the problem violates, redesign recommendations, and the severity of the problem determined through the discussion between the usability specialist and the designer of FEM2DGUI. Usability problems marked with an asterisk are problems that were fixed for the next iteration of FEM2DGUI. Problems marked with a plus sign were indirectly corrected during redesign and redevelopment.

### 4.3 Thinking Aloud Protocol

#### 4.3.1 Participants

Four participants were invited for the Thinking Aloud Protocol, and three of these participated. All of the participants were graduate-level geophysicists; two PhD students and one MS student. All users had at least eight years of general computer experience. Two of the participants currently study electromagnetic geophysics and one studies seismology. All of the participants have previous experience with geophysical model-building software. None of the participants had previously interacted with FEM2DGUI.
4.3.2 Equipment

All of the Thinking Aloud Protocol tests were conducted in a classroom at Virginia Tech. The users were asked to interact with FEM2DGUI using a mouse and keyboard for input, and the monitor for output. Participants were also video and audio recorded to allow for the review of the tests in greater detail than what could be recorded through note-taking by the observer as the tests were taking place. The video camera was focused on the computer monitor to watch how the users moved around the interface with the mouse. This particular video camera has a timestamp accurate to the second, so a stopwatch was not needed for the collection of quantitative data. The audio was recorded with the same device as the video. The test participant and observer are clearly heard through the audio track of the recording device. The audio portion of these tapes proved useful because some of the most useful data collected during Thinking Aloud Protocol comes from the users talking about their interaction with FEM2DGUI throughout the testing process. Transcription of the video after the tests was facilitated due to the audio and video being in sync.

4.3.3 User Tasks

As described in 3.3.2, after users were given a brief introduction to magnetotellurics, the forward-modeling problem, and the software, they were given a short demonstration of FEM2DGUI by the observer (who also acted as the test administrator). This demonstration was employed to show users how to operate FEM2DGUI, such as location of menu items and their behavior when clicked. Next, the users were asked to take some time for free exploration of FEM2DGUI, as was done during the Heuristic Evaluation. This exercise allowed the users to become more familiar with FEM2DGUI after the demonstration, as well as to mention any immediately apparent usability problems. Two different geological models were given to the users by the test observer for the users to draw and analyze. The first of these was a simple model of a dipping layer under the Air-Earth Interface (AEI). The second geologic model contained more complicated geology involving multiple dipping layers and anomalous bodies.
4.3.4 Pilot Study and Changes to Evaluation Method

An informal pilot study was conducted as practice to prepare for potential, unforeseen complications that could arise during the tests. The test administrator and test observer was the same person, so care had to be taken when determining a location for the video camera, an appropriate speaking volume for the participants, and an appropriate pace for the testing to ensure that all of the tasks could be conducted in a timely and efficient manner. The pilot study also helped to determine specifics, such as the level of geologic model complexity for user drawing and the tasks the users would be asked to complete while participating in the tests.

The usability test described in 3.3.2 was conducted in Derring Hall at Virginia Tech. As described above, a pilot test was conducted to ensure that the evaluations were performed with as few difficulties as possible. The pilot study provided some useful information into how these tests were to be conducted, such as the optimum relative position of observer, study participant, and video recording device was determined. The position of the observer changed from sitting next to the participant to sitting in a row of desks behind the participant. There are three reasons for this modification:

1. The video camera was placed in a location over the left shoulder of the participant forcing the participant to sit a little to the right of the monitor. Sitting next to the participant was only comfortable for the participant and observer if the observer sat to the left of the participant. This placement often led to the observer blocking the view of the monitor for the video camera.

2. With the observer sitting next to the participant the conversation between them often became mumbled and inaudible. This seemed to occur because the observer and participant began to talk in a volume appropriate for two people sitting next to one another. After some time, neither observer nor participant spoke in a volume that could be recorded clearly by the video camera.

3. The participant in the pilot study was very dependent on the observer to explain in what order and how to accomplish tasks. However, the intent of this study is for the users to explain their thought process as they work through the interface. Although users should
be able to ask the observer questions when stuck or confused, they should not rely on the observer to answer every question (Nielsen, 1993).

4.3.5 Results

Although there has been work in developing software interfaces for potential field methods including Geisler & Krieger, 2009; Pidlisecky et al., 2007; Schmidt et al., 2007, there is no published usability test data pertaining to these, or other, potential-field software packages. During the tests, quantitative data such as task completion times, error counts, error recovery times, and questionnaire scores were collected. However, because this is the first iteration Usability Evaluation of FEM2DGUI, and there is a lack of usability test results from other similar projects, no specific timing targets were set. Thus, the quantitative data were primarily used to support the qualitative results for the goal of making the interface more usable and to facilitate model building. Qualitative results were recorded through note taking while the user interacted with the software, transcribing the video and audio recordings, and through questionnaires. Quantitative results were gathered from the video recordings of the tests. Quantitative and qualitative results are found in Appendix E: Think Aloud Protocol and Questionnaire Results.

4.4 Strengths

Users found that one of the biggest strengths of this interface was the overall simplicity of model building while using this software. Two of the users commented on how the existing system of text file parameterization of the input for Triangle and FEM2D is much more complicated and involved than drawing models with a mouse. Users also felt that the instructions given to them through prompts were easy to understand and use. The interface’s desktop metaphor appeared to be understood by all of the users, as none of them had any trouble with the menus, toolbars, and actions contained within the menus. Also, this interface has input methods for drawing similar to other drawing programs, so what the users’ expected to experience while
drawing was not drastically different from established mental models. Finally, users seemed to immediately understand the representation of a 2D cross-section of the Earth on the monitor.

4.5 Usability Problems

The quantitative and qualitative data revealed 54 usability problems contained in the interface (0). Seven of these problems were repeated by all three users, eight of them repeated by two users, and 39 by a single user (Table 4.3). Most of the 39 problems experienced by one user were opinions or judgments concerning future features they would like to see in the interface and are generally considered low priority issues. A number of the usability problems found during the Thinking Aloud Protocol were the same or similar to those discovered during the Heuristic Evaluation. Table 4.2 illustrates the overlap between problems found by the usability specialist and those found by the test participants. The usability problems found by multiple users that were similar to problems found by the usability specialist and given a high rating, and those believed to most directly affect model quality were given a higher importance concerning interface redesign. Presumably, if a particular problem is found by multiple test participants, that problem has a high frequency of occurrence. Furthermore, the problems found by multiple test participants tended to be problems that most greatly impacted their interaction with FEM2DGUI. Usability problems deemed important for redesign and redevelopment were those that were found repeatedly, those rated severe by the usability specialist, and those that would directly or indirectly aid in resolving other usability problems found during the evaluation and testing. The only usability problem that was given a high severity rating and not addressed was the problem concerning the difficulty of drawing for left-handed people. This problem could potentially be solved by positioning the mouse to the left of the keyboard.

Many of these usability problems were combined to fix multiple issues during interface redesign. The nine problems believed to most affect the use of the interface were assigned numbers (although these numbers do not correspond to relative importance) and are listed below along with potential redesign methods. The listed problems also include the usability problems discovered during the Heuristic Evaluation and include redesign recommendations based on the
discussion with the usability specialist. Usability issues one through nine as well as the other refinements listed in 4.5.10 were redesigned and developed before another round of testing.

<table>
<thead>
<tr>
<th>Problems found by specialist and at least one end-user</th>
<th>Problems found only by specialist</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. It is unclear what steps to make and in what order to make them throughout the use of the interface.</td>
<td>1. Locations of “OK” and “Cancel” buttons in dialog boxes are backwards.</td>
</tr>
<tr>
<td>2. Icons are confusing.</td>
<td>2. Drawing models is hard for left-handed people because interface is designed for right-handed input.</td>
</tr>
<tr>
<td>3. When the “Add Attribute” button is clicked it is unclear what the user is expected to do or where the attribute will be located.</td>
<td>3. Progress bars are available for running Triangle and FEM2D but not when deleting meshes.</td>
</tr>
<tr>
<td>4. “Nodes per skin depth” is ambiguous.</td>
<td>4. The output graph can be viewed but not saved.</td>
</tr>
<tr>
<td>5. When drawing large meshes the user has to wait or cancel to start over (B2 experienced crash and had to restart FEM2DGUI).</td>
<td></td>
</tr>
<tr>
<td>6. Drawing points is unintuitive and difficult, especially placing the points on the side of the model.</td>
<td></td>
</tr>
<tr>
<td>7. It is difficult to make straight lines.</td>
<td></td>
</tr>
<tr>
<td>8. “Add Receiver” dialog box is unintuitive. It is difficult to tell when receivers are added to the screen.</td>
<td></td>
</tr>
<tr>
<td>9. It is difficult to quickly add an air-earth interface.</td>
<td></td>
</tr>
</tbody>
</table>

Table 4.2: List of problems found by usability specialist and end-user participants, as well as list of problems found only by the specialist.
### Table 4.3: List of problems found by end-user participants, categorized by how many users found the problems. For the problems only found by one end-user, and only the problems most affecting usability have been listed.

<table>
<thead>
<tr>
<th>Problems found by all three end-users</th>
<th>Problems found by two end-users</th>
<th>Most important problems found by one end-user</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. It is unclear what steps to make and in what order to make them throughout the use of the interface.</td>
<td>1. It is difficult to quickly add air-earth interface. 2. Desire for a screen that displays conductivity/resistivity ranges for common rock types. 3. Unclear of the meaning of “nodes per skin depth”. 4. Insert coordinate dialog is ambiguous. 5. Difficulty in deleting points. 6. Desire for a way to stop drawing a line when it has started. 7. Ability to specify the angle of dipping layers. 8. Ability to define an air region without its low resistivity value being a part of the color scale in the mesh.</td>
<td>1. Entering attributes should be able to accommodate conductivity values as well as resistivity values. 2. Hard to create exactly horizontal lines. 3. Unable to add points directly to lines. 4. Unable for the ability to change color scales. 5. Unable to insert predefined shapes, such as circles or rectangles.</td>
</tr>
<tr>
<td>2. Drawing points is unintuitive and difficult, especially on the side of the model.</td>
<td>4. Icons are confusing. 5. Attribute location is ambiguous. 6. Progress bar that tells user when Triangle is running often does not disappear, even after completion. 7. Scale bar spacing and units cannot be controlled by the user.</td>
<td></td>
</tr>
</tbody>
</table>

| 3. “Add Receiver” dialog box is unintuitive and difficulty setting receiver positions. | 3. “Add Receiver” dialog box is unintuitive and difficulty setting receiver positions. | |
| 4. Icons are confusing. | 5. Attribute location is ambiguous. 6. Progress bar that tells user when Triangle is running often does not disappear, even after completion. 7. Scale bar spacing and units cannot be controlled by the user. | |
| 5. Attribute location is ambiguous. | 6. Progress bar that tells user when Triangle is running often does not disappear, even after completion. 7. Scale bar spacing and units cannot be controlled by the user. | |
| 6. Progress bar that tells user when Triangle is running often does not disappear, even after completion. 7. Scale bar spacing and units cannot be controlled by the user. | | |
| 7. Scale bar spacing and units cannot be controlled by the user. | | |

#### 4.5.1 Issue #1: Unintuitive and ambiguous icons in the toolbar

![Toolbar with difficult to understand icons. The circled icons are those that gave evaluator and participants the most difficulty.](image)

**Figure 4.2:** Toolbar with difficult to understand icons. The circled icons are those that gave evaluator and participants the most difficulty.

**Description**

All of the users during the usability testing showed confusion or frustration over the “look-and-feel” of the icons located in the toolbar. Certain icons, such as the “Insert Geology”
and “Delete” icons did not give users any trouble. The “Insert Attribute”, “Generate Mesh”, “Run Forward Model”, and “Clear Screen” icons all created confusion for the users (Figure 4.2). The “Insert Attribute” and “Clear Screen” icons were problematic because users said they did not look like what they are supposed to do. To deal with this problem, users either found the action they needed in the menus or they clicked on the icon in the toolbar and tried it to see what would happen. Further, the “Run Forward Model” icon was grey in color and users assumed this meant it was disabled. In this case, users avoided the icon button and found what they needed in the menus. The “Generate Mesh” icon suffered from both not looking like its task and being dimly colored.

Refinement Strategy

The most obvious refinement strategy is to talk with potential end-users of this software about how they think the icons should be designed and make the icons to their specifications. Changing the color of the “Generate Mesh” and “Run Forward Model” icons would keep the users from assuming they are disabled. Redevelopment of the icons could be augmented with some redesign recommendations given by the usability specialist. For example, adding the icons to the menu lists would associate those icons with the words in the list. Another redesign recommendation from the usability specialist was to disable and enable the icons depending on where the user is in relation to the task, such as enabling the “Run Forward Model” button only when the user has accomplished all of the necessary tasks for complete FEM2D input generation. Further, the addition of icons for navigating to and from the mesh viewing screen was suggested in user testing.

4.5.2 Issue #2: The user’s progression through tasks from start to finish is ambiguous

Description

Throughout the use of the interface, users became confused or stuck and were unclear about how to proceed during the completion of a task. Often, this resulted in the user asking the observer where they were to proceed or the action of randomly clicking buttons and icons until they came across what they expected and/or needed. There was general confusion about when to
do each of the specific tasks before mesh generation, such as adding attributes. Order of operations errors led to instability problems with FEM2DGUI, especially concerning mesh generation. When the mesh was generated, users found it difficult to view because there was no prompt from the interface indicating to the user whether or not the mesh had been properly created and where to proceed to continue with the task. Most often, after users showed prolonged confusion, they were told where to go by the observer. Not only were the users confused about the overall flow of a task, they also had trouble navigating back to the draw screen from the view screen so that it could be cleared to start on a new task.

Refinement Strategy

There are only a few, although crucial, steps in the flow from building a model from scratch to generating the desired forward model output; therefore, some simple methods of workflow control could ensure the user is following a particular process. As mentioned earlier when discussing how icons can be improved, enabling or disabling icons based on the users’ progression through the workflow would help keep the user from getting overwhelmed with choices and keep the flow streamlined. Another redesign strategy is to employ documentation explaining to the user in step-by-step instructions how to accomplish the task from start to finish. The use of dialog boxes directing the user at particularly ambiguous points could also help. For example, after a mesh is generated, a dialog could appear explaining whether or not the mesh has been successfully created and that the user needs to navigate from the draw screen to the mesh viewing screen to see this mesh.
4.5.3 Issue #3: Placing points on the edges was difficult for users

Figure 4.3: Users must be very precise to correctly place points on drawing edge.

Description

Users repeatedly had problems with placing points on the edges of the model space. The points are placed by the user when the point currently controlled by the mouse comes close to an edge. It turns red indicating that the point is close enough to “snap” to the edge. It is integral that the point be placed at exact horizontal locations (left-most and right-most positions of the model space) for Triangle to correctly compute a mesh. The “snap” feature was introduced to enforce this horizontal placement. However, there is little room for error, and if the user does not click correctly while close enough to the edge the point will not “snap”. If the point is not correctly placed onto the edge of the model view control of the drawing tool will either not be released or the point will be placed in an incorrect location. Point placement along the right edge of the scene was especially problematic due to the dimensions of the scene and low error tolerance.

Refinement Strategy

Multiple strategies for refinement were suggested by the users and the usability specialist. A seemingly simple, yet effective method for correcting this problem would be to increase the “snap” tolerance at the edge. A foreseeable problem with this solution would be that a user would be unable to place a point very close to the edge without actually adding it to the edge. Other suggestions for solving the problem include:
1. Holding a key such as Shift or Ctrl while clicking to make the current point “snap” to the edge at the mouse’s current vertical value.

2. Adding a “magnifying glass” feature that would zoom in on the current point to allow for more precise point placement by the user. This could also allow for people with visual impairments to more effectively use the software.

3. Adding a text box that would follow the mouse on the screen that could alert the user to what exactly will happen if they place a point at that particular location.

4. Include a double-click feature that would place a point at the location under the mouse without “snapping”. A single-click of the mouse would “snap” the point to the edge assuming it is within the tolerance of the edge.

The magnifying glass and text features are too involved and would require more development than possible under current time and resource constraints. However, this issue could be resolved using a combination of increasing the tolerance of when a point would “snap” to an edge, holding a key that would tell the next point to be placed on the edge at the current vertical location of the mouse, and implementing a double-click feature that would release the mouse from drawing and prevent the point from being added to an edge.
4.5.4 Issue #4: Placing attributes is unintuitive and their location can be ambiguous

Description

When the users click on the “Add Attribute” button, a conductivity value is added at the next location they click in the model space. Problems with this method of placing attributes stem from the lack of indication for users whether or not this button has done anything. When the attribute is added to the screen, a dialog box appears in the middle of the screen asking for the conductivity value of the attribute. The attribute is represented by a draggable box containing the conductivity value and units (screenshots in Chapter 2). The draggable box that holds that text is anchored by a point at the top-left of the box. This method of attribute representation works well for large regions but proves to be confusing when applied to regions smaller than the attribute box. In these cases, users tended to drag the attributes over regions so that the middle of the attribute was in the region rather than the top-left of the box.

Refinement Strategy

Redesigning attribute insertion can be accomplished by changing the mouse cursor to something other than the normal mouse arrow. This would indicate to users that an attribute is ready to be added to the screen. When the attribute is added the mouse will change back and the dialog box that asks for a conductivity value from the user will pop up near where the attribute will be added instead of the middle of the screen. This will be helpful because it will reiterate to the user that the attribute is being placed where they clicked. To ensure correct placement of attributes within a small region the anchor point of the attribute can be changed from the top-left of the box to the middle point of the box. Based on how users interacted with the attribute boxes, placement of the attribute box anchor point would be more natural and usable.

4.5.5 Issue #5: “Triangle is Running” dialog box does not work correctly

Description

When users click the “Generate Mesh” button Triangle is called to create the mesh for the finite elements method. A mesh containing many elements can take some time to compute; thus,
a progress bar appears to indicate to the user that Triangle is actively running. However, if Triangle runs very quickly, the progress bar will not correctly disappear and will stay on the screen. This behavior was also noticed with a similar progress bar that appears while FEM2D is running.

Refinement Strategy

It is believed this happens because the progress bar is being called in the code after Triangle has already finished. The remedy for this issue is to check to see if Triangle is running before showing the progress bar.

4.5.6 Issue #6: Scale bar spacing and units are not controlled by the user

Figure 4.4: Users are unable to change the default spacing and units for scale bars.

Description

There are three scale bars in this interface: a bar on the left for conductivity values, one on the bottom for horizontal position, and one on the right for depth. All of the values for these scale bars are automatically calculated based on the maximum and minimum values. All of the users during the testing expressed a desire for control over how the scale bar values are spaced and in what units they are displayed. The scale bars showing depth and position only show these
values in meters and are always set to a particular spacing leading to potentially unattractive scale bars.

Refinement Strategy

Create a menu item that will give user control over the spacing and units used in the scale bars. The automatic scale bar generation could also be more intuitive by choosing a more appealing spacing than what is created in this iteration of the interface.

4.5.7 Issue #7: Receiver dialog box is unintuitive

![Figure 4.5: When the "Add" button is clicked new receiver elements appear at the bottom of the list without the scroll bar adjusting making it difficult to recognize the new receiver elements.](image)

Description

Often, many receivers will be applied to a given model; therefore, the “Add Receiver” dialog box was designed so that multiple receivers can be added quickly and efficiently without having to insert receivers one at a time. However, users found flaws with how this dialog box works. All of the end-users and the usability specialist had difficulty, because when the dialog box appears, it is big enough only for one receiver. When receivers are added to the box with the “Add” button, a scroll bar appears but the box does not grow, leading the users to believe that the button did nothing. The dialog box must be manually resized to accommodate the added receivers. Further, some users were confused with how the receivers should be placed. When the
“OK” button is clicked in the dialog box all of the receivers appear on the screen; however, some users thought this was unintuitive and clicked “OK” after adding only one receiver at a time.

Refinement Strategy

To make the “Add Receiver” dialog box more intuitive, the box will need to appear on the screen in a larger size to accommodate more receivers when they are added, the scroll bar should automatically scroll down to the lowest receiver, and an “Apply” button should be added. The “Apply” button would allow the user to add the receivers to the screen without closing the dialog box. The functionality of the “OK” and “Cancel” buttons should stay the same.

4.5.8 Issue #8: Insertion of AEI and exactly horizontal lines is difficult

Description

All of the users had difficulty adding lines that were exactly horizontal. This is problematic because there are often multiple horizontal lines in a model and the AEI is almost always horizontal and there almost always needs to be a defined AEI in a model.

Refinement Strategy

Two strategies have been suggested that would greatly improve point placement accuracy to aid in the insertion of exactly horizontal lines:

Figure 4.6: Solid white background and no reference lines make it difficult for users to add horizontal lines.
1. Allow for the user to insert points to the model by typing horizontal and vertical coordinates. This could also prove useful for non-horizontal lines where point placement needs a higher degree of accuracy than the mouse can provide.
2. Insert a grid in the background that would act as reference for the users.

Quick insertion of the AEI could be accomplished by a menu button that would ask the depth to place the AEI.

4.5.9 Issue #9: Users cannot add points to lines

Description

Points must be attached to other points when connecting layers to one another. This adds an unnecessary burden on the user because they must think ahead when drawing layers. If a layer is drawn with only two end points, no more lines can be attached to that layer without the user being forced to delete the original layer and redraw it with multiple points along the line. All of the users of the usability testing ran into this problem repeatedly and therefore unnecessary time was added to their total model completion time.

Refinement Strategy

Redesign the drawing tool so that users can click on a line with a point and that point will be added to the line. Point addition would “snap” in a similar fashion to how the points “snap” to edges and other points, and be able to be dragged in the same fashion as the other points.

4.5.10 Other Usability Problems

Other refinements that would increase the usability of this interface based on user feedback include:

1. Insert a button that when clicked shows a table with common rock types and their conductivity/resistivity values.
2. Allow resistivity values as well as conductivity values to be specified when inserting attributes.

3. Provide documentation on the meaning of “nodes per skin depth”

4. Allow for the air region to be removed from the color scheme of the conductivity values to keep it from altering the colors of all the other regions. Apply to it a color of light grey.

5. Add a button on the output screen that allows the user to save the output graph as an image file.

6. Reverse the location of the “OK” and “Cancel” buttons in the dialog boxes. The right-most button is the most likely to be clicked by the user, and putting “Cancel” on the right would prevent unwanted insertions or deletions.

There were also some usability problems and suggestions provided by the users and usability specialist that were not under consideration for redesign due to time constraints or resources:

1. Addition of a pause button while waiting for meshes to be drawn.

2. Changing the interface from mouse input to stylus input.
Chapter 5: Discussions and Conclusions

5.1 Design and Development

One of the goals of this project was to design and develop a Graphical User Interface (GUI). To accomplish this task the C++ programming language was used along with the Qt widget toolkit (widgets are elements of GUIs with which a user interacts, such as buttons and text boxes). Qt has proven to be an appropriate toolkit for this work because it provides the programmer with an array of widgets for quickly developing and implementing GUIs. Widget locations are pre-defined in Qt and automatically adhere to the styles and standards of the particular operating system under which a GUI has been compiled and is being run, adhering to Nielsen’s Usability Heuristic of consistency and standards. Once programming with the Qt widget toolkit was learned, most widgets were straightforward to implement and use.

The initial design was loosely based off of existing model-building interfaces and the geophysical experience of the designer. The designer of FEM2DGUI had previous experience with geophysical modeling tools and MT modeling. Therefore, certain design considerations were made based on that previous experience. The models constructed by the users of this software consist of drawing points connected by lines in a two-dimensional model space representing a cross-section of the Earth. This is a popular method for geophysical modeling tools to represent the subsurface. These points and the lines connecting the points make up the vertices and segments of planar straight graphs for Triangle and, more abstractly, these points and segments represent geological layers and anomalous bodies in the sub-surface for the users drawing them. During the development of FEM2DGUI discussions with potential end-users were had with the inclusion of each new feature ensuring each feature had the ability to accomplish a particular task and met the set of established goals outlined in 2.2.

Because FEM2DGUI is intended to be a fully functional interface, more time was spent in the development phases of the project as opposed to the design phase. However, it is believed that the design and subsequent development could have been facilitated and streamlined with the use of paper prototypes. Paper prototypes, or “low-fidelity prototyping”, have been shown to
provide designers with test data that can be implemented during the development phases (Nielsen, 1993; Donahue, 2001). It is unknown how much paper prototyping would have accomplished during the design phases because many of the usability problems found during testing dealt with how well users were able to draw points on the screen and how they interacted with dialog boxes, problems that need users to interact with the interface to diagnose as opposed to speculating on potential problems. However, some of the point-drawing usability problems, such as double-clicking a point to provide a dialog box for defining detailed horizontal and vertical values, might have been discussed before evaluation. This is unknown though because paper prototypes were not used during the design of FEM2DGUI. Certainly, paper prototyping would have provided a more firm basis for particular aspects of the interface, such as the look-and-feel of the icons used and how the menus were organized, and provided designers and developers a stronger basis for what end-users desire and expect from a geologic model-building interface, such as FEM2DGUI.

5.2 Conclusions from Usability Evaluation

There were expected to be many usability problems found during both the Heuristic Evaluation and the Thinking Aloud Protocol, but the types of problems uncovered were often surprising. For example, users thought the “Generate Mesh” icon to be disabled because the lines used to draw the icon’s picture were thin which gave the icon the appearance of being grayed out, indicating a disabled icon. Some usability problems noted by the developer were never suggested by the users, such as the lack of save/load and undo/redo features. The usability specialist was expected to be unfamiliar with the problem domain during the Heuristic Evaluation; whereas, the end-user participants were selected for the Thinking Aloud Protocol because they are experts in geophysics and in general computing. However, the Thinking Aloud Protocol participants often encountered problems when attempting to create models, namely drawing models onto the screen. Only after users became well acquainted with FEM2DGUI did they show confidence in what they were attempting to draw and how they were adding the different objects to the screen. This lack of confidence from users is mostly attributed to a lack of usability concerning particular aspects of the interface.
Often, the usability of FEM2DGUI had a direct impact on the utility. One example of this was when users were trying to add receivers to the model through the “Add Receiver” dialog box. Multiple users, even after using the dialog box more than one time, tried to add receivers in an incorrect manner leading to receivers being misplaced or nonexistent. Receivers not located where users expected led to erroneous output; thus, the utility of the interface has broken down due to a failure in usability concerning the “Add Receiver” dialog box. Finding that usability affects utility in this way confirms earlier assumptions made about the relationship between usability and utility.

One deviation from the Usability Evaluation methodology was made during testing. Due to time constraints and scheduling conflicts, the Heuristic Evaluation was conducted after the Thinking Aloud Protocol. Ideally, the Heuristic Evaluation would have provided the experimenter with a list of usability problems that could have been more rigorously tested by the end-user participants during the Thinking Aloud Protocol. However, because the end-user participants had already completed testing and their results had been analyzed, some of their suggestions were discussed with the usability specialist during the redesign recommendation phase of the Heuristic Evaluation. This proved useful because there was only one usability specialist that took part in the Heuristic Evaluation and insights from the end-user participants of the Thinking Aloud Protocol supplemented the debriefing stage that would normally be conducted by multiple usability specialists.

The Usability Evaluation conducted on FEM2DGUI employed a usability specialist and end-users during testing. As seen in Table 4.2 and Table 4.3 many of the usability problems discovered by the usability specialist overlap with those found by the end-users. This calls into question the necessity to conduct both Heuristic Evaluation and Thinking Aloud Protocol. While most of the problems found during Heuristic Evaluation did overlap with some found during Thinking Aloud Protocol, there were some important problems that may have only been found by a person with a usability background. For example, only the usability specialist noticed that the output graph could be viewed but not saved, and that there was no progress bar for deleting meshes. The participants of the Thinking Aloud Protocol also found problems that were unlikely to have been found by a usability specialist, such as entering parameter values as resistivity as well as conductivity. FEM2DGUI was testing during an early prototype stage of development, so it is not surprising that the most egregious of usability errors were noticed by both groups of
testing participants. Both groups of participants noticing many of the same problems indicated that these problems were the most pressing issues to correct. With only one usability specialist available, conducting Heuristic Evaluation and Thinking Aloud Protocol concurrently worked well, especially when it came to ranking the aspects of FEM2DGUI that needed redesign and redevelopment. However, if the recommended 3-5 specialists were employed for Heuristic Evaluation, it would have been preferable to conduct Heuristic Evaluation, followed by interface refinement, then Thinking Aloud Protocol. If this was the case then end-user participants would have been less likely to spend time struggling with problems such as confusing icons and more time with working through model drawing and mesh generation.

FEM2DGUI has the potential to be a powerful tool for creating two-dimensional models and running forward-model simulations for magnetotelluric methods, but the Usability Evaluations conducted show there are still usability and utility concerns to resolve before this tool could be expected to become widely adopted amongst the potential field community. The redesign and redevelopment of the usability problems discussed in 4.5 brings FEM2DGUI closer to a highly productive GUI. In the next section, the refinements made to the interface based on the refinement suggestions described in 4.5. The numbers of the usability problems are correspond to the numbers in 4.5 to avoid confusion. After these refinements were made, another informal, end-user test was conducted on FEM2DGUI to determine whether or not the redesign made the interface more useful. This evaluation is discussed in section 5.5.

5.3 Refinements Completed

5.3.1 Issue #1: Unintuitive and ambiguous icons in the toolbar

Figure 5.1(a) shows the icon toolbar before refinement. Users found the icons contained within this toolbar to be unintuitive and ambiguous.
Figure 5.1: (a) Tool bar before refinement. All icons are enabled. (b) Toolbar after refinement during program startup. Some of the icons are disabled.

To remedy this usability problem several changes were made to the icons and toolbar (Figure 5.1 (b)) to make each more understandable to users:

- “Insert Geology” icon remained unchanged
- “Add Point to Line” icon has been added to toolbar since this feature was added to the interface. The icon is similar to the “Insert Geology” icon but it has a point inside of the line instead of one at each end making a line segment. Because no users expressed concerns over the “Insert Geology” icon, the “Add Point to Line” icon is believed to be straightforward and clear to users.
- “Add Attribute” icon has been updated from a simple “A” to an image that more appropriately represents what the button does.
- “Generate Mesh” icon was updated to look more like the type of triangular mesh users create, and the lines have been made darker as to not give the appearance of being disabled.
- The color of “Run Forward Model” icon was changed to green so it does not appear to be disabled.
- Buttons were added for switching back and forth from drawing to viewing screens.
- All of the icons were added to the left of each action in the drop-down menus to aid in user association of icons to actions.

These changes make this aspect of the interface adhere more closely to Nielsen’s Usability Heuristics. Now, the icons make a closer match between FEM2DGUI and the real world, they will prevent some of the errors found in user testing, and they are more aesthetically pleasing.
5.3.2 Issue #2: The user’s progression through tasks from start to finish is ambiguous

Users were found to have difficulty moving from one task to another during the MT forward-modeling process. The follow changes have been implemented to aid in user progression through tasks. When the interface is started, the icons in the toolbar, and their associated actions in the drop-down menus, are disabled (Figure 5.1 (b)) to prevent users from clicking on a process step at an inappropriate time. As a user progressed through a forward-model, specific buttons are enabled or disabled to help the user with making the correct steps. The disabling and enabling of actions will help with the visibility of system status and the prevention of errors. Documentation was added to the Help menu, to which users can refer, outlining the different steps a user should make while working through FEM2DGUI. Providing appropriate documentation is one of Nielsen’s Usability Heuristics. Dialog boxes were created to pop-up at key stages in the workflow to keep users informed on what has happened, where they are to proceed, and what is expected of them. An example of one of these dialog boxes is seen in Figure 5.2. These dialog boxes will help in preventing user error and in reducing what a user needs to remember between sessions. A future refinement of this concept would be to allow users to temporarily or permanently disable these boxes so they do not slow down users familiar to FEM2DGUI.

Figure 5.2: Example of a dialog box that helps users progress through tasks
5.3.3 Issue #3: Placing points on the edges was difficult for users

Three refinements were made to FEM2DGUI in an effort to reduce user frustration over the placement of points along the left and right-hand edges of the model space. The first of these refinements was to increase the collision tolerance of each one of the points. This means that the point does not need to be as close to the edge before it is ready to “snap” to it as users needed before. Second, the user can now hold down the Shift key while clicking in the drawing area and a horizontal line will be draw from the current mouse location to the nearest edge. Finally, a double-click feature was added so users can double-click to stop drawing and regain control of the mouse from drawing mode. These refinements should increase user control and freedom as well as help to prevent errors.

5.3.4 Issue #4: Placing attributes is unintuitive and their location can be ambiguous

Users had difficulty using the “Add Attribute” button and recognizing when FEM2DGUI was ready for their input. To ensure the user knows FEM2DGUI is ready to accept an attribute location and value from them, all icons have and menus are disabled and the mouse changes from the normal mouse cursor to a crosshair cursor. The dialog box that appears to accept user conductivity input shows at the attribute location the user picks instead of in the middle of the screen. This change gives users better recognition of where they put the attribute. These changes make the visibility of the system status more clear.

5.3.5 Issue #5: “Triangle is Running” dialog box does not work correctly

The progress bar that appears to show users that Triangle is running does not always disappear when Triangle has finished. This problem occurred because if the mesh being generated was simple, Triangle finish running before the progress bar was called in the interface code. This problem was solved by writing checks into the code to ensure Triangle is still running before the progress bar is called.
5.3.6 Issue #6: Scale bar spacing and units are not controlled by the user

Giving users control over scale bar spacing and units was one of the most discussed features during testing. FEM2DGUI was redeveloped to give users control over the scale bars by allowing users to double-click on a scale bar to manually change the spacing and units, meters or kilometers, of that particular scale bar. This refinement gives users more flexibility while using FEM2DGUI.

5.3.7 Issue #7: Receiver dialog box is unintuitive

Testing showed that users were often confused by how they were expected to interact with the “Add Receiver” dialog box. Confusion stemmed from the box being too small when it appeared on the screen and it not being immediately apparent if new receiver objects were added to the box. This usability problem was fixed by adding and “Apply” button to the dialog box that will add receivers to the model view without closing the dialog box, having the scroll bar for the dialog box automatically scroll down to the most newly added receiver object, and by increasing the default size of the dialog box itself. These changes improve the visibility of system status and should prevent errors.

5.3.8 Issue #8: Insertion of the AEI and exactly horizontal lines is difficult

To aid users with the insertion of horizontal lines, FEM2DGUI has been refined to allow for users to insert a horizontal line by specifying a certain depth value. A grid (Figure 5.3) can be turned on and off to add a reference frame for the user. These changes give users more flexibility and control over the system.
5.3.9 Issue #9: Users cannot add points to lines

Points can now be added to lines directly. This fix addresses users’ concerns over the drawing mechanic and having to think ahead when drawing lines. However, adding points to lines does not happen through use of the drawing tool, but rather, through its own action. Users can click the “Add Point to Line” button and they can then place that point anywhere on a line. The point then becomes a permanent fixture on that line and an anchor point for attaching other points and lines. This change gives users more freedom and flexibility and prevents them from having to add a completely new line if they forgot to draw a point originally.

5.3.10 Other Refinements

Other refinements were made based on user testing. The output graph can now be saved as an image file. Meshes are drawn more quickly and take up less computational resources. Resistivity values, as well as conductivity values, can be entered into the “Add Attribute” dialog.
box. The user can now save and load models. The save files are binary files with an “.f2d” extension. The File Menu contains a list of the most recently saved files to make it easy for users to quickly load a previously saved file. One change that was mentioned to be changed in 4.5 was the reversal of the “OK” and “Cancel” buttons in each of the dialog boxes. This change was not implemented. Upon further inspection of how buttons should be placed in these boxes, it was discovered that each different type of style guide prefers a different way. Qt automatically places these buttons in the correct locations based on the system which the interface is currently running. This conforms to Nielsen’s (1994) Consistency and Standards principle in that an interface should follow platform conventions.

5.4 Walkthrough

The walkthrough of FEM2DGUI presented in this section follows the same process presented in Section 2.3 with different model parameters to show how this interface changed as a result of the usability testing undergone on FEM2DGUI. The application still opens in the same manner as it did previously (Figure 5.4). However, the dialog box asking for model coordinates has been simplified, a background grid has been added to the drawing space, the buttons in the toolbar at the top of the application have been redesigned, and now users can load previously saved models from the File menu. While drawing straight lines proved to be difficult before FEM2DGUI underwent testing Figure 5.5 shows straight lines marking zero depth and the borders of a rectangular resistive body. These straight lines are now much easier to create because users can specify a particular depth at which to insert a horizontal line, and because points in the model can be double-clicked so that users can adjust position (x) and depth (z) values. In the “Add Receiver” dialog box, the buttons have been reorganized, and an “Apply” button has been added (Figure 5.6). Adding conductivity attributes to the model (Figure 5.7) has been simplified and now users can enter an “Air Region” parameter instead of having to enter a distinct air region value. The “Nodes per skin depth” dialog box (Figure 5.8) now allows users to select which MT mode they would like to use. Figure 5.9 and Figure 5.10 show the look of a completed mesh as well as apparent resistivity and phase output plots.
Figure 5.4: Opening screen for FEM2DGUI

Figure 5.5: Simple resistive body in a half-space
Figure 5.6: Adding receivers to the model

Figure 5.7: Completed model with receiver and conductivity attributes
Figure 5.8: Setting the "nodes-per-skin-depth", frequency, MT mode

Figure 5.9: Completed mesh
5.5 Formative Evaluation of the Redesigned Interface

It was deemed necessary to conduct a formative evaluation of the redesigned interface after the usability problems listed in 4.5 had been redesigned and redeveloped. This evaluation involved going through the same procedure and participants as the Thinking Aloud Protocol except without the video camera, pre-test presentation, and pre-test questionnaire. Users were given a short demonstration of the new features and then asked to go through two scenario-based walkthroughs of different complexity while talking aloud what they are thinking and doing while using FEM2DGUI with an observer taking notes (this was the same as the Thinking Aloud Protocol). Participants were asked to fill out the same post-test questionnaire administered at the end of the Thinking Aloud Protocol.

An analysis of the questionnaire results between the Thinking Aloud Protocol and this evaluation displays some interesting results. Even after the demonstration from the test administrator explaining the new features found in FEM2DGUI, test participants still had difficulty learning them. In particular, participant B3 repeatedly mentioned the desire for more documentation and/or help files to aid in the learning of new features and concepts, as well as
older features and concepts not remembered in between the two tests. Throughout the rest of this section qualitative and quantitative results from the post-test questionnaires are compared and contrasted to determine if the usability refinements made at the conclusion of the Usability Evaluation improved the usefulness of FEM2DGUI.

The Thinking Aloud Protocol highlighted the considerable difficulty for all participants to move through the process steps in the workflow of FEM2DGUI. During this formative evaluation, participants found the interface to be improved, but the new features required a learning curve and more documentation is needed, despite the addition of help files explaining workflow progression and a rock properties table after the first stage of testing. Participants commented on how the symbols used for icons made more sense during this test and how the inclusion of certain new features (i.e. “add horizontal line” feature) eased the model creation process. Despite how the new features made it easier to accomplish certain tasks, participant B3 still desired more documentation on how these buttons worked even after this participant commented after the first test that FEM2DGUI was easy after repeated use. Adequate, available documentation complies with Nielsen’s (1994) *Help and Documentation* usability heuristic. Usability Evaluation of FEM2DGUI has shown that it is better to have more documentation and help files to aid users of all experience through the process steps of an interface. As far as the instructional prompts are concerned for the second test, two out of three of the participants had no problem understanding them, while participant B1 did not recognize they existed. However, there was most definitely a prompt added after the first round of Usability Evaluation, as described earlier in this chapter. Recovering from errors was not easy for participants during the Thinking Aloud Protocol. According to post-test questionnaires, this was much easier during the formative evaluation of the redesigned interface. FEM2DGUI errors from which participants would have needed to recover were not as prevalent during the second round of testing. When participants came across errors, they were easier to diagnose and recover from. Participants found during the first round of testing that drawing with FEM2DGUI could be improved to facilitate drawing shapes on the screen. The post-test questionnaire given to participants at the end of the formative evaluation revealed that changes made after the Heuristic Evaluation and Thinking Aloud Protocol were conducted eased drawing for the participants. Some of the features mentioned participants as good additions to FEM2DGUI for the facilitation of drawing shapes on the screen are the ability to change values or points so they can be placed in specific
locations, adding points to lines, and double-clicking to terminate line generation. As mentioned earlier, test participants encountered difficulty moving through the FEM2DGUI workflow during the Thinking Aloud Protocol. Less difficulty was encountered in this evaluation. Streamlining the workflow by enabling and disabling buttons as users move through the process steps, as well as adding a dialog box that can display the process steps in the Help Menu was seen to have had a positive effect on the decision making of the participants. Participant B3 had the least amount of trouble with the FEM2DGUI workflow. However, this participant also had the advantage of working with FEM2D and the text-based interface between the Thinking Aloud Protocol and the second-round test, so this participant was intimately familiar with the process steps of FEM2D and the MT forward-modeling problem. Mesh drawing saw an increase in usability between the Thinking Aloud Protocol and this evaluation for two out of three of the participants. All of the participants found the mesh generation to work; however, one participant needed some trial and error to produce the desired mesh. Also, participant B1 desired more control over the mesh generation and refinement instead of just defining a generic “nodes per skin depth”.

Quantitative results from the post-test questionnaire results (see Table 5.1) of the Thinking Aloud Protocol and this evaluation display a positive trend in participant answers concerning the usability of FEM2DGUI. Only participant B2 showed a negative trend from the first to second post-test questionnaire with answers decreasing by 0.57 over the questionnaire. B1 showed the highest average answer increase with 1.43 and the second highest was B3 with 0.43. The overall change in questionnaire answers over all test participants is positive with an average change of 1.29 across the entire questionnaire.
Table 5.1: Answers to post-test questionnaires from Thinking Aloud Protocol and the informal final test. Results from participants B1, B2, and B3 are shown as well as averages and changes from the first questionnaire to the second.

<table>
<thead>
<tr>
<th></th>
<th>B1</th>
<th></th>
<th>B2</th>
<th></th>
<th>B3</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>First:</td>
<td>Last:</td>
<td>Mean:</td>
<td>Change:</td>
<td>First:</td>
<td>Last:</td>
</tr>
<tr>
<td>1.</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2.</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3.</td>
<td>2</td>
<td>3</td>
<td>2.5</td>
<td>-1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4.</td>
<td>4</td>
<td>1</td>
<td>2.5</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>5.</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>6.</td>
<td>3</td>
<td>2</td>
<td>2.5</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>7.</td>
<td>3</td>
<td>2</td>
<td>2.5</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>8.</td>
<td>3</td>
<td>2</td>
<td>2.5</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

The short-answer questions asked of participants also yielded more positive results from the formative evaluation of the redesigned interface than from after the Thinking Aloud Protocol. All of the participants comment that FEM2DGUI has improved from the refinements made after Heuristic Evaluation and Thinking Aloud Protocol; however, B1 commented that FEM2DGUI would benefit from the ability to iterate through multiple frequencies instead just one per mesh, and B2 commented that FEM2DGUI is still not streamlined enough. The streamline issue could perhaps be fixed through adding or improving existing documentation as suggested by participant B3.

### 5.6 Implications of Research on Geophysics

Geophysical problems are often non-unique and computationally expensive. Interfaces acting as tools for scientists to use for different geophysical problems should facilitate users with building models of appropriate complexity. As discussed in Chapter 1, a focus of this research was to understand usability problems of a two-dimensional MT forward-modeling GUI, and to understand how these problems affect users as they progress through their workflow. Furthermore, can principles of Usability Engineering and User-Centered Design be applied to help design and develop a more effective GUI for the two-dimensional MT forward-modeling simulations?
To accomplish these goals the two-dimensional MT forward-modeling GUI, FEM2DGUI, was designed and developed incorporating principles of Usability Engineering and User-Centered Design. Next, two different Usability Evaluations, Heuristic Evaluation – employing a usability specialist as the test participant – and Thinking Aloud Protocol – employing potential end-users as test participants – were conducted on FEM2DGUI. Usability Evaluation helped to evolve FEM2DGUI into a more complete and usable system. Finally, another evaluation was conducted to test whether or not the Heuristic Evaluation and Thinking Aloud Protocol working in improving usability for end-users. The final evaluation showed that participants had less trouble drawing models, creating meshes, and generating output. Although usability problems of FEM2DGUI were reduced through both Heuristic Evaluation and Thinking Aloud Protocol, it is difficult to discern how much “better” the refined iteration of the interface is compared to the version that was tested. Results from subsequent usability testing would have to be compared to the results from this work to more accurately determine how FEM2DGUI is changing throughout the development cycle (Nielsen, 1996).

Although FEM2DGUI is not perfect – it still contains usability problems and is not as feature-rich as some users may desire – its design and development has been facilitated by Usability Engineering and User-Centered Design through improving usability, affecting utility and overall usefulness. Performing Usability Evaluation, similar to the methodology presented in Chapter 3 could benefit all geophysical GUIs in increasing usefulness, especially usability. As found with the analysis of the design, development, and evaluation of FEM2DGUI, certain steps could be taken on other geophysical GUIs not performed here, such as paper prototyping (Nielsen, 1993; Hall, 2001) and employing additional usability specialists during Heuristic Evaluation. Also, for a GUI similar to FEM2DGUI with a similar development cycle, designers may prefer to perform Heuristic Evaluation with 3-5 specialists followed by iterative design, then Thinking Aloud Protocol, as suggested by Nielsen (1994). The reason for this alteration in methodology from this project is for snuffing out glaring and obvious errors with Heuristic Evaluation before Thinking Aloud Protocol to maximize the amount of time end-users spend working with the interface pertaining to the scenarios, instead of wrestling with usability problems. This shows that usability problems really do affect utility.

Essentially, FEM2DGUI can be divided into three different modules: a model-builder, a triangular mesh generator, and a FEM2D output visualizer. Of these three modules, most of the
usability problems discovered during the Usability Evaluations originated from the model-builder. For example, the icon design for the model-building tools was a significant problem for the Thinking Aloud Protocol participants. Because there are few graphical user interfaces for 2D geophysical model-building, there seems to be no standardization found between icons or drawing tools amongst these geophysical model-building interfaces. Partly due to this lack of standardization, FEM2DGUI was designed and developed from scratch with no point of reference from which icons and drawing techniques were designed.

One possible way in which usability and utility could be improved for 2D geophysical model-building interfaces would be the development of a 2D widget toolkit for the drawing and parameterization of 2D models, such as the cross-sections drawn with FEM2DGUI. Widget toolkits for other uses in GUI design exist and were utilized in FEM2DGUI. Qt, for example, is a widget toolkit for creating the generic menus, drop-downs, and windows for GUI projects. A widget toolkit created for 2D geologic model-building could contain methods for inserting many different geologic structures, formations, and attributes. Designers and developers of a geophysical interface could then simply import the components of the toolkit as needed for their interface. This would allow for designers to focus on the unique aspects of an interface such as the mesh generation tools and simulation output. Furthermore, the use of a consistent widget toolkit would aid users in more effective model building, while also conforming to existing usability guidelines such as Nielsen’s (1994) Consistency and Standards usability heuristic.

The overarching question asked in Chapter 1 was: How does the structure of an environment or application affect how the user reasons through the 2D model-building process? Namely, if an analysis environment affects how a user reasons through this process, the outcome of the process could also be affected. This work shows how improving the usability of a GUI decreases the number of errors a user encounters while working through the 2D model-building process, and increases user satisfaction with the GUI. On the other hand, poor interface design leads to an increase in user mistakes, dissatisfaction, and may ultimately drive them from using the interface entirely (Galitz, 2007). During testing of the first iteration of the interface, users struggled with drawing models, especially complex shapes. This led users to become frustrated with the interface, often leading them to give up on attempting complex shapes or they could become impatient, causing an error. If this type of unrefined GUI was to be used throughout the EM community, it would most likely lead users to drawing simpler models or to abandon the
interface, returning to the more familiar text files and command line. In this case, simpler models could lead scientists to interpreting apparent resistivity and phase curves omitting particular geological details. With the second iteration of FEM2DGUI, there was a noticeable difference in the time it took for users to complete model-building. Two or three more complex models were produced with the second iteration in the same time it took for users to produce a simple model with the first iteration. This shows that the time it takes for a user to complete models with similar complexities decreases with improvements to usability. Higher levels of usability could then allow for users to be more comfortable with producing highly complex models quickly and efficiently. Presumably, the only time constraints with highly usable model-building interfaces would be the generation speed of meshes and compute speed of the forward-modeling algorithms.

While the ability to produce models efficiently is important, especially when comparing the forward-modeling output to the outputs of other models, this does not answer the question of how users’ inferences of model and output (apparent resistivity and phase) could be altered by the analysis environment used. Salom et al. (2009), as mentioned in Chapter 1, showed that by changing a seismic picking paradigm from a static 2D system to a dynamic 3D system, users were able to conceptualize the data more effectively, thereby producing better results. Similarly, I believe that if a model and its output can be created effectively, a user would be more inclined to attempt adding geologies they would otherwise omit. This, in turn, could lead to more accurate representations of the subsurface and its magnetotelluric response. However, the decision to create complex versus simple models is made by the user with interfaces such as FEM2DGUI. In the case of FEM2DGUI, users are presented with a blank drawing area, and how they decide to draw models is left completely up to them. While FEM2DGUI guides users through the process of the forward-modeling process, there are still areas where a more sophisticated interface could guide users in how they choose model parameters, such as layer conductivities and source frequencies. Also, FEM2DGUI allows for input from the mouse and keyboard. Now, there exist technologies that allow for input from a stylus or human fingers, which could presumably alter how a user would build or perceive models due to these more natural drawing methods. Furthermore, as computer power increases, one can envision a FEM2DGUI-like interface that displays input and output simultaneously, and changes in output could be viewed nearly instantly as users alter their models. Unfortunately, this work only probes into the usability of one
interface; however, if other interfaces were built utilizing different design paradigms, a study could be conducted as to how these different designs affected the output of the 2D MT forward-modeling problem.

5.7 Future Work

From this point forward, the best course of action for FEM2DGUI is to put it through a wider range of testing and allow anyone who is interested in using the program to use it. Since refinements were made to fix usability problems found by local users, it should be broadcasted to a larger audience, and test how a larger user-base responds to the interface. After users have had an adequate time to familiarize themselves with FEM2DGUI, they could be asked to participate in Field Observation testing or in general queries about the program. Field Observation is a usability test method involving visiting users’ workspaces and observing how they use the software under normal working conditions (Holzinger, 2005). The advantages of this method include how simple it is to conduct and the number of usability errors users commonly encounter. The biggest disadvantage of this method is the emergence of false positives if the users are over aware of the observer. Query answers from users could also be used to forward the development of FEM2DGUI. Using questionnaires and surveys would be useful for gathering information such as feature desires, satisfaction with using FEM2DGUI, and user anxieties. According to Holzinger (2005), these testing techniques suffer from some disadvantages, such as the number of responses that must be collected for uncovering fewer problems other than other usability test methods. However, these test methods would provide data from a large set of users. The evaluations conducted in this study were on a small group of users, and to forward the development of FEM2DGUI and its usability, more users are needed to be involved with the development process. Data from this project can be used for subsequent work involving FEM2DGUI. The collection of quantitative data will prove more useful because of the benchmarks found during the Usability Evaluations in this project.

As well as these test methods, usability inspection methods should also be utilized for the continuing development of FEM2DGUI. As discussed in 1.2.7, combining usability inspection
methods with test methods provides the most complete evaluation of the usability of an interface. Therefore, it would be beneficial to conduct another Heuristic Evaluation with the recommended three to five usability specialists. Heuristic Evaluation would be useful finding problems previously hidden behind the glaring errors now rooted out of FEM2DGUI. Furthermore, according to Holzinger (2005), Heuristic Evaluation can be effectively applied to an interface throughout the development cycle of an interface. There is a question as to how many of these Heuristic Evaluations can be performed before there are diminishing returns as far as usability is concerned. Nielsen (1996) states that an interface is ready, or “good enough”, for deployment when the design has been refined to address users’ concerns, such as likes, dislikes, and what they think works. As long as new features are added to FEM2DGUI, Heuristic Evaluation is a useful tool for refining usability.
Appendices

Appendix A: User Consent Form

VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY
Informed Consent for Participants in Research Projects Involving Human Subjects

Usability Engineering Applied to an Electromagnetic Modeling Tool

Samuel Fortson

I. Purpose of this Research/Project
You will be one of approximately ten participants in this study. All participants are graduate students in either Industrial Systems Engineering or Geophysics. The purpose of this study is to evaluate a geophysical model building graphical user interface and document usability errors found while interacting with the interface.

II. Procedures
In a single session in Derring Hall, you will be given a brief presentation on the domain and context of the geophysical problem in which this interface intends to aid the user in solving, asked to evaluate the interface and write down any usability problems you find. After the evaluation, you may be asked to join in an evaluation session where all of the participants will discuss the results of the individual evaluations and speculate on potential improvements to the interface. Finally, a questionnaire will be sent to you via email asking you to rate the severity of the usability problems found. The session should last no longer than two hours.

III. Risks
The risks associated with this study are the same that you would experience during normal computer usage. To minimize risk, you will be permitted to take a break at any time you wish. Merely inform the test administrator that you would like to do so.
IV. Benefits
The benefits to study participants are a better understanding of usability problems found in geophysical software and software in general. If you are interested in the results of this study please contact the researcher Samuel Fortson smfrtsn@vt.edu and a copy of the results will be sent to you once they are complete.

V. Extent of Anonymity and Confidentiality
The results of this study will be kept confidential. Due to the nature of the interviews you may be identified, but your name will not appear in any of the research results. You may be videotaped in the process of this study. The video will not be viewed by anyone other than the researcher and will be destroyed as soon as research is complete. It is possible that the Institutional Review Board (IRB) may view this study’s collected data for auditing purposes. The IRB is responsible for the oversight of the protection of human subjects involved in research.

VI. Compensation
Participants will be compensated $15.00 for this study. Participants will receive the full amount for attending the test and providing some feedback on the interface. The amount compensation amount will be received whether or not the final questionnaire is completed.

VII. Freedom to Withdraw
Participants are permitted to withdraw from the study at any point throughout the study without penalty.

VIII. Subject's Responsibilities
I voluntarily agree to participate in this study. I have the following responsibilities:
- Adhering to study procedures
- Answering interview and questionnaire questions honestly
- Attending briefing session
IX. Subject's Permission

I have read the Consent Form and conditions of this project. I have had all my questions answered. I hereby acknowledge the above and give my voluntary consent:

_______________________________________________ Date__________
Subject signature

_______________________________________________
Printed Name

Should I have any pertinent questions about this research or its conduct, and research subjects' rights, and whom to contact in the event of a research-related injury to the subject, I may contact:

Samuel Fortson (540-231-8827)  -  smfrtsn@vt.edu
Chester Weiss (540-231-3651)  -  cjweiss@vt.edu
David M. Moore  540-231-4991  moored@vt.edu

Chair, Virginia Tech Institutional Review
Board for the Protection of Human Subjects
Office of Research Compliance
2000 Kraft Drive, Suite 2000 (0497)
Blacksburg, VA 24060
Appendix B: Test Briefing

USABILITY ENGINEERING APPLIED TO AN ELECTROMAGNETIC MODELING TOOL

Investigator: Sam Fortson

AGENDA

• Signed consent form
• Pre-test questionnaire
• Briefing of domain and test procedures
• Conduct the test

TEST PURPOSE

• To evaluate the usability of a geophysical modeling tool
• Usability is defined here as the ease of use and learnability of this tool

DESCRIPTION OF MODELING TOOL

• Interface to already existing command line programs that will allow for input from the mouse as opposed to text files
• Within this interface users can draw subsurface geologies, create triangle meshes for finite difference methods, and run a forward modeling simulation (numerical hypothesis testing)

GEOPHYSICS OVERVIEW

• “Geophysics is the study of the planet earth using methods of physics”
• Geophysical studies can range from large-scale global geophysics (earth’s structure and behavior) to exploration geophysics (applying geophysical techniques to find subsurface targets, i.e., oil/gas, water)
• The technique this interface uses is magnetotellurics

MAGNETOTELLURICS (MT)

• MT method uses both electric and magnetic field observations to infer the electrical conductivity structure of the subsurface
• MT is based on natural sources originating in Earth’s ionosphere and magnetosphere
FORWARD MODELING OVERVIEW

- A forward model is one where the parameters (i.e., source frequency, conducting bodies, etc.) of the model have been defined and a software simulation is run to approximate real-world results.
- This particular forward model solver uses finite elements to compute predicted data.
- Finite elements require the generation of a triangular mesh.

USABILITY OVERVIEW

- For this study, participants will be asked to keep in mind Nielsen’s Usability Principles (Nielsen, 1993).
- These principles exist to provide guidelines for designers, developers, and testers of software to strive for when creating software.

USABILITY OVERVIEW (CON’T.)

- Visibility of system status:
  - The system should always keep users informed about what is going on, through appropriate feedback within a reasonable time.
- Match between system and the real world:
  - The system should speak the users’ language, with words, phrases, and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.
- User control and freedom:
  - Users often choose system functions by name and will need a clearly marked ‘emergency exit’ to leave the unwanted state without having to go through an extended dialogue, support undo and redo.
- Consistency and standards:
  - Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow pattern conventions.
- Error prevention:
  - Even better than good error messages is a careful design which prevents a problem from occurring in the first place.

USABILITY OVERVIEW (CON’T.)

- Recognition rather than recall:
  - Make objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily recoverable whenever appropriate.
- Flexibility and efficiency of use:
  - A flexible system saves step-by-step translation for the user since it can cater to both infrequent and frequent users. Allow users to take frequent actions.
- Aesthetic and minimalist design:
  - Dialogue should not involve information which is visually counterintuitive. Every element of information should contribute to the total purpose of the interface and should not appear merely to support other goals.
- Help and documentation:
  - Even though it is better that the system can recover without any intervention, it may be necessary to provide help and documentation. Such information should be easy to search, forward in the interface, and accessible whenever needed.

USABILITY TEST

- Participants will be asked to provide written feedback on what specific usability principle a certain aspect of the interface violates.
- Participants will be asked to step through 3 tasks as they use the interface:
  - Draw a 2D model that represents a cross-section of the Earth and apply conductivity values to each of these layers and/or anomalous bodies.
  - Generate a 2D triangular mesh of the model.
  - Run the forward model simulation and generate necessary output.
Appendix C: Pre-test Questionnaire

What is your job title?

How would you describe yourself? ("Programmer", "geophysicist", etc.)

How long have you been doing this kind of work?

How long have you been using personal computers?

How often do you use a personal computer?

Have you ever used a geophysical model builder that employed a graphical user interface?

If so, what have you used?
Appendix D: Post-test Questionnaire

Participant #:

Date:

This questionnaire is designed to tell us how you feel about the product you used. Please circle the number that most clearly expresses how you feel about a particular statement. Write any comments you have below each question.

1. Using the software was:

   1  Very Easy
   2  Easy
   3  Neither Easy Nor Difficult
   4  Difficult
   5  Very Difficult

   Comment:

2. Finding the features I wanted in the menus was:

   1  Very Easy
   2  Easy
   3  Neither Easy Nor Difficult
   4  Difficult
   5  Very Difficult

   Comment:

3. Understanding the instructions in the prompts was:

   1  Very Easy
   2  Easy
   3  Neither Easy Nor Difficult
   4  Difficult
   5  Very Difficult

   Comment:
4. Recovering from errors was:

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Very Easy</td>
<td>Easy</td>
<td>Neither Easy Nor Difficult</td>
<td>Difficult</td>
<td>Very Difficult</td>
</tr>
</tbody>
</table>

Comment:

5. Drawing points was:

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Very Easy</td>
<td>Easy</td>
<td>Neither Easy Nor Difficult</td>
<td>Difficult</td>
<td>Very Difficult</td>
</tr>
</tbody>
</table>

Comment:

6. Moving through the process steps was:

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Very Easy</td>
<td>Easy</td>
<td>Neither Easy Nor Difficult</td>
<td>Difficult</td>
<td>Very Difficult</td>
</tr>
</tbody>
</table>

Comment:

7. Drawing a mesh that you expected/wanted to see was:

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Very Easy</td>
<td>Easy</td>
<td>Neither Easy Nor Difficult</td>
<td>Difficult</td>
<td>Very Difficult</td>
</tr>
</tbody>
</table>

Comment:

8. What did you like most about this interface?

9. What did you like least about this interface?

10. What would you like it to do that it doesn’t do?
Appendix E: Think Aloud Protocol and Questionnaire Results

<table>
<thead>
<tr>
<th>Pre-test Questionnaire Results:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. PhD Student in Geophysics</td>
</tr>
<tr>
<td>2. Applied geophysics with moderate programming background</td>
</tr>
<tr>
<td>3. Has done this type of work for 3 years</td>
</tr>
<tr>
<td>4. Has used computers regularly for 8 years</td>
</tr>
<tr>
<td>5. Uses computers very often</td>
</tr>
<tr>
<td>6. GMSYS, Elementary Geophysics 1D modeling package, SPW, PROMAX, PULSEEKKO</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Think-aloud Results:</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Quantitative Measures:</strong></td>
</tr>
<tr>
<td>• Quit the program twice as opposed to clearing the screen to start a new model</td>
</tr>
<tr>
<td>• Confusion over mesh icon because they believed it to be disabled</td>
</tr>
<tr>
<td>• Time spent to recover from errors during a 30 minute test session: 3 minutes</td>
</tr>
<tr>
<td>• Approx: 5-10 minutes to draw a model from start to finish and generate the mesh (dependent on model complexity)</td>
</tr>
<tr>
<td>• Needed help on how to navigate from the mesh screen back to the draw screen and to clear the draw screen</td>
</tr>
<tr>
<td>• Expressed frustration over putting points on the edge of the model space</td>
</tr>
<tr>
<td>• Some attributes were placed in the wrong area and not moved because the participant did not know they were draggable</td>
</tr>
</tbody>
</table>

| **Qualitative Measures:**                                                                    |
| • Wants scale bars in km as well as in m                                                    |
| • Ability to insert AEI automatically at 0 depth                                             |
| • When adding lots of receivers the dialog box grows into an unmanageable size              |
| • Allow for input of resistivity as well as conductivity for attributes                     |
| • Automatically append .poly to the end of poly files when opening them for saving           |
| • The generate mesh icon appeared to be constantly disabled                                 |
| • “Triangle is Running” dialog does not close if Triangle computes too quickly               |
| • Desire for a table with known conductivity ranges to avoid looking them up in a different location or having to remember them |
| • Desire for the ability to automatically define an air region                               |
| • “snap to horizontal” for finishing drawing                                                |
| • Ability to add points directly to the lines                                                |
| • Scale bar value spacing cannot be controlled by the user                                  |
| • Angles for dipping layers                                                                  |
### Post-test Questionnaire Results:

1. 3 – See usability suggestions
2. 4 – Not always apparent where to go through the flow and how to go back. Icons sometimes don’t make sense.
3. 2 – N/A
4. 4 – Not always apparent where to go and what to do. This became easier with more usage.
5. 3 – Drawing was easy but placing points at specific locations was difficult. Horizontal lines were hard to draw.
6. 3 – Was not apparent what came next
7. 3 – Did not draw own mesh
8. Great graphical interface and is more user friendly than the standard text file.
9. See previous comments.
10. Manipulate the mesh by hand after creation. Work in time domain. Cool plots at end.

### Second Test Questionnaire Results

1. 1 – Much improved, very self-explanatory
2. 2 – Good symbols! I love the inclusion of the “add horizontal line” and the ability to change points (by double-clicking and manually setting x and z values)
3. 3 – Not sure that there was usable instructions
4. 1 – N/A
5. 1 – Changing values, deleting, adding to lines, “select all” are good
6. 2 – No problem, but then I already knew how to use FEM2D
7. 2 – Just a little trial and error. It would be nice to be able to control the mesh density over the space a little more
8. Easy to use and much better than older version
9. N/A
10. Iterate through multiple frequencies.

*Table 0.1: Thinking Aloud Protocol and questionnaire results from participant B1*
### Participant: B2

#### Pre-test Questionnaire Results:

1. Geophysicist graduate student  
2. Seismologist  
3. 4 years experience with this kind of work  
4. 12 years experience with computers  
5. 3-4 hrs/day pc usage  
6. Landmark’s SeisWorks

#### Think-aloud Results:

<table>
<thead>
<tr>
<th>Quantitative Measures</th>
<th>Qualitative Measures</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Difficulty setting receiver positions (set by receiver spacing instead of ((x, y)) coordinates. Repeated 5 times</td>
<td>- Ability to specify angles of dipping layers</td>
</tr>
<tr>
<td>- Used coordinate dialogs incorrectly one time by placing values in wrong boxes</td>
<td>- Set AEI to specific points as opposed to estimating</td>
</tr>
<tr>
<td>- Twice had trouble placing points along the right-hand side of the model space</td>
<td>- Use different mouse click to pause drawing</td>
</tr>
<tr>
<td>- Found it ambiguous as to how attribute points are placed on the screen (are they just at an ((x, y)) location, or are they tied to a particular region)</td>
<td>- Have a table containing known ranges of conductivity/resistivity values for common rock types</td>
</tr>
<tr>
<td>- Confusion over “triangle is running box”, did not behave as expected by either closing or the bar moving back and forth as triangle was running</td>
<td>- Generate mesh terminology is a bit too ambiguous</td>
</tr>
<tr>
<td>- Unclear how and when to move from draw screen to view screen</td>
<td></td>
</tr>
<tr>
<td>- Unclear of the meaning of “nodes per skin depth”</td>
<td></td>
</tr>
<tr>
<td>- Deletes points one by one as opposed to using the mouse drag function or ctrl button (described this as a problem without realizing it is already possible)</td>
<td></td>
</tr>
<tr>
<td>- Very large meshes with very high conductivity values caused program to crash and/or unable to draw the mesh elements to the screen. The time spent trying to recover from this crash was 5 minutes before I intervened. The only way to recover from this crash was to terminate the program</td>
<td></td>
</tr>
<tr>
<td>- Approximately 5-10 minutes to complete a model depending on complexity</td>
<td></td>
</tr>
</tbody>
</table>
### Post-test Questionnaire Results:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1 – N/A</td>
</tr>
<tr>
<td>2</td>
<td>1 – N/A</td>
</tr>
<tr>
<td>3</td>
<td>1 – N/A</td>
</tr>
<tr>
<td>4</td>
<td>1 – N/A</td>
</tr>
<tr>
<td>5</td>
<td>1 – N/A</td>
</tr>
<tr>
<td>6</td>
<td>1 – N/A</td>
</tr>
<tr>
<td>7</td>
<td>1 – N/A</td>
</tr>
<tr>
<td>8</td>
<td>Fluid, ease of use. Simple</td>
</tr>
<tr>
<td>9</td>
<td>No complaints</td>
</tr>
<tr>
<td>10</td>
<td>More reference for non-EM geophysicists (i.e. reference to conductivity properties of common sub-surface rock types).</td>
</tr>
</tbody>
</table>

### Second Test Questionnaire Results

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2 – New features learning curve</td>
</tr>
<tr>
<td>2</td>
<td>1 – The symbols made sense</td>
</tr>
<tr>
<td>3</td>
<td>1 – Well explained prompts</td>
</tr>
<tr>
<td>4</td>
<td>2 – The “clear all” button was a good feature</td>
</tr>
<tr>
<td>5</td>
<td>1 – Double-click to terminate line was a good addition</td>
</tr>
<tr>
<td>6</td>
<td>2 – Well documented instructions</td>
</tr>
<tr>
<td>7</td>
<td>2 – Mesh works</td>
</tr>
<tr>
<td>8</td>
<td>Simple usage</td>
</tr>
<tr>
<td>9</td>
<td>Not quite streamlined enough</td>
</tr>
<tr>
<td>10</td>
<td>The software has improved all-in-all</td>
</tr>
</tbody>
</table>

*Table 0.2: Thinking Aloud Protocol and questionnaire results from participant B2*
**Participant: B3**

**Pre-test Questionnaire Results:**
1. PhD Student
2. Geophysicist
3. 2 years as geophysicist
4. Has used computers for a very long time
5. Everyday
6. Surfer, RESIX

**Think-aloud Results:**

<table>
<thead>
<tr>
<th>Quantitative Measures</th>
<th>Qualitative Measures</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Insert coordinate dialog box is ambiguous, accidentally reversed axis on first pass</td>
<td>• Double click to release control of points</td>
</tr>
<tr>
<td>• Expected shapes and layers to scale when the coordinates changed</td>
<td>• Allow for predefined shapes (rectangles, circles)</td>
</tr>
<tr>
<td>• Attributes are tied to top left of box, this user expected them to be in the middle</td>
<td>• Receiver dialog should grow as receivers are added</td>
</tr>
<tr>
<td>• Moving between draw and view screens is ambiguous</td>
<td>• Ask user if they want shapes to scale with coordinate changes</td>
</tr>
<tr>
<td>• Didn’t know when to move to view screen after mesh was generated</td>
<td>• Color bar on left-hand side should only be present in the mesh view screen</td>
</tr>
<tr>
<td>• Showed frustration when attempting to put points on the edges, repeated the same error 6 times</td>
<td>• Choose frequency first, show skin depth, then ask for nodes per skin depth to give the user a better sense of how many they should add</td>
</tr>
<tr>
<td>• Confusion over knowing when points are on the edges</td>
<td>• Ability to change color scales</td>
</tr>
<tr>
<td>• Attempted to delete objects not by the point but by the lines</td>
<td>• Ability to change axis scaling</td>
</tr>
<tr>
<td>• Time spent trying to understand how receiver dialog box works: 1 minute</td>
<td>• Ability to add title to the meshes</td>
</tr>
<tr>
<td>• Unable to quickly find the icon for generating the mesh</td>
<td>• Ability to remove mesh lines so they can see the color conductivity pictures without the lines</td>
</tr>
<tr>
<td>• Confused how z-coordinate should be entered</td>
<td>• Ability to delete mesh scene and return to draw screen with an icon in the toolbar</td>
</tr>
<tr>
<td>• Tried to use help…doesn’t exist</td>
<td>• Warn user if there are points that aren’t on the edges when they should be</td>
</tr>
<tr>
<td>• Difficultly navigating the input receiver dialog box</td>
<td>• Allow for reversing axis</td>
</tr>
<tr>
<td>Table 0.3: Thinking Aloud Protocol and questionnaire results from participant B3</td>
<td></td>
</tr>
<tr>
<td>---</td>
<td></td>
</tr>
</tbody>
</table>

### Post-test Questionnaire Results:

1. 2 – N/A
2. 2 – It is easy when you get used to it.
3. 3 – N/A
4. 3 – N/A
5. 2 – See suggestions
6. 2 – N/A
7. 3 – N/A
8. See video.
9. See video.
10. See video.

### Second Test Questionnaire Results

1. 2 – It helps to add documentation
2. 2 – More help
3. 2 – N/A
4. 2 – More description
5. 2 – N/A
6. 2 – N/A
7. 2 – N/A
8. I like it in general. It is improved from last time I used it.
9. N/A
10. N/A
Works Cited


