2007-01-01

Wireless Data Transmission System for Real-time 3D Geophysical Sensing

David Anthony Viggiano
University of Miami, dave.viggiano@gmail.com

Recommended Citation

Viggiano, David Anthony, "Wireless Data Transmission System for Real-time 3D Geophysical Sensing" (2007). *Open Access Theses*. 88.
https://scholarlyrepository.miami.edu/oa_theses/88

UNIVERSITY OF MIAMI

WIRELESS DATA TRANSMISSION SYSTEM FOR REAL-TIME 3D GEOPHYSICAL SENSING

By

David Anthony Viggiano

A THESIS

Submitted to the Faculty
of the University of Miami
in partial fulfillment of the requirements for
the degree of Master of Science

Coral Gables, Florida

December 2007

UNIVERSITY OF MIAMI

A thesis submitted in partial fulfillment of
the requirements for the degree of
Master of Science

WIRELESS DATA TRANSMISSION SYSTEM FOR REAL-TIME 3D GEOPHYSICAL SENSING

David Anthony Viggiano

Approved:

_________________ Dr. Xiadong Cai Assistant Professor of Electrical and Computer Engineering	_________________ Dr. Terri A. Scandura Dean of the Graduate School
_________________ Dr. Mark Grasmueck Associate Professor of Marine Geology and Geophysics	_________________ Dr. Manohar Murthi Assistant Professor of Electrical and Computer Engineering

VIGGIANO, DAVID A. (M.S., Engineering)
WIRELESS DATA TRANSMISSION SYSTEM FOR REAL-TIME 3D GEOPHYSICAL SENSING (December 2007)

Abstract of a thesis at the University of Miami.

Thesis supervised by Dr. Xiadong Cai and Dr. Mark Grasmueck.
No. of pages in text. (68)

A wireless data transmission system was developed and implemented for use in real-time 3D geophysical sensing. Server and Client applications were designed to run on a stationary computer and a mobile computer attached to the geophysical sensor, respectively. Several methods for optimizing communication over wireless networks using commonly available hardware were tested and compared, and a scheme for varying the size of transmissions in accordance with the recent performance of the wireless network was chosen. The final system was integrated with a 3D Ground Penetrating Radar (GPR) system and tested in a field experiment that spanned two weeks and involved the acquisition of 16 data volumes. The system performed successfully throughout the experiment and provided necessary feedback to assist in the error-free acquisition of all data volumes. Future development of the system is discussed in order for the system to support an automated wireless geophysical sensor network with multiple client sensors moved around the survey area by automated robots.

Acknowledgments

I would like to thank both of my parents for supporting me through my many years of higher education. I would also like to thank my girlfriend Dana and her family for their support and for listening to all of my technological talk during this project. Finally, I would like to thank Dr. Mark Grasmueck for the wonderful opportunity to learn about the geological sciences and geophysical sensing while I have been involved in this project. The work presented here would not have been possible without the support of the people above.

TABLE OF CONTENTS

Page

Chapter

LIST OF FIGURES

Chapter 1: Introduction

1-1) Definition of Problem

The science of geophysical imaging has shown the ability to study various aspects of the shallow subsurface. Several technologies exist for such studies including Ground Penetrating Radar, magnetometry, and acoustic sensors. Commercially available products are designed for use by a single user with a digital interface for displaying acquired data in real-time. Data is acquired by moving around with the sensor and recording measurements at different points around a chosen survey area, but very little off-the-shelf support is provided for arranging the acquired data into a spatial map for detailed analysis in real-time.

A new system has been developed to aid in the acquisition of spatially oriented geophysical data. The user is guided along closely-spaced parallel acquisition lines similar to a lawnmower pattern to build a surface map showing a selection of the acquired data. Due to the enhanced requirement of the user to follow the predefined acquisition lines accurately, it is not possible for the user to monitor a display to verify data quality and acquisition success. Therefore, a system is desired which will allow a remote user to monitor progress without affecting the acquisition of the geophysical sensor.

1-2) Intended Solution

The goal of this thesis is to design a client/server application to wirelessly transmit geophysical data from a mobile client computer attached to the sensor to a stationary server computer placed adjacent to the survey area. The applications will be implemented in National Instruments LabView [1]. In addition to the transmission of the acquired data, the system should be capable of sending control messages from the server to the client to adjust settings remotely during acquisition. In order to make the system widely available, the solution will use currently existing and commonly available networking technologies. The primary challenge in the implementation of the system is to adjust transmissions in order to compensate for variations in the quality of the wireless network, transferring as much data as possible while minimizing failed transmissions and the resulting waste of system resources.

The completion of the system described in this thesis will also serve as a necessary step in the development of a wireless geophysical sensor network. This future aim includes acquiring data with multiple geophysical sensors simultaneously as well as using robotic pullers to automatically move the sensors around the survey area. Such a system will require centralized control in order to ensure that the sensors acquire the entire survey area efficiently without colliding into one another, and the server designed in this thesis is a first step toward making centralized control a reality.

1-3) Thesis Organization

Development of this thesis begins in Chapter 2 with a discussion of previous developments that have contributed to the technologies used in wireless networking and geophysical sensing today. These technologies are examined to study existing methods for optimizing data transmission over computer networks, as well as to ensure that the developed solution uses the optimal protocol for data transmission. The development of the final solution is detailed in Chapter 3, and several optimization schemes are attempted and compared before determining the method for the final implementation of the system. A detailed description of the final implementation is included in Chapter 4, including chosen data structures and optimization of system parameters. The final system was tested during an extended geophysical experiment, and a description of the experiment as well as results are presented in Chapter 5. Finally, the significance of the final system and plans for future research are outlined in Chapter 6.

Chapter 2: Background

2-1) Review of Geophysical Sensing

The desire to quickly gain insight into the shallow subsurface without physically disturbing the underlying material has led to the technological development of several kinds of geophysical sensors. The data acquired by these sensors can be used to aid in many studies related to the near-surface. For archaeology, the use of geophysical sensors allows an area to be non-destructively scanned for areas of interest [2]. This allows excavations to be performed on isolated target areas and reduces the risk of damaging unforeseen archaeological artifacts. Similarly, the location of subsurface utility systems such as power lines, water pipes, and irrigation systems can be identified using geophysical sensors when existing records are insufficient or missing [3]. Building materials such as concrete can also be analyzed for structural integrity [4, 5]. In the presence of Unexploded Ordinances (UXO) or land mines, geophysical sensors have shown the ability to locate such hazards without disturbing them so they may be safely disarmed [6]. Finally, geophysical sensors can be used to study the static and dynamic properties of the near-surface. For instance, 3D GPR can be used to characterize the internal geologic architecture of a subsurface volume [7]. Also, research has shown that these sensors can be used to track water content changes related to infiltration or leaking pipes [8, 9, 10], providing a new understanding of hydrological processes and enabling studies such as the efficiency comparisons of several irrigation methods in the future.

2-2) Methodology for Subsurface Exploration

To this day, the most common way to explore the shallow subsurface is through a combination of drilling and digging. While these methods produce reliable results of the investigated volume, they are accompanied by several drawbacks. Excavation is often slow and expensive, yet provides information for only a relatively small volume. Furthermore, the in situ structure of the subsurface material is permanently destroyed upon excavation. Because of these drawbacks a faster, cheaper, and non-destructive method for exploring the subsurface is desired and exists in geophysical sensors. Most geophysical sensors today are used by moving around a target area and detecting anomalies based on the physical properties of the underlying objects. Although much more information is typically provided by the sensors, the geophysical expertise required to interpret the information is often not available due to the extra associated cost and the tendency to get unreliable results. A variety of geophysical sensors can be used for subsurface exploration including magnetic, gravimetric, acoustic, and electromagnetic sensors. The two most common electromagnetic geophysical sensors are EM sensors for low frequencies (deeper penetration) and GPR sensors for high frequencies (higher resolution).

The current procedure for acquiring data with geophysical sensors is very tedious, requiring the placement of measuring tapes to guide the user along predefined straight lines over the target area (Figure 2.1). Due to the time overhead of staking out measuring tapes, many researchers compromise by acquiring less data in lieu of dense data and interpolating the results between measurements. Commercially available sensors usually come with a monitor and only display instant sensor responses to the user as the data is acquired (Figure 2.2), making it necessary for the user to watch the display while simultaneously moving with the sensor. Although a map view of the results is more intuitive and easier for the user to interpret, most systems do not provide a method for recording sensor locations and therefore can only display the data as a vertical cross-section or *profile view*. Some systems use GPS receivers to record sensor locations [12], but these systems can only function in open fields with clear visibility of the sky and require post-processing of the GPS data in order to attain locations with sufficient precision. Even in the absence of GPS data, significant post-processing of the geophysical data is often required which prevents useful results from being available while still at the survey area. In addition, most geophysical sensors are provided as isolated devices with no functionality for integration with other sensors to allow for faster and more diverse measurements.

Figure 2.1: Photo showing typical GPR data acquisition with the assistance of measuring tapes staked out on the ground to guide the user along target transects [11]. Such acquisition is tedious and compromises are often made to save time by recording less data and interpolating.

Figure 2.2: Photo showing GPR data acquisition using a commercially available digital display for data monitoring [13]. One user must simultaneously perform the two tasks of positioning the GPR antenna and monitoring the acquired data on the digital display. Results are displayed in profile view rather than a more intuitive map view.

Processing and interpretation of data acquired with existing geophysical sensing systems is also associated with several limitations. Complete processing of the data often requires all of the acquired data and therefore cannot be accomplished in real-time. Because of this, only basic information can be used while in the field for quality control. One consequence of this is that acquisition errors may not be discovered until days after the measurement, resulting in missing data in the final result. Interpolation of this missing data along with unrecorded data due to compromises for time, can result in the introduction of artifacts and overall reduced data quality. Therefore, a geophysical expert is often required to interpret the results and distinguish data artifacts from true subsurface anomalies.

2-3) Proposed Geophysical Sensing System

Upon completion of this thesis, wireless networking will be integrated with geophysical sensors to improve several aspects of subsurface exploration. The most significant improvement will be to separate the two tasks of acquiring data with the geophysical sensor and monitoring quality during acquisition so that the users can focus on the tasks individually. Wireless networking will be used to transfer data to a remote computer during acquisition so one user can focus on the progress of acquisition while the second accurately positions the geophysical sensor. A real-time guidance system will be used to guide the user without the use of measuring tapes, which will also greatly improve efficiency and avoid missing data due to compromises for time. These improvements will allow the acquisition of dense geophysical data which will result in easier interpretation with less missed targets and false identifications.

2-4) Ground Penetrating Radar Overview

GPR senses anomalies in the shallow subsurface using an electromagnetic pulse with a frequency between 25 MHz and 2 GHz (Figure 2.3). Separate dipole antennae in one enclosure are used to transmit a short pulse and receive the reflection from the subsurface using an amplifier and sample-and-hold circuit. The GPR reflections result mainly from changes in the dielectric properties of the subsurface which are mostly controlled by water content [14]. The depth of the reflections depends on the material of the subsurface, but typically ranges from 5-10 m. *2D* GPR data is acquired by continuously moving in a straight line with the antenna to acquire evenly spaced GPR *traces* which can be displayed as a vertical cross section (Figure 2.4). However, GPR antennae have a conic radiation pattern that extends 45-60 degrees from vertical, causing reflections from objects out of the vertical plane which are subject to misinterpretation.

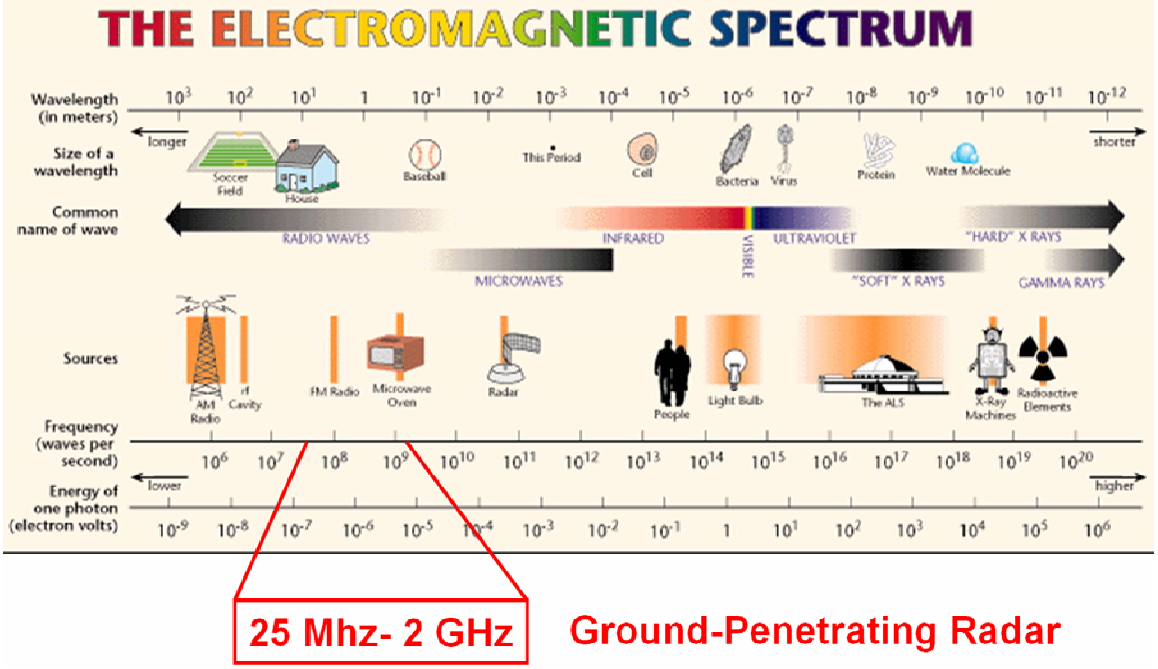


Figure 2.3: The electromagnetic spectrum highlighting the common frequency range used by GPR antennae. Commercially available equipment uses frequencies between 25 MHz and 2 GHz.

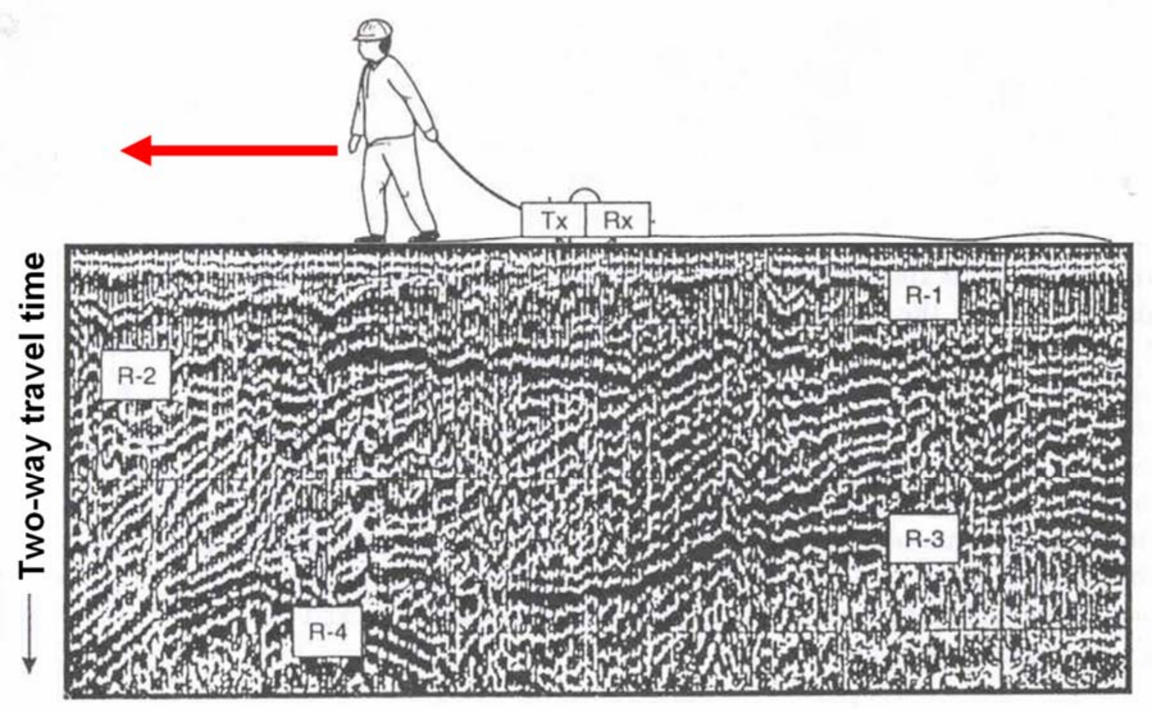


Figure 2.4: Illustration of data acquisition and display for 2D GPR. Radar traces are acquired at uniform spacing to build a vertical cross section of the subsurface.

With high-resolution 3D GPR it is possible to focus the out-of-plane signals using *migration* (a.k.a. SAR) algorithms for a more accurate representation of the subsurface. Parallel GPR lines are acquired with uniform spacing which is determined by the frequency of the GPR antenna as defined by Nyquist sampling theory. For example, a 250 MHz GPR antenna with a wavelength of 40 cm is sufficiently sampled at 10 cm [15]. The out-of-plane signals can only be properly focused if the electromagnetic field is fully sampled, but existing commercial equipment does not support acquisition of such dense data. The wireless system developed in this thesis is a necessary step in improving the efficiency of 3D GPR surveying and enabling the development of the next generation of GPR field equipment.

2-5) Wireless Networking Review

In order to develop a working solution that utilizes existing networking technology as efficiently as possible, it is first necessary to review and understand some common protocols. Most computer networking schemes can be related to the 7-layer Open Systems Interconnection (OSI) Network Model [16] (Figure 2.5). This model defines seven distinct layers that build on one another to provide complete networking functionality. The bottom layers of the stack contain the low-level functionality needed for data transmission, while the top layers provide high-level functions that are easy to use without requiring a detailed understanding of the layers underneath. Interaction between layers in the stack is handled so that each layer of the stack needs only basic information about the neighboring layers in order to interact with them. This allows for different realizations of each layer to be used without affecting the choice of the surrounding layers.

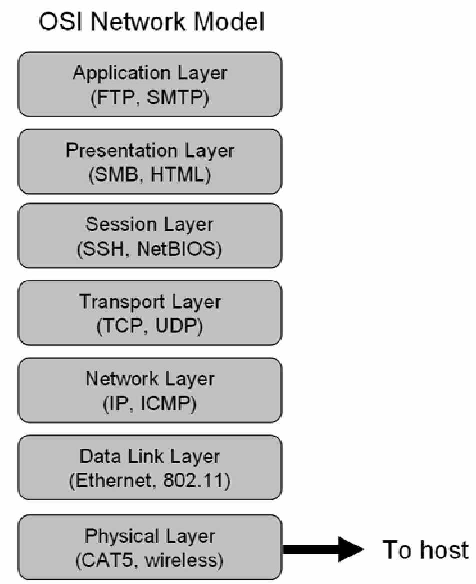


Figure 2.5: The Open Systems Interconnection (OSI) Model for network architecture. The seven layers defined by the model are shown with example realizations of each layer.

The bottom layer of the stack, known as the Physical Layer, defines the physical configuration of the network hardware. The Data Link Layer represents the channel that is used to transfer data between the computers. Realizations of this layer include wired solutions such as Ethernet [17] (802.3), or wireless solutions that use different channel structures and encoding schemes such as 802.11a or 802.11g [18]. The commonly-used Internet Protocol (IP) [19] is a realization of the third layer in the stack, or the Network Layer. Two protocols that are often used in conjunction with IP reside at the fourth layer (Transport Layer) and are called the Transmission Control Protocol (TCP) [20] and the User Datagram Protocol (UDP) [21]. These protocols will be discussed and compared in depth later in this chapter. The fifth, sixth, and seventh layers of the stack are known as the Session, Presentation, and Application Layers, respectively. These layers contain many realizations for various high-level tasks, including Hypertext Transfer Protocol (HTTP) [22] for transferring files over the internet, and Simple Mail Transfer Protocol (SMTP) [23] for transferring email messages between a mail server and client.

The Internet Protocol (IP) is the most commonly used Network Layer protocol today. IP is a connectionless protocol, meaning that a connection does not need to be established before data is sent [24]. This concept can be illustrated along with its counterpart (connection-oriented transmission) by comparing the phone and postal systems. The phone system is connection-oriented; a connection must be established between the two parties before anyone is able to start communicating. In the (connectionless) postal system, individual messages are sent via a shared delivery system. There is no connection to establish before sending the first data or close after sending the last data. However, each message must contain all the necessary information (the address) in order to be delivered to the destination.

The two most common Transport Layer protocols used with IP are TCP and UDP, and of these two TCP is the most commonly used [25]. This is because TCP attempts to provide the reliability of a connection-oriented transmission while running on top of the connectionless IP protocol. The protocol accomplishes this by using packet numbering, error-checking codes, and acknowledgments sent from the receiver back to the sender to confirm or deny successful transmission. However, the gain in guarantee-of-service comes with an increase in network traffic. The packet numbering and error-checking codes add overhead to the size of each transmitted packet, and acknowledgment packets add entirely new transmissions to the flow of communication. The result is that TCP can not transfer data as quickly as its counterpart protocol. In UDP, the connectionless transmission of IP is preserved with few changes. Although this can provide a significant increase in data rate, it also comes with the downfalls of connectionless transmission which include packets arriving late, out of sending order, or not at all.

An important decision in the development of this thesis is whether to use TCP, UDP, or a combination of the two as the transmission protocol for the system. Although UDP provides faster data transmission than TCP, it will require manual implementations of features included with TCP in order to provide the reliability needed for this system. For this reason, TCP will be used as the primary transmission protocol. However, the benefits of UDP may still warrant its use in some parts of the system and the decision of which protocol to use will be revisited later in the thesis. The chosen implementation for the first four layers of the OSI Network Model is illustrated in Figure 2.6. The overhead of this implementation relative to the Session Layer is illustrated in Figure 2.7.

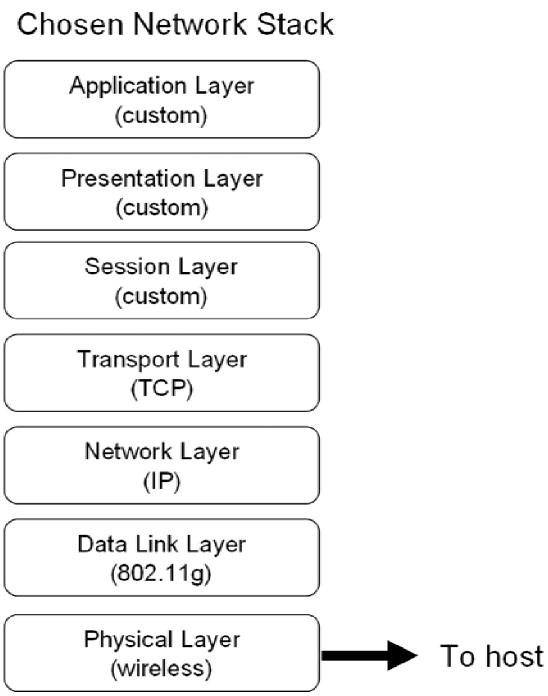


Figure 2.6: Network stack chosen for the system in this thesis. The three highest layers are customized for the geophysical sensing application, and TCP is used for the Transport Layer.

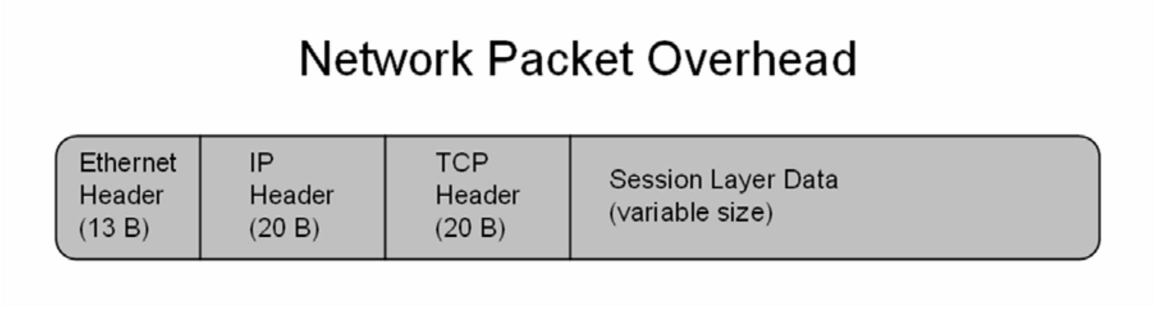


Figure 2.7: Packet layout relative to the Session Layer of the network stack. Each protocol encloses the data from the higher-level protocols in a larger data structure with a header specific to the protocol. This results in extra data overhead relative to the amount of data sent from the Session Layer as defined in the illustration.

2-6) Modifications to TCP

There are several modifications to the basic implementation of TCP that aim to improve its efficiency. Some of these modifications, such as TCP Tahoe and TCP Reno

[26], focus on improving the mechanism for congestion avoidance. This is accomplished by maintaining a congestion window based on the recent success or failure of transmitted packets. The congestion window controls how many consecutive packets can be sent before an acknowledgment packet must be received. A low window value means that only a small amount of data can be sent between acknowledgments. The window is increased at a defined rate when communication is successful and decreased at a different rate after packet failure. The definition of these rates depends on the implementation of TCP. For example, TCP Tahoe increases the window first exponentially for each successful transmission then linearly after a specified threshold is reached. The window restarts at one system-configurable *Maximum Segment Size* (MSS) after a single packet failure and halves the threshold as illustrated in Figure 2.8. The concept of adjusting the size of data transmission based on the conditions of the network will be revisited in Chapter 3 during the development of the system.

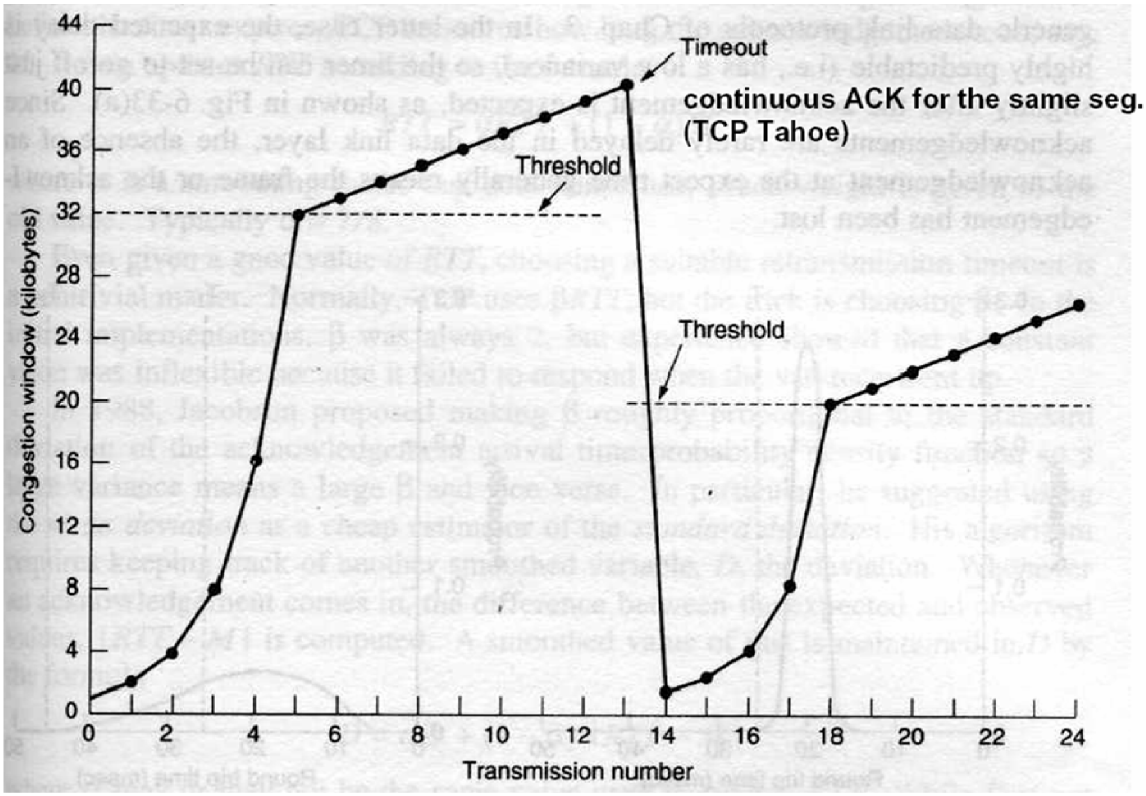


Figure 2.8: Illustration of transmission scheme for TCP Tahoe [27]. The congestion window is increased after successful transmissions and decreased after failed transmissions in order to maximize data transfer while minimizing network congestion. Variations can be made to the rate at which the congestion window is increased and decreased. In this figure the congestion window is increased exponentially until a threshold is reached, at which point the window begins to grow incrementally. When a timeout is reached the congestion window is reset to one kB in order to avoid network congestion, and the threshold is reduced to half of the value when the timeout occurred for future transmissions.

Another series of modifications to the implementation of TCP deals with optimizations for mobile wireless clients. These modifications attempt to deal with the fact that packet loss is often due to transmission errors or handovers in a mobile setting, and not just network congestion as traditional TCP assumes [27]. A packet lost due to signal fading on the wireless channel should be retransmitted immediately, whereas packet loss due to network congestion on the wired network requires a reduction in the rate at which data is sent. However, as there is no simple way to determine the cause of packet loss the decision whether to retransmit a lost packet immediately or compensate for network congestion can not be made. One solution to this problem is to split the TCP connection into two by adding a third host as a *relay* between the original hosts (Figure 2.9). This additional host provides the bridge between the fixed and wireless network so the source of packet loss can be safely assumed. If a packet is lost between the relay and mobile host it is immediately retransmitted, while packet loss between the relay and the fixed host is treated as network congestion. Some example protocols that use this concept are Snooping TCP [28] and Indirect TCP [29]. Despite what the name suggests, these concepts are not currently applicable to the geophysical sensing system as all packet loss is related to wireless signal fading.

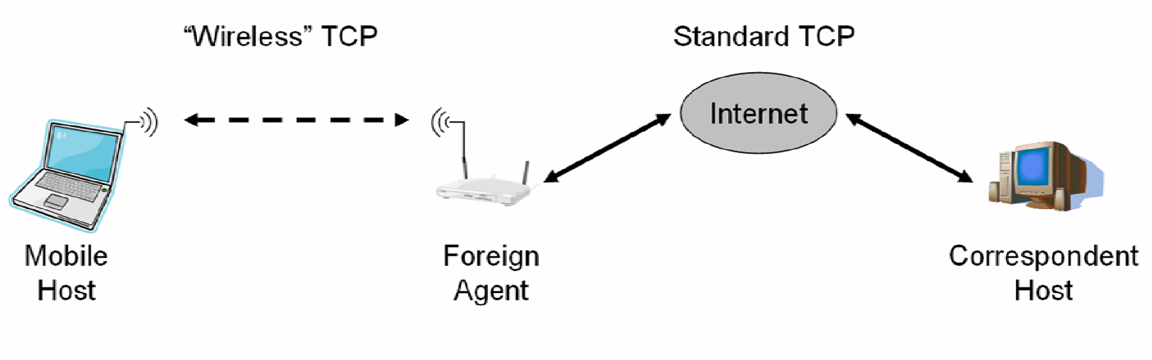


Figure 2.9: Conceptual scheme for Mobile TCP. The connection between the Mobile Host and Foreign Agent (relay) is isolated from the internet so that the source of packet loss can be safely identified. Lost packets on the "wireless" TCP connection are a sign of wireless transmission errors, while lost packets on the standard TCP connection are a sign of network congestion.

2-7) Wireless Networking in Microsoft Windows XP

The third party software provided with most geophysical sensors for data acquisition control requires Microsoft Windows XP. Therefore the computers in the proposed system must run this operating system and use its built in network stack for wireless networking. There are several issues associated with the Windows network implementation that must be addressed. The first issue deals with selecting a wireless network when several are available. The operating system maintains a list of recognized wireless networks and assigns a priority to each, then connects to the available network with the highest priority. However, if another network with a higher priority is detected while connected to the first network the operating system will drop the first connection and establish a connection to the higher priority network. This action must be prevented in order for stable runtime performance in the presence of multiple wireless networks.

Another issue arises when the signal strength of the wireless connection is low. In this case, the operating system may sometimes drop the connection and then reconnect after several seconds. However, the disconnect/reconnect procedure uses extra resources on the mobile host computer that are not always available, and sometimes this causes the geophysical sensor to stop functioning.

The most intriguing problem associated with wireless networking in Windows XP is the network overhead related to Microsoft's implementation of the NetBIOS protocol [30]. NetBIOS is a higher-level network protocol that is used in Windows to share information between computers. The information transferred via the NetBIOS protocol includes data for file sharing (as seen in "My Network Places") and printer sharing. However, transmitting this data between the computers can require significant network resources depending on how much data needs to be shared between the computers. The Windows XP operating system provides very little control over the transmission of this data, so scheduling the operation for a specific time is not possible. Instead, the most thorough way to stop these extra transmissions on the wireless network is to disable any file or printer sharing on the two host computers so that the NetBIOS protocol has no information to send.

2-8) TCP/IP Issues Related to 3D Geophysical Sensing

In the proposed system with one wireless router and only the wireless mobile client and stationary server as network hosts, there are several factors that provide the challenge in attaining efficient and reliable data communication. The most significant of these factors is that the mobile host is continually in motion, meaning that the condition of the wireless channel changes rapidly. The consequence of this is that channel performance measurements are rendered obsolete before they can be used to optimize transmission.

Although rapid changes in the wireless channel can not easily be compensated for, slower changes due to large-scale signal fading are more manageable. Signal fading refers to the decrease in received signal power as the distance between the transmitter and receiver is increased. Because this loss is mostly dependent on distance, optimizing the use of the wireless channel can be done in two ways. The first method is to minimize the movement of the mobile host relative to the wireless router. For the acquisition pattern applied in 3D geophysical imaging, this can be done by placing the base station on the side of the survey area as shown in Figure 2.10. Walking lines perpendicular to the line between the mobile host and the router results in signal fading that changes more slowly than when walking directly to and from the base station. The second method is to use recent information about the channel to slowly adjust the size of future transmissions.

This concept is commonly used in wireless technology and will be further discussed in Chapter 3.

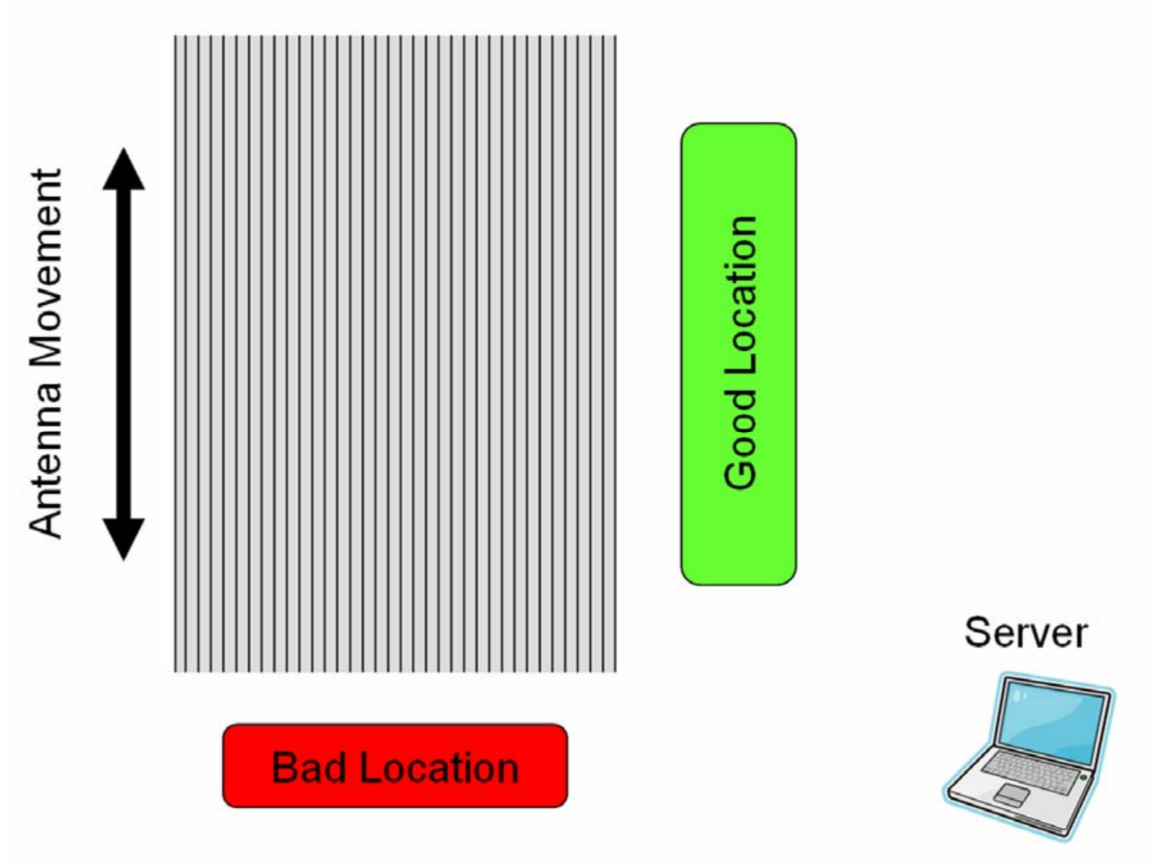


Figure 2.10: Optimal locations for the wireless router relative to the survey orientation. Locations on the side of the survey area are preferred to minimize the movement of the mobile client relative to the router.

Because 3D geophysical measurements are often recorded in populated areas with other wireless networks present, it is important to also consider the effects of these networks and ensure that they can successfully coexist. In the case of interference with another wireless network, changing to another of the 12 available frequency bands will solve the problem if a channel can be found which is not already in use. This problem is more prevalent in a geophysical sensing system because surveys are acquired in many different locations with different existing wireless networks. If all channels are in use, a less-desirable solution is to use a different modulation scheme such as 802.11b instead of 802.11g if both of the involved wireless devices support multiple schemes. It is also important to configure Windows XP so that it will only connect to the desired network when in populated areas. This ensures that even if Windows temporarily drops the connection, it will not connect to a different wireless network before it reconnects to the desired network.

Chapter 3: System Development

3-1) 3D GPR System Description

Development of this thesis was accomplished with the aid of a 3D Ground Penetrating Radar (GPR) system [31] (Figures 3.1 and 3.2). 3D GPR is acquired using off-the-shelf GPR antennae [13] to acquire many parallel cross-sections, building a high-resolution data cube which represents the subsurface [32]. Integration with a Rotary Laser Positioning System (RLPS) [33, 34] provides centimeter-precise coordinates in real-time for fusion with the radar data as well as user guidance along each target line.

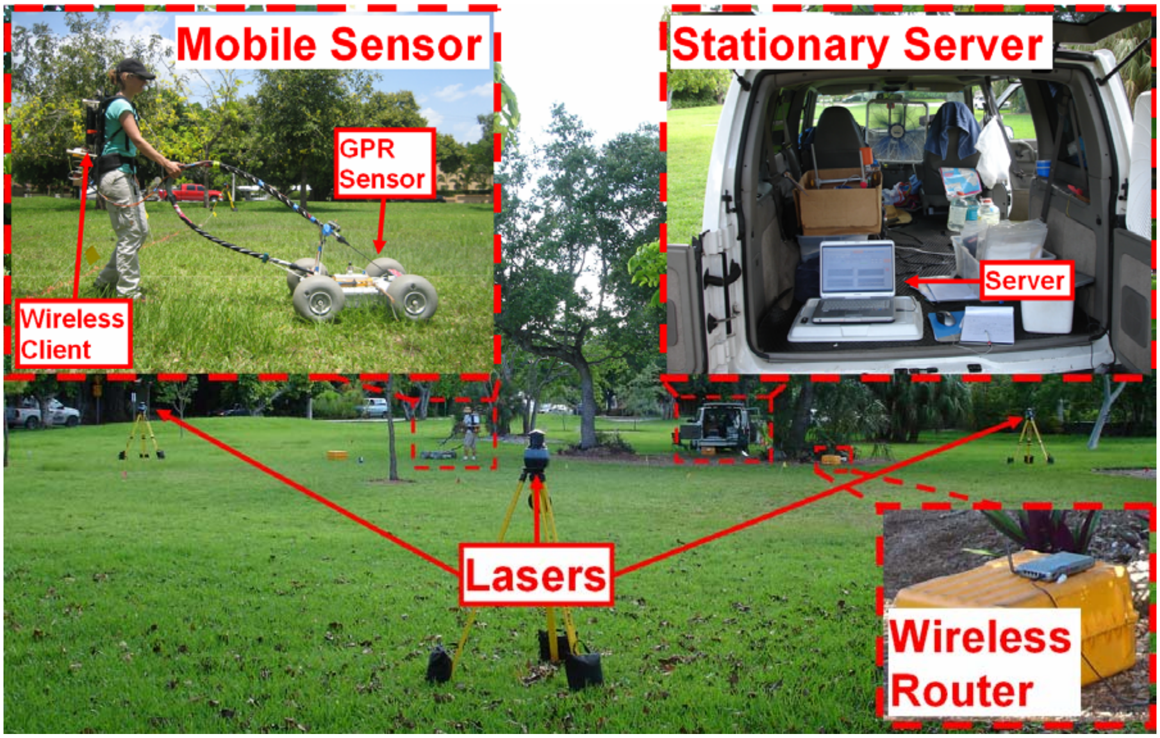


Figure 3.1: Overview of 3D GPR system. Integration of off-the-shelf GPR antennae with RLPS provides real-time fusion of radar data with centimeter-precise coordinates. Acquisition of densely spaced parallel lines yields a 3D data volume characterizing the shallow subsurface. A wireless router and stationary laptop are included with the mobile acquisition laptop to form a small wireless network for remote monitoring and control.

There are several additional features that facilitate acquiring the dense 3D GPR data. A guidance system provides real-time feedback from the positioning system using two strips of LEDs to indicate the location of the target line to the user. Sounds are used to help the user maintain a desired speed, indicate when to start the next target line, and notify the user when problems are detected.

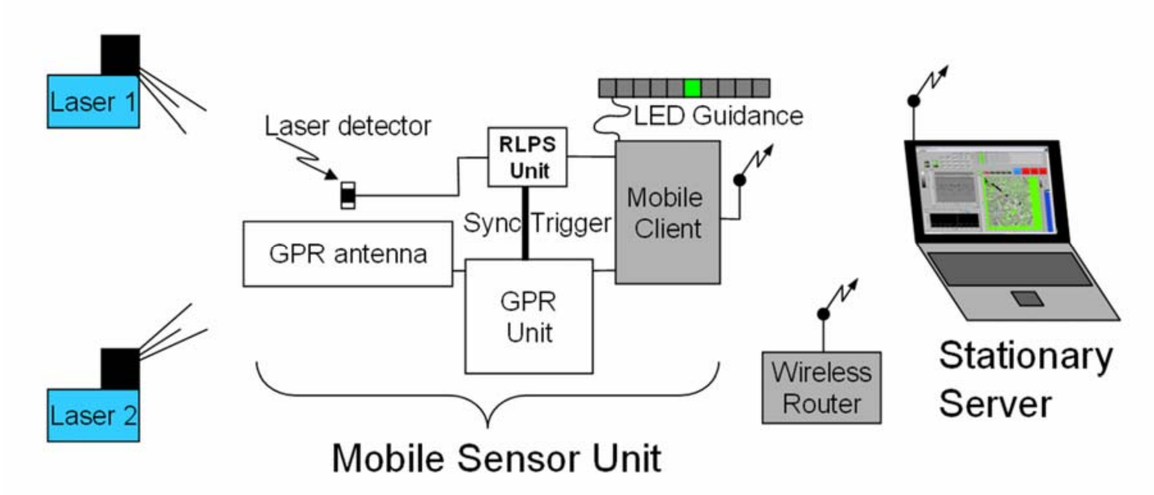


Figure 3.2: Schematic for wireless geophysical sensing system using GPR. Laser coordinates are fused with the acquired geophysical data in real-time. LED strips are used to provide real-time feedback to guide the user during acquisition. A wireless router is used to transfer data to and from a remotely placed stationary server computer.

3D GPR is ideal for the development of a wireless communication system for geophysical sensors because it acquires data at a higher rate than other geophysical sensors and therefore represents the largest amount of data to be transmitted over the wireless network. Adapting the system for use with other geophysical sensors is made simple once the challenges presented by the 3D GPR system are overcome. This work also lays the foundation for multiple and diverse geophysical sensor networks.

3-2) Wireless Communication Tasks

Before the start of data acquisition, the wireless network will be used along with Microsoft's Remote Desktop Connection to remotely initialize the third-party software and allow interactive procedures such as calibration of the laser system from the stationary server. Data acquisition via the third-party acquisition software is also started using Remote Desktop Connection, and all following traffic during acquisition will be performed by the implemented wireless system. Because transmission of all the acquired geophysical data requires too many system resources to be accomplished without disrupting the geophysical sensor, a more efficient scheme for data transmission will be designed.

The proposed solution will perform several functions during data acquisition. The first function is for real-time monitoring of the mobile sensor. Completion of this task requires some information to be sent with every transmission such as battery status, user statistics, and line number. This information will be packaged in the header of each update transmission. Although the size of each transmission will be varied to optimize the usage of the wireless channel, this information must always be transmitted and therefore specifies the minimum size of each transmission.

The second function of the proposed solution is to transfer the acquired data from the mobile client attached to the geophysical sensor to the stationary server on the side of the survey area in order to produce maps during acquisition for quality control. The main challenge associated with this is to transfer as much of the data as possible without disrupting the performance of the geophysical hardware and software. The current 3D GPR system typically acquires data at a rate of 250 kbps, which is a higher data rate than can be transferred across the wireless network without affecting the geophysical sensor. It is therefore not possible to transfer all of the raw data across the wireless network in real-time, so some processing must be applied to the data on the mobile host before it is packaged and transmitted. Several methods for accomplishing this data transmission are implemented and compared later in this chapter.

The final task of the system is to support short control messages sent from the stationary host to the mobile host to allow adjustment of some settings during data acquisition. Although the control messages are small in size, they have a high priority and need to be transmitted and applied as quickly as possible. While the previous two functions required transmission only from the mobile host to the stationary host, this function requires communication in the opposite direction and therefore increases the probability of packet collisions on the network if synchronization is not performed properly.

Aside from the high data rate, development with the 3D GPR system is complicated by a second factor. Data acquired by the GPR system is transferred to the mobile host computer via a parallel interface with no buffering. If this data transfer is not given enough resources by the computer, the GPR system will stop recording and report a communication failure. One result of these GPR failures is that the proposed wireless system must minimize its use of resources on the mobile host computer in order to maintain stable system operation. This characteristic imposes an upper limit on the aggressiveness of the wireless system.

3-3) 1-way vs. 2-way Communication

The first realization of the system used communication from the mobile client to the stationary server only. While this did not allow for control messages to be sent from the server to the client as desired in the final system, it represented a simplified solution and enabled basic issues to be investigated before expanding to the more complex system with communication in both directions. It is important to clarify that this first version of the software used one-way communication at the application layer, but two-way communication at the layers below. Because TCP was used for the transport layer, acknowledgment packets were frequently sent from the server to the client to help ensure reliable transmission of the data. However, these packets did not introduce packet collisions on the wireless network as they were small and synchronized with the flow of transmission. Finally, the first realization of the system used a fixed size for each transmission which could be chosen by the user before running the system. Although this solution made no attempt to optimize channel usage during the survey, it provided a simple and robust method for data transfer which could be used as a benchmark for future system realizations.

The addition of Application Layer communication from server to client introduced several problems involving synchronization of the two hosts. The first of these problems involved a 20 second delay whenever initiating communication on a new port. Investigation of this delay revealed that the client host was performing network discovery by sending a broadcast packet and allowing 20 seconds for other hosts on the network to identify themselves. The server responded to the request almost immediately by sending information about its shared files, but as there were only two hosts on the network the rest of the 20 seconds was spent waiting without purpose. Once the cause of the delay was known, the simple solution was to disable file and printer sharing from within Windows on both of the host machines.

A second problem introduced with bi-directional communication involved the specified connection and transfer timeouts for the TCP functions. If these timeouts were not set identically on both hosts, communication would occasionally break down in one of two ways. If the client used a shorter timeout than the server, the client would sometimes timeout while waiting for an acknowledgment of the last transmission and would therefore abort the connection and attempt to establish a new connection. However, because the server received the transmission without timing out it would attempt to respond with the existing connection and would be forced to wait another timeout period before determining that the client had aborted. During this time the client would make several unsuccessful attempts to establish a new connection while the server was not accepting incoming connections, and the overall result would be several consecutive transmission failures. Although this problem was difficult to diagnose, the solution of ensuring that the timeouts on the two hosts matched was straight forward and prevented the series of failures.

3-4) Fast-varying Transmission Size

The second realization of the system and first attempt at channel optimization varied the transmission size of each update independently of the previously used size. Instead of modifying the previous size, each transmission size was chosen based solely on the amount of buffered data to be sent. This method is particularly advantageous when the mobile client is not moving. Because there is no new data to send while stationary, the size of each transmission can be reduced to the minimum. It also may seem that this method would yield the most efficient use of the channel as it never sends more or less data than it needs to in order to keep the data buffer from overflowing, but two significant problems hinder the efficiency of the method.

The first problem with this method is exposed when analyzing the communication flow conceptually. As previously mentioned, most transmission failures can be attributed either to large-scale signal fading or small-scale dropouts related to multi-path or blockage. The small-scale dropouts change too quickly to be used for channel optimization, so only the large-scale dropouts related to signal fading can be used for adjustments. The conceptual problem with this method arises when a transmission fails. Assuming the failure is related to signal fading, the best approach is to reduce the size of the next transmission. However, this algorithm will almost double the size due to the extra buffered data remaining from the previous failed transmission. The channel usage will only be reduced when there is less data to send, which is related to either the channel performing well or the client standing still. Both of these situations are conducive to increased channel usage, so again the algorithm for optimization is unsuccessful. Although this transmission scheme would be ideal with sufficient system resources, it is not suitable for the geophysical sensing system.

3-5) Importance of Synchronization

Varying the size of each transmission introduced a second problem which would need to be addressed for any scheme with a varying transmission size. This problem concerned the synchronization of the transmission size between the client and the server. There are two possibilities when the server assumes a size other than that used by the client, and in either case the result is a breakdown in communication. If the server tries to receive more data than the client sends, it will timeout and wait for a new connection to be established. If the server receives less than the client sends, the remaining data will be buffered within the TCP protocol and will be erroneously interpreted as the beginning of the next transmission. In this case simple error checking will catch the synchronization loss and again require the connection to be reestablished. The discrepancy in transmission size between the client and the server was usually caused by a transmission error, but correcting the error took too much time and resources and would often cause data acquisition to fail. Because of this, the solution could not be used for reliable data acquisition and a more robust algorithm was chosen for the third implementation of the system.

3-6) Fixed Transmission Size

The third implementation of the system focused on successful and robust communication rather than optimizing channel usage. This was accomplished by using a user-specified fixed size for every transmission, thereby avoiding the problem of synchronizing transmission size. Null data was used to pad transmissions when there was not enough buffered data available. Although this method provided successful communication by avoiding the size synchronization problem, it had several flaws including the fact that only basic optimization could be performed and had to be determined manually by the user. The only available way to optimize communication was to vary the pre-determined size of transmissions, but the chosen size had to be determined by the user based on recent performance with previously chosen values. Furthermore, this optimization did not provide for adjustments to channel usage during data acquisition. An additional problem concerned the initial synchronization of the client and server. This was accomplished by specifying the transmission size independently on both computers, but if conflicting values were accidentally specified the system would fail to perform. Finally, poorly chosen values for the transmission size could result in poor performance for two reasons. If the chosen size was too low, the data buffers would overflow and not all data could be transmitted by the system. If the size was too high, a significant amount of null data would be sent which resulted in inefficient channel usage as well as an increased probability of transmission errors. Despite these problems, this