Proceedings of the 8th Winona Computer Science Undergraduate Research Symposium

April , 2008

Table of Contents

Effective Security Enhancement for Radio Frequency Identification Tags

Andrew Biermaier
Department of Computer Science
Winona State University
Winona, MN 55987
Abbierma7009@winona.edu

ABSTRACT

Radio frequency identification (RFID) technology provides a low cost, highly energy efficient and convenient solution to many everyday problems. Due to these benefits, RFID tags are becoming more popular. As this technology becomes more popular, it will attract the attention of those who wish to misuse it. The best way to prevent malicious attacks is to identify weaknesses and fix the vulnerability before the attacker has time to identify the vulnerability. This paper outlines the research that has been done in this area to identify vulnerabilities in RFID technologies. This paper also provides research into a proof of concept to better protect the information on proximity RFID tags, such as the tags that exist in newer implementations of passports and credit cards.

General Terms

Measurement, Performance, Economics, Reliability, Experimentation, Security, Legal Aspects

Keywords

RFID, Security, Radio frequency, E-passport, Measurement

1. INTRODUCTION

Radio frequency identification (RFID) technology has several potential applications in everyday life. There are more than 20 million Americans who own credit or debit cards that contain RFID tags [3]. These cards enable customers to make purchases faster and more conveniently by allowing them to lightly tap the credit card against the reader rather than swiping it through the magnetic strip reader [5]. There are even plans to put RFID tags into cell phones in order to hold credit card information. This would allow people to use their cell phone as a credit card [3]. There are several other applications of this technology. You can find RFID tags in passports, identification cards and packaging for merchandise. They are even used for security keys and implanted into pets as means of identification. Though this technology has been increasing in popularity, it is not a new concept. The idea of using radio frequencies for identification started during World War II where this concept was used to identify aircrafts [1].

1.1 Past Problems

The use of radio frequencies for transmitting data has broad applications. Unfortunately, this wide range of applications and growing popularity makes RFID tags a prime candidate for malicious attacks. Finding these vulnerabilities before they are exploited is a crucial step in security. Knowing about these vulnerabilities will help to patch current vulnerabilities and prevent further mistakes in the implementation. We can look at the 802.11 protocol and see clearly that when new technologies are adopted quickly, the time spent on security is usually minimal and not well thought out. The original implementation of 802.11 was vulnerable to a wide array of attacks. Some of these attacks were message injection, message modification, message interception, denial of service attacks, and several other attacks. The main reason these attacks are possible is due to several vulnerabilities associated with the poor security implementation. What is worse is that the security implementation is optional, leaving some networks completely open to attackers. Many of these same issues have been identified in newer technologies, such as Bluetooth and the focus of this paper, RFID.

Historically, security has been an afterthought in the implementation of a technology rather than one of the main concerns. Unfortunately, it appears that history is repeating itself with RFID technology. We have not taken enough time to test implementations of this technology and because of this we will experience many of the same security issues we have seen with IEEE 802.11 and many other protocols. This is not to say there is no security for RIFD. Encryption schemes exist for this technology, but they still have problems. One of the main problems is key exchange. Van Le, Burmester, and Medeiros have proposed a solution to this problem with authentication [6]. Even with the existence of this security implementation that can be used, many RFID tags still go unprotected, because of this something more is needed.

1.2 Objectives

The primary objective of this research is to illustrate some of the potential vulnerabilities that are associated with RFID technology, describe how attacks may be carried out and how one can protect themselves against these attacks. We addressed the proximity characteristic as a means of security of RFID tags. Proximity RFID tags are designed to be read only at a range of about 4 inches from the reader. Though the normal read range is about 4 inches there exist devices that can read tags from much greater distances, these devices are extended range sweepers [4]. One may also increase the power to a normal RFID reader to attempt to increase the read range. With our test materials we observed read ranges between 3 and 10 inches. In this study, we tested the effectiveness of the reader under several conditions that would be present if an attack were to be applied to everyday life. These conditions included reading through a wallet, backpack and a desk drawer to see at what range and under what circumstances RFID tags are most vulnerable. We will present a proof of concept solution to the range vulnerability that exists with the proximity RFID technologies. One of the main ways that RFID tags protect ones information is by limiting the range. With this limited range, the tag should be secure against attacks that may prove malicious. We will cut the effective range from the standard 3-10 inches to less than an inch.

2. HYPOTHESIS

Using easily acquired materials, one is able to easily implement an effective solution to reduce the read range of RFID tags by at least 70%. This solution will increase security of information and attempt to stop ranged attacks against RFID technology. Furthermore this solution can be implemented inexpensively and by people who are non-specialists to the computer science field.

3. METHODS

Through this research, we attempted to strengthen the security of proximity RFID tags by limiting the maximum range from which the tag can be read. This experiment was designed to either block the transmission from the tag completely or to reduce the transmission to a “safe” range. We identified a strong approach to blocking the RFID transmissions. This method is to shield the tags, similar to how faraday cages work. For this experiment we compared several different materials that provide various amounts of shielding for radio frequencies. In particular, we used 26 gauge copper sheet metal, 26 gauge steel sheet metal, 26 gauge aluminum sheet metal, and aluminum foil. We preformed this testing using only RFID tags that are embedded in items we own, or have been approved to use. We used two RFID tag readers along with six RFID tags to perform the testing. The tags are of varying sizes, from a one inch diameter to a 2 7/8 X 2 inch rectangle. These varied sizes gave us more accurate results in our testing thus allowing for our analysis to be more complete and stronger conclusions can be made. We have broken down our methods into several phases that were preformed to complete our research.

3.1 Equipment

The first phase of this research was to obtain the materials needed. We obtained a RFID reader from 3M that we used to perform some of the tests. This reader also came with several RFID tags that we used as part of the research experiment. A lower end RFID reader that also came with several tags of differing sizes and shapes was also used in the experiments for this research. This reader and tags were obtained from a company called RFIDeas Inc. The product purchased is a PC ProX RFID reader.

3.2 Equipment Specifications

In the next phase, we identified what protocols and standards that are used for each tag and reader. This is important to note since we are not using tags that contain sensitive information. It is important that the tags that we test ran at the same frequencies as the tags that could be attacked by malicious individuals. The 3M tags operate at 13.56 MHz. Similarly, the PC ProX tags operate at 13.56 MHz. To operate at 13.65MHz means that the tags are communicating by using radio waves at the frequency of 13.56 MHz. Most implementations of RFID tags in practice use the 13.56 MHz frequency for communication. Because the tags that we used in testing run at the same frequency as most tags in the market, they are a good representation of the read ranges tags that exist in personal items.

3.3 Identification of Required Data

We had some difficulty measuring the read ranges accurately with small metrics. As such, we decided to measure with an eighth of an inch precision. This allowed for a reasonable precision while significantly increasing the accuracy of the measurements. We also preformed some base line testing. The base line testing consisted of testing the read ranges of an RFID tag with no shielding, both inside of and outside of a wallet. For the actual tests, we tested the tags with the shield up against the tag and with the tag and shield within a wallet without them touching.

3.4 Environment Design and Construction

Before we could conduct these experiments, we had to set up our test environments. For our testing environments, we used the RFID tag readers described in Section 3.1, six RFID tags, and tools for measuring the distances of the tags. The reader was oriented in a vertical manner in order for the tag to be slowly moved towards the reader alongside a tape measure. This allowed for greater precision and a more organized testing environment. This is shown in Figures 1 and 2.

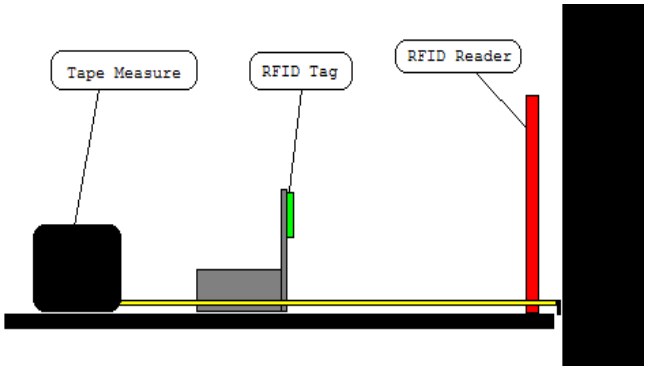


Figure 1. Side View of Testing Environment

After setting up the testing environment, we ran our experiments which consisted of the base line testing, followed by the testing of the application each shielding material to a wallet. To do this, we applied a metal plate to the inside of a wallet between the tag and the reader. We then tested the maximum read distances that each reader is able to produce with each of the tags. We repeated these experiments with each of the test materials, copper, steel, and aluminum.

3.5 Resolved Issues

There are a few problems which had to be addressed for this project. The first issue to be resolved was setting up a stable environment for doing range measurements on the RFID tags. Simply holding a tag in ones hand and measuring the distance at the same time is inconsistent, inaccurate and difficult. We decided to use a block of wood to stabilize a piece of cardboard that held the RFID tag. This stability allowed for more stable movement of the RFID tag across the table which made measurement more accurate.

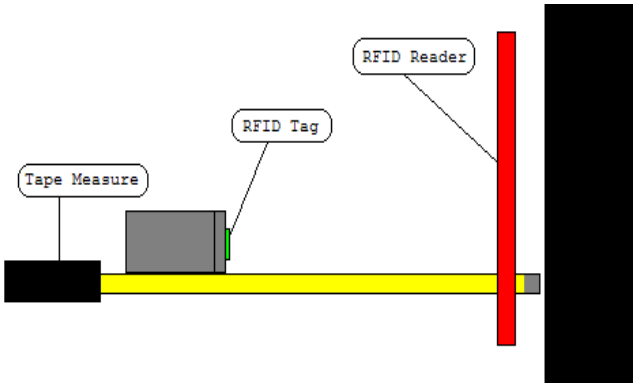


Figure 2. Top View of Testing Environment

For a measuring tool, we ended up using a tape measure running along the top of a table to the edge of a RFID reader which sat vertically on the table. This setup is shown in Figures 1 and 2. To measure the distance, we placed the tag on a stable surface and moved the RFID tag closer to the reader until the tag was read. The next problem to be resolved was determining what measurements are significant. Obviously, the distance from the reader is a significant measure but the question is, are there other significant measures such as speed at which the tag is moving, or the angle that the tag is to the reader. In the initial testing, some of these measurements were done to see if there is any significance to them. This testing determined that the angle the tag approaches the reader does not affect the read range. There was also some correlation between speed the tag was moved and the ability for it to be read. Another problem is determining what normal environment for these tags may be. This is very important in setting up the baseline testing. If the tags can be read from their normal environment, then there is a need for protection. Therefore determining the environment that they will be read from is a very important step in this process. The normal environments that were suggested for RFID tags were purses, desks, wallets and back packs.

Some initial testing was done to see if tags within these environments could be read at a reasonable range. The read ranges were fairly consistent with what we found with the tag being in the open, thus these environments should be considered insecure. This data can be found in Appendix A. Our results were obtained through extensive testing that included ten trials for every tag for each part of the experiment. This included ten readings for each card in the baseline testing, ten readings with just the shield for each material, ten readings with the tag in the wallet and ten readings for each shielding material within the wallet for ever tag. We used eighth of an inch increments for the maximum read range measurement. These readings were compiled into a spreadsheet for later analysis. Our method of measurement as described in the methods section was to use a stable surface to move the tag slowly towards the reader until the tag was read and then we record the measurement of this range. This system seemed to work well, however, it only allowed for an eighth of an inch precision. Each row of data was then averaged to get a better look at what our data actually means.

4. RESULTS

The averages obtained from the experiments where compiled into Tables 1-3. Table 2 subcategorized to show the difference between the 3M and PC ProX readers and their performance as shown in Table 3. Tables 2 and 3 show that shielding RFID tags does reduce the maximum read range and may be an applicable implementation for securing sensitive information held on these tags. Once the tables were filled out and our data was collected we were able to create graphs to show this data in a more reasonable format shown as Figure 3 and Figure 4.

Table 1. Baseline Testing: Maximum read range for each tag with no shielding.

3M Tag 1	Antenna Size: 47 X 45 mm	Average Max Read Range: 11.088″
3M Tag 2	Antenna Size: 47 X 45 mm	Average Max Read Range: 8.457″
3M Tag 3	Antenna Size: 47 X 45 mm	Average Max Read Range: 8.163″
PC ProX Tag1	Antenna Size: 50 mm	Average Max Read Range: 2.975″
PC ProX Tag2	Antenna Size: 73 X 50 mm	Average Max Read Range: 4.100″
PC ProX Tag3	Antenna Size: 25 mm	Average Max Read Range: 2.558″

Table 1 shows the read ranges for the different tags that were used for testing. It is apparent from the data shown in this table that the read range for different tags is variable and thus several tags of differing antenna sizes were tested. It is also important to note that the 3M reader has its own power supply while the PC ProX RFID reader runs off the power supplied to it through USB. This difference in power likely has some effect on the potential maximum read range of the devices. Another difference to note is the antenna size of both the reading devices. The 3M reader has a much larger antenna measuring approximately 5" X 5" while the PC ProX reader has a much smaller antenna measuring approximately 1" X 2". The reason for the experiments with these different readers and tags is that it is necessary to test a broad range of antenna sizes and power levels in order to be able to have some generalizations. We feel that due to the broad range testing, the results should appropriately reflect the results that would be found by testing extended range readers and sweepers that were described in Section 1.2.

Table 2. Average maximum read range of all RFID tags for shielded wallet.

	Copper	Steel	Aluminum	Aluminum Foil
Average	87.9%	81.3%	88.9%	79.1%

With the use of a 26 gauge copper shield within a leather wallet, the average read range of all tested tags was reduced by 87.9 percent. With the use of the 26 gauge steel shield, there was an average reduction in maximum read range of 81.3 percent. With the use of the 26 gauge aluminum shield, there was a 88.9 percent reduction in maximum read range. Finally, with the aluminum foil, we observed a maximum read range reduction of 79.1 percent. All of these reduction percentages are above the range we were looking for when identifying a valid shielding mechanism. As such, we feel that any of these shielding materials would work sufficiently in the added protection of these tags.

Table 3. Percent reduction in maximum read range with shielded wallet.

	None	Copper	Steel	Aluminum	Aluminum Foil
3M 1	0.000	0.771	0.767	0.776	0.798
3M 2	0.000	0.776	0.795	0.792	0.755
3M 3	0.000	0.738	0.799	0.769	0.745
Average	0.000	0.762	0.787	0.779	0.766
PC ProX 1	0.000	1.000	0.632	1.000	0.824
PC ProX 2	0.000	0.989	0.888	1.000	0.624
PC ProX 3	0.000	1.000	1.000	1.000	1.000
Average	0.000	0.996	0.840	1.000	0.816

As shown in Table 3, the average percent reduction in maximum read range is both dependent on the tag being used and the reader. As discussed earlier in this paper, the tags with the larger antennas have larger read ranges. These tags are also less affected by the shielding that was implemented in these experiments. Likewise, the 3M reader has a larger antenna which allowed the read ranges of the tags to be greater. This also had an impact on the effectiveness of the shielding. The 3M tags antenna measured 2" X 2", while the antenna for the reader measured approximately 5" X 5".

5. ANALYSIS

The results of this experiment have shown us that with the application of a shield, one is able to reduce the maximum read range of a RFID tag. In this section, we will look at what this means and how significant the reduction is. As described in the previous section, we used averages of the data collected to represent our findings. Tables and graphs have been compiled to reflect these averages.

In Table 3, the overall averages have been expanded to give you a better view of the data. In this table, we have shown the breakdown of averages for each tag with each type of shield. One can see that there is a significant difference between the maximum read ranges for both the 3M reader and the PC ProX Reader. Both readers still show a significant reduction in read ranges on average. Though hey have not completely shielded the tag from being read, each of the shields has reduced the range that it can be read to within a "safe" range by use of a standard RFID reader. We can see that the 3M reader had a reduction in max read range by between 76 and 78 percent while the PC ProX had a reduction of read range of between 81 and 100 percent on average. These differences are consistent with the differences in the actual readers. As described earlier, the 3M reader is more powerful and has a larger antenna which should reduce the effectiveness of the shielding.

We can see from Table 3 that even with the variations in maximum read ranges the overall averages on the low end still falls within the significant maximum read range of 70% that we proposed in Section 2. However, the PC ProX tag with steel being used as a shield was less than 70% and would be considered not a significant reduction in maximum read range. Another example that does not fall within the significant reduction is the PC ProX tag two with the use of aluminum foil. Though these two examples exist, the data shows that the shielding does provide a significant reduction in range in most cases. To better display these findings, graphs showing the data in different views have been generated.

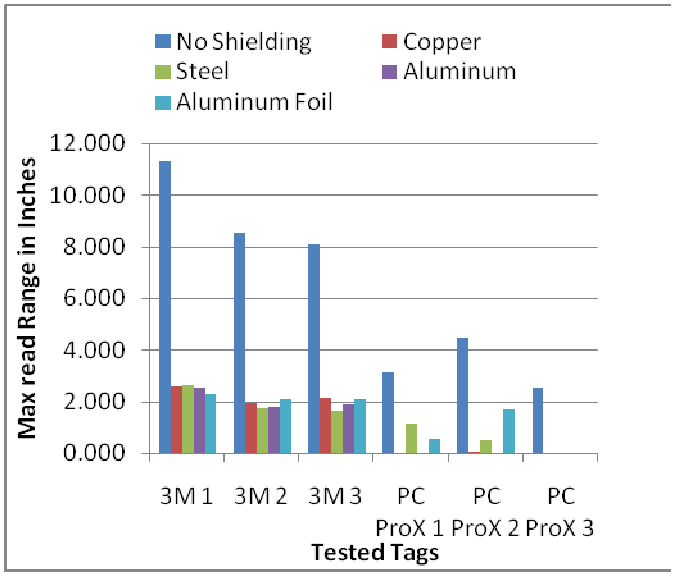


Figure 3. Maximum read ranges of tags within unshielded and shielded wallets.

As Figure 3 shows, there is a rather large reduction in the average maximum read range of a given tag when shielding is applied to the wallet containing the RFID tag. The dark blue bars represent the maximum read range of a tag within a wallet with no shielding. All other colors represent the maximum read ranges for varying shielding materials. The graph shows very clearly that there is a difference between an unshielded and shielded wallet. For a more detailed look, you can refer to Appendix A which has a table showing the actual values that these bars represent.

Viewing this data with percentages, we get the graph shown in Figure 4. This graph represents the percent reduction in maximum read range for each of the shielding materials. This alternate view clearly shows that most of the test cases were well above the significant percentage range and helps to show that the shielding worked well enough to be a worthwhile application.

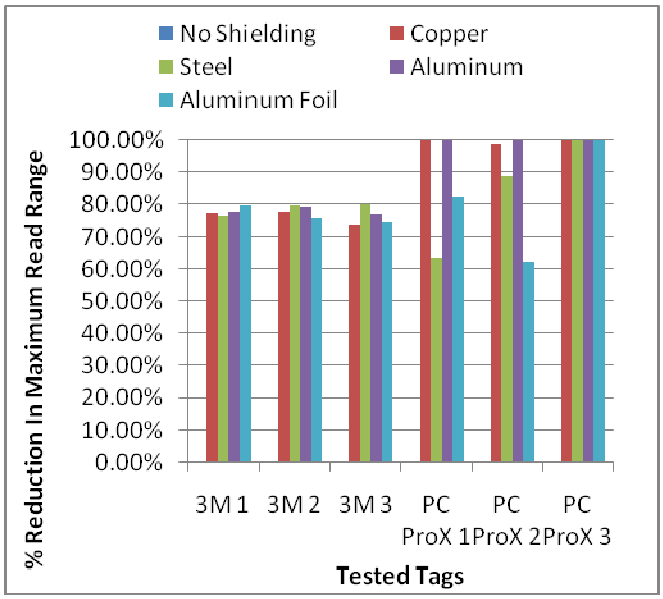


Figure 4. Average percent reduction in maximum read range given a particular tag and shielding material.

Through these experiments and data analysis, sufficient evidence was gathered to conclude, even with the variance in results for each material, that implementation of a shielding material will significantly reduce the maximum read range of RFID tags and help enhance the protection of your sensitive information. Our criteria for significant reduction in maximum read range and sufficient supplemental protection was a reduction in the maximum read range of an RFID tag by at least 70 percent. There was variance in the performance of each tag though most of the time the shielding results with averages above the expectations The averages consistently showed a reduction of over 70 percent for each tag, and ranging above 90 percent. The outliers should not be ignored and this is why the implementation of a shield has been suggested as a enhancement, not a solution to the insecurities in RFID technology. However, we feel that the shielding techniques do provide significant reduction in maximum read range. With the results of the experiment supporting our hypothesis, we have shown that one can implement a supplemental means of security for their RFID tags that hold sensitive information. Furthermore, this implementation is cheap and easy to do.

6. CONCLUSION

In this paper, we have identified that one of the major insecurities associated with RFID technology is the read range. Because the limited read range is intended to act as a security mechanism, it is crucial that the read range truly is limited. With our testing of RFID tags, we were able to read them from almost a foot way without any modifications to the equipment we used. Though this distance is relatively short, it is not short enough to prevent attackers form stealing information without physical contact. The ability for this to happen provides attackers with a way of stealing information without being noticed. Our research considered the feasibility of providing a radio frequency shield in order to reduce

or completely stop the transition of sensitive information from RFID tags.

During the testing, we noticed some interesting results that are not directly part of the research but are worth mentioning. If there are multiple tags next to each other, they make it more difficult for the reader to read any of the tags. The number of tags and the distance at which this stacking of tags affected the reader was different for the two readers tested. The 3M reader only had a minimal reduction in read range when two tags were stacked, however, the PC ProX reader could not read either of them at any distance when any two tags were stacked. It took four or more tags to cause significant problems with the 3M reader, however, with four tags the reader could not read any of the tags unless the tags were actually against the reading surface.

We presented a methodology for performing a RFID read range shielding experiment. We also presented the results of this experiment to support our analysis of the shields tested. The analysis of our results leads us to the conclusion that the implementation for a small conductive shield is an effective enhancement to securing the data contained on RFID tags that contain sensitive information. We also identified ways of increasing the range that we did not test. There are several new ideas that we have come across through this research. We feel that, even though our experiments were successful, there is much work to be done in this area.

Though we are satisfied with the results we obtained from this research, we do not think that this topic should be closed. There is still a lot of work to be done associated to security of RFID technologies. Some necessary future work is the testing of extended range sweepers and readers. There should also be more testing with a larger variety of tags, not just passive proximity tags. There are tags that are designed to have longer ranges, such as the new passport that will be used for frequent travelers from the Unites States to Canada. This RFID tag is designed to be read from several meters rather than just a few inches [7]. Tags such as these have even more security implications.

Even though this research can be applied to other things people carry, such as purses and back packs, more testing needs to be done in this area to confirm these beliefs. There should also be some work done in implementing an efficient and easily implemented cryptography scheme for RFID tags. Currently, many RFID tags are left un- encrypted which creates even greater incentive for attacking these tags. However, with an easily implemented cryptography system, more companies will accept the added complexity. Another area of testing that should be looked into is at what ranges RFID tags can be read while moving at various speeds. This topic would have significant influence on tracking individuals via RFID. We also feel that there are several improvements that could be done to refine our methodology and obtain even more precise results. One of the main things that could be done to improve the methodology is to conduct the testing in an area where there would be no radio frequency interference such as within a faraday cage. Another thing that could have been done is have more than one person take the measurement for each test in order to increase accuracy. This project has been a good starting point for looking at security in RFID technology and we hope that work continues in this field.

Acknowledgements

Our thanks to Brian Hackerson at 3M for providing us with a RFID reader and several RFID tags for our research. We would also like to thank Richard Moore at 3M for providing us with some technical support in using the RFID reader. Brian Hackerson and Richard Moore have been a great help to this research.

References

[1] Bendavid Ygal, Wamba Samuel Fosso, Lefebvre Louis A, Proof of concept of an RFID-enabled supply chain in a B2B e-commerce environment . ICEC'06, August 14–16, 2006, Fredericton, Canada.

[2] Bernardi, Gandino, Montrucchio, Rebaudengo, Sanchez, Design of an UHF RFID Transponder for Secure Authentication. GLSVLSI'07, March 11–13, 2007, Stresa-Lago Maggiore, Italy.

[3] Bray Hiawatha, Credit cards with radio tags speed purchases but track customers, http://www.boston.com/business/technology/articles/2006/08/14/credit_cards_with_radio_tags speed_purchases_but_track_customers_too/ August 14, 2006. February 2, 2008

[4] Kirschenboum Ilan, Wool Avishai. How to Build a Low-Cost, Extended-Range RFID Skimmer. Security 2006: 15th USENIX Security Symposium

[5] Swedberg Claire. AmEx Adds RFID to Blue Credit Cards. RFID Journal, June 7, 2005

[6] Van Le Tri, Burmester Mike, de Medeiros Bren. Universally Composable and Forward-secure RFID Authentication and Authenticated Key Exchange . ASIACCS'07, March 20-22, 2007, Singapore.

[7] Office of the Spokesman. Department of State to Introduce Passport Card, Media Note Office of the Spokesman. Washington, DC October 17, 2006

Appendix A: Raw Data

Read Range of RFID tags with no obstacles:

Tag # 3M Tags

	1	2	3	4	5	6	7	8	9	10
1	11.000	10.875	11.125	11.000	11.125	11.125	11.250	11.250	11.125	11.000
2	8.375	8.375	8.500	8.625	8.500	8.750	8.250	8.375	8.500	8.500
3	8.000	8.125	8.000	8.250	8.125	8.125	8.125	8.125	8.375	8.375

Tag # PC ProX

	1	2	3	4	5	6	7	8	9	10
1	3.000	3.000	3.000	3.000	2.875	3.000	3.000	3.000	2.875	3.000
2	4.375	4.125	4.000	4.000	4.125	4.125	4.125	4.125	4.000	4.000
3	2.500	2.625	2.625	2.625	2.625	2.625	2.500	2.625	2.625	2.500

Maximum read range from within a wallet:

Tag # 3M Tags

	1	2	3	4	5	6	7	8	9	10
1	11.250	11.250	10.875	11.375	11.500	11.125	11.625	11.125	11.750	11.250
2	8.625	8.625	9.000	8.875	8.125	8.250	8.375	8.625	8.500	8.375
3	8.250	8.000	8.250	8.000	8.125	8.000	8.250	8.125	8.000	8.000

Tag # PC ProX

	1	2	3	4	5	6	7	8	9	10
1	3.125	3.000	3.000	3.250	3.125	3.125	3.000	3.125	3.250	3.250
2	4.500	4.500	4.625	4.375	4.375	4.375	4.375	4.500	4.500	4.375
3	2.375	2.375	2.500	2.500	2.625	2.625	2.750	2.625	2.500	2.375

Maximum range testing for 3M tags within unshielded backpack:

Tag # 3M Tags

	1	2	3	4	5	6	7	8	9	10
1	11.000	11.500	11.000	11.500	11.500	10.750	11.000	11.500	11.000	11.000
2	8.500	8.500	9.000	9.000	8.000	8.500	8.500	8.500	8.000	8.000
3	8.000	8.000	8.000	8.000	8.000	8.500	8.000	8.500	8.750	8.000

Tag # PC ProX

	1	2	3	4	5	6	7	8	9	10
1	3.000	3.000	3.000	3.500	3.000	3.000	3.000	3.250	3.250	3.250
2	4.500	4.500	4.500	4.000	4.500	4.000	4.500	4.500	4.500	4.000
3	2.000	2.250	2.500	2.500	2.500	2.500	2.750	2.500	2.500	2.250

Maximum range testing for 3M tags within unshielded purse:

Tag # 3M Tags

	1	2	3	4	5	6	7	8	9	10
1	11.125	11.500	11.375	11.125	11.375	11.000	11.000	11.125	11.500	11.250
2	8.250	8.500	8.250	8.750	8.125	8.250	8.375	8.500	8.250	8.750
3	8.250	7.750	8.250	8.000	8.125	7.750	8.125	8.125	8.125	8.000

Tag # PC ProX

	1	2	3	4	5	6	7	8	9	10
1	3.125	3.000	3.000	3.250	3.125	3.125	3.000	3.125	3.250	3.250
2	4.500	4.500	4.625	4.375	4.375	4.375	4.375	4.500	4.500	4.375
3	2.375	2.375	2.500	2.500	2.625	2.625	2.750	2.625	2.500	2.375

Maximum range testing for 3M tags within unshielded desk:

Tag # 3M Tags

	1	2	3	4	5	6	7	8	9	10
1	10.750	10.500	10.875	10.375	10.500	11.000	10.500	11.000	10.750	10.000
2	8.000	7.625	8.500	8.000	8.500	8.500	8.000	8.000	8.500	8.250
3	8.000	8.250	8.500	7.750	8.250	8.000	8.000	8.000	7.750	8.000

Tag # PC ProX

	1	2	3	4	5	6	7	8	9	10
1	3.000	3.000	3.000	2.750	3.000	2.500	3.000	3.000	2.750	3.250
2	4.000	4.500	4.000	4.500	4.000	4.000	4.375	4.250	4.000	4.000
3	2.000	2.500	2.250	2.000	2.250	2.250	2.500	2.000	2.250	2.250

Maximum range testing for 3M tags within shielded wallet:

Copper 26 Gauge

Tag #

	1	2	3	4	5	6	7	8	9	10
1	2.000	2.500	2.875	2.500	2.875	2.500	2.750	2.875	2.125	2.875
2	1.750	1.750	1.750	2.000	2.000	2.000	2.000	1.875	2.000	2.000
3	2.000	2.875	2.750	2.000	1.500	1.875	2.125	2.000	2.250	1.875

Steel 26 Gauge

Tag #

	1	2	3	4	5	6	7	8	9	10
1	2.125	2.250	2.250	2.750	2.875	2.750	2.875	2.875	2.750	2.875
2	1.750	1.875	1.750	1.625	1.875	2.000	2.000	1.500	1.500	1.625
3	1.500	1.500	1.750	1.250	1.875	1.375	1.500	1.625	2.000	1.875

Aluminum 26 Gauge

Tag #

	1	2	3	4	5	6	7	8	9	10
1	2.875	2.750	2.500	2.125	2.500	3.625	2.000	2.125	2.500	2.375
2	1.500	1.625	1.500	1.500	1.875	1.750	2.000	2.000	2.000	2.000
3	2.125	1.750	1.875	1.875	1.875	1.750	1.875	1.750	2.000	1.875

Aluminum Foil

Tag #

	1	2	3	4	5	6	7	8	9	10
1	2.000	2.500	2.375	2.375	2.375	2.375	2.250	2.125	2.250	2.250
2	2.000	2.000	2.000	2.375	1.375	2.125	2.000	2.375	2.375	2.250
3	2.000	2.000	2.125	2.000	2.000	2.250	2.125	2.000	2.125	2.000

Maximum range testing for PC ProX tags within shielded wallet:

Copper 26 Gauge

Tag #

	1	2	3	4	5	6	7	8	9	10
1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
2	0.250	0.250	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
3	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Steel 26 Gauge

Tag #

	1	2	3	4	5	6	7	8	9	10
1	1.250	1.000	1.000	1.125	1.250	1.000	1.125	1.250	1.250	1.250
2	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500	0.500
3	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Aluminum 26 Gauge

Tag #

	1	2	3	4	5	6	7	8	9	10
1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
2	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
3	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Aluminum Foil

Tag #

	1	2	3	4	5	6	7	8	9	10
1	0.625	0.750	0.875	0.875	0.875	0.500	0.250	0.250	0.250	0.250
2	1.625	1.625	1.750	1.750	1.750	1.500	1.625	1.750	1.750	1.625
3	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

User Awareness of Personal Information Security on Social Networking Websites

Kristina Durivage
Winona State University
P.O. Box 5838
Winona, MN 55987
1-651-238-3605

kdurivage@gelicia.com

ABSTRACT

This study sought to quantify how concerned social network users are with their personal information and how knowledgeable they are about what can be found about them with an automated tool. We looked at previous studies for trends regarding what information users provide, and for information about what combinations of information that can lead to security issues. We were unable to surprise subjects with finding, but attribute this to a misunderstanding of an individual's online social network and a questionnaire that allowed for hindsight bias. We feel that with further refinement of the study, these can be eliminated to find more data about a user's awareness of their privacy.

Categories and Subject Descriptors

H.4.3 [**Information Systems Applications**]: Communications Applications

General Terms

Security, Human Factors, Legal Aspects.

Keywords

Social networking, personal information, security

1. INTRODUCTION

The development of the Internet was led with the idea of a "global information space" that would allow people to link computers together and share information. In recent years, developers have taken this idea to the user level by developing social networking websites. This allows people to create profiles about themselves and link to people and places they associate with. The two major players currently are Facebook[1] and mySpace.com[2], averaging over 32 and 65 million visits respectively for December of 2007 [1]. This growth in popularity has also led to interesting questions regarding security and privacy among users, as different sites encourage different standards of privacy [2]. Despite reports of social networking usage leading to unintended consequences such as identity theft and stalking [3], the numbers still grow- more than 14 million pictures are added daily to Facebook's platform, along with 250,000 new registrations a day [4]. Social networking is catching on even quicker with young people – 49% of English children between the ages of 8-17 have an online profile, and an even greater percentage use social networking sites to make friends [5].

Much of the data displayed on social networking websites can be used for malicious purposes. Among college students, real world stalking can be facilitated by knowing the user's current residence and at least two classes. Cyber stalking can be done much more easily. The AOL instant messaging (AIM) client allows someone to add another to his or her contacts and access to the other's availability, without that person's permission or knowledge. This means that without the proper privacy options set, cyber stalking can be done just knowing a person's AIM screen name. Knowing someone's zip code, gender, and birth date can allow them to be identified across multiple databases, and a social security number can be determined with a birth date, hometown, current residence and current phone number [2].

Despite these risks, people join social networking websites specifically for the purpose of meeting others. A positive correlation exists between availability of personal information and how many friends a user has, suggesting a user values finding "common ground" with others and facilitating friendships before worrying about malicious use of their information [6]. The appeal of making new friends was shown in a study done by the security firm Sophos, who showed that 41% of users would give personal information to a fake profile that added them as a friend first, indicating that they had something in common. Of these respondents, 78% provided their current address or location [7].

On Facebook specifically, users are encouraged to sign up under a certain network (often a metropolitan area, school, or company), and security defaults allow others under the same network to see more of a user's information. In a study done in 2005 within one network, it was found that 50.8% of people listed their current address, available to anyone within that network even if the two didn't know each other. This study noted that "it would appear that the population of Facebook users we studied is, by large, quite oblivious, unconcerned, or just pragmatic about their personal privacy" [2].

The information people disclose about themselves has been looked at in areas other than academically. One company is working on a search engine that begins by getting a name of the user, crawling social networking sites for that user's friends, and giving precedence to search results liked by the people in that user's social network [8]. Artists with an interest in portraying the abundance of information available with technology have explored privacy as well. The art installation iTea was premiered at a conference where members were asked to provide personal

[1] Http://www.facebook.com

[2] Http://www.myspace.com

information when they registered. This information was placed in an RFID tag embedded in their conference badge and could be dropped in a teacup in the middle of a table with an LCD screen under it. Information would be found out about them from what was on the RFID tag and displayed on the tea table to inspire conversation [9]. Running on the same premise, this study aims to better quantify and define how concerned users are with their security online.

2. METHOD

2.1 Setup

Three things were developed for this study: an initial questionnaire to get preliminary data and guide development, a program to find information about a person and a final interview session to use the tool and gather responses.

2.1.1 Exploratory Survey

The first questionnaire had three questions. The first two questions were written to differentiate what information people would feel comfortable giving to a stranger on the internet versus someone who was closer to them. A list of options was provided using categories of information similar to [2] and [6]. The questions were:

-What information are you likely to give to someone you just met on the internet?

-What information are you likely to give someone online that indicates they work for the same company and/or goes to the same school as you?

The third question listed many popular social networking websites and asked subjects to identify those in which she or he has a profile. This was written to provide evidence to the popularity of certain websites among subjects and ascertain what websites the program should focus on.

The survey was delivered to 50 people on a local college campus. No demographic information was taken, but the location of delivery was a weekday evening near a campus cafeteria, so it would be a fair assumption that most subjects were college students between the ages of 18-28.

2.1.2 The Program

A program was developed in the Java programming language to extract personal information from social networking profiles. Up to five screen names would be provided, and the program would search the internet (using Google) for websites that contained the screen name and whose domain fell under a list of supported domains. Based on our results from the initial survey but limited by some of the privacy limitations in place on Facebook, supported domains for this experiment were limited to two mySpace domains, Digg.com and Youtube.com. The generator would use the internal structure of the page and regular expressions to extract personal information and return it as a table. These returned tables would be used to create a generated profile. See Figure 1 for a diagram of the program's generation process. While the program is currently limited, it is extendable. Other search engines could be integrated to add to the number of results found, and other websites could be added by adding the domain to the list of supported domains, and creating a process to extract information. The program runs as a Java Servlet, and in this experiment, was run on Apache Tomcat 4.1.

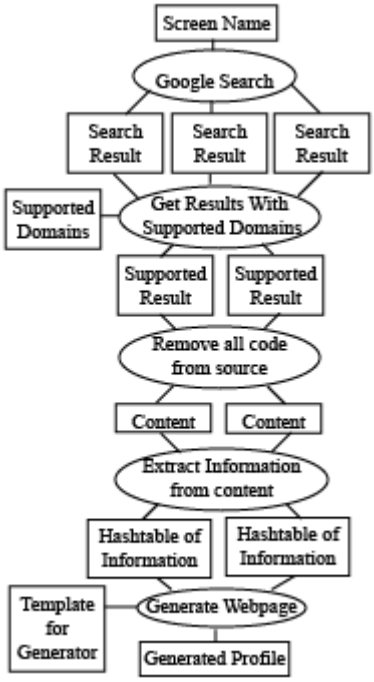


Figure 1 – Process Diagram of Generator

2.1.3 Final survey

The final survey was conducted using similar questionnaire as the first, only it delivered more as a structured interview so researchers could gather feedback about subject's security choices and ask questions based on responses. The first two questions were the same as the initial survey only as an open ended question instead of being provided with discrete answers. Subjects were asked to elaborate on behavior. The third was "Given what information you've provided, what would you expect someone to be able to find out about you?" The program was then run with any screen names given, and subjects were asked to confirm the data was accurate. This approach was meant to gather more information from fewer subjects so this survey was only performed on ten people. The delivery of the survey was changed from the exploratory survey to receive more thorough answers in a manner not possible with static surveys.

2.2 Results

Results of the initial survey were consistent with what was found in [6]. 80% of those surveyed indicated that they would give more information to someone who indicated they were near them. Specifically, a stranger may indicate nearness by bringing up local landmarks, or a plausible story the subject would have no reason to distrust. The highest categories of information that people were willing to provide to anyone was gender and AIM screen name. Between strangers and strangers in close proximity, the largest differences in disclosed information were in the full name, high school, and address categories.

The third question showed the accuracy of national statistics: only two people out of those surveyed didn't have a Facebook profile, and a little under half didn't have a mySpace profile. Numbers after that were much smaller-the third largest was YouTube[3] with 15 people, followed by Yahoo Spaces[4] with five and Digg[5] with four.

[3] Http://www.youtube.com

[4] Http://360.yahoo.com

[5] Http://www.digg.com

Results of the final survey showed some problems with our approach to measurement- our program was unable to find any information about a subject they did not expect. Our team attributes this to a lack of understanding about user's social networking choices and hindsight bias rather than a lack of a problem.

3. ANALYSIS

There are many factors we believe influenced our findings.

3.1 A Personal Social Network is Not Scale-Free

Our program operated under the assumption that a user's online social network would act as many real world networks and fall under the scale-free network model. In scale-free networks, as nodes are added, the probability a node is linked to another depends on how many other links that node has. Some nodes have many links and are termed a hub. When a node is added, the probability it will be linked to that hub is much higher than the probability of it being linked to only another, less linked node. An often used example is the internet in general, where each node is a website. Search engines like Google act as the hubs, with many more links to other websites than the average site has to other sites [10]. This is not the case with one person's social networking activity for several reasons. For one, many social networks do not allow search engines to crawl their pages, and thus the traditional hubs on the internet don't work. Users also don't tend to explicitly link their various profiles to each other. They may end up implicitly linking sites to each other by using one email address to register or choosing one screen name across websites. Hubs may exist within Facebook due to the ability to add applications that provide information from other sites, but his has yet to be explored. Secondly, most users do not join a lot of different social networking sites, so network may only be one or two nodes- not enough to establish a network model. This may change as popularity of various social networking websites broadens.

3.2 Non-Structured Content

A second source of trouble with our project was getting data from profile sites. Most sites have a very strict structure that allow for little personalization, but mySpace.com profiles are very flexible. This allows for people to elect to leave out information about themselves, add in more information, and change the style of their pages with their own code. Since mySpace was the second most used website from our surveys and profiles contain many important pieces of personal information about a user, they were a main focus for our program. We found irregularities in the code, however, that made parsing difficult. We were able to find where most of the data asked about in the survey would be on a page and how it would be structured and display it in a table for easy viewing. However, if users opt to place their personal information in freeform sections, it would be much harder to generate that data. Outside of mySpace, social networking sites that are focused on a certain area- like videos or news articles, allow users to omit almost all information about themselves, and most elect to. The data that is contained offers little that can be used maliciously on its own. It could be used with more personal data for social engineering purposes.

4. CONCLUSION

Our study found that users are aware of the information they provide on social networking websites. They are often very restrictive when disclosing information when involved in an active chat conversation, but tend to be more open when filling out a profile that can be found through searching. This doesn't necessarily lead to surprise, as they recall filling out the profile and providing information. It is, however, a discrepancy that warrants further research.

Overall, data was more difficult to find than initially thought. Our program depended on a more tightly woven network model, where web sites of interest are accessible through a few widely known hubs. Facebook, the most popular social networking site in this area, does not allow its pages to be crawled and most profiles can't be viewed unless a user is logged in and certain location criteria are met. Fake profiles and automatic scripting violate Facebook's Terms of Use [11], so gathering information through Facebook went out of the scope of this study. This does not mean, however, that Facebook profiles are safe. There are three types of profiles on the Facebook network, most users falling under one of the first two:

-Users may belong to a specific network that they only join by having access to a particular email domain. For example, a user can join the Winona State University network by having an email that ends with winona.edu. Facebook privacy defaults allow most information about a person to be visible to others within their own network [12].

-Users can join an area (such as Minneapolis/St. Paul) network with any email.

-Users may join Facebook and not be under any network.

Friending strangers can be a dangerous way to harvest information, and as area networks can be joined by any email, creating a series of fake accounts to various area networks could gather personal data on anyone within that area network whose privacy settings allowed it. Hackers have also found ways around privacy restrictions, resulting in personal photos from celebrities to Facebook CEO Mark Zuckerburg to be released [13]. Users should be warned against having information available to anyone besides who they know, and warned about possible attacks to access information they specify as private.

4.1 Future Work

Future work needs to be done in broadening the scope of the research. While many social networking websites prohibit running automatic scripts on their site, it has been arranged in the past with site owners [6] and could perhaps be done again. Also, future research depends on the trends of users. At the present, results from our preliminary survey show a very narrow range of social networking sites used. This could be due in part to the narrow sample of people we gathered data from – various locales of the world have culture specific social networking sites, as do different ethnic groups and people who are participating in social networking for varying reasons.

Another way our program could be improved is by having more iterations. For example, if additional screen names are found that do not match those given, searches can be done on those until all

possible searching has been exhausted. This requires more of a sophisticated crosschecking algorithm than what is present.

Further, our researchers felt that the questionnaire/interview process was too explicit, and often caused users to think more about what they have made available rather than what they really feel comfortable with people knowing. We gathered this by the discrepancies that occurred between what the subject would provide a stranger and what they expected could be found out about them. It would be reasonable to assume that someone who is private with their personal data upfront would limit the information available through a simple search, but this is oftentimes not the case. A more obfuscated study may help prevent this form of hindsight bias.

Finally, use of a questionnaire often leaves out over other differences that can cause different privacy levels. One male subject, when asked what information he would give to a stranger online, remarked it depended greatly on gender- "If it was a guy, I would tell him to go away, but if it was a female, I would tell her more." This kind of thing is never addressed in a questionnaire and may play more of a role of determining how private people are than is given credit.

Use of this information could help bridge the gap between users' willingness to participate in social networking websites and security experts who warn not to. The user's mindset of "it can't possibly happen to me" applies to their involvement in social networking, and so reports of identity theft, termination of employment and other interference between their online life and offline life are often brushed off. Showing people how easily their information is available, the minimum amount of information that is needed for malicious purposes, and the consequences of malicious actions with their data would help raise awareness that for every new friend made by showing a common ground, new enemies could be made as well.

Source code and documentation for the project is available at [14].

5. ACKNOWLEDGMENTS

The author would like to thank Dr. Joan Francioni, Dr. Nicole Anderson, Dr. Tim Gegg-Harrison, Dr. Gerald Cichanowski, Dr. Drakoski-Johnson, Dr. Peter Sternberg, Chris Burg, Kelly Torkelson, Andrew "Bailey" Biermaier, Heather Espersen, Wade Hanson, Chris George, the teachers of America and the authors of the first Amendment.

6. REFERENCES

[1] Complete, Inc. http://siteanalytics.compete.com/ Accessed 2/3/2008.

[2] Gross, R., Acquisti, A., and Heinz, H. J. 2005. Information revelation and privacy in online social networks. In *Proceedings of the 2005 ACM Workshop on Privacy in the Electronic Society* (Alexandria, VA, USA, November 07 - 07, 2005). WPES '05. ACM, New York, NY, 71-80. DOI= http://doi.acm.org/10.1145/1102199.1102214

[3] Kornblum, J. Marklein, M. What You Say Online Could Haunt You; Schools, Employers Scrutinize Social Websites Such as MySpace and Facebook. USA Today, March 9th, 2006: 1A.

[4] Statistics. http://www.facebook.com/press/info.php?statistics Accessed 2/5/2008.

[5] Waters, Darren. 2008. Children Flock to Social Networks. BBC News. http://news.bbc.co.uk/2/hi/technology/7325019.stm Accessed 4/8/2008.

[6] Lampe, C. A., Ellison, N., and Steinfield, C. 2007. A familiar face(book): profile elements as signals in an online social network. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (San Jose, California, USA, April 28 - May 03, 2007). CHI '07. ACM, New York, NY, 435-444. DOI= http://doi.acm.org/10.1145/1240624.1240695

[7] Sophos Facebook ID probe shows 41% of users happy to reveal all to potential identity thieves. http://www.sophos.com/pressoffice/news/articles/2007/08/facebook.html. Accessed 2/5/2008

[8] Naone, Erica. 2008. Social Search. Technology Review. http://www.technologyreview.com/Infotech/20138/?a=f Accessed 4/23/2008.

[9] Blaaw, D., Borra, E., Kousemaker, D., van Oosterbosch, D., Trifa, V., and Weltevrede, E. 2008. iTea: cosy tea-table with "facts" about you. http://www.mediamatic.net/artefact-22745-en.html. Accessed 3/24/08.

[10] Barabasi, A., Bonabeau, E. Scale-Free Networks. Scientific American, May 2003, 50-59. http://www.nd.edu/~networks/Publication%20Categories/01%20Review%20Articles/ScaleFree_Scientific%20Ameri%20288,%2060-69%20(2003).pdf Accessed 4/20/2008.

[11] Terms of Use. http://www.facebook.com/terms.php. Accessed 4/8/2008.

[12] Privacy Policy. http://www.facebook.com/policy.php. Accessed 4/8/2008.

[13] Havenstein, Heather. 2008. Update: Facebook fixes Security Lapse that Exposed Photos. *ComputerWorld*. http://www.computerworld.com/action/article.do?command=viewArticleBasic&articleId=9071458. Accessed 4/8/2008.

[14] Senior Seminar Project Source. Gelicia.com. http://www.gelicia.com/SenSem/source.zip

Automation of Chord Identification in Tonal Music

Michael Merkouris
Michael Merkouris
Winona State University
Department of Computer Science
P.O. Box 5838
Winona, MN 55987
mmerkouris06@winona.edu

Abstract
Computers have many practical applications that simplify our daily lives. One way in which computers achieve this is through automation of routine tasks. We believe that the harmonic analysis of music, as it is taught in the classroom to first-year college students, provides a mechanism for chord identification that yields itself kindly to computer automation. The goal of this research paper is to describe and implement a system that is designed to automate the task of chord identification in tonal music.

Keywords
music theory, jfugue, tonal harmony, Java, chords, triads, automation

1. Introduction

1.1 Automation of Chord Identification
Computer automation greatly simplifies our daily lives by relieving us from the need to perform certain mundane tasks. These tasks are usually very systematic. Computers have become an invaluable research tool for many academic fields. Music theory is one such field in which computers have had a profound impact on research performed [7].

Tonal harmony is considered a subset of the body of knowledge called music theory. Training in tonal harmony lies at the core of a majority of the music degree programs offered by higher education institutions. In music theory courses, students often learn a systematic way of analyzing the harmonic form of music. Such an analysis requires identification of chord structures within the music.

It is our belief that the systematic methods of chord identification, taught to first-year music theory students, lends itself very kindly to computer automation. The purpose of this project is to discuss how to apply computers to the task of chord identification, for the purpose of harmonic analysis of tonal music.

The aim of section 1.2 is to identify practical applications of the desired system. Section 1.3 then defines the scope of the project. Section 2 offers a discussion on related work in the area of computer automated harmonic analysis. Section 3 describes our method for the implementation of the system. Section 4 is an overview of the results and analysis of the system verification. And Section 5 offers concluding remarks.

It should be noted that this paper assumes that the reader has a minimal understanding of standard music notation practices. The reader should also be familiar with musical terms such as pitch, intervals, scales, triads and chords. If the reader is unfamiliar with any of these areas of music theory, it is recommended to read the introduction to these elements in Appendix A, which is located in section 9.1. Appendix A will provide an introduction to the elements of music that will help the reader to more easily understand the content presented in subsequent sections of this paper. Appendix B, found in section 9.2, provides data about the system verification. The material found in this section illustrate the system input and output.

1.2 Practical Applications
An automated system for chord identification has numerous practical applications, primarily in the field of music theory. One such application of this type of system is to assist students in their education of music theory. As a Music faculty member at the University of Illinois at Urbana-Champaign, Heinrich Taube states his observation that faculty members are experiencing a greater teaching load than in previous years while incoming freshmen seem to be less prepared for their courses in music theory [7]. Taube believes that because this generation of students has increasing access to computers, a system with the ability to automate the process of harmonic analysis of tonal music can serve as an extension of the classroom. When students perform harmonic analysis exercises, they can get immediate feedback from the computer system. In this situation the course instructor does not need to intervene in order for the student to receive critical feedback.

Another possible application for such a system in the field of music theory is in the realm of research. A system for chord identification can be used as a means to automate the mundane task of chord identification within a musical work, for the purposes of further analysis of the work. When a music theorist wishes to analyze the harmonic form of music, historically they would need to manually perform such identifications. This process is pretty straight forward for compositions that are tonal in nature and follow good compositional practices. For musical compositions that are more complicated in nature, the task of manually identifying chords may be rather time consuming. A researcher may use the analysis output of the automated system as a basis for a much more detailed analysis of the music.

1.3 The scope of this Project
It is our intention to implement a system to automate the process of creating a harmonic analysis of a given piece of music, whose elements follow a specific set of criteria. The said criteria are as follows:

1. All notes contained within a chord have the same rhythmic duration
2. The chords are diatonic to the key signature
3. All chords can be expressed in a reduced form as triads or seventh triads

4 The tonality can only be major or minor

By implying the first restriction on the music, the task of identifying chords is significantly simplified. In this context, the musical example will offer the chords as blocks of data. Each chord can then be processed with little regard to its rhythmic duration. This restriction precisely states that acceptable musical examples will have no overlapping of notes, which start and/or end at different times.

The second restriction on the music for this system says that all chords contained within the work are diatonic to the key signature of the musical work. This means that for every chord in the musical work, none of the pitches have been altered in any way. Each of the chords in the musical work will then be built using only notes that are native to the key signature of the work.

The third restriction on the musical work is that each chord in the work must have either a triad representation or a seventh triad representation, in which all pitch classes contained in the chord must be present. This means that no chord shall contain a note whose pitch class is not native to the chord. This kind of note is called a non-chord tone and it is used sometimes in practice to make the musical work sound more interesting. For the sake of simplicity, we will not analyze music that makes use of non-chord tones.

The fourth restriction is that only musical works composed using pitches native to either a major or minor scale are valid. This will provide consistency when we wish to define the relationship of a chord to the tonal center of the musical work.

Using the above-stated criteria, it was our goal to develop a computer program to effectively identify chords within a musical work and express the chords in terms of their relationship to the key signature of the musical work. The information about the key signature of the musical work is offered as input to the system. This project was carried out with the belief that this system can be implemented to accurately identify chords within a 90 percent degree of success. As a means to measure such success, our musical input was restricted to musical examples found in published, peer-reviewed music theory textbooks, which are accompanied by a preexisting harmonic analysis. The system takes as input a representation of the musical work along with information as to the tonal center of the musical work. The resulting output from the program is compared to the preexisting analysis of the music. Our success rate is calculated as a ratio which describes the number of correctly identified chords in the musical example divided by the total number of chords in the musical example.

2. Related Work

There are numerous examples of research performed in the area of computer analysis of tonal harmonic music. In some earlier work in this subject, Terry Winograd applied theory of linguistics to the harmonic structure of tonal music. This work was based very heavily on the works of linguist Noam Chomsky, in particular a work of Chomsky's titled *Syntactic Structures*. Winograd noted similar features between music and natural language. This prompted the notion of treating the computer processing of music similarly to the parsing of natural language. Winograd proposed a systemic grammar for parsing musical works. The grammar was implemented in a LISP program and was used to perform a chord identification analysis on chorales written by J. S. Bach [9].

Other research performed by Stephen Somilar is based on the works of music theorist Heinrich Schenker and linguist Noam Chomsky. Somilar draws upon similar ideas between Schenker and Chomsky. Schenker proposed a method of harmonic analysis which makes use of a musical construct called a proto-structure. This musical proto-structure can be expanded expand into a musical composition through a set of "rewriting rules" which Schenker defined in his work. Somilar saw a similarity between Schenker's notion of a musical proto-structure and Chomsky's notion of a base component, which is part of a natural language sentence [5]. Chomsky offers a set of transformations, similar to the rewriting rules offered by Schenker. Using the transformations on the base component can in return derive a surface structure of a natural language sentence. Somilar believes that the derivation of a musical composition from a musical proto-structure is very similar to the process of deriving a natural language surface structure from a base component. His research concludes that transformational grammars can serve as a basis for implementing a system to perform Schenkerian analysis of tonal music [6].

James Meehan offers some insight into using artificial intelligence to develop a system for the automatic analysis of tonal music. He discusses the drawbacks of using the linguistic approach to find the musical proto-structure of a musical work. His argument against using linguistics is that that using that it is easy to make sense of sentences that are not grammatically sound and thereby lessening the effectiveness of the grammar. Meehan ultimately determines that the best approach to analyze music with the computer is to use artificial intelligence. Meehan suggests that we look to music theory textbooks, and read the material with a mindset to express the ideas within the text in a computer program. This is his alternative method to develop a system to perform analysis of tonal music.[2].

3. Methods

3.1 Representations of Music

To develop the system that takes music data as input and produces an analysis of chords as output, it was first necessary to decide how the musical data should be represented. There are a number of options available for such a task. One possible way in which to process musical data is through the representation of the graphically notated musical score as a digital image. There are examples of commercial software, such as Make Music's *Finale music notation* software suite, which allow a user to scan a piece of graphically notated music. The software will then process the digital image and analyze each note represented in the score and convert it into an editable form. This style of musical data processing is very complicated and is currently in state where it is not very reliable. Using this form of input may also provide significant overhead in this project. This form of technology is more useful for software that is marketed toward consumers who wish to edit and print sheet music that they already have in printed form. This is clearly beyond the scope of this project. For this reason and for the purposes of keeping the nature of this project simple, we did not use the graphically notated musical score as input to our system.

Another option for processing musical data is to use the computer to analyze the sound of the music in real-time as it is performed. Music is a temporal art form, and this process of music analysis relies on the temporal aspect of the music. There are numerous

examples of studies that use sound data to produce a harmonic analysis of music. This form of analysis of music is very unique, as it processes the music in the way the way we naturally perceive it. However, it is also a very complicated way in which to analyze musical data. There are examples of commercial software packages which have the ability to recognize musical chords based on the harmonics produced by the sound of the music. An upcoming release the software package *Melodine*, created by the German company *Celemony Software GmbH*, will feature a new tool called direct note access. This feature can analyze the sound data of a musical recording and effectively identify each individual note played throughout the recording of the music. Even though multiple notes were played simultaneously, the direct note access feature enables the software to identify each note in terms of its pitch, duration and articulation. This software also provides a graphical interface with which the musician can identify and change properties of any of the notes that were played. The implications of such technology are that if a single note was not articulated correctly, or the wrong pitch was played during the course of the recording, the musician can use this feature to correct the mistake at a later time. Likewise the duration of each note can also be easily edited. In some ways, this technique of processing musical data represented as sound data is more complicated than processing digital images of a musical score. Because this technology has a very comfortable place in the market of consumer-based music editing software and because of the complexities associated with this form of music processing, this method of musical input was not used for the purposes of this project.

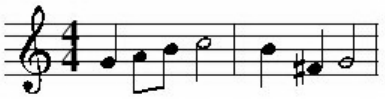
A third option that was considered was to determine a text-based representation of music to act as input to the automated system. This text-based representation would need to be well-formed, in order for it to capture critical details about the music. It should be reiterated that we are only interested in a small set of the information provided by the musical score. Because of this fact, the text-based representation seems like the best approach within the scope of this project. In the next section we will introduce the exact representation of musical data chosen for this project.

3.2 JFugue API

Jfugue is an open-source Application Programmer's Interface (API) written in Java that is designed to simplify the task of programming Musical Instrument Digital Interface (MIDI) applications. Jfugue does a very good job of abstracting away from the complexities of the java MIDI library by offering a rich set of classes that can be used to program computer music in a way that is similar to the practices of representing music in a musical score. One of the most attractive features of Jfugue is its notion of a musical string. A musical string is a way to represent musical score data in the form of a string. This feature alone made the Jfuge API very attractive for the purposes of this project. The musical string is able to capture relevant information about the notated music, which simplifies the process of producing a harmonic analysis.

Figure-1 shows a graphically notated musical example with its equivalent musical string representation directly below. The musical string in this example contains seven notes. Each note is represented as its own entity in the musical string format by using white-space as a delimiter to make the distinction between notes. Each note in the musical string format has three parts that describe different properties held by the note.

The first part is the pitch class of the note. If we examine the music in its notated-score representation, we can see that the first note is located on the G line. Likewise, the first note in the musical string starts with G. So the first character of a note in its musical string format describes the pitch class of that note. It should also be noted that the sixth note in this example is an F-sharp. The musical string accounts for this property by adding a '#' character after the pitch class label. Likewise, if a note was represented as a flat, the musical string representation would include the character 'b' to indicate it as such.



G4q A4i B4i C5h B4q F#4q G4h

Figure-1

The second property of a note which the musical string defines is the octave register in which the pitch occurs. By examining the staff on which the first note is located we can determine that it is located in the octave 4 register. The musical string makes this distinction by labeling the note with the number 4. We can effectively describe the G note in the fourth octave register using the notation G4. Likewise, the F-sharp in our example would be labeled F#4 in its musical string format.

The third property of the note described by the musical string construct is the rhythmic duration of the note. This is done in a way that mirrors the convention used in standard music notation of Western culture. Since the G described earlier is notated as a quarter note, the letter *q* is used in that note's musical string representation. Figure-2 illustrates the characters used to indicate other significant note durations.

After careful consideration of the musical string it becomes clear why this representation is so effective. It provides a mechanism for representing a musical score as a string that mirrors the conventions used in the graphically notated musical score. So far we have discussed how to convert a single melodic line into a musical string format, but now we must consider how we can convert a more complicated musical texture into an equivalent string format.

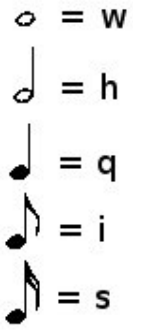


Figure-2

Since this project is aimed at chord identification, we will consider the example shown in Figure-3. The music in this example differs from the previous example in that this musical example has chords. This means that we must have a way to indicate that multiple notes are to be sounded together. We can use the techniques described earlier to identify each of the notes, but we must have a way to indicate each chord as a collection of notes. The musical string format uses the character '+' in the place of the white-space delimiter to indicate the chord relationship between notes. Consider the first chord in the example. It is made up of the notes G, B, and D. Each note has a quarter note duration. The G and B are both in the fourth octave register, and the D is in the fifth octave register. Using the techniques described earlier, we are able to convert each note independently into its musical string representation. Using the '+' character to indicate that these three notes sound as one unit we can derive the following musical string representation "G4q+B4q+D5q".

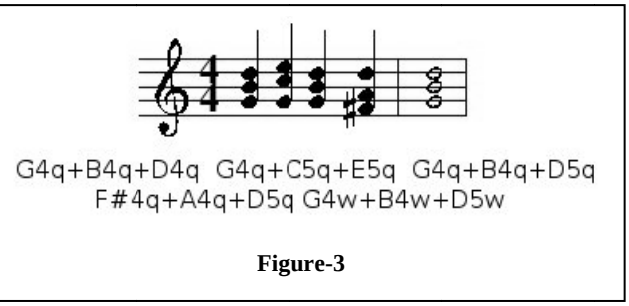


Figure-3

3.3 Implementation of the System

The process of chord identification described in this section was designed with the goal of modeling the process of chord identification as taught to first-year music theory students. It is a systematic way of identifying chords in terms of their root, their inversion, and their significance within their key signature. The ultimate goal of this process is to automate the process of chord identification in a systematic way.

Using features of JFugue, we were able to implement the process of chord identification using five classes. This process has two phases. The first phase is to identify the inversion of the chord. When the inversion is realized, we are be able to effectively identify the root of the chord. The second phase is to identify the relationship of the chord's root to the tonal center of the musical work. Based on this relationship, and the scale on which the work is composed, a resulting roman numeral analysis is generated to indicate the quality of the chord and the relationship between the root of the chord and the tonal center of the musical example. Information about the inversion is also attached to the output analysis of the chord.

The first class we shall discuss is the *Chord* class. This class serves as an abstraction of a chord. We defined a chord as a collection of notes and a set of methods to return relevant information about those notes. Upon initialization a *Chord* object takes a *String* element as input. The input *String* follows the format of the JFugue musical string as described earlier. Because the musical string describes a chord in terms each of its notes and uses the '+' character as a delimiter between notes, the musical string is processed in order to be further broken up into discrete notes. Each note is then stored as a single element in an array of strings. The notes are still in the form of the musical string. This is done to simplify the process of referencing each note individually for further processing.

Upon initialization of a Chord object, a second array is also initialized. This array stores byte values for each note. The byte value is the native representation of the pitches in the *javax.sound.midi* package. The byte values are important, because they assist in the identification of exact intervals between notes. Using byte values as a representation of pitch, it is very easy to determine lower pitches from higher pitches. The byte value representation simplifies the process of determining the distance between notes.

Each Chord object is given an assigned inversion upon initialization. This inversion identification process is defined in the *ChordAnalysis* class. The *ChordAnalysis* class has a static method called *getInversion*, which takes in as parameters the array of byte values and the array of String values which are representative of the Chord object. This method then uses a static method defined in the *IntervalAnalysis* class to realize the intervals between each of the notes contained within the Chord object. Using this data, the *ChordAnalysis* class can make a logical decision as to the inversion of chord. The *getInversion* method then returns a String to the Chord class which identifies the inversion of the Chord object.

As an example of this phase of chord identification we will consider the initialization of a Chord object. Our Chord object will take as input the first chord of our example in Figure-3. The constructor method for the Chord object takes as input the musical string "G4q+B4q+D7q". This String is then separated into three distinct String elements, representing the three notes in the chord, all of which are placed into an array. The resulting array will resemble the form {"G4q", "B4q", "D7q"}. Each of these elements is then translated into a corresponding byte value representative of the exact pitch of the note. The byte values are also stored in an array. The Chord object will then initialize its inversion property by calling the *getInversion* method, passing it the String array and byte array representing its notes. The *ChordAnalysis* class will then use the *IntervalAnalysis* class to determine that the Chord object contains an interval of a third

between the first and second notes, an interval of a third between the second and third notes, and an interval of a fifth between the first and third notes. The chord analysis class will then conclude that a chord made up of those intervals must be in root position. This is then made clear when the representative String, "5:3" is returned to the Chord class, which represents root position.

The preceding example illustrated the first phase of chord identification. This phase was only concerned with identifying the inversion of the chord. The second phase of chord identification is concerned with producing a symbol to identify the quality of the chord, and identify the relationship between the root of the chord and the tonal center of the musical example.

Identification of the quality of the chord is relatively straight forward within the scope of this system. Recall the restrictions placed on the musical input examples within the system. Each chord must consist only of notes that are native to the key signature of the musical work. Because of this fact and because the key signature is taken as input to the system, we can easily identify the quality of each chord with a simple lookup mechanism. To do this we must simply compare the root of the chord to the key signature of the musical example. This is done by using a method called *getRomanNumeral*, which is defined in the Chord class. This method relies on the *NoteAnalysis* class, which defines a static method called *getInterval*. The *getInterval* method a String object as its first argument.. This argument represents the pitch class of the tonal center of the musical example. The second argument of the *getInterval* method is a String object, representative of the pitch class of the root of the chord. The return type of the *getInterval* method is an integer, representing the number which characterizes the degree of the root of the chord and the tonal center of the musical work. This integer is then compared to the key signature. A roman numeral is then printed with inversion information about the chord. This process of identification is repeated for every chord processed by the system.

4. Results and Analysis

The results of the test were overall successful. The first musical example was in the key of Bb Major, and consisted of 17 chords. Out of the total 17 chords, 16 were correctly identified as the music theory text book indicated. We believe the reason that one chord could not be correctly identified was because the system is designed to identify chords that are fully spelled. That means that a C major chord is identified by the presence of the notes C, E, and G. One of the chords in this example was not completely spelled out. As a result, the system was not able to correctly identify the chord.

A solution to this error in the system would include a more elaborate process of identifying chords within a musical work. When a chord is not fully spelled, it is good practice to identify the chord according to the context in which it occurs. By examining the chords before and after the ambiguous chord, an educated music theorist can determine the most logical label for the chord. This sort of logic could potentially be incorporated into this system to avoid such failures. When the system reached the ambiguous chord in the first example, it could have used such logic to examine the chords before and after that chord and determine the harmonic function of the chord in question.

The second example was in the key of B minor and consisted of 10 chords. Out of the 10 chords, 9 were correctly identified according the text book from which the example was taken. The chord that was not correctly identified was a B minor chord. The system identified it as being in first inversion, but the book identified it as being in root position. Upon further investigation we came to the determination that the reason for this discrepancy was that the label in the textbook is erroneous. A root position B minor chord is spelled B, D, F#, but the chord in this example has a D in the bass note. Therefore the system correctly identified the chord, but the text book was in error.

The third example was in the key of F major and consisted of 8 chords. Every chord in this example was correctly identified according to the book from which the example was taken.

The fourth example was in the key of D Major and consisted of 12 chords. Every chord in this example was correctly identified according to the book from which the example was taken.

The fifth example was in the key of G minor and consisted of 9 chords. Every chord in this example was correctly identified according to the book from which the example was taken.

5. Conclusion

We consider the overall effectiveness of the system to be successful. Our goal was to implement a system that could automate the identification of chords in musical input. We wanted the system to be within 90 percent of accuracy. Out of the five musical examples selected and processed, we achieved an average of 96.824 percent accuracy. Each of our individual test cases was within 90 percent accuracy. The success of the system is partially due to the simplicity of the musical examples, described by the constraints on the system. Because the constraints on the system significantly simplified the musical input, the processing of that input was significantly easier. Out of the five examples, there were only two misidentified chords. The reason for the first error is that the misidentified chord did not follow the criteria for allowable input to the system. As a result, the system was not designed to process that input. Upon investigation of the second misidentified chord, it came to our intention that misidentified chord was not correctly identified in the text from which the example was taken. The identification of the chord from the automated analysis was in fact correct.

6. Future Work

The system implemented in this project was designed with simplicity in mind. Such a system is of little use on its own. Future work for this project will involve developing a more robust system for harmonic analysis. This will be accomplished by gradually removing the constraints from the allowable input data.

7. Acknowledgements

We wish to extend our thanks to the department of Computer Science at Winona State University. In particular, we wish to thank the members of the faculty involved in the Computer Science Research Seminar for spring semester 2008. This includes Dr. Nicole Anderson, Dr. Joan Francioni, Dr. Gerald Cichanowski, and Dr. Tim Gegg-Harrison. Thank you all for your support and criticism.

8. References

[1] Kostka, Stefan AND Payne, Dorothy: Tonal Harmony With an introduction to 20th Century Music, McGraw Hill June 2003

[2] Meehan, J. R. 1979. An artificial intelligence approach to tonal music theory. In *Proceedings of the 1979 Annual Conference* A. L. Martin and J. L. Elshoff, Eds. ACM 79. ACM, New York, NY, 116-120

[3] Piston, Walter. Harmony. Third Edition. W. W. Norton & Company, Inc. 1962

[4] Prather, R. E., Harmonic Analysis from the Computer Representation of a Musical Score. Communication of the ACM, Vol. 39, No. 12, Dec. 1996

[5] Smoliar, S. 1976. SCHENKER: a computer aid for analysing tonal music. *SIGLASH Newsl.* 10, 1-2 (Dec. 1976), 30-61.

[6] Smoliar, S. W. 1979. A computer aid for Schenkerian analysis. In *Proceedings of the 1979 Annual Conference* A. L. Martin and J. L. Elshoff, Eds. ACM 79. ACM, New York, NY, 110-115.

[7] Taube, H. Automatic Tonal Analysis: Toward the Implementation of a Music Theory Workbench. *Computer Music Journal* 23, 4 (Winter, 1999): 18-32

[8] White, Gary. The Harmonic Dimension. Wm. C. Brown. Publishers. 1991

[9] Winograd, T. Linguistics and the computer analysis of tonal harmony. *Journal of Music Theory*, Vol. 12, No. 1 (Spring, 1968), pp. 2-49

9. Appendix

9.1 Appendix A

9.1.1 About This Appendix

The purpose of this appendix is to introduce the reader to some concepts associated with tonal harmony. The concepts covered in this appendix are aimed at preparing the reader with enough information on the subject to understand the material presented in this paper.

9.1.2 Identifying Pitch

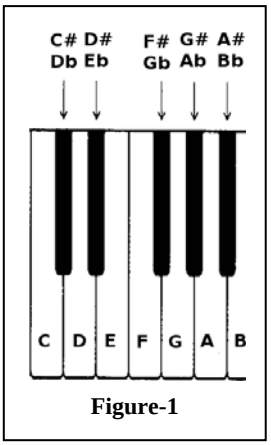


Figure-1

The first element of music which we will consider is pitch. Pitch refers to the highness or lowness of the sound, in terms of its frequency. Faster sounding frequencies result in higher sounding pitches. In standard Western music practice, pitch is described in terms of seven possible pitch classes. These classes are labeled by the letters A, B, C, D, E, F, and G. Formal lessons in basic music theory are accompanied by aural examples of pitches, but because such examples are not possible in this form of presentation, we will consider pitch as it is organized on a piano keyboard. This graphic organization is illustrated in Figure-1. This figure represents the organization of the seven pitch classes as keys on a keyboard. This collection of keys is a single series of the pitch classes, starting with *C* and ending with *B*. This is the typical organization of pitches. A piano keyboard usually consists of several adjacent series called octave registers. Figure-1b shows the organization pitch classes as keys on a keyboard spread out over three octave registers. Each octave register follows the same organization of pitch classes as depicted in the Figure-2. As we move toward the right side of the keyboard, the pitches occur in higher octave registers. Likewise, the pitches contained within the higher octave registers are sounded at higher frequencies than those of the lower octave registers. Similarly, as we approach the left side of the keyboard, the pitches occur in lower octave register and these pitches sound at lower frequencies. A full keyboard consists of eight full octave registers. We can identify precise pitches from any of these octaves by using the naming convention which identifies the pitch class followed by the number of the octave register in which the pitch occurs. For example, the note *G5* represents the *G* that occurs in the fifth octave register. To find this specific pitch, we can count the fifth *G* key starting from the left side of the keyboard. It is important to point out that there are white keys and black keys on the keyboard. Some white keys are adjacent to other white keys, while some are separated by black keys. This significance will be explained in section *9.1.6.*

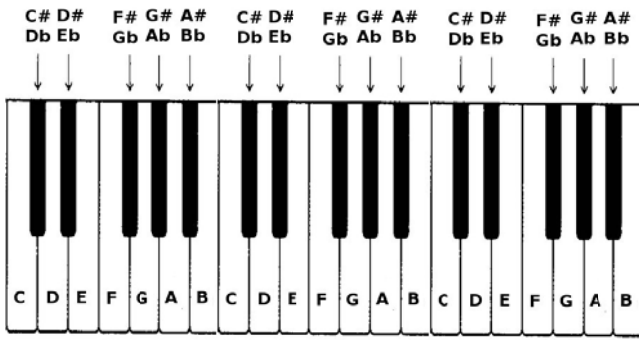


Figure-2

9.1.3 Notating Music

We will next consider the notation of music. The practice of notation of music in Western culture has evolved from early forms, used in Medieval times, into the sophisticated system that it is today. We will consider musical notes in terms of two important properties. The first of these properties is the pitch of the note. The second property is the length of time for which the pitch is meant to sound, known as rhythm. The music notation system used by the majority of the Western world captures these two properties for each note and effectively describes them to the musician using an element called a staff. S. Kostka and D. Payne describe similarities between a musical staff and a graph, "in which time is indicated on the X axis and pitch is shown on the Y axis."[1] A staff is illustrated as a grouping of five parallel horizontal lines between which there are four spaces. Each line and space effectively represents a different pitch. With five lines and four spaces present on a staff, we are only able to represent nine distinct pitches. Because there many more than nine pitches that this notation system must represent, we must use a feature of the notation system called ledger lines. A ledger line is a line that is placed above or below the staff in order to act as an extension of the staff, used to express pitches above and below the pitches on the staff.

Because most musical instruments are able to sound pitches over several different octave registers, it is not sufficient to notate their music on a single static staff. For this reason Western music notation uses the facility of a clef to indicate which pitches are associated with each of the lines and spaces on the staff. Clefs achieve this by matching a single line with a single pitch. All lines and spaces can then be identified by the musician by their vertical position relative to the clef's identified line. Although there are several different clefs used in the notation of Western music, only the two most frequently used clefs will be considered in this paper. The treble clef is illustrated in Figure-3. This clef is sometimes referred to as the *G-clef*, because of two reasons. The first reason is that this clef at one time was illustrated as an ornamented letter G. The second reason this clef is sometimes called a G-clef is because the inner "loop" of the treble clef circles the second line from the bottom of the staff. The treble clef associates this line with the pitch G4.

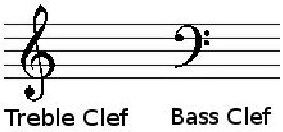


Figure-3

The second clef we will consider is the bass clef. Figure-3 illustrates the notation of a bass clef. This clef is sometimes referred to as the *F-clef*, for two reasons. The first reason is that at one time, this clef was drawn as an ornamented letter F. The second reason for this unique name is because the two dots on the right side of this clef surround the second line from the top of the staff. The bass clef associates this line with the pitch F3.

Using the treble clef to identify the pitch G4 and the bass clef to identify the pitch F3, we can determine the representative pitch for each of the lines and spaces associated with the staff. Some instruments, such as the piano, have the ability to sound pitches over a wide range of octave registers. Music notated for such instruments often uses both the treble clef and the bass clef to indicate which notes should be played. This requires that two staffs are used. Since the piano is a single instrument this notation style makes it clear that the two staffs are associated by connecting them with a vertical line at the left side of the staffs. This is known as a *grand staff*. Figure-4 shows a series of notes on a grand staff, and indicates the pitch name associated with those notes.

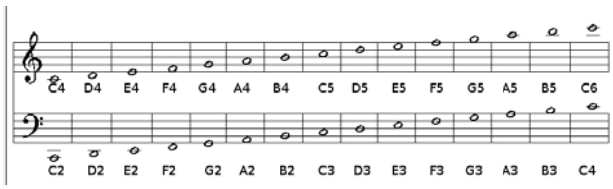


Figure-4

9.1.4 Rhythm

The second property of a note to which the notation system must accommodate is the rhythmic information about the note. As stated in section 9.1.3, the horizontal organization of the notes represents temporal characteristics of the notes. Sequentially sounding notes are placed side-by-side, with each note sounding individually from left to right. This organization convention effectively communicates sequences of notes, but does not indicate the duration of time for which the note is to be sounded. To illustrate this characteristic the music notation system has several different graphical representations of the notes. The different representations do not directly communicate amounts of time for which the notes should be sustained, rather they provide a mechanism to describe how long a note should sound relative to the other notes in the musical work. Figure-5 illustrates some of the possible note representations. The notes shown in this figure are perhaps the most commonly used notes. The names of each of the note symbols indicate the relative rhythmic duration of each of the notes. For example, a whole has twice the rhythmic duration of a half note. Similarly, a quarter note has half the rhythmic duration of a half note. This pattern continues for the eighth note and sixteenth note. The note names are representative of fraction

values. The relationships of each of the notes, in terms of their duration, are proportionate to that of the fraction values indicated by their names.

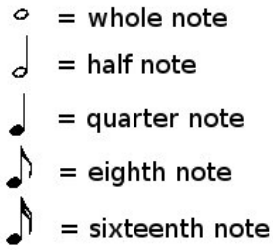


Figure-5

9.1.5 Accidentals

It was stated in section 9.1.2 that there are seven distinct pitch classes. The organization of each of the pitch classes within an octave register on the keyboard is depicted in Figure-1. It should be noted that there are white keys and black keys on the keyboard. Each of the white keys is associated with one of the pitch classes and each of the black keys is associated with two different pitch names.

It was mentioned in section 9.1.2 that some adjacent white keys have black keys between them. It should be noted that each white key on the keyboard is considered to be one step away from its adjacent white keys. However, there are two types of steps possible. The first type of step is called a *whole-step*. The pitches C and D are considered to be a whole-step apart, since they are separated by a black key. This means that there is a pitch that exists between the pitches C and D. The second type of step is a called a *half-step*. The pitches E and F are considered to be a half-step apart, since they are adjacent with no black key between them. This means that there is no pitch between the pitches E and F. We can further generalize this distinction to say that each key on the keyboard is one half-step from each of its adjacent keys. In this description, we do not consider pitches C and D to be adjacent, since they are separated by a black key.

The notation of the pitches represented by the black keys is achieved through the use of symbols called accidentals. The most commonly used accidentals are called sharp and flat. Referring back to Figure-1, we see that the black key between pitches C and D has a dual label. One way we can specify this pitch in the tradition of Western music notation is by classifying it as a raised C, or a C#, pronounced “C-sharp”. This notation specifies that a pitch should be sounded that is one half step above C. The pitch in this example is half way between C and D. An alternative way to specify this pitch is by classifying it as a D ♭ , pronounced “D-flat”. This alternatively specifies that the pitch should be sounded that is one half-step below D. Although it may seem like a trivial detail, it makes a difference in our analysis of the relationship between notes, depending on which way such a note is identified.

The sharp and flat symbols are not limited to the representation of black keys on the keyboard. Since the pitches E and F are one half-step apart, we can alternatively specify the pitch F by expressing it as E#. Likewise the pitch E can be equivalently expressed as F ♭ . The same relationship exists between the notes B and C, because they are adjacent pitch classes on the keyboard, with no black keys between them. The alternative representations of pitches described in this example are rarely used in practice. More advanced music theorists may find practical reasons as to why we should harmonically use such a notation, but such details are beyond the scope of this project.

9.1.6 Intervals

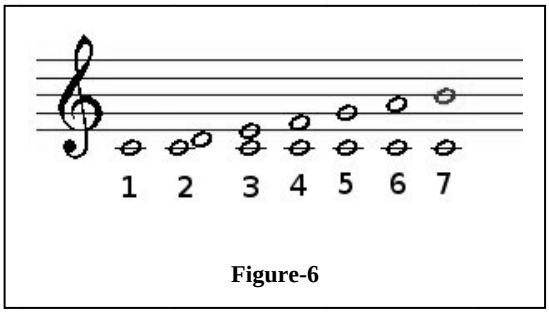


Figure-6

An interval is a unit of measurement which describes the distance between two pitches. A general description of intervals is depicted in Figure-6. The intervals shown in this figure detail the degree of the intervals between the pitch C4 and each of the other pitches within the fourth octave register. This example provides a general description of the distance between each note by representing it as a numeric value. The examples depicted in this figure use unaltered each of the unaltered pitch classes, ones that are not raised or lowered. Notice that the degree of the interval between C4 and itself is measured as a degree of 1. This distance is often referred to as unison. Likewise the degree of the interval between C4 and D4 is considered a degree of 2. It may seem strange to think of intervals in this manner, since it might seem logical to think of the distance between C4 and C4 as zero. Music theorists do not agree. It may be helpful to consider the organization of the pitches on the piano keyboard when determining intervals. When calculating the degree of the interval between pitch x and pitch y, it may be helpful to think of the degree as the sum of the white keys on the piano between and including pitches x and y. As an example, the degree of the interval between the pitches E4 and A4 is measured as 4, since that interval would include the keys E4, F4, G4, and A4. In practice, intervals must express distance between two pitches in a much more precise way. They accomplish this by using s set of modifiers that are attached to a number expressing the degree of the interval. There are three distinct modifiers which we shall now consider.

The first modifier which we will consider is the *perfect* modifier. The perfect modifier can be attached to intervals with a degree of 1, 4, 5 or 8. The perfect modifier is simply a label that music theorists use to describe the nature of the sound produced by such intervals. This means that we can classify the interval between C4 and F4 as a perfect-fourth. When representing an interval with the perfect modifier, the word perfect is often abbreviated by the letter *P*. In other words, the perfect-fourth interval can be abbreviated as P4. Furthermore, we can classify the distance between C4 and G4 as a perfect-fifth, otherwise labeled as P5. If we refer to the keyboard in Figure-1, we can look at the notes C4 and F4 and define some characteristics of the P4 interval. Notice that if we count the number of white keys and black keys between these two pitches, we find F4 is a total of five half steps above C4. Recognizing this distinction will help us to further characterize other intervals. We can think of the interval of a perfect fourth as the relationship between two pitches. The two pitches have an interval with a degree of 4 and they are 5 half steps away from

each other. Likewise in the P5 interval, the two pitches are seven half steps apart.

The next two significant modifiers are called major and minor. These modifiers are directly related, because they are both attached to intervals with a degree of 2, 3, 6, and 7. It is common practice to abbreviate the minor modifier with a lower case *m*. It is also common practice to abbreviate the major modifier with an upper case *M*. Consider the interval between the pitches C4 and E4. This interval has a degree of 3. The two pitches are also four half-steps away from each other. We classify this as a major-third, abbreviated M3. If we were to instead consider the interval between C4 and E ♭ 4, the degree of the interval stays the same, but this time the two pitches are three half-steps apart. This interval is described as a minor third, abbreviated m3. In general we can use the modifiers described above as they are attached to a label indicating the degree of the interval. Figure-7 shows a table of intervals as they are associated with a number of half steps.

Interval	# Half Steps
P1	*0*
m2	*1*
M2	*2*
m3	*3*
M3	*4*
P4	*5*
P5	*7*
m6	*8*
M6	*9*
m7	*10*
M7	*11*

Figure-7

The table relates the abbreviated interval symbol with a number of half steps associated with the interval. We can use this table, and the keyboard depicted in Figure-1 to accurately determine most intervals. For example, upon consideration of the precise interval between D4 and G4, it can be noted that this interval has a degree of 4. It can be further noted that these two pitches are five half steps away from each other. Using this information we can conclude that the two pitches are a perfect-fourth away from each other. It should be mentioned that there are some additional predicates that can be applied to intervals. Such intervals will however not be considered within the scope of this paper.

9.1.7 Scales

Scales are used to define a series of notes with which a musical work can be built. There are several types of scales used in Western music. Only two of these scales will be considered relevant to this project. The two scales are the *major* scale and the *natural minor* scale.

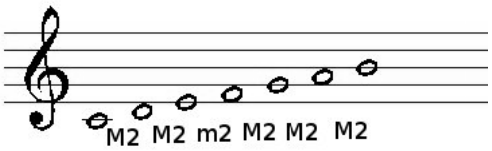


Figure-8

The major scale can be characterized by a series of distinct intervals and a starting pitch. Figure-8 shows each pitch in a major scale built on the pitch *C* spanning over one octave register. The intervals between each adjacent pitch are labeled in the figure. These intervals are what characterize the sound of the major scale. To build a major scale starting with any other pitch, the series of intervals depicted in Figure-8 must be followed. It should be noted that if the first note of the scale were to be translated up one octave register, the distance between the seventh note of the scale and the translated first note of the scale is a minor-second. The major scale in Figure-8 is built on the pitch *C* and is therefore classified as the *C-major* scale.

The idea behind this scale can be used to find all the notes in any major scale built on any pitch. For example, if we followed this pattern to build the major scale starting with the pitch *G* we would find the notes G, A, B, C, D, E, and F#. Notice that the final note in this scale is raised in order to preserve the interval of a major-second between the sixth and seventh notes.

The natural minor scale can be characterized in a similar way as the major scale. Although the natural minor scale appears to be a slight variation on the major scale, the two scales sound drastically different when used in musical compositions.

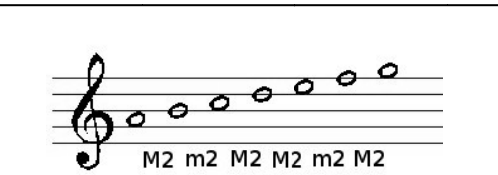


Figure-9

Figure-9 shows each note in the natural minor scale built on the pitch A, with each of the intervals labeled between adjacent pitches. To identify the notes within a scale built on any pitch, it is necessary to find the pitches that exhibit the same series of intervals with the starting pitch. For example, the natural minor scale starting with the pitch B, contains the pitches B, C#, D, E, F#, G, and A. Notice the two pitch classes C and F were raised in this example in order to preserve the interval relationships with the pitch B.

9.1.8 Key Signatures

The tonal center of a musical work can be classified as the key in which the work is written. A work that is composed using the pitches of the C-major scale is said to be in the key of C-major. Likewise, if a musical work is composed using the pitches in the A-minor scale, that work is said to be in the key of A-minor. Key

signatures are a way to communicate the key in which a musical work is written. Key signatures are directly related to scales. When composing music using pitches from scales that frequently raise or lower pitch classes, it is inefficient to notate the sharp or flat symbols every time those particular pitch classes are used. For example, the scale of B-major requires that the pitch classes F, C, G, D, and A are all raised. It is not efficient to use the '#' symbol each time one of these five pitch classes is used. For this purpose it is common practice that music is notated with a key signature. The key signature is usually located on the left side of every staff, illustrated to the right side of the clef, used in the notated music. It is a way for the composer or publisher of music to communicate to the musician that a set of pitches should be sounded as raised or lowered, unless otherwise expressed.

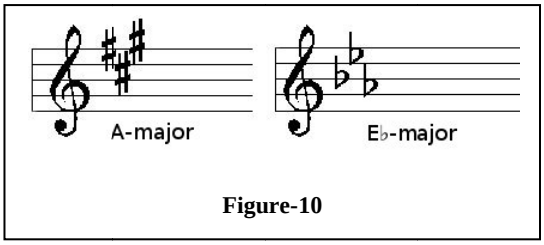


Figure-10

Figure-10 shows two examples of key signatures used in practice. The first example is the key signature for A-major. This key signature requires three pitch classes to be raised. The second example in the figure illustrates the key signature associated with the key of E ♭ -major. This key signature requires that three notes are lowered. It should be noted that even though the key signature indicates altered pitches within a single octave register, this property is implied for pitch classes over all octave registers.

9.1.9 Triads

Triads play an important role in the field of tonal harmony. As the name triad indicates, triads consist of three pitches. Triads are notated as three stacked notes. To build a triad from the bottom-up, we must first find a note to be the bottom note. The bottom note is the most significant note of the triad. It is referred to as the root of the triad. The second note in the triad is an interval of a third above the root. Because of its relationship to the root, this note is called the third. The final note in the triad is placed an interval of a third above the second note of the triad. This note is an interval of a fifth above the root. Because of this characteristic, this pitch is often referred to as the fifth of the triad.

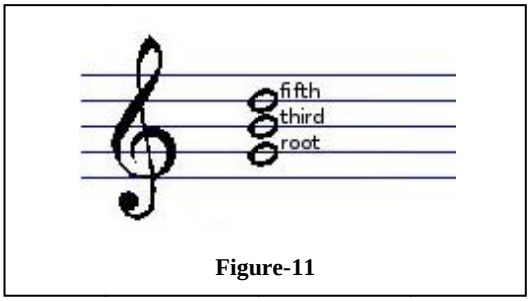


Figure-11

Figure-11 shows the triad built bottom up on the pitch G. In this example G is the root of the triad. Notice that the interval between the root and the third is a degree of a third. Also the distance between the root and the fifth of the triad is a degree of a fifth. In this example, the precise interval between the root and the third is a major-third. The precise interval between the root and the fifth is a perfect-fifth. A triad that consists of pitches with these relationships to the root is classified as a major triad.

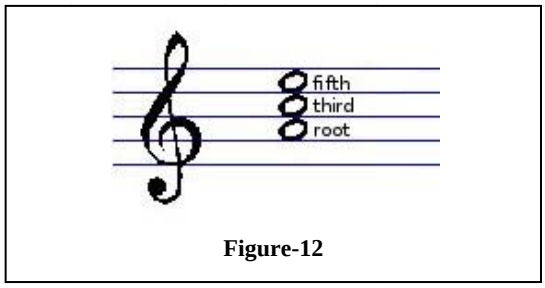


Figure-12

As a second example, Figure-12 shows a triad built on the pitch A. This triad appears very similar to the previous example. The difference between the triad in Figure-12 and the previous example is that the relationship between the root and the third of the triad is a minor-third interval. Although this significance may seem trivial, this relationship changes the classification of the triad. A triad built on an interval of a minor-third and a perfect-fifth is classified as a minor triad.

Because major and minor triads are so similar, it is easy to find one if the other is known. A C-major triad consists of the pitches C, E, and G. This is true because E is a major-third above C and G is a perfect-fifth above C. This triad can be modified to become a C-minor triad by modifying the third of the triad. If this pitch is lowered by one half-step, it becomes the pitch E ♭ . The distance between the pitch C and the pitch E ♭ is a minor-third interval. This illustrates that the C-minor triad consists of the notes C, E ♭ , and G.

A triad is a set of three pitches with distinct properties. Using these properties allows us to determine each of the pitches that belong in a triad. Figure-13 shows alternative ways to represent a triad. The triad in this example is the C-major triad. Notice that the first example follows the convention used previously. The root is the lowest pitch in the triad, and the other pitches are stacked an interval of a third and an interval of a fifth above the root. This representation of a triad is called root position. The label below the triad is a labeling mechanism used. It indicates that the root of the chord is C. The numbers 5 and 3 represent the intervals between C and the other pitches in the triad. A 5-3 label indicates root position.

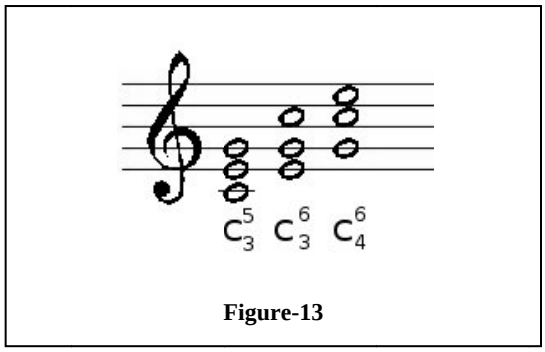


Figure-13

The second representation of the C-major triad in Figure-13 deviates from root position. In this spelling, the root is displaced, such that it is no longer the lowest note. Instead, the third of the triad is now the lowest note. This representation of the triad is known as first inversion. The characteristic of this triad is that the third of the chord is the lowest note, the fifth of the triad is a third

above that and the root is an interval of a sixth above the lowest note. The symbol below this inversion of the triad indicates that the pitch C is still the root of the chord. The numbers 6 and 3 indicate that third of the triad is the lowest pitch and the other two notes are intervals of a third and a sixth above that pitch.

The third spelling of the triad has the fifth as the lowest note. This spelling of the triad is known as second inversion. Like the previous examples, the symbol below the triad indicates that C is the root, and fifth of the triad is the lowest pitch. The number 6 and 4 represent the intervals between the other pitches and the lowest pitch.

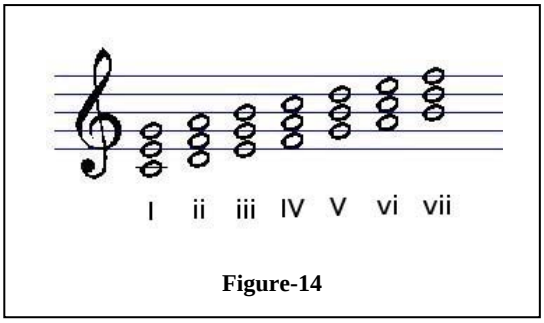


Figure-14

Figure-14 shows a series of triads built on the notes of the C-major scale. Fundamentally, the triads depicted in Figure-14 make up the harmonic building blocks of a musical work composed in the key of C-major. The roman numerals under each of triads indicate the sequential number in which the root of the triad occurs in the C-major scale. Some of the roman numerals are represented as upper-case and some are lower-case. The upper-case roman numerals are used to symbolize the major triads. The lower-case triads are used to indicate the minor triads. These roman numeral will appear exactly the same for all major scales. The quality of the triads, major or minor, will be the same in any major key. It should be noted that the seventh triad is a special case which was not covered earlier this triad cannot be classified as major or minor. In an attempt to stick to the basics with this introduction to tonal harmony the significance of this triad will not be described any further.

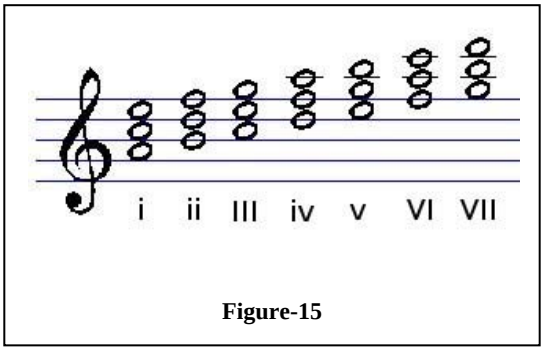


Figure-15

Figure-15 shows the triads built on each pitch of the A-minor scale. The roman numerals used to identify the triads exhibit the same properties as in our previous example. The same exact sequence of major and minor triads will likewise occur in any minor key. It should also be noted that the second triad in Figure-15 a special case, just like the seventh triad in our previous example.

It is noteworthy to mention that triads are sometimes elaborated to include an additional pitch. This pitch is a third above the fifth of the triad. When the pitch is placed at this location, it is an interval of a seventh above the root. This type of triad is classified as a seventh triad.

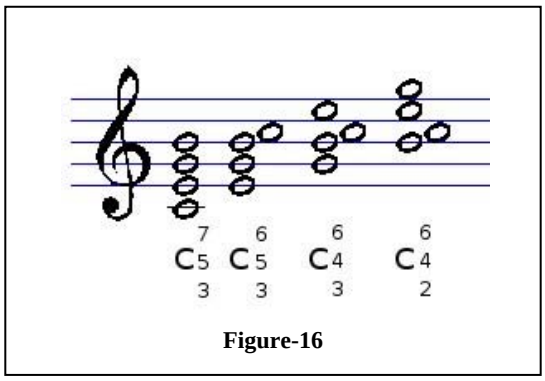


Figure-16

Like the triads in the previous examples, the triads with the added seventh intervals can be expressed in inverted forms. The first illustration of the C-Major seventh triad in Figure-16 shows the triad in root position. The symbol below the staff indicates that C is the root of the triad. The numbers 7, 5, and 3 label this as a seventh triad in root position. This notation expresses that the intervals between the lowest pitch and the other pitches are seventh, fifth and third degree intervals.

The second representation of this triad shows it in first inversion. Just like in our previous example, the first inversion triad has the third as the lowest pitch. The symbol below the staff for this inversion characterizes it as a seventh triad, with C as the root, and pitches occurring at the intervals of a sixth, fifth and third above the lowest pitch.

The third representation of this seventh triad shows the second inversion spelling of it. With the fifth as the lowest pitch, the symbol below the staff represents this inversion using the numbers 6, 4, and 3 to indicate that the other pitches occur at intervals of a sixth, fourth and third above the lowest pitch.

The final representation of this seventh triad shows its spelling in third inversion. Because the seventh triad has one more pitch than regular triads, it has this fourth representation. The third inversion is depicted with the seventh of the triad as the lowest pitch. The symbol below this representation illustrates that the other pitches occur at intervals of a sixth, fourth and second above the lowest pitch.

9.1.10 Chords

Chords are directly related to triads. A chord is a set of distinct pitches. The pitches of a chord are determined by the triad which defines the chord. A C-major chord consists of the pitches found in a C-major triad. The difference between a triad and a chord is that a chord is not limited to three pitches. A chord can be spelled out over several octave registers, doubling pitches in some cases. The inversion of a chord is treated the same as discussed in the previous section with triads. The lowest note is considered when determining the inversion of a chord.

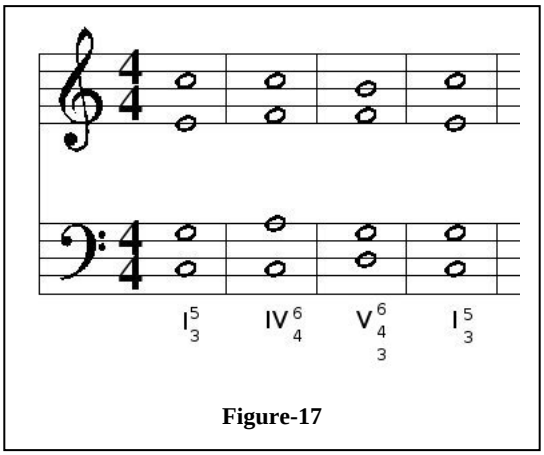


Figure-17

Figure-17 shows several chords in the key of C-major. The chords can be identified by determining each of the pitch classes contained in the chord and using them to spell out the equivalent triad. In the first chord, the pitches are C3, G3, E4 and C5. This means the triad of this chord contains pitch classes C, E, and G. In the key of C-major, this chord is built on the first pitch of the C-major scale. Because the lowest pitch in the chord is a C, the chord is in root position. Therefore, the analysis symbol for the chord uses an upper-case roman numeral I, to indicate this is a major chord with the first degree of the scale as the root. The numbers 5 and 3 indicate that the chord is in root position.

The second chord in Figure-17 contains the pitches C3, A3, F4, and C5. The triad formed with these pitches is spelled with the pitch classes F, A, and C. This is the F-major triad. Because the F-major chord is spelled with a C as the lowest note, the chord is in second inversion. The roman numeral analysis of this chord is an upper-case *IV*, illustrating that it is a major chord, with the fourth scale degree as the root.

The third chord in this example is spelled using the pitches D3, G3, F4, and B4. The pitches of this chord spell out a G-Major seventh chord. Because the fifth of the chord is in the bass, the chord is in second inversion. In the key of C-major, the roman numeral analysis of this chord uses an upper-case *V*, since the root of the chord is the fifth degree of the C-major scale. The inversion numbers are 6, 4, and 3.

major scale. Because the lowest pitch in the chord is a C, the chord is in root position. Therefore, the analysis symbol for the chord uses an upper-case roman numeral I, to indicate this is a major chord with the first degree of the scale as the root. The numbers 5 and 3 indicate that the chord is in root position.

The second chord in Figure-17 contains the pitches C3, A3, F4, and C5. The triad formed with these pitches is spelled with the pitch classes F, A, and C. This is the F-major triad. Because the F-major chord is spelled with a C as the lowest note, the chord is in second inversion. The roman numeral analysis of this chord is an upper-case *IV*, illustrating that it is a major chord, with the fourth scale degree as the root.

The third chord in this example is spelled using the pitches D3, G3, F4, and B4. The pitches of this chord spell out a G-Major seventh chord. Because the fifth of the chord is in the bass, the chord is in second inversion. In the key of C-major, the roman numeral analysis of this chord uses an upper-case *V*, since the root of the chord is the fifth degree of the C-major scale. The inversion numbers are 6, 4, and 3.

Appendix B

9.1.11 Sample 1

Musical String:

F4h+A5h+C6h
F4q+A5q+C6q
F4q+A5q+C6q
Bb4h+Bb5h+D6h
F4q+A5q+C6q
D4q+B5q
F4w+A5w+C6w
D4w+F5w+F6w
Eb4h+G5h
C4q+Eb5q+A5q
C4q+Eb5q+A5q
D4h+F5h+Bb5h
Eb4q+G5q+C6q
Eb4q+G5q+C6q
F4w+Bb5w+D6w
F4w+F5w+C6w
Bb3w+D5w+Bb5w

Key Signature:

B♭ Major

Calculated Analysis:	Actual Analysis:
V: 5:3	V: 5:3
V: 5:3	V: 5:3
V: 5:3	V: 5:3
I: 5:3	I: 5:3
V: 5:3	V: 5:3
I: 6:3	I: 6:3
V: 5:3	V: 5:3
iii: 5:3	I: 6:3
IV: 5:3	IV: 5:3
viio: 6:3	viio: 6:3
viio: 6:3	viio: 6:3
I: 6:3	I: 6:3
ii: 6:3	ii: 6:3
ii: 6:3	ii: 6:3
I: 6:4	I: 6:4
V: 5:3	V: 5:3
I: 5:3	I: 5:3
Success:	94.12%

9.1.12 Sample 2:

Musical String:

F#4h+A#4h+F#5h+C#5h
B4h+F#5h+D6h
A4h+C#5h+F#5h+C#6h
G4h+E5h+G5h+B5h
F#4h+A4h+A5h+C#6h
E4h+B4h+G5h+E6h
D4h+B4h+F#5h+F#6h
C#4h+E5h+A#5h+E6h
B3h+F#5h+B5h+D6h
F#4w+F#5w+A#5w+C#6w

Key Signature:

B minor

Calculated Analysis:	Actual Analysis:
V: 5:3	V: 5:3
i: 5:3	i: 5:3
V: 6:3	V: 6:3
iv: 6:3	iv: 6:3
V: 5:3	V: 5:3
iv: 5:3	iv: 5:3
i: 6:3	I: 5:3
VII: 6:3	VII: 6:3
i: 5:3	i: 5:3
V: 5:3	V: 5:3
Success:	90.00%

9.1.13 Sample 3:

Musical String:

F4h+C5h+F5h+A5h
C4h+C5h+E5h+G5h
D4h+A4h+D5h+F4h
E4h+G4h+C5h+G5h
F4h+C5h+A5h
A3h+A4h+F5h+C6h
C4w+G4w+Bb5w
F3w+F4w+F5w+A5w

Key Signature:

F Major

Calculated Analysis:	Actual Analysis:
I: 5:3	I: 5:3
V: 5:3	V: 5:3
vi: 5:3	vi: 5:3
V: 6:3	V: 6:3
I: 5:3	I: 5:3
I: 6:3	I: 6:3
V: 7:5:3	V: 7:5:3
I: 5:3	I: 5:3
Success:	100.00%

9.1.14 Sample 4:

Musical String:

D4w+A4w+D5w+F#5w
A3h+A4h+C#5h+E5h
B3h+F#4h+D5h
G3h+B4h+D5h+G5h
A3h+A4h+C#5h+E5h
D4w+A4w+D5w+F#5w
D4w+D5w+F#5w+A5w
D4h+A4h+D5h+F#5h
G4h+B4h+D5h
E4h+G4h+B4h+E5h
A3h+A4h+C#5h+E5h
D4w+F#4w+A4w+D5w

Key Signature:

D Major

Calculated Analysis:	Actual Analysis:
I: 5:3	I: 5:3
V: 5:3	V: 5:3
vi: 5:3	vi: 5:3
IV: 5:3	IV: 5:3
V: 5:3	V: 5:3
I: 5:3	I: 5:3
I: 5:3	I: 5:3
I: 5:3	I: 5:3
IV: 5:3	IV: 5:3
ii: 5:3	ii: 5:3
V: 5:3	V: 5:3
I: 5:3	I: 5:3
Success:	100.00%

9.1.15 Sample 5:

Music String:

G3h+G4h+Bb4h+D5h
G4h+Bb4h+D5h+G5h
F#4h+A4h+D5h+A5h
G4h+D5h+Bb5h
D4h+F#4h+D5h+A5h
Eb4h+G4h+Bb5h+G5h
C4h+C5h+Eb5h+A5h
D4w+A4w+D5w+F#5w
D4w+F#4w+A4w+D5w

Key Signature:

G minor

Calculated Analysis:	Actual Analysis:
i: 5:3	i: 5:3
i: 5:3	i: 5:3
V: 6:3	V: 6:3
i: 5:3	i: 5:3
V: 5:3	V: 5:3
VI: 5:3	VI: 5:3
iio: 6:3	iio: 6:3
V: 5:3	V: 5:3
V: 5:3	V: 5:3
Success:	100.00%

An Effective Web-Based Tool to Help Select Promising Lung Cancer Treatments

Scott Olson
Department of Computer Science
Winona State University
Winona, MN 55987
SROlson7730@winona.edu

Abstract

Selecting treatments for patients with lung cancer takes time and effort on the part of the medical professionals. A web-based tool was developed to allow different lung cancer treatments to be compared on a patient by patient basis in real time. The success of the tool was then judged along with an analysis of the tool.

Keywords

web-based tool, prediction, lung cancer

1. Introduction

As it stands today, treatment plans for patients with lung cancer are selected after review of historical data of previous patients that had lung cancer. Being that medicine is moving towards personalized treatments [3] and it takes time to review the historical data, a tool to aid in the lung cancer treatment selection process would be beneficial. Specifically, such a tool would benefit the oncologists, clinicians, physicians and patients who are the ones normally involved in the treatment selection process. We believe it is possible to create an effective web-based tool that would allow different lung cancer treatments to be compared real time on a patient by patient basis.

To strengthen the success of this project, research on other web based tools was used to gain insight into what strategies, components and features have been proven successful and also which have not been successful. This knowledge was then applied during the design of this tool to maximize the tools overall effectiveness. Another goal during research was to identify proper guidelines for displaying information on web pages to the end users, being that part of the effectiveness of this tool will depend on how well the user is able comprehend what is shown to the user through the user interface (UI). The third goal during research was to determine common components of good usability studies. The information was helpful in guiding the creation of a usability study on the UI developed for this project.

So far the basis for a web-based tool to aid in the lung cancer treatment selection process has been identified. The following sections will further define what is being proposed, how it is to be accomplished, how the success was determined and an analysis of results obtained.

1.1 Web-Based Tools

Web-based tools are being used more commonly for a wider variety of applications. A web-based platform is appealing for use in this project for a few reasons. First and foremost, web-based tools can be easily accessed and setup in most current day environments. Another benefit is that updates to web-applications only require updates to the server where the application is deployed; nothing needs to be updated on the client side. A drawback to using a web-based platform is that they require access to the server to be used. The two most common cases where this will prove to be a problem is the network connection fails or the server fails.

1.2 The Model

Simulation based on models has been used in the medical field before, specifically in emergency rooms of hospitals. A model required of this project will lay more on the fixed process end of the spectrum because the model will only be required to predict specific criteria related to lung cancer treatments and lung cancer patients. This will allow for greater ease of use as depicted in Figure 1 which is borrowed from a paper on a simulation tool for emergency rooms [6]. There is a lack of easy to use tools to manipulate simulation or prediction models which is having a negative effect on the adoption of those types of technologies [6]. This web-based tool is meant to be a good, easy to use tool for manipulating the prediction models used. The model itself only has to model one type of situation, a patient with lung cancer and a given lung cancer treatment. This type of model is called a fixed process model.

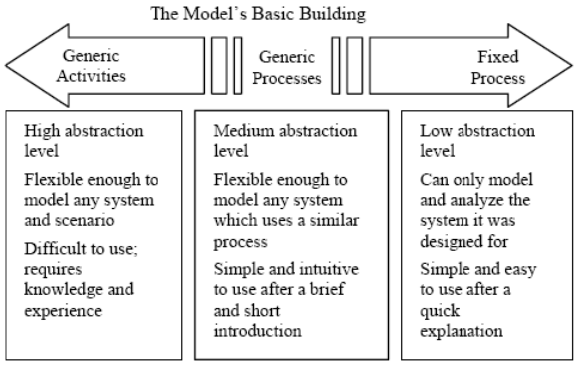


Figure 1. Visual and explanation of characteristics of generic activities, generic models and fixed models.

1.3 The Existing Tool

There is an implementation of this tool that existed before the start of this project, but it falls short of being effective to the criteria posed to this project. The pre-existing tool was developed as a Java servlet and it used Tomcat 6.0 as the application server. Since the tool was required to predict the survivability of each patient, a method of doing so was required. The predictions were made possible through the use of an R model provided by Mayo Clinic. R is a statistical modeling language with an accompanying work environment. The R model is loaded into the work environment, the required libraries are also loaded in the work environment, and then the survivability function may be used to predict the survivability of a patient with lung cancer. To facilitate information flow between Java and R, an interface from Java to R, called JRI, was used. JFreeChart was used to create images of the charts to graphically show the information on the web page. In addition to charts, tables were created and displayed through HTML tables on the web page.

The pre-existing tool falls short in a few areas. The largest problem is that the pre-existing tool only has support for a single lung cancer treatment. One of the driving points for this project is to allow for multiple lung cancer treatments to be compared so that an appropriate treatment for a given patient may be chosen. This also gives rise to needs for other features that need to be accounted for in the user input forms and also in the workings of the way graphs and tables are created. Another concern with the pre-existing tool is how the patient information is stored. Currently, patient information is stored within R. This required change because R is not meant to serve as a database. The documentation for R states that "R is not well suited to extremely large data sets" [5]. Given that the number of patients entered into the system is potentially large and R stores its working data in memory, patient information should be saved elsewhere. That being said, the existing tool will still serve as a good resource in designing and implementing a new tool to extend and improve upon the existing tool, but it is not without flaws.

2. Hypotheses

The primary goal of this project is to create an effective web-based tool to allow different lung cancer treatment plans to be compared in real time at the benefit of clinicians, physicians, oncologists and ultimately lung cancer patients. Effective will be judged on two criteria: (1) the ease of use and accuracy in the interpretation of the output of the R model, and (2) the speed at which the results can be obtained. The success of the project will judged through two methods. A usability study on the UI of the tool will provide a degree of insight as to the usability by users, that are not medical professionals, and software testing methods will provide some proof of accuracy and reliability of the tool. We expect the tool to allow users to: (1) enter or change patient information with little to no confusion and within the matter of minutes, and (2) compare lung cancer treatments on a patient by patient basis with little to no confusion and within the matter of seconds

.

3. Methods

Good software applications normally sprout from an applicable software development process. An incremental object oriented software development process was used during the development of this tool. Another layer on top of the software development was the usability study conducted on the UI of this tool. The initial UI had to be developed early on to allow time for the usability study to complete. The first increment was split into 3 phases to give a feel for the timeline in which events took place. First, the requirements were gathered and analyzed. As well, the initial UI was developed for use in the usability study. During the second phase, the usability study was conducted on the UI and the backend of the tool was developed. The third phase consisted of testing and validation of the backend and examination of the results of the usability study

.

3.1 Analysis of Requirements

During the analysis of the requirements, the use cases and initial functional test cases were documented. Use case diagrams were created to show how different actors were to use the system. Only 3 actors are involved with the system; Server, Admin and User. The Server actor has two use cases; initialize and destroy. The Admin actor has 3 use cases. The User actor has 7 use case interactions. The User is able to add a new patient to the system, return to an existing patient that is within the system already, view a patient's information, modify a patient's information, compare lung cancer treatments for a given patient through a graph and compare lung cancer treatments for a given patient through a table. From those 12 use cases, the initial functional test cases were documented to show how the use cases were connected and also the expected results given an initial state and an input.

3.2 Design

The design of the tool involved taking the analysis documents and creating class designs. However, the UI was given priority. Documents and classes were designed first depending on whether or not they were thought to be crucial to the usability study or not. Then the initial UI was developed for the usability study, and then the remainder of the design was completed. A few resources in specific were used as guides when designing the UI. Following the recommendations in "Research-Based Web Design & Usability Guidelines", we were able to keep pages from feeling too cluttered, keep pages consistent, and

draw attention to important parts of the pages [2]. Another resource that had influence on UI design was the article about how a UI was redesigned because it was not appealing to the user and the design was not user friendly [7]. The UI for the AT&T site was much larger scale, but design ideas such as centering design on the user as much as possible and keeping the design concerns of the system in second place were still applicable to this project. This increases the complexity of the design, but it allows for a more natural UI for the end user. In the end, six user interface pages were included in the prototype that was used in the usability study. To keep the site as easy to understand as possible, all pages have a uniform header and footer in which the page content is displayed in between. The header contains a banner with the site title and slogan along with the menu bar located underneath the banner. The different pages included in the UI were the index page, a "New Patient" page, an "Existing Patient" page, a "Patient Information" page, a "Help" page and a "Links" page.

The index page is fairly straight forward; it contains a short description of the site and also a guide to indicate where the user should start. The index page is meant to be the first page the user will see. The guide indicating where to start was included on the index page with the intent that new users will not have to spend time wondering where they should go next. Figure 2 is a screen shot of the index page to give a feel for the layout of the site.

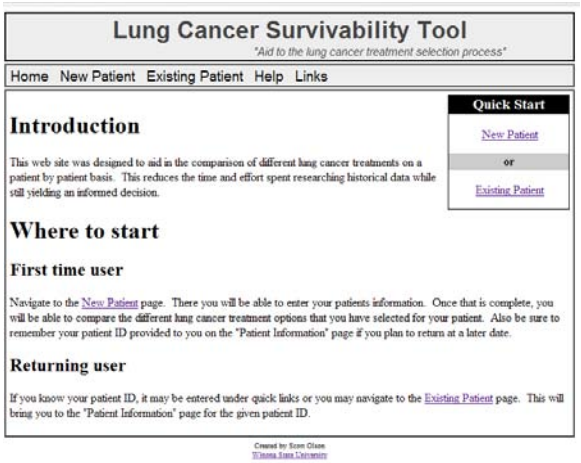


Figure 2 Screen shot of index page.

The "New Patient" page contains a short description of the page located at the top portion of the page to give insight as to what is required for the new patient form which is located under the description. The new patient form needs to be filled out so that a prediction is made based on information that best matches the patient. In addition, patient information is stored for later retrieval. The only identifying information that is stored is the Patient ID that is generated by the system. No data is stored other than what is necessary for the prediction to be made. Once the user has submitted valid data, the user is brought to the "Patient Information" page and the patient's ID is assigned. The patient ID is what is used by the system to keep track of patient information. Figure 3 shows a partial screen shot of what the "New Patient" page looks like.

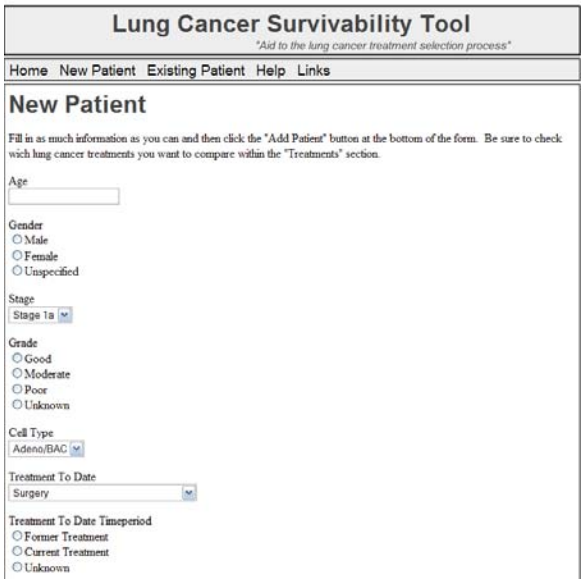


Figure 3 Partial screen shot of the "New Patient" page.

The "Existing Patient" page contains a short description of what needs be entered in the form located on the bottom portion of the page. This page is used by returning users to view patient information of a patient that was previously entered into the system through the "New Patient" page. Figure 4 shows a screen shot of the "Existing Patient" page.

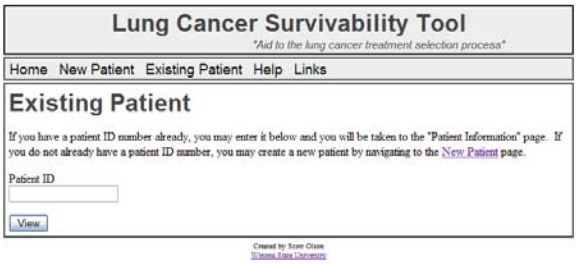


Figure 4 A screen shot of the "Existing Patient" page.

The "Patient Information" page has the patient's ID displayed at the top in large text so it is very noticeable. Below the patient ID are 3 tabs titled "Graph", "Table" and "Patient Information". The "Graph" tab contains an image of the different predictions for the lung cancer treatments that are selected for that patient. Off to the side of the graph, there is a means of unselecting and reselecting treatments to be included or excluded from the graph. This is to allow the treatments to be more comparable by removing treatments from the graph or re-including treatments in the graph. The next tab is the "Table" tab which contains a table showing the same data as the graph, just in table format. The need for the table is to have a numerical representation of the predictions as well as a visual representation. Similar to the graph, treatments may be excluded and re-included through options off to the side of the table. The final tab is the "Patient Information" tab which shows the information that the system has for the given patient ID. Through this tab, the user is able to change the current patient's information as needed. Upon the successful update of the patient's information, the "Graph" tab and the "Table" tab are also updated to reflect the changes the updated information had on the different predictions. Figure 5 is

a screen shot of the table tab of the "Patient Information" page to give a feel of how the page is laid out.

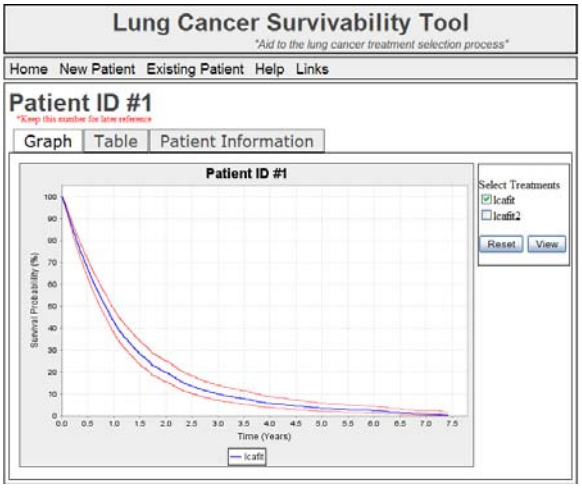


Figure 5 A screen shot of the "Patient Information" page showing the graph tab. The predicted survivability for various lung cancer treatments is shown visually through the graph.

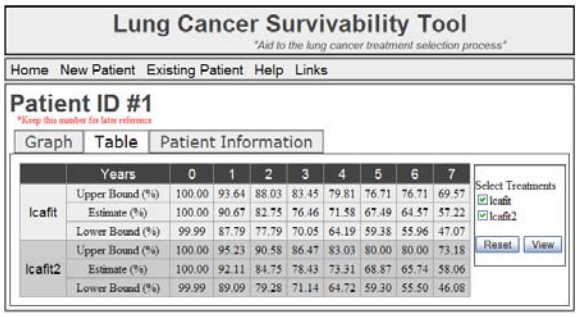


Figure 6 A screen shot of the "Patient Information" page showing the table tab. The predicted survivability for various lung cancer treatments is shown in tabular form through the table.

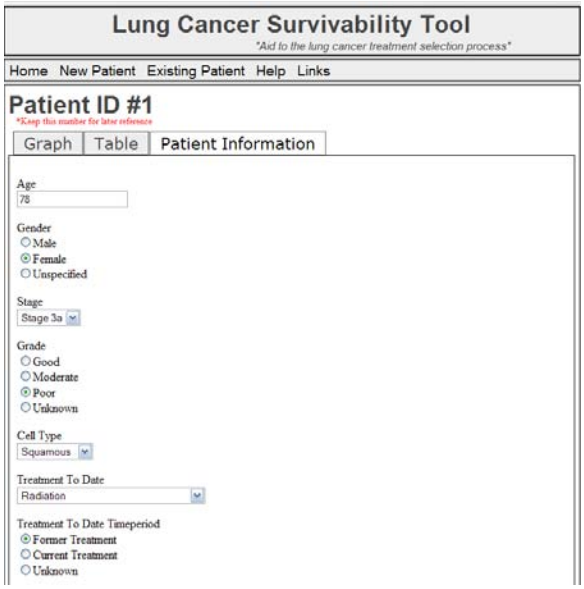


Figure 7 A screen shot of the "Patient Information" page showing the patient information tab. Patient information can be seen as well as updated on this tab.

The "Help" page contains information on the operation of the site, which is divided into topics. There was an index of the topics located at the top of the page. Below the index is where the list of each topic and its explanation. The topics ranged from the overall site functionality to how a particular task is accomplished.

The "Links" page contains a few links to other external sources of information on lung cancer. This was primarily added for the usability study; however it was probably not required.

The usability study requirements were formed after the UI prototype was developed. The usability study was designed and carried out by a team of people to in which we served as their clients. Review of other usability studies was taken into consideration when forming requirements for a usability study for this project. The requirements for the usability study were to assess how intuitive it was for users who were not necessarily knowledgeable in the lung cancer field, to:

- Add a patient to the system
- Change existing patient information
- Add treatments to a given patient and also remove treatments from a given patient
- Compare lung cancer treatments visually
- Compare lung cancer treatments in table format
- Return to an existing patient and to change from one patient to another patient

Further information on what was confusing to the user and what could be added, changed or removed are also goals of the usability study.

After the UI was designed, a prototype was developed and the usability study requirements were formed, and the remainder of the design was then completed. The main changes kept in mind when developing the backend were: (1) the patient information was not to be kept in R, (2) the UI is tailored for the user, and (3) multiple lung cancer treatments are to be supported. One of the difficult items to design was the form system for data entry.

The patient information inputs required by the model may change if the model changes, which means that the forms should allow for easy modification. Also, the inputs required by the R models do not show grouping information, nor can they be used as a complete list of inputs. A form generation system was designed to handle this problem. This allows the forms to be changed easily, forms to be created on the fly and for the forms to be pre-populated with values as required. The form items with their suitable values are specified in XML format along with a mapping showing what should be used when interacting with R. To remove the patient information from the R environment while still persisting data for later retrieval, the system is now designed to store patient information in XML format and the XML files are saved on the server's file system which is better suited than R to be used as a repository for large amounts of information. XML has been used before to fill similar roles in other web applications with a clinical support emphasis [e.g. 1, 4]. Class diagrams along with sequence diagrams were developed to document the design process.

3.3 Usability Study Design

The usability study that was performed on the initial UI is designed to serve as an indication to its usability, not specifically its usability by medical professionals. The study consisted of an entrance exam, a think-aloud procedural test using the site, and an exit exam. The entrance exam gathered a few bits of information on the tester such as:

- Gender
- Age group
- Medical knowledge
- Internet usage
- Field of study

They then proceeded to the think-aloud test where a facilitator verbally communicated to the tester what they are to do next and the tester then verbally said what they were thinking. Upon completion of the testing portion, the tester was asked to give their thoughts of the tool.

3.4 UI Implementation

Since the project had to follow a timeline, which included a usability study, the UI was developed first. The UI in specific has a lot of complexities that make operation of the site easier to understand and use. These niceties to the user cause the UI to be more complex during design and implementation. The design calls for some form validation, forms to be built on the fly, asynchronous JavaScript and XML calls (AJAX), and tabbed panels.

The web application was developed through the use of JSPs and JavaScript. The jQuery 1.2.3 and the jQuery.ui 1.0 JavaScript libraries were used to help create tabs on the "Patient Information" page and also to help with performing AJAX calls. While jQuery did help with the AJAX calls and attaching a callback function for when the AJAX calls completed, the response then had to be parsed to retrieve values which posed a problem for some time to do so in a generic format that could be applied to multiple pages. This problem was an effect of creating forms on the fly, normally semi-static pages can be created to show validation errors for inputs that are known ahead of time. The problem was resolved through the use of following the JavaScript Object Notation (JSON), format and JavaScript functions to accomplish required tasks. JSON was used to format the responses in a way that was easy parsed into JavaScript Objects that could then update the Document Object Model (DOM), as needed.

The UI included features such as: a validated form to add users to the system, a validated form to return to existing users, a page with 3 tabs which held the graph, the table data, and the patient information. Tabs were chosen to allow the user to quickly and easily navigate between the chart, graph and patient information while keeping page length to an easily manageable length. The graph could be changed by selecting or deselecting different lung cancer treatments which were listed on the side of the graph. If the selected lung cancer treatment options were modified, the graph would also be updated. The table data followed a similar format as the graph tab did. The table showed data for the selected lung cancer treatment options and the selection of lung cancer treatment options could be changed. The patient information page allows for the user to view and change the patient information that is stored by the system.

For the usability study, a complete UI was developed, but it was only semi-functional. The main functionalities not included for the usability study were storing patient information and the ability to communicate with R. To account for the missing functionalities, tables and images were simply created ahead of time to mimic what would happen if those functions were present. The reason for this was to allow time for the usability study to take place. Once the initial UI was completed for the usability study and the rest of the design was completed, then the functionalities were added to the UI as they were completed. The color scheme chosen was a white background for the content area and dark shades of gray or black for text and borders to create a high contrast color scheme that is easy to look at. It is important to have a high contrast color scheme so users can view information on the page with as little difficulty as possible [2].

3.5 Backend Implementation

The backend of the system primarily exists in the form of servlets. There are 4 main backend components: a patient manager, an R manager, a graph generator and a table generator. The patient manager is responsible for adding new patient information to the system, retrieving existing patient information given a patient ID and it also allows patient information to be updated. Patient information is stored in XML format on the file system of the server instead of being stored within the R environment. The R manager handles all communications between R and Java. The R manager is also responsible for setting up the initial R environment and loading all the required models and libraries. Required libraries and models are specified within properties files that are parsed upon startup of the web application. The graph generator creates graphs and streams them to the UI. This component relies on JFreeChart and is used to generate many different graphs for a given patient depending on what the user whishes to view. New graphs are generated when the user selects or deselects different lung cancer treatment options via controls located on the UI. Figure 4 shows an example of what the generated graphs look like. The table generator is similar to the graph generator, but it creates

tables instead of graphs. It creates tables based on the different lung cancer treatments that are selected via controls on the UI.

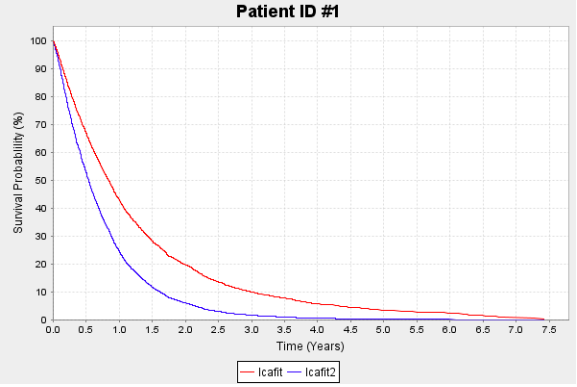


Figure 8 An example of what graphs look like that are generated by this tool.

3.6 Testing and Validation

Part of the testing of the software was carried out in the form of a usability study. The usability study was used as a means of assessing how intuitive the UI was and also how well information was displayed. Test cases were created to test individual classes and special test cases were created to test special cases or circumstances. Lack of time prevented further testing.

4. Success Criteria

The success of the project will be partially determined through the usability study and also partially determined through software testing methods. The usability study will provide evidence that the site is usable if:

1. 98% of all testers are able to complete the required tasks
2. 85% of all testers find the site to be easy to use in addition to comfortable after having used the site
3. There are no serious deficiencies reported by testers

Software testing and validation will provide evidence that the web application is functioning correctly if:

1. Tables that are created are accurate according to information received from R
2. Graphs that are created are accurate according to information received from R
3. Patient information is consistent from page to page and from visit to visit
4. No major bugs are left present that are caused by the web application

5. Results from Testing

The results gained from validation of the backend tool show that it is producing results as expected. The major components that were tested were the R manager, the patient manager, and the graph and table generators.

The graphs generated by the tool were compared to graphs generated by R using the same inputs for both the tool and R. Graphs created from 2 different treatment options were compared and also graphs created using different sets of inputs were used. The graphs created by the tool and the graphs created by R were on a similar scale, but the resulting sizes of the graphs were different. To enable the graphs to be easily and more accurately compared to each other, the graphs created from R were resized to fit the size of the graphs created from the tool. From a purely subjective standpoint, the graphs created by the tool and the graphs created by R, both using the same inputs, will create identical graphs. In addition, a hand crafted data set was passed to the graph generator and it produced an appropriate graph for the given data set.

The table data was also verified against the data that was received from R by the tool. The tables created by the tool were found to be correct when taking into account that the final number is rounded to two decimal places. The table generator was tested by printing out the resulting table from a given result set from R and manually verifying each of the values given in the table. As a second measure, a test case created to test the methods of the graph generator class. The test case passed without problems.

The patient manager was tested to ensure it properly was able to save new patient information, retrieve old patient information, update patient information, and test if a patient exists on the system. The system was found to successfully save, update, locate, and retrieve patient information that exists on the system.

The connection between R and the tool was tested by creating a special test case that simply determined that R was being initialized properly and that results could be received from R. The results from the R model used by this tool could not be tested because we would not know what values are expected.

5.1 Results from Usability Study

Overall, the results from the usability study were very positive. There were a total of 15 students from Winona State University involved in the usability study. All of the testers were able to complete the required tasks, but a few areas of difficulty were encountered for some users.

The tool was found very easy to use and navigate by the testers. Comments were made such as:

- "Easy to use…"
- "Easy to navigate…"
- "Straight forward"
- "Simple"

Table 1 shows the features that were most liked of the tool.

Pros	**Frequency**
Simple to navigate	15 of 15
Easy and quick to use	9 of 15
The graph is useful	11 of 15
Quick start	4 of 7

Table 1 A table showing features and functionalities that testers liked about the tool.

But there were also a few problems that the testers were able to identify.

The largest problem encountered was many testers did not know how to select multiple lung cancer treatment options that are to be compared. During the walkthrough, there was also confusion between treatment options that are being compared vs. the list of all the possible treatment options. Table 2 shows the problems reported along with the number of testers that reported each problem.

Cons	Frequency
Selecting multiple, disjoint, treatment options	10 of 15
Confusion between treatments being compared and treatments to compare	8 of 15
Delays during stage selection	4 of 15
Plain site/lack of color	2 of 15
Multiple links to "New Patient" page of the front page	3 of 15
Difficulty seeing yellow line of graphs when used	2 of 15
Inability to read graph	7 of 15
Screen resolution problems	3 of 15

Table 2 A table showing features of the tool that testers didn't like or problems that testers had with the tool.

One note worth mentioning is that this usability study cannot be used as a direct indicator as to how usable doctors will find the site to be. Of the 15 students, 4 were nursing majors, 2 were biology majors, and 6 of the students were enrolled in an entry level computer science course at the time.

6. Analysis of Features

This tool provides many features for allowing different lung cancer treatments to be compared effectively on a patient by patient basis. Features included in the system were:

- the ability to effectively compare multiple lung cancer treatment options on a patient by patient basis
- the ability to store patient information for later use
- the ability to update patient information
- the ability to return to previously entered patient information
- a system that allows different models to be used and for lung cancer treatments to be removed or added
- a system that allows the user input form to be modified
- a system that is web accessible.

Each of these features required that many component technologies be used collectively to provide the desired effect.

Ultimately, the user interface is viewed by the user in a web browser in which HTML, images, and JavaScript each play their own roles. To produce the HTML for the web pages, JSPs and Servlets were used. JSPs were primarily used to create the framework for the page and to keep a uniform appearance throughout the site. On the other hand, Servlets were used to create the dynamic content for the site such as the user input forms and other form elements. JavaScript was used to modify the document object model on the client side. JavaScript was not used to create content, just modify content.

Another feature is the effective comparison of the different lung cancer treatments on a patient by patient basis. This is made possible through the graphs and tables that are presented to the user on the "Patient Information" page. Once the patient has been added to the system and the appropriate treatment options have been selected, the graphs and tables are then generated. The graphs are created using a Servlet, R, and JFreeChart. The Servlet controls the graph generation operation and responds to the user appropriately. R provides the prediction data based on the patient information passed to it by the Servlet. The data is then formatted to a friendly format and used to create the graph. JFreeChart is used to create the graph and store it as a PNG image on the server. The table generation is somewhat similar to the graph generation, but it does not use JFreeChart. Instead, there is simply a Java class that is responsible for taking the formatted data from R and turning it into HTML tables that are more user friendly format. Tables and graphs are presented to the user to allow for visual comparison as well as allowing for more of a pragmatic comparison.

User input is obtained through the use of HTML forms that the user submits to the web-tool. This feature is brought up through two sub features: one being a system for dynamically creating HTML forms, and the other is a method for specifying what must be presented in the user input forms along with any validation the fields may require. Java classes are used to dynamically create the HTML for the user input forms. An XML file that is parsed upon the initialization of the Servlet is used to specify what fields are to be presented in the user input forms along with any validation that must take place. This form is then copied as needed and can then be populated with values or simply rendered as HTML. Lung cancer treatment options are also defined in the XML file that specifies what fields are included in the user input forms, so the treatment options are easily modifiable.

A properties file is used to specify many of the remaining configurations. This includes what the working directory for R is along with what the filename for the R model is. If a new R model is to be used, all that needs be changed is the appropriate attribute in the properties file. The server application must also then be restarted before the changes will be put into effect.

There are also features that were not included in the system. There is only limited support for mobile devices (e.g., cell phones), a database is not used, and there is no authentication. Many doctors and other medical professionals currently use mobile devices for a variety of activities at work. Not all mobile devices are able to process HTML pages that include JavaScript or AJAX. This is largely due to the limited processing power along with the fact that mobile devices have a small screen in comparison to laptops and desktops. However, including Wireless Application Protocol support would be a nice addition to this web-tool to allow for a larger variety of devices access to the tool. Another feature not included in this tool is the use of a database. A database would prove useful in storing and retrieving patient information, but it would be another additional technology to maintain. In the end, the complexity added from maintaining a web server as well as a database was thought to be too great of a burden. Especially since a full-fledged database was not needed and XML files are simple to work with and they accomplish the job. Finally, there is no support for authentication.

6.1 Analysis of Usability Study

The usability study did bring to light a few important problems. The main problem is with selecting multiple treatment options. The problem is believed to have been caused by a lack of indication that multiple treatment options are allowed to be selected on the corresponding HTML form element. This problem was fixed by adding text explaining how to select multiple items to HTML form elements that allow for this type of functionality. The next problem addressed was the confusion between treatment options being compared and all treatment options available to the user. The text that accompanied these fields was changed to more appropriately explain the intended meaning of the field. To help the delay in selecting the stage of lung cancer, it was suggested to use Decimal numbers as opposed to Roman numerals. The next problem solved was the difficulty in seeing the yellow line on the graph if enough treatment options were chosen. It has now been changed so that there are no bright yellow colors displayed which should increase the visibility of any one of the different treatment options on the graph. Finally, during the usability study, the page width was fixed at 800px across. When a screen resolution of 800x600 is used, there is a problem of the right hand scroll bar occupying some of the screen which would require the user to scroll to the left or right to view the contents of the entire page. This was fixed by changing the width of the pages from 800px to 750px and adjusting any other elements to fit and appear correctly within that limit.

There were also a few concerns noted by the usability study that were not deemed significant. The first was a few of the users commented on the fact that the theme of the site is not very colorful and the site is fairly plain. Being that the site is made for oncologists and physicians, not the general public, the aesthetics of the tool are not as important as the usability and functionality the tool provides. Another problem noted by from the usability study was that there are multiple links to the "New Patient" page on the index page. This is an issue from the usability study that we are a little unsure how it posed a problem. All of the links that are named similarly go to the same exact page. We also feel that the links should help the efficiency for returning users via the "quick start" section and for new users via the "New users" section. The final issue that was be noted from the usability study is the fact that many of the testers were unable to accurately read the graph. Again, this is probably due to the fact that few, if any, of the testers have sufficient knowledge in the area of lung cancer and lung cancer treatment. However, a section may be added to the help page to further explain the graph and the table.

7. Conclusion

The goal of this project was to develop an effective web-based tool to compare different lung cancer treatments on a patient by patient basis. This was accomplished by first researching other web-based tools and identifying a proper set of guidelines for web pages and web applications. This then lead to an analysis and design phase where the requirements were formulated and design documents were created. The user interface was then created so the usability study could take place during the time that the backend was being developed. After the backend was developed, the results from the usability study were analyzed along with results from testing of the backend.

The results from the usability study and from testing of the backend are in favor of an effective web-based tool. The graphs and tables are generated as expected. Patient information is stored, retrieved, and updated as expected. Forms are easily modified being that the model is subject to change. Web pages are uniform in appearance and are easy to follow along with easy to understand. If help is needed for a page, there is a help page that explains the pages in a fair amount of depth. In addition, there were no serious problems that hindered the site from being usable judging from the results obtained from the usability study. A few concerns were noted, and a few fixes were made, but that was the purpose of the usability study in the first place.

We are happy with the results we have seen so far and hope the trend continues. We put a lot of time and effort into the project and we have produced something usable to show from it. But as with any software application, there is always room for improvement. Most notable is improved support for mobile devices.

8. References

[1] CATLEY, C., PETRIU, D. and FRIZE, M. 2004. Software Performance Engineering of a Web Service-Based Clinical Decision Support Infrastructure. *ACM SIGSOFT Software Engineering Notes*, 29, 1, 130-138.

[2] LEAVITT, M. and SHNEIDERMAN, B. 2006. *Research-Based Web Design & Usability Guidelines*. U.S. General Services Administration, Washington, D.C.

[3] PETERSON, A. 2006. The genetic conception of health: is it as radical as claimed? *Health*, 10, 4, 481 – 500.

[4] PETROVSKI, A. and MCCALL, J. 2005. Smart Problem Solving Environment for Medical Decision Support. In *Genetic and Evolutionary Computation Conference (GECCO)'05*, Washington, D.C., June 2005, ACM, New York, NY, 152 – 158.

[5] R DEVELOPMENT CORE TEAM. 2007. R Data Import/Export. In *R Documentation*, R Development Core Team.

[6] SINREICH, D. and MARMOR, Y. 2004. **A Simple and Intuitive Simulation Tool for Analyzing Emergency Department Operations. In** *Proceedings of the 36th Conference on Winter Simulation*, Washington, D.C., 2004,.

[7] SHEIKH, A. and TARAWNEH, H. 2007. **A Survey of Web Engineering Practice in Small Jordanian Web Development Firms. In** *Proceedings of the 6th Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on The Foundations of Software Engineering*, Cavtat, Croatia, September 2007, ACM, New York, NY, 481 – 490.

Network Performance of Next Generation TCP/IP versus Windows XP TCP/IP

Kyle T. Peters
Computer Science Department, Winona State University 14765 Sumter Ave
Savage, MN 55378
612-600-7148
Peters4385@yahoo.com

ABSTRACT

Since the 1990's, Microsoft has made many modifications to its existing TCP/IP stack to meet the needs of homes and businesses around the world. With the release of Windows Vista, Microsoft has not only made significant changes to the user experience of its operating system, but also included a newly developed TCP/IP stack named Next Generation TCP/IP. The changes made to the TCP/IP stack consist of a dual IP stack, one for IPv4 and one for IPv6, receiving window auto tuning, compound TCP, and enhancements for high packet loss, high latency and high bandwidth environments.

We tested Windows Vista TCP/IP performance compared to Windows XP TCP/IP under WAN and LAN network conditions. In most cases, we found the Windows Vista TCP/IP stack shows an increase in transfer speeds and decrease in transfer times. We found Windows Vista shows a significant decrease in transfer times under high latency and high bandwidth network conditions.

KEYWORDS

TCP/IP Performance, Next-Generation TCP/IP, Windows XP, Windows Vista

1. INTRODUCTION

With the rising usage of the Internet, there is an increasing need for faster ways to download data. To solve this challenge, there have been increases in bandwidth (the speed at which computers and other devices connect to the Internet). With the increase of bandwidth, there needs to be software as well as hardware to utilize these speeds.

On the software side, TCP/IP connections are one of the most common connections found on the Internet. IP or the Internet Protocol is used to route data from the sending computer or device, to the receiving device. TCP or Transmission Control Protocol is used to provide a stable and reliable connection for applications such as an internet browser. [1,12]

Microsoft designed a new piece of software called Next-Generation TCP/IP, which increases the amount of data transferred on high bandwidth connections and connections between computers and devices far away from each other. This was needed because the Windows XP version of TCP/IP was created in a time when bandwidth was significantly lower then it is now.

Next Generation TCP/IP was created by using both old and new technology to increase the performance of both TCP and IP. With the use of Next Generation TCP/IP in Windows Vista, how do the new changes compare to the old Windows XP implementation of TCP/IP in terms of transfer speed and transfer time?

We predicted Windows Vista to have an increase in network performance in both LAN and WAN conditions. To test the performance of both Windows XP and Windows Vista, we setup a network using a network simulator to have a reproducible environment for our test cases. For each test, we used ftp to transfer 50 kb and 50 mb files. We recorded the transfer time and speeds of each test.

2. TCP

TCP is a connection-oriented communication protocol, which means it provides end to end reliable connections.[1,12] A common example of this is using a circuit-switch telephone network system. When someone picks up the phone and talks with another person, everything being said is heard by each party. In other words, TCP guarantees all communications by one party to be received by another in the order in which it was sent. This is useful because IP, which delivers TCP data to the receiver, does not guarantee this.

TCP can be described by the TCP/IP reference model, basic data transfers, reliability provided and flow control.

2.1 TCP/IP Model

The layers of the Transmission Control Protocol/Internet Protocol (TCP/IP) Reference Model are Host-to-Network, Internet, Transport and Application layers.

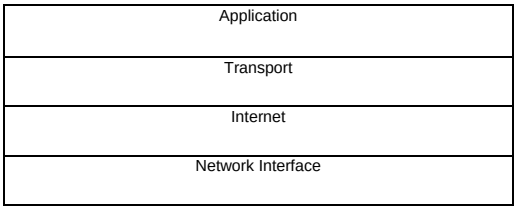


Figure 1 TCP/IP Reference Model.[1]

The Host-to-Network layer offers a medium to communicate between systems, whether it is by a twisted copper pair or fiber optics cabling.[1,12] The Internet layer consists of IP, which is defined as an unreliable, best effort packet delivery service.[1,12] This means IP tries its best to deliver a packet of information sent from one node to the next, but there is no guarantee that it will get to the intended destination.

The Transport layer offers a connection-less (UDP) or a connection-oriented service (TCP).[1,12] The User Datagram Protocol (UDP) provides a connection-less service, which is a best effort service and is not considered reliable. [1,12] The data being sent to another node may not reach the target node or data may not be received in the order that it was sent.

The second connection type is Transmission Control Protocol, which provides a reliable connection between two systems.[1,12] Some of the features included in TCP are guaranteed in order data delivery, congestion control and flow control.[1,12] Congestion control is a mechanism used to prevent over loading a network with data. Flow control is used to prevent overloading the receiving node with more data than it can handle. File Transport Protocol (FTP) is an example of an application that uses TCP.

Finally, the application layer of the TCP/IP Reference Model is used to provide users an easy way to communicate using TCP/IP layers.[1,12] An example of an application is an Internet Browser.

2.2 BASIC DATA TRANSFERS

The most important feature of TCP is its ability to continuously stream data between parties. It does this by breaking the stream of data up and inserting the data into segments to be sent across the network.[12] Segments are also used to establish connections, send acknowledgments, advertise window size and close a connection.[8]

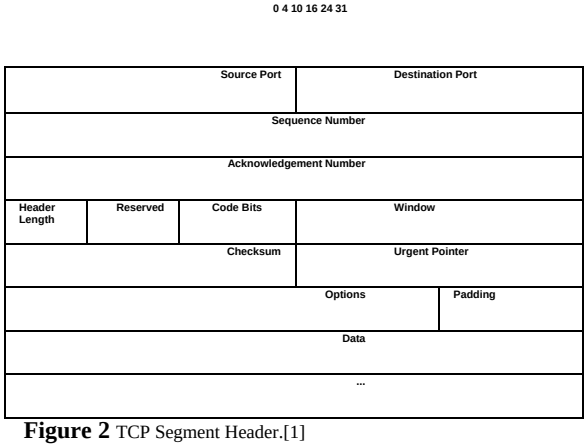


Figure 2 TCP Segment Header.[1]

Figure 2 shows a TCP segment consisting of a 20 byte header containing source and destination port, sequence number, acknowledgement number, windows size, various other options and data.[8] The maximum amount of data that can be sent in a segment is determined by the IP maximum transfer unit or MTU.[12] When establishing a connection, a port needs to be specified. Source and Destination Ports are used in TCP for two reasons. The first reason is to direct the received segment to the specified process in the node. The second reason is TCP allows multiple connections to be open while receiving and sending segments simultaneously.[8] TCP has some standardized port numbers for applications such as HTTP and FTP. These port numbers are port 80 for HTTP and port 21 for FTP.[1]

Sequence numbers are used in TCP to verify the order of each segment sent.[8] TCP guarantees in-order transfers and needs a way to determine the ordering of segments. The segment field is 32 bits long and allows for over 4 billion different bytes to be transmitted.[1] If the sequence number goes higher than 32 bits long, TCP wraps the sequence number and starts over.[8]

Being a reliable source of communication, TCP needs a way to ensure the receiver gets all the segments the sender has sent. The way TCP achieves this is by acknowledgements. An acknowledgement is TCP's way of saying to the sender that the receiver has received the segment sent.[1,12] A sender sends a segment containing information, and when the receiver gets that piece of information, it sends a note back to the sender saying that the receiver received the information. Since TCP needs to acknowledge the received segment, the field for acknowledgments is 32 bits in length as well.[1]

The other features of the segment header are header length, reserved space, code bits, checksum, urgent pointer, options and padding. The header length serves as a way to tell how many bits are in the header to accommodate options having variable lengths.[1] The reserved space is reserved for future use. Code bits are used to tell the receiver the urgent pointer and acknowledgement fields are significant and the next bit is used to set the push function to tell the sender not to buffer data.[1] The last three fields are used when establishing a TCP connection. Checksum is an inner detection mechanism and is used to verify that the contents of the segment are correct.[1] Finally, the options field provides a way for extra changes to be made that are not specified in the segment header.[1]

2.3 RELIABILITY

Wide Area Networks or WANs are a very dynamic environment; nodes and links come and go. In order to provide a reliable service, TCP needs to recover from segments that are lost, out of order, damaged or duplicated. The way TCP supports this environment is by using acknowledgements, sequence numbers and checksums.[8] If a segment is not acknowledged during a certain period of time, the sender resends the segment. If the segment is sent out of order or damaged, the receiver does not acknowledge the data and the timer expires, forcing the sender to resend the segment.[1]

2.4 FLOW CONTROL

In Local Area Networks (LAN) and Wide Area Networks (WAN), nodes have different connection speeds, as well as different hardware configurations limiting the ability to receive and send segments. TCP uses a sliding window to limit the amount of data that can be sent at one time.[8]

A sliding window is a mechanism window allowing only so many segments to be sent or in transit at one time. A pointer points to the next segment to be sent.[1] When a segment is sent, the pointer number is increased by one. As long as the number of segments in transit or unacknowledged is less than the size of the window, a segment is sent. Once the window is full, the sender must wait for the receiver's acknowledgements before it can send another segment. When an acknowledgement is received, the sliding window moves to the next segment number.

It is worth noting that the receiver has a sliding window used to keep track of the number of acknowledgments sent. Figure 3 is an example with a window size 7, meaning it can send 7 segments until it has to wait. The bold number 8 is where the pointer is currently and is the next segment to be sent.

1 2 3 4 5 6 7 **8** 9 10 11 12

Figure 3 Sliding Window.[1]

How does this control the flow of segments and what are the benefits? When a connection is established, the receiver tells the sender the size of the window. The receiver may increase or decrease the window size to control the number of segments going across the network. This is beneficial because different nodes on the network have different abilities to read and send packets. If the receiver is busy, it can decrease the window size, which tells the sender to slow down. This can also be used to control the congestion in the network, which is discussed in the next section.

2.5 CONGESTION CONTROL

Congestion occurs when high numbers of segments go through a small subset of the network. This is a local phenomena not global. The symptoms of congestion are high delays in segment transfers and segment loss.[8] These symptoms only effect traffic going through that affected part of the network.

Because segments can be lost in WAN environments and during congestion, TCP uses a retransmission timer to know when to retransmit a segment. If the timer reaches zero, the unacknowledged segment is resent to the receiver. This retransmission timer is dynamically determined using samples from segments sent and the amount of time it took to receive an acknowledgement.[8]

How does TCP flow control deal with congestion? There are two ways TCP deals with congestion; slow-start and multiplicative decrease. Slow-start is a preventative algorithm, meaning it tries to prevent congestion from happening.[8] Slow-start applies to new connections and connections that are recovering from congestion.[8] The methodology behind slow-start is to start the flow control window at one and increase the window size by fifty percent every time an ACK is received until the window reaches maximum size.[1] The point of this is to not overload the network when adding a new TCP connection or when congestion is experienced.

Multiplicative Decrease Algorithm is used when congestion is experienced.[1] When a segment is lost, the multiplicative decrease algorithm assumes congestion has happened in the network.[1] The first step in the multiplicative decrease is to reduce the window size by half each time a segment is lost. [1] This reduction can decrease down to one if enough segments were lost. The next step the algorithm takes is exponentially increasing the retransmission timer.[1] By doing both steps, the amount of segments sent on this connection dramatically decreases.

3. PROBLEMS WITH TCP

There are several problems with TCP that can impact performance. The major problems with TCP include TCP sequence numbers, flow control window size and slow start.

3.1 FLOW CONTROL WINDOW

In a standard TCP connection, the default maximum window size is 64 kilobytes, which is limited by the 16 bit window field in the segment header. With today's technology to service The Internet and other networks via transcontinental networks and even via satellites, this window size becomes an issue.[11] For example, if a geosynchronous satellite link is used to service the network, the round trip time of a segment, which includes the data and the ACK, would take roughly 300 milliseconds. This window size limits the connection speed to 1 mbps under favorable connections.[11] This limit occurs because of the time it takes to send a packet and receive the acknowledgement. With high bandwidth and high latency, the useable space in small windows are quickly exhausted. Therefore, the number of segments sent is limited on the time it takes to receive an acknowledgment.

One solution to this problem is to increase the maximum window size, but due to the header maximum size, the ability to increase this is limited.[11]

3.2 SLOW START

The slow start algorithm leads to performance problems. The first problem is the part of the algorithm which increases the window size by fifty percent with each ACK. The amount of time it takes to get up to speed on a high speed network can be really long. The equation to figure out the amount of time it takes to get up to speed is $R(1 + \log_{1.5}(DB/l)$, where R is the round trip time and DB is the connection speed and l is the segment length.[11] It is important to note that as the bandwidth goes up and the delay goes up, the amount of time it take to get up to speed drastically increases.[9] It turns out to be a lot of wasted bandwidth, which could be better utilized.

4. WINDOWS XP IMPLEMENTATION

The Windows XP Service Pack 2 TCP/IP stack includes standard features like the slow start algorithm. There are some features not found in a standard TCP/IP stack, which include algorithm changes for increased performance in high

loss and high delay networks, scalable window sizes, fast retransmission, fast recovery and selective acknowledgments.

4.1 LOSS AND DELY NETWORK PERFORMANCE

There are two major changes to TCP in Windows XP that increase the performance in high bandwidth and high delay networks.[4] The first change is the increase of the default window size. The default window size depends on the connection speed. If the connection speed is below 1 mbps then it is 8KB.[4] If the connection is between 1 mbps and 100 mbps then it is 17KB. When over 100 mbps, the default window size is 64KB.[4]

The second change is to the slow start algorithm. This change increases the number of segments sent right away to two instead of one. This increases the rate at which the slow start algorithm increases window size.

4.2 SCALABLE WINDOW SIZES

Windows XP supports scalable window sizes as specified in RFC[1] 1323. This allows the window size to be increased up to 1 GB in size.[4] Most importantly, it allows both sender and receiver to communicate with each other during the transfers to determine what is the best size of the window.[4] It is important to note that this option must be agreed upon by both parties.[4]

The messages sent to change window sizes are initiated by using a SYN segment. In the option field, the receiver or sender enters the size that it wants the scale of the window to be.[4]

4.3 FAST RETRANSMISSION AND FAST RECOVERY

As described earlier, when a receiver gets a segment out of order, the receiver pretends that it did not receive the segment. With fast retransmission, the receiver sends an ACK with the sequence number it was expecting instead of the sequence number of the next segment to send.[4] This increased the time in which TCP recovers from packet loss. Along with this, the segments which have not been received yet, do not need to be resent; therefore this is called fast recovery.

4.4 SACKS

SACKS or selective acknowledgements is the ability of the receiver to acknowledge segments, which are not the first segment in the window.[1] This allows the receiver to send specific information about what segments have been received and what segments need to be resent.[4]

SACKS are very helpful when dealing with large window sizes. Normally the entire window or a large majority of the window would need to be resent if a segment was lost, but now only the specified pieces need to be resent to the receiver.

1 Request For Comments are the standards documents for the Internet Protocol.

5. WINDOWS VISTA IMPLEMENTATION

The Windows Vista TCP/IP implementation code name Next Generation TCP/IP, has many new features, which include Receiving Window Auto Tuning, Compound TCP, ECN, and many other improvements to increase performance in high packet loss environments.[13]

5.1 RECEIVING WINDOW AUTO TUNING

In Windows Vista Receiving Window Auto Tuning, both parties need to agree on a window size using a scale factor. With use of receiving window auto tuning, the receiver continually monitors the incoming segments and makes modifications on a per connection basis.[2,3]

The auto tuning algorithm monitors the perceived bandwidth times the latency plus the application retrieve rate.[2,3] After all the information is collected, the receiving window is automatically updated to the best maximum size. The default range of the window sizes are 64 kb to 16 mb.

5.2 COMPOUND TCP

Compound TCP is another modification that monitors the bandwidth, latency and packet loss to determine how many segments can be sent at one time.[2,3] It aggressively increases the number of segments sent at one time when the network conditions permit it.[2,3] In addition, compound TCP makes sure that the modifications made do not effect other TCP connections.

It is important to note that this feature is not enabled by default on Windows Vista. This feature is enabled by default on Windows 2008 Server.

5.3 ECN

ECN or explicit congestion notification is used as an early warning detection to congestion. When a router starts to get congested, the router marks the segments with an ECN option. When this option is set, the sender and receiver backs off their transmission to hopefully prevent packet loss due to congestion.[3] If packet loss is prevented, the congestion control algorithms do not come into play.

Although Windows Vista supports this feature, most routers do not have this feature enable in practice.

5.4 HIGH PACKET LOSS ENVIRONMENT IMPROVEMENTS

There are five major changes made to Next Generation TCP/IP that improve the recovery of TCP due to packet loss. The first algorithm is NewReno Fast Recovery, which is a modified version of fast recovery. One major flaw with the normal fast recovery algorithm is that it only works well when one packet is lost.[3] NewReno is a modification that increases throughput when multiple packets are lost.[5]

When TCP enters a state of fast recovery, NewReno allows for new and unsent segments to be sent when an ACK is received.[5] This means while the holes in the already

transmitted data are repaired, new information is being sent to keep transfer rates up.[5]

The next change made to improve performance under high loss environments is an extension to SACK.[3] The extension allows a TCP connection, with the SACK option enabled, to make modifications to the sender's behavior when it has unnecessarily sent multiple segments.[5] This will generate less traffic because the sender will know when not to send multiple packets.[5]

Conservative Selective Acknowledgements allows for information to be stored on all the ACKS received on a per connection basis when in fast recovery.[3] It uses this information to determine which segments need to be sent, which in turn increases the recovery process.

Finally, the Forward Round Trip Timeout Recovery algorithm is used when segments are lost due to a sporadically changing round trip time.[3] When the round trip timer expires, F-RTO sends one packet to see if sporadic network latency is the problem.[3] If the packet is acknowledged, the round trip timer is not changed.

6. METHODOLOGY

The method we used to test the performance of the Next Generation TCP/IP and Windows XP TCP/IP stacks is similar to studies done by Jensen, McGregor and Gibson.[6,7] Their study used a network emulator, with which they could change the speed of the connection between nodes and could also change the propagation delay as well as the number of packets lost during the connections.

In our study, both wide area network (WAN) and local area network (LAN) conditions were tested. A WAN emulator called WANem, which is built on top of the Knopix distribution of Linux, was used to create reproducible network conditions to simulate WAN and LAN networks. The network setup is shown in by figure 4.

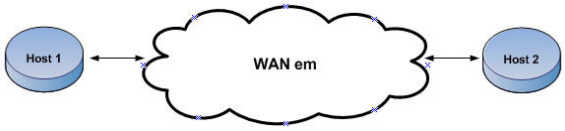


Figure 4 Network setup used.

Performance was measured using the transfer speeds and times from the default ftp clients used on each operating system. Each test case was done using transfers from XP to XP, Vista to Vista, XP to Vista and Vista to XP. To see how Windows performs with other operating systems, there were transfers between Linux to XP and Linux to Vista.

The file sizes transferred were 50 megabytes and 50 kilobytes. The reason we chose this is because it will take some time to transfer, but not all day for some of the tests as well as testing how fast each slow start algorithm increased the window size. We used the default window sizes for both Windows XP and Windows Vista. Windows XP uses a default window size of 17 kb for connections less 100 mbps and 64 kb for connections greater than 100 mbps. Windows Vista on the other hand uses Receiving Windows Auto Tuning, so there will be no default window size.

The network conditions were controlled using three parameters, which are connection speed, latency, and packet loss. The connection speeds used were 100 mbps, which is simulating LAN conditions, five mbps and one mbps. One and 5 mbps simulated connection speeds found when purchasing a typical connection from an internet service provider.

The latency delays we used was 1 ms, which simulates a LAN environment, 70 ms, which is found in a U.S. coast to coast transfer and finally a 300 ms delay, which simulates a satellite connection delay.

The final network condition change made was the number of packets lost. Zero packets lost will simulate LAN conditions, 1 in 100, 1 in 12 and 8 in 100 packets simulate WAN conditions.

The last test was a combination of high bandwidth usage or connection speed and high latency. These are simulated conditions that are very hard on the TCP algorithms and exploit the major weaknesses in TCP.

To ensure the performance differences weren't due to the amount of resources Windows Vista uses compared to Windows XP, software called nLite and vLite was used. [9,10] This software allows a user to choose which software is to be installed on a clean version of Windows. nLite is the version used for XP and vLite is the version of the software used for Vista. For example, one can remove all the games and fonts that come with Windows Vista and XP.

7. RESULTS

We expected to see significant improvements across the board when doing TCP connections with Windows Vista. What we found is there is little to no improvement when doing small amounts of data. We also saw little to no improvement over LAN conditions. Where we began to see a notable difference is when bandwidth was low and latency was high, as well as high packet loss. We felt that this was to be expected.

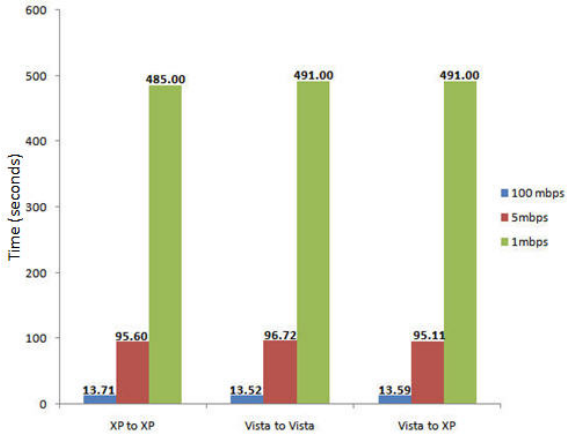


Figure 5 Transfer times for XP to XP, Vista to Vista and Vista to XP for connection speeds of 100 mbps, 5 mbps and 1 mbps using a 50 mb file.

In Figure 5, the test results of connection speeds of 100 mbps, 5 mbps and 1 mbps are shown. We found little difference between Windows Vista and XP in transfer times for connections of 100 mbps and 5 mbps. Window XP did have a decrease in transfer time with transfer speeds of 1 mbps.

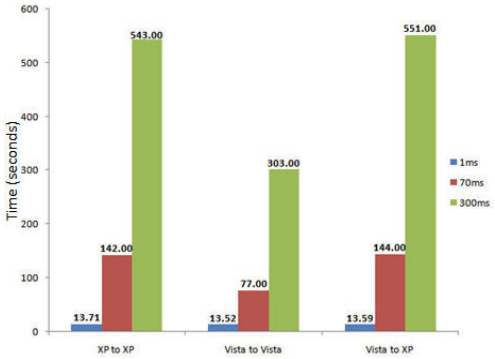


Figure 6 Transfer times for XP to XP, Vista to Vista and Vista to XP for latencies of 1 ms, 70 ms and 300 ms using a 50 mb file.

When testing the latencies of 1 ms, 70 ms and 300 ms, we saw a significant decrease in transfer times by Windows Vista. The results of these test cases are shown in Figure 6.

Windows Vista did show a decrease in transfer time when packet loss occurred. However, WANem does packet loss randomly, so the results of each test was significantly different. After repeating these test several times, we felt the results of our tests were not significant because the result were not consistent.

The result of our hybrid test case of both latency and bandwidth conditions being changed is shown in Figure 7. Windows Vista shows a significant decrease in transfer times under these test cases. The only exception is the test case of 1 mbps and 300 ms.

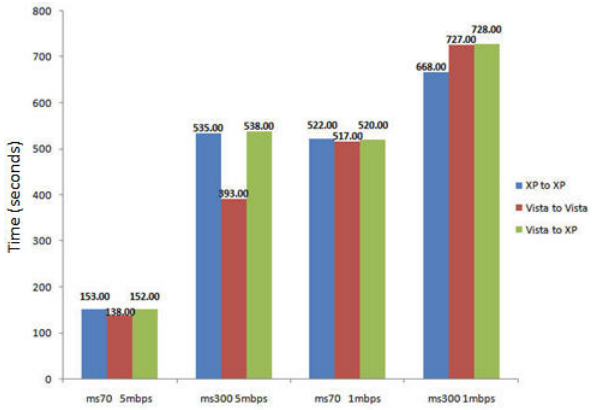


Figure 7 Transfer times for XP to XP, Vista to Vista and Vista to XP for latencies of 1 ms, 70 ms and 300 ms with connection speeds of 5 mbps and 1 mbps using a 50 mb file.

We also found that in most cases, transfers between XP to Vista and Vista to XP took the same amount of time or a similar amount of time to transfer between XP to XP. There were some cases where XP to Vista had faster transfer times then XP to XP which was in high latency situations, low bandwidth and high packet loss.

Finally, we found that Linux to Linux transfer times were similar to that of Vista. As with the test cases using Vista to XP and XP to Vista, we found Linux to XP transfers to perform at similar speeds as XP to XP transfers. It is also important to note that the Linux to Linux and Linux to Vista transfers were significantly shorter than Linux to XP when introducing high latency and high packet loss.

8. ANALYSIS

There are two important points learned from the results of the experiments. The first key point is that Windows Vista has improvements in transfer time in all test cases besides the case of 1 mbps bandwidth and LAN conditions. This shows that in fact that the modifications done to Windows Vista improved transfer times in cases of packet loss, high latency and high bandwidth connections.

The reason for these improvements are because of the new features like Receiving Window Auto Tuning and the Slow Start algorithm changes. In the case of high latency, having an increase in window size as well as a dynamically changing window, there can be an increase in bandwidth used. When experiencing high packet loss, the improvements to the Slow Start and fast recovery algorithms decrease the amount of time spent in the startup and recovery phase of TCP connections.

In the case of slower connections, Windows Vista's Next-Generation TCP/IP stack was more concerned about high bandwidth connections then low bandwidth connections. Unfortunately, when there is a combination of low bandwidth and high latency, the improvements of Windows Vista actually creates slower transfer times, negating any improvement made to increase the bandwidth usage. We speculate that the increase of overhead needed for Next Generation TCP/IP is the cause of this result.

We expected not to see a significant increase in performance under LAN conditions. This is because 50 kb and 50 mb files take very little time to transfer under LAN conditions.

The second point learned from the study is the transfer time observed in Vista to Vista transfers can be created in other operating systems outside of the Windows family. The data collected with Linux to Vista transfers shows that these improvements can be seen using other operating systems.

9. CONCLUSION

In conclusion, the Windows Vista Next-Generation TCP/IP stack does show improvements in high latency, high bandwidth and high packet loss environments. Windows Vista does suffer in low bandwidth environments. As the connection speeds of LANs and WANs go up, this problem is less of an issue.

To get even more of an idea of how Window Vista performs in various environments, some future work would be to increase the set of network conditions that are tested. It would also be useful to increase the set of TCP/IP stacks that are tested with Windows Vista.

Reference List

[1] Comer, Douglas E. (2006). *Internetworking with TCP/ IP: Principle, Protocol, and Architecture*. (5th Edition). Upper Saddle River, New Jersey: Pearson Education, Inc.

[2] Davies, Joseph. (2006, February). New Networking Features in Windows Server 2008 and Windows Vista. *TechNet Magazine*. Microsoft Corporation.

[3] Davies, Joseph. (2005, November). Performance Enhancements in the Next Generation TCP/IP Stack. *TechNet Magazine*. Microsoft Corporation.

[4] Davies, Joseph. (2006). *TCP/IP Fundamentals for Microsoft Windows*. Microsoft Corporation. Microsoft Corporation. (2003). *Microsoft Windows TCP/IP Implementation Details*. Microsoft Corporation.

[5] Floyd, S. and Henderson T.(1999). *NewReno Modification to TCP's Fast Recovery* (RFC 2582). U.C. Berkley, CA.

[6] Gibson, Bill. (2000). *Congestion and Delay Hamper the Global Internet*. Boulder, CO: Niwot Networks, Inc.

[7] Jansen, Sam, & McGregor, Anthony. (2005). Comparative Performance of TCP Stacks. *Lecture Notes in Computer Science*. (3431 Vol., pp. 329-333). Springer Berlin.

[8] Information Science Institute. (1981). *Transmission Control Protocol* (RFC 793). Marina del Ray, CA.

[9] Nuhagic, Dino. (2008). Nlite. www.nliteos.com.

[10] Nuhagic, Dino. (2008). Vlite. www.vlite.net.

[11] Partridge, Craig and Shepard, Timothy. (1997). TCP/IP Performance over Satellite Links. *IEEE Network*. September/October.

[12] Tanenbaum, Andrew. (2002). *Computer Networks.* (4th Edition). Upper Saddle River, New Jersey: Pearson Education, Inc.

[13] Tolley Group Inc. (2007, June). Enhanced Network Performance with Microsoft Windows Vista and Windows Server 2008. *White Paper*. (Doc. 207180).

Analysis of Using Force Fields with a Pen Input Device

Clay Smith
Computer Science Student
Winona State University
Winona, MN 55987
Crsmith3923@winona.edu

ABSTRACT

This paper conducts a usability study with 30 participants on the concept of graphical user interface "force fields." The study compares usage with a tablet PC (Personal Computer) pen input device with and without force fields enabled. The results were an average of a .32 seconds increase in selection speed with a 1% decrease in accuracy. Additionally, 18 of the user's impressions were negative, while only 9 of the user's impressions were positive.

Keywords

Pull-down menus, Menu navigation, Selection, Fitts' Law, Input devices, "Force fields", tablet, pen, GUI, usability

1. INTRODUCTION

Force fields are a concept of helping the user steer the mouse cursor to increase their selection times in a graphical user interface. Force fields have already proven useful for touchpad and track point devices. Our research indicates whether force fields prove useful for the tablet PC pen input device.

This paper talks about the usability of the pen with and without the force fields enabled. The term 'usability' is used throughout our paper. Usability is used to mean an increase in the user's selection speed and accuracy.

1.1 Force Fields

The concept of "force fields" originated from David Ahlström's paper entitled "Modeling and Improving Selection in Cascading Pull-Down Menu's Using Fitts' Law, the Steering Law and Force Fields." Ahlström develops an algorithm to help speed the time it takes for the user to select items in a menu by using a concept called force fields. The force field takes the user's mouse cursor position and processes it through an algorithm that Ahlström develops. The result of the algorithm is a new cursor position. The algorithm is designed to reduce the difficulty of the cursor steering task, which in turn speeds up the user's selection time [1].

Force fields that help a user navigate through a menu can be best explained with an illustration. Let's take a look at Figure 1.

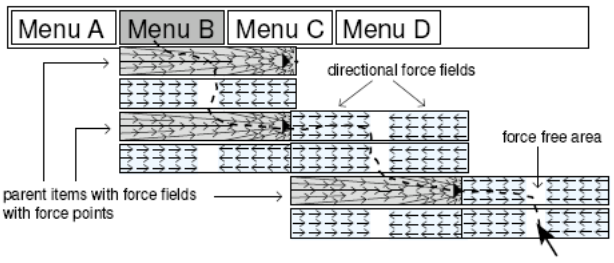


Figure 1. Depiction of force fields [2].

Figure 1 illustrates a menu with two different kinds of force fields. The first type force field in Figure 1, called a force field with a force point, is the menu item with a large arrow on the right hand side that allows you to access submenu items. The entire field of force leads to a single force point, which is located on the middle right side of the menu. As the user moves their cursor, their cursor is pulled towards the force point. The idea of the force point is to help the user access submenus within the linear menu more efficiently. The second type of force field is the directional force field. As illustrated in Figure 1, the directional force field lacks a force point. Instead, there are two force fields working to push the cursor towards the center of the menu. As the cursor reaches the center of the menu, there is a force free area where no force acts on the cursor. The idea of the directional force field is to speed user's selection times by keeping the cursor in the center of the item.

The selection time for the force field enhanced menus have proven faster than the standard menus for both touch pad and track point devices. Benefits include a 30.6% speed increase for novice touch pad users and an 18.3% speed increase for novice track point users. On average, force field enhanced menus were shown to have an 18% speed increase in comparison to the standard menu [2].

1.2 Force Field With Pen Input

The concept of the force field enhanced menus have already been shown to increase the user's selection time for both touch pad and track point users. Extrapolating on those results, our conjecture is that a force field enhanced menu increases user selection times and improves the accuracy for users of the Tablet PC pen input devices.

Stated more formally, the hypothesis of this research is that the increase of usability of touchpad and track point devices that

comes with a force field enhanced menu provides an increase of usability when combined with the tablet PC pen input device.

2. METHODOLOGY

2.1 Informal Usability Pretest

The pretest was an informal test that consisted of a quick approximation of how forces work in software. The pretest was conducted in order to get a better understanding about how people may react to the idea of force fields in a linear menu. The pretest was implemented using our tools and 10 random college student participants to test our implementation and give us their subjective impressions of the system. The software consisted of a simple menu implementation of our approximation of force fields.

Each participant was first explained the concept of force fields as well as their benefits. Each participant used the force enhanced menu with both a touch pad and a pen input device. The participants then made mental notes of both good and bad qualities of using the pen combined with the force enhanced menu. After they had finished, they were asked for their subjective impressions of the system.

2 of our 10 participants exhibited a strong dislike for the force field enhanced menu combined with a pen input device. Reasons cited were that it lowered their threshold of accuracy, and that the mouse cursor became too jumpy around the pen. One of our participants had neither strong likes nor strong dislikes for the pen combined with the force field enhanced menu. 5 of our users cited both good and bad qualities for using the force field enhanced menu with the pen. The good qualities included that the force enhanced menu was helpful in selecting items, the pen becomes easier to steer, and that the mouse doesn't stray. The bad qualities included that the cursor was too jumpy and the system was unfamiliar. 3 of our users cited no ill effects when using the force field enhanced menu with the pen input device.

Our pretest provided an opportunity to learn a number of things. First, the force field combined with the pen could yield some benefit. Second, there was an overwhelmingly positive attitude towards force fields. Part of this positive attitude may have been from telling the user the supposed benefits of the force fields before the user uses it. This may suggest to the users mind to expect benefits even if they don't personally experience any. The real test was designed not to tell our users the benefits of the system before they had used it.

2.2 Formal Usability Test Design

The usability test was designed based off of the University of Texas at Austin's usability testing guidelines [3]. Our first plan was to design our usability test. Our metrics for usability became the user's selection speeds and accuracy. Then a timeframe to complete the usability test was designed. Within one week the design of the usability test was conducted. The next week the implementation of the usability test was complete. The third week was the week the usability study was conducted.

2.2.1 Development of the Usability Test

The first part of our usability test was to write our introduction survey questions that the users would be asked before they took part in our test. The questions asked were their age, hours of computer and tablet pen usage per week, their major, whether they had corrected or normal sight, and whether or not they have had prior experiences with the concept of force field enhanced menus.

Then a program was developed for allowing the user to execute various selection tasks. The program starts by briefing the users on our study and giving them instructions to complete the task. Verbal instructions were given as necessary when the user had questions or became confused.

The program that was developed then had users take part in 3 rounds of selecting items in a menu. Each round required the user to select a total of 13 unique items in the menu. The order that the user is required to select these items is randomized by our program, and the user ends up selecting each item from the menu once. The first round had the user use a touch pad for the input device and the menu with the force fields enabled. This round was used to give the user a basic feeling for the force field enhanced menus that yield a significant usability increase [4]. The second round had the user use the Tablet PC pen with the force field disabled as a standard to compare the force field enhanced menu with the pen against. The third round had the user use the pen with the force fields enabled. This round was used to gather data on the usability of the pen combined with the force enhanced menu.

In a given round, the user is asked first asked to press a 'Start' button to begin their selection task. Once the user selects the 'Start' button, they are then asked to select a specific item from the menu. When the user clicks on the start button, the program starts recording the time it takes for the user to select the item in the given task. The program also records if the user selects the correct item. After all items are selected in a given round, the user is given a unique identifier and the user's selection times and accuracy are then recorded to file with a corresponding user identification number.

Upon completion of the test, the users were then presented with exit survey questions. The questions asked were whether the user felt the force field enhanced menus helped them use the pen, what they liked about the force enhanced menu with the pen, what they didn't like about the force enhanced menu, and any suggestions they had in order to improve the pen combined with the force enhanced menu.

2.2.2 Conducting the Usability Test

Winona State University was the location where our usability test was conducted. Winona State University proved to be the most efficient place to find subjects. All of our subjects were college students. Participants were involved in the study by simply approaching them and asking them to participate in the study.

The usability test included a total of 30 random participants. They were each asked the entry and exit survey questions and conducted the usability testing program. The test was conducted in a day, starting at 11 in the morning to 8 at night.

3. RESULTS

3.1 Selection Time Results

Figure 2 below shows the difference between the individual user's selection times with the pen without forces vs. the selection time of the user with the pen with forces. If the difference is positive, that means that overall the pen with forces helped the user by the amount of seconds shown. If the difference is negative, it means that the pen with forces hindered the user by the amount of seconds shown.

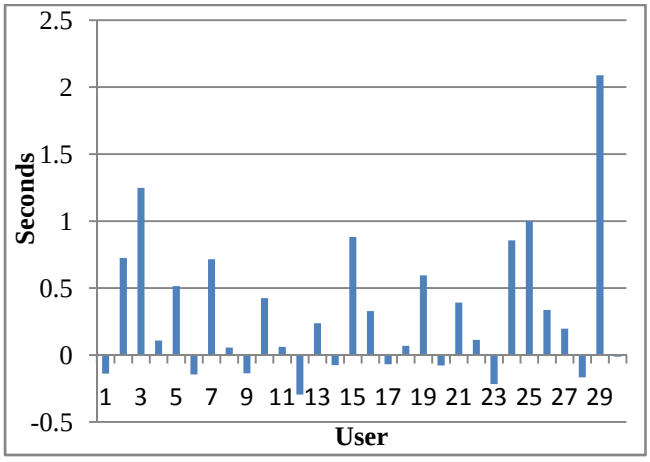


Figure 2. Selection Time Differences.

Table 1 below shows the extreme values of the differences from Figure 2. It contains the lowest difference and the highest difference, as well as the average difference.

Table 1. Selection Difference Extremes.

	Difference
Low	-.296 sec
High	2.088 sec
Average	.32 sec

3.2 Accuracy Results

Figure 3 below shows the difference of accuracy between the pen with forces and the pen without forces. The accuracy was rated on a scale of 1 to 100. If the difference is positive, that means that the user was helped by the pen with forces by the percent shown. If the difference was negative, that means forces hindered the user by the percentage shown. If there was no difference, the pen with forces neither helped nor hindered the given user.

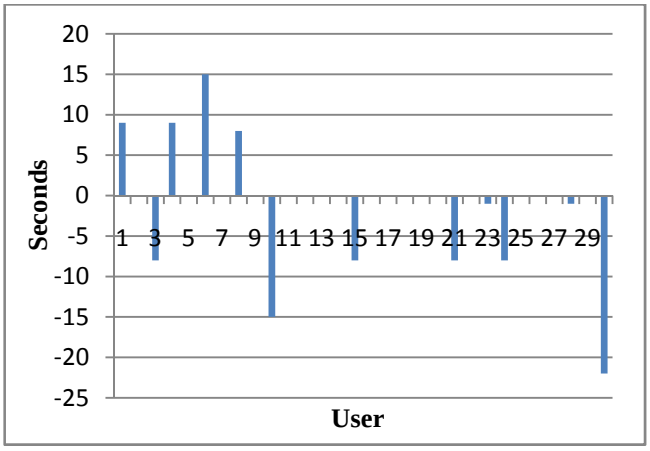


Figure 3. Accuracy Differences.

Table 2 below shows the accuracy extreme values from Figure 3 above. It contains the lowest difference and the highest difference, as well as the average difference.

Table 2. Accuracy Difference Extremes.

	Difference
Low	-22%
High	15%
Average	-1%

3.3 Entry Survey Questions

28 out of 30 participants were found to have either corrected or normal sight. The youngest person in the study was 18 years old, the oldest person was 50 years old, and the average age of participants was 21 years old. The average amount of time spent on the computer by participants was 33 hours per week. On the low end was a user using the computer for only 4 hours per week, and on the high end was a user using their computer for 112 hours per week.

20 out of 30 of our participants said they use the tablet pen for 0 hours per week. 6 of our participants used the pen for 2 hours per week. 1 of our participants used the pen for 3 hours per week, and 1 of our participants used the pen for 4 hours per week.

3 of our participants said they have heard about force enhanced menus, and the 27 other participants said they have not heard about force enhanced menus.

The participants in our study belonged to a wide variety of majors. 1 was on the premed pathway, 2 were nursing majors, 1 was an accounting major, 2 were business majors, 5 were engineering majors, 2 were undecided, 4 were computer science majors, 1 was

a Spanish major, 2 were chemistry and biochemistry majors, 1 was a public relations major, 2 were biology majors, 1 was an environmental science major, 1 was a graphic design major, 1 was an elementary education major, and 2 were finance majors.

3.4 Exit Survey Questions

18 out of 30 participants said that the pen combined with the force field enhanced menu did not help. 9 out of 30 participants stated that the pen combined with the forces did help them.

User's stated that they didn't like how the mouse cursor was jumpy and didn't stay where the pen was. Another common complaint was that the pen was difficult to click with.

User's stated that what they liked about the pen combined with the force field was that it may have saved a little bit of time and it was nice if you could anticipate it. User's stated that they thought it helped direct the mouse cursor in the correct direction.

Finally, user suggested improvements to the system include both making the force fields more and less powerful which was suggested by different users, making the forces fields less jumpy, making the pen click easier, and making the pen smoother.

4. ANALYSIS

4.1 Selection Time Analysis

Figure 2 shows the selection time differences of users 1 through 30. According to the chart, there is an interesting trend taking place. It appears that for 20 of our 30 participant had an increase in selection times using the pen with force fields. Out of those participants, 8 of our users had marginal gains (greater than .5 second difference), while 13 of our participants had slight gains (.5 seconds or lower). 10 of our users were impeded by the force field combined with the pen, but out of the 10 of them, none of them were impeded by more than .5 seconds. Therefore, according to Figure 2 it appears that the pen increased the selection times of more users than decreased the selection times of users. Out of the selection times that it did impede users, it never impeded users for more than .5 seconds. Therefore, user's selection times increased more than they decreased.

Table 1 seems to agree with our above analysis. The high time is a lot larger than the low time, and on average there is an increase in selection time using the pen with forces.

4.2 Accuracy Analysis

Figure 3 is used to give an accuracy analysis. This figure shows the accuracy differences, with positive values meaning the pen with forces was more accurate. According to Figure 3, it looks as though it actually decreased user's accuracy with the system. Only 4 of our 30 users had any gains in accuracy with the pen combined with forces. For 18 of our users, the accuracy stayed the same for both pen with forces and pen without forces. For 8 of our users, accuracy was decreased. It appears that while for the majority of our users the accuracy was neither increased nor decreased, there were more users that had a decrease in accuracy rather than an increase in accuracy.

According to Table 2, the greatest decrease in accuracy was by 22%, while the greatest increase in accuracy was only 15%. On average, it shows that there is actually a -1% decrease in accuracy. So, while there is a decrease in accuracy with the pen combined with forces, the accuracy decrease is minimal.

4.3 Subjective Data Analysis

Surprisingly, the subjective impressions did not quite line up with the qualitative data. While there are some quantitative improvements, the improvements on the way the user subjectively see's them aren't as large. 18 of our participants though that the pen with forces didn't help at all. The reasons stated were that the pen was too jumpy and they didn't like the loss of control caused by the force fields. 9 of our users said it did help, saying that the pen with force field enabled menus saved some time if they could be anticipated. Let's keep in mind that 20 of our 30 participants never use the pen, and 27 never used force fields before, so perhaps this is a concept that could be gotten used to. Perhaps what may be required is to tweak the strength of the force field to work better with the pen. The strength of .8 was used based off of Ahlström's recommendation [2]. With the pen, it may be required to figure out a way to smooth out the forces because of the jumpy motion of the cursor may just cause the user to think the software is buggy and therefore dislike the force field concept. Therefore there needs to be work done to have the user appreciate the concept on face value.

5 CONCLUSIONS

Based upon our analysis of quantitative data, it seems that our hypothesis was correct. The selection times of the users were faster with the pen combined with forces for the majority of our users, showing an increase in selection times. While accuracy was worse with the pen combined with forces, the average the accuracy was impeded by only 1%, and therefore the accuracy decrease is not significant.

Based upon our analysis of the subjective data, the results are not as clear. While there are some quantitative improvements, 18 out of 30 of our user's impressions were negative, and only 9 were positive. While there are quantitative improvements, there still needs to be future work done on getting the user to view the system more positively.

However, developer's can implement the force field concept on Tablet PC's, mobile devices, and other devices supporting pen input and yield an improvement in the usability of the system, they just need to worry about upsetting user's subjective impressions of how the system should work.

6 ACKNOWLEDGMENTS

I would like to thank the Winona State Computer Science Department, Dr. Nicole Anderson and Dr. Francioni for guiding me through the research and writing process, and Davyion Crossland for moral support.

7 REFERENCES

[1] David Ahlstrom, Martin Hitz, and Gerhard Leitner. 2006. An Evaluation of Sticky and Force Enhanced Targets in Multi-Target Situations. NordiCHI 2006: Changing Roles, 14-18 October. Oslo, Nordway.

[2] David Ahlstr¨om, Rainer Alexandrowicz, and Martin Hitz. 2006. Improving Menu Interaction: a Comparison of Standard, Force Enhanced and Jumping Menus. CHI 2006 Proceedings – Menus. April 22-27. Montréal, Québec, Canada.

[3] The University of Texas at Austin. 2007. Usability Testing. http://www.utexas.edu/learn/usability/

[4] David Ahlstrom. 2005. Modeling and Improving Selection in Cascading Pull-Down Menus Using Fitts' Law, the Steering Law and Force Fields. CHI 2005 - PAPERS: Basic Level Interaction Techniques, April 2–7 ı Portland, Oregon, USA.

ZigBee Enabled Device Location through Trilateration

Kelly Torkelson
Winona State University
314 West 4th Street
Winona, MN 55987
(651) 792-6062

KATorkel4715@winona.edu

ABSTRACT
In large-scale wireless networks, we often times have a mechanism for determining the position of a node based off of other nodes' locations. This has been done for 802.11 and in cellular networks, but not yet for the ZigBee wireless standard. Perhaps this is due to the fact that it is a fairly new technology which has yet to gain popularity. We will explore different ways in which we can locate a node in a ZigBee network using its distance to other nodes with known locations. The methodology we will use is trilateration, which uses three devices to estimate the position of another device in a two dimensional plane.

General Terms
Algorithms, Measurement, Performance, Experimentation.

Keywords
Trilateration, ZigBee, Localization, Positioning, Wireless Networks, RSSI

INTRODUCTION
The focus of this paper will be on the means by which we determine the location of a ZigBee enabled device in a network. Using trilateration, we can calculate an estimated location based on received signal strengths from other nodes in the network to the unknown node.

ZigBee enabled devices are made to be low-power and low-data rate. This makes them very useful in large-scale networks where minimal data needs to be transferred at a given time. ZigBee operates on the Industrial, Scientific, and Medical (ISM) band of the radio spectrum, which is unlicensed, so this helps ZigBee enabled devices remain inexpensive. [2]

Although Bluetooth and ZigBee are both designed for Wireless Personal Area Networks, ZigBee is intended for much longer ranges. The modules we will use for our experiments have an indoor range of up to 40 meters and an outdoor range of up to 120 meters. ZigBee is also made to be power efficient, making it useful in many situations.

Proceedings of the 8th Winona Computer Science Undergraduate Research Seminar, April 30, 2008, Winona, MN, US.

The ZigBee standard is used for Wireless Sensor Networks where long ranges are desired. The major areas where these networks are used is in home and industrial automation. An example of home automation is putting a temperature sensor in the refrigerator to make sure it stays cold enough. For industrial automation, motion sensors can be used to know if the lights should be on or off.

ZigBee is an open wireless standard that is managed by an organization called the ZigBee Alliance. Many companies are members of this organization including Ember, Honeywell, and Texas Instruments. One major advantage of ZigBee is that it allows for mesh networks. A mesh network is self-forming and self-healing, meaning that a node can join or leave the network at any given time and the network is able to adapt to the change. This may make it hard to know what nodes are on the network and where they are located. So, it would be useful to be able to find the positions of multiple nodes while only needing a few nodes with known positions. This is where trilateration using estimated distances can be beneficial. [1]

When only considering two dimensions, trilateration is normally carried out using three devices with known positions in an attempt to pinpoint the position of another device. Distances between devices can be estimated through various techniques, such as the received signal strength. Using the positions of the known devices, the estimated distances based off of the received signal strength, and trilateration, a position can be estimated for the device with the unknown location. Trilateration is done by creating three circles with center points at the locations of the devices and radii that corresponds to the signal strengths from those devices to the unknown device. The location of the unknown device can be estimated by figuring out where the three circles intersect. This will be explained in more detail in the Methods section. [5, 6]

Due to the dynamic nature of ZigBee networks, it would be useful to have the ability to put in place a few devices with known positions and then use those devices to calculate the positions of devices as they join the network. This is why trilateration is a relevant topic in the field of ZigBee.

Being able to identify the location of a device is a problem that is seen in multiple areas. One approach we took in figuring out how to implement a solution to this problem was to look into these similar problems. This included how trilateration of nodes is done in a cellular network and also in an 802.11 network. We found that studies have been done in these two areas, but we didn't find comparable studies done with ZigBee networks. We will discuss the research we found for cellular network and 802.11 networks in

the next section.

Overall, we will show that it is feasible to use trilateration techniques to figure out the position of a node to a certain degree of accuracy. We will show this on a small scale and speculate on how this implementation could be refined and applied to large-scale networks. Throughout the rest of our paper, we will review related work, explain the methods we used to carry out our experiment, and summarize the results that we obtained from the experiment.

RELATED WORK

In an 802.11 network, trilateration can be done to figure out the position of a node. This is possible because the received signal strength of a device is exposed by the 802.11 standard. Once we can find out the signal strength from three nodes to the node, we can use trilateration to estimate a location. [4]

Cellular networks also use a similar way to determine the position of a cell phone user. The received signal strength is measured from base stations. After this information is acquired, trilateration can be used to figure out the location of the cell phone user. [3]

HYPOTHESIS

Given three ZigBee enabled devices with known positions, it is possible to estimate the position of a fourth ZigBee enabled device within two meters of accuracy using trilateration.

METHODS

For our experiment, we first implemented trilateration and tested it using simulated data. To do this we used standard trilateration techniques in a Java application. The application takes in the positions of the three known nodes and the signal strengths from them to the unknown node. It then outputs the estimated position of the unknown node.

The way we implemented trilateration was by using the intersection points of the circles created by the signal strength (Figure 1 in the appendix). The equation of a circle is $(x-h)^2+(y-k)^2 = r^2$ where (h,k) is the center point and 'r' is the radius. We will take the equation of the first circle and subtract it from that of the second circle. Solving for 'x' will give us the estimated 'x' coordinate of the unknown node. We then substitute 'x' back into the equation for the first circle, giving us a circle which passes through both intersection points of the first and second circles. We can find where this circle and the third circle intersect by setting them equal to each other and solving for 'y'. Now we have the 'x' and 'y' values of the estimated position.

When we set up ZigBee networks to test our positioning methods, we envisioned the devices in a coordinate system. The units that we used for this system were meters. This scheme made it easy for us to estimate the location of one node and see if our estimation was close to the actual distance. The reason that we chose to design our own coordinate system, as opposed to using one such as GPS, is because we feel that it will be more accurate and therefore yield better results.

Once we had our network of four ZigBee enabled devices set up, we recorded the location of each device using our custom coordinate system. We then queried for the Received Signal Strength Indicator (RSSI) from the last received transmission from each of our three known devices, or landmark devices, to the unknown device. We took the positions and RSSI values and entered them into our trilateration application and recorded the estimated location that it returned.

Finally, we saw how close the estimated location from our application was to the location in the coordinate system. We did this process multiple times and in different environments while carefully recording all of our results in a table. This table shows us how accurate our estimated locations were to the actual locations, making it possible to answer the question that we asked in our hypothesis. For the scope of this experiment, we will exclude testing different interferences.

Trilateration Data Representation

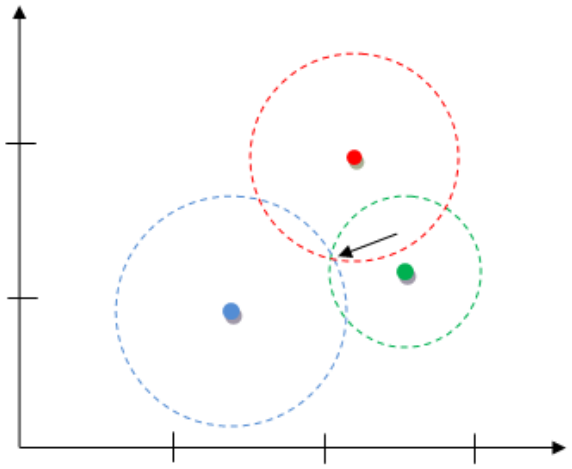


Figure 1. Intersection of signal strengths of landmark nodes

Figure 1 shows three devices (red, blue, green dots) and their received signal strength to an unknown device (respective dashed circles). Since we don't know in which direction the unknown node is located from any of the three nodes, we draw a circle to show that the unknown node is theoretically located somewhere along the circumference of the circle. The arrow shows the intersection of the three signal circles, which is the estimated location of the unknown device. Even if the circles don't intersect at once point, our equation can still estimate the location.

Before we could actually start our experiments, we had to set up all of our devices. We used a configuration tool called X-CTU to do this. Through this tool, we can write firmware, read parameters, and send commands. On the three landmark (known) devices we loaded the XB24-B ZNET 2.5 ROUTER/END DEVICE AT firmware and on the unknown device we loaded the XB24-B ZNET 2.5 COORDINATOR AT firmware. To communicate with the devices, we used X-CTU to send AT commands.

When setting up our coordinate system, we used a 30 meter tape measure. We put one landmark node at the origin, another along the positive side of the x-axis, and the last one somewhere else in the fourth quadrant. The unknown node was also placed in the

fourth quadrant of our coordinate system.

For each of the landmark devices, we set the destination address to the serial number of the unknown device using the ATDH and ATDL (destination high and destination low) commands. We set the destination address of the unknown device to one of the landmark node's serial numbers, sent data to the node, and then queried the node for the received signal strength of the transmission. We got the signal strength by sending the command ATDB which returns the signal strength. We repeated this process for both of the other landmark devices. In order to get the received signal strength from a device, we had to first receive something from it. To do this we attached a serial loop back adapter to the node so that the unknown node could send data and the landmark node would send it back. For each test run, we carefully recorded the position of each node and the received signal strength from each of the landmark nodes to the unknown node.

We decided to experiment in two different environments, one outdoors and one indoors. To do the experiment outdoors, we went to a vacant parking lot in order to test over long distances. For the inside experiment, we went to an event hall on campus.

During the experimenting we did outdoors, we put the devices at different positions within an area of 60 square meters. We used a device with a wire antenna for the unknown device (Figure 2) and devices with RPSMA, or omni-directional, antennas (Figure 3) for the three landmark nodes. Due to the nature of the RPSMA antennas, we found that we were unable to communicate with the landmark nodes unless they were off the ground. To handle this we elevated the nodes at a height of 1.65 meters, which allowed the devices to communicate with each other.

When doing our inside experiments, we used devices with wire antennas for all of the nodes. To facilitate measuring the positions of the devices, we put tape markers on the floor at each point in the coordinate system. This allowed us to do more tests runs and in a shorter amount of time. We did this testing in a 7 meter by 3.5 meter area to see the feasibility of trilateration using devices that are within close range of each other.

For both experiments, we used two devices hooked up to laptops through USB and two devices with the serial loop back adapters. We had one of the devices that was hooked up to a laptop as the unknown device so that we could send commands to the other devices and get the data we needed. The USB devices were powered through USB and we had 9V batteries for the serial devices.

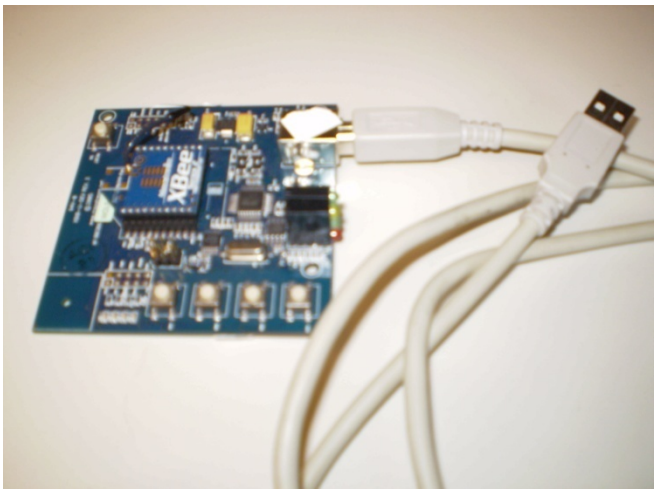


Figure 2. USB ZigBee enabled device with wire antenna

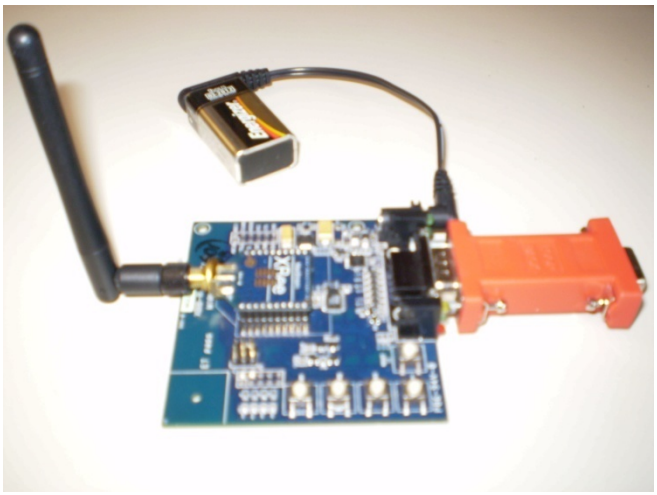


Figure 3. Serial ZigBee enabled device with RPSMA antenna and serial loopback adapter

RESULTS

When collecting data and recording the results, we made sure to be very careful and meticulous to decrease the chance of error in our resulting data. We kept the outdoor and indoor results separate from each other so we could distinguish between the two.

We needed to figure out how the received signal strength correlated to distance. For both the indoor and outdoor experiments, we took all of the distances we had and the signal strengths associated with each of these distances. We used Microsoft Excel to populate a scatter plot with this data and then added a trend line to it. We used the equation for the trend line to

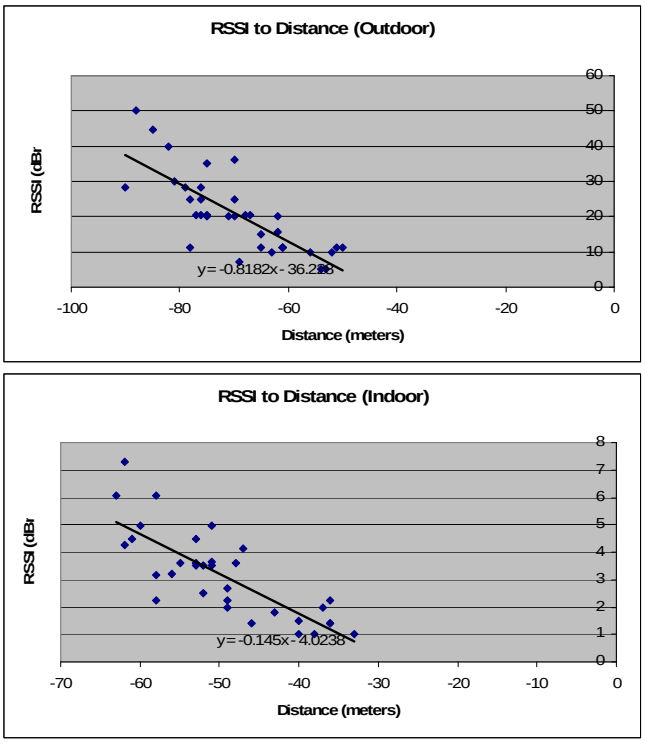


Figure 2. Trend line showing how RSSI values roughly correspond to measured distances.

get a distance that best correlates to a given RSSI value. Figure 2 shows the scatter plots produced from both the indoor and outdoor experiments. It also shows how a trend line can be used since it was clear that there was a relationship between the RSSI values and their corresponding distances.

Once we had an equation to convert an RSSI value into a distance, we were able to calculate the estimated position of the unknown node using our trilateration equation. We used the distance formula ($\sqrt{((x_2 - x_1)^2 + (y_2 - y_1)^2)}$) to figure out what the difference between the actual position and our estimated position was.

For each experiment, we recorded the positions of all four nodes and their antenna types, the received signal strength from each landmark node to the unknown node, the estimated position of the unknown node, and the distance difference between the actual and estimated positions. We kept all of this data in a spreadsheet to make it easy to read and analyze (see Tables 1 & 2 in the appendix).

ANALYSIS

After doing our experiments and recording all of our data, we analyzed our results. To do this we look at the amount of error in the indoor and outdoor experiments and also considered the area in which we tested for each of the experiments.

We found that in the outdoor experiment, the distances were off from the actual positions of the unknown node from 4.07 meters to 14.76 meters. The average difference in the distances between the actual node and the estimated position was 7.71 meters.

For our indoor experiment, the distances between the actual node and the estimated positions ranged from 0.12 meters and 3.36 meters. The average distance that we were off from the actual position of the unknown node was 1.56 meters.

Looking at the data from our outdoor experiment, we noticed one resulting distance that seemed to be an anomaly. This was the distance difference of 14.76 meters. The rest of the numbers were between 4.07 and 9.42 meters, making 14.76 meters seem like an error or an abnormal case. If we were to exclude this number, our average difference in distance would be 6.48 meters.

From our results, we demonstrate our hypothesis only to be true in our indoor experiment. We were able to estimate the position of a device within 2 meters of accuracy in this environment. However, we found that we were unable to meet our hypothesis in our outdoor experiment. More work in this area could help improve the overall accuracy of locating a device in a network.

FUTURE WORK

Since this is an area where not too much research has been done, it would be beneficial to have work carried out in the future. To get a better average accuracy, we could try using more than three landmark nodes. In this situation, we would do trilateration between three devices at a time and then average those results. Experiments could also be done to compare the accuracy of different antenna types and also different types of ZigBee modules. One area where we decided to not look into was how interference impacts the ability to use trilateration with received signal strengths. Also, exploration into different ways to convert RSSI values into distances and different implementations of trilateration could be very beneficial.

CONCLUSION

Throughout this paper, we discussed the possibility of using trilateration to estimate the location of a device in a network. We used the received signal strength from each of three landmark nodes and correlated the value to a distance. We used the locations of the devices and the estimated distances to do the trilateration. From this we came up with an estimated position of an unknown node. We found that by using three ZigBee enabled devices, we could successfully estimate the distance of a fourth device within 2 meters of accuracy in our indoor experiment. However, in our outdoor experiment, our estimated location was further than 2 meters from the actual position of the unknown node. Further work and research would be very beneficial to this field and could potentially find solutions to this problem that yield better average accuracy.

ACKNOWLEDGEMENTS

We would like to thank Chris Popp and Digi International for the use of their equipment in our experiments.

REFERENCES

[1] Baker, Nick. "ZigBee and Bluetooth strengths and weaknesses for industrial applications." Computing & Control Engineering Journal 16 (April-May 2005): 20-25.

[2] Baronti, Paolo, Prashant Pillai, Vince Chook, Stefano Chessa, Alberto Gotta, and Y. Fun Hu. "Wireless Sensor Networks: a Survey on the State of the Art and the 802.15.4 and ZigBee

Standards." Computer Communications 30 (Dec. 14, 2006): 1655-1695.

[3] Lhomme, E., Frattasi, S., Figueiras, J., and Schwefel, H. 2006. Enhancement of localization accuracy in cellular networks via cooperative ad-hoc links. In Proceedings of the 3rd international Conference on Mobile Technology, Applications &Amp; Systems (Bangkok, Thailand, October 25 - 27, 2006). Mobility '06, vol. 270. ACM, New York, NY, 60. DOI= http://doi.acm.org/10.1145/1292331.1292400

[4] Schloter, C. P. and Aghajan, H. 2006. Wireless symbolic positioning using support vector machines. In Proceedings of the 2006 international Conference on Wireless Communications and Mobile Computing (Vancouver, British Columbia, Canada, July 03 - 06, 2006). IWCMC '06. ACM, New York, NY, 1141-1146. DOI= http://doi.acm.org/10.1145/1143549.1143778

[5] Sharma, N. K. 2006. A weighted center of mass based trilateration approach for locating wireless devices in indoor environment. In Proceedings of the 4th ACM International Workshop on Mobility Management and Wireless Access (Terromolinos, Spain, October 02 - 02, 2006). MobiWac '06. ACM, New York, NY, 112-115. DOI= http://doi.acm.org/10.1145/1164783.1164804

[6]Wu, H., Wang, C., and Tzeng, N. 2005. Novel self-configurable positioning technique for multihop wireless networks. IEEE/ACM Trans. Netw. 13, 3 (Jun. 2005), 609-621.

APPENDIX

Outdoor Experiment Results

Experiment Run	Node 1		Node 2		Node 3		Antenna Type	Node 1 RSSI	Node 2 RSSI	Node 3 RSSI	Estimate		Actual		Antenna Type	Distance Difference
	x	y	x	y	x	y					x	y	x	y		
1	0	0	20	-60	60	0	RPSMA	46	51	58	23.04	-21.83	20	-30	wire	8.72
2	0	0	20	-40	60	0	RPSMA	4C	48	55	26.37	-13.8	20	-20	wire	8.89
3	0	0	20	-40	40	0	RPSMA	5A	4B	4F	27.39	-20.88	20	-20	wire	7.44
4	0	0	20	-40	40	0	RPSMA	4E	4C	46	23.97	-14.11	20	-15	wire	4.07
5	0	0	20	-40	40	0	RPSMA	3E	52	47	16.66	-7.4	20	0	wire	8.12
6	0	0	20	-40	40	0	RPSMA	4C	4B	43	24.1	-13.48	20	-5	wire	9.42
7	0	0	20	-20	40	0	RPSMA	4D	41	44	24.24	-6.46	20	-5	wire	4.48
8	0	0	20	-10	25	0	RPSMA	4B	36	45	16.94	-19.44	20	-5	wire	14.76
9	0	0	20	-10	25	0	RPSMA	33	3D	3E	8.92	0.64	10	-5	wire	5.74
10	0	0	20	-10	20	0	RPSMA	32	3D	40	4.07	-8.64	10	-5	wire	6.96
11	0	0	10	-10	20	0	RPSMA	41	35	3D	12.5	-9.27	10	-5	wire	4.95
12	0	0	10	-10	20	0	RPSMA	34	3F	38	8.71	8.93	10	0	wire	9.02

Table 1. Results from Outdoor Experiment

Indoor Experiment Results

Experiment Run	Node 1		Node 2		Node 3		Antenna Type	Node 1 RSSI	Node 2 RSSI	Node 3 RSSI	Estimate		Actual		Antenna Type	Distance Difference
	x	y	x	y	x	y					x	y	x	y		
1	0	0	7	-3.5	4	0	wire	30	3E	24	2.89	-0.66	3	-2	wire	1.34
2	0	0	7	-3.5	6	0	wire	30	3E	37	2.42	-1.62	3	-2	wire	0.69
3	0	0	7	-3.5	7	0	wire	30	3E	35	3.16	-0.14	3	-2	wire	1.87
4	0	0	5	-3	4	0	wire	30	3A	24	2.9	0.93	3	-2	wire	2.93
5	0	0	5	-3	6	0	wire	30	3A	37	2.42	0.13	3	-2	wire	2.21
6	0	0	5	-3	7	0	wire	30	3A	35	3.16	1.36	3	-2	wire	3.36
7	0	0	0	-2	7	0	wire	24	24	3F	1.74	-1	1	-1	wire	0.74
8	0	0	0	-2	5	0	wire	24	24	2F	1.86	-1	1	-1	wire	0.86
9	0	0	0	-3.5	7	0	wire	24	31	3F	1.74	-0.6	1	-1	wire	0.84
10	0	0	0	-3.5	5	0	wire	24	31	2F	1.86	-0.6	1	-1	wire	0.95
11	0	0	3	-3.5	7	0	wire	24	38	3F	1.74	0.64	1	-1	wire	1.8
12	0	0	3	-3.5	5	0	wire	24	38	2F	1.86	0.75	1	-1	wire	1.95
13	0	0	6	-3.5	6	0	wire	3A	34	21	4.55	-0.07	6	-1	wire	1.72
14	0	0	6	-3.5	7	0	wire	3A	34	2E	4.37	-0.38	6	-1	wire	1.74
15	0	0	7	-3	6	0	wire	3A	31	21	4.55	-0.66	6	-1	wire	1.49
16	0	0	7	-3	7	0	wire	3A	31	2E	4.37	-1.08	6	-1	wire	1.63
17	0	0	3.5	-3.5	7	0	wire	34	35	33	3.57	0.22	3.5	0	wire	0.23
18	0	0	0	-3.5	7	0	wire	34	33	33	3.57	-1.89	3.5	0	wire	1.89
19	0	0	0	-1	7	0	wire	34	33	33	3.57	-0.1	3.5	0	wire	0.12
20	0	0	7	-3.5	7	0	wire	34	3C	33	3.57	-0.25	3.5	0	wire	0.26
21	0	0	7	-3.5	7	0	wire	3E	28	31	4.58	-2.66	7	-2	wire	2.51
22	0	0	7	-3	7	0	wire	3E	28	31	4.58	-2.56	7	-2	wire	2.48
23	0	0	7	-1	7	0	wire	3E	26	31	4.58	-4.14	7	-2	wire	3.23
24	0	0	7	-3	4	0	wire	3D	3A	25	4.68	0.59	4	-2	wire	2.68
25	0	0	7	-3	7	0	wire	3D	3A	35	4.2	-0.53	4	-2	wire	1.48
26	0	0	3	-3.5	4	0	wire	3D	2B	25	4.68	-1.65	4	-2	wire	0.76
27	0	0	3	-3.5	7	0	wire	3D	2B	35	4.2	-2.06	4	-2	wire	0.21

Table 2. Results from Indoor Experiment

The formulas of the three circles:

$C1: r_1^2 = x^2 + y^2$

$C2: r_2^2 = (x - h_2)^2 + y^2$

$C3: r_3^2 = (x - h_3)^2 + (y - k_3)^2$

Subtract C1 - C2 and solve for 'x':

$r_1^2 = x^2 + y^2$

$- \; r_2^2 = (x - h_2)^2 + y^2$

This gives us the x-coordinate of our point:

$$x = \frac{(r_1^2 - r_2^2 + h_2^2)}{2h_2}$$

Take 'x' and put it into the formula for C1:

$$r_1^2 = \left(\frac{(r_1^2 - r_2^2 + h_2^2)}{2h_2}\right)^2 + y^2$$

This gives us a circle at the intersection of C1 and C2:

$$y^2 = r_1^2 - \left(\frac{(r_1^2 - r_2^2 + h_2^2)}{2h_2}\right)^2$$

Solve the formula of C3 for y^2 :

$$y^2 = r_3^2 - (x - h_3)^2 + 2yk_3 - k_3^2$$

Then set this equal to the formula of the circle:

$$r_1^2 - \left(\frac{(r_1^2 - r_2^2 + h_2^2)}{2h_2}\right)^2 = r_3^2 - (x - h_3)^2 + 2yk_3 - k_3^2$$

Then we solve for 'y', this is the y-coordinate of our point:

$$y = \frac{(r_1^2 - r_2^2 - x^2 + (x - h_3)^2 + k_3^2)}{2k_3}$$

Figure 1. Derivation of Trilateration Equation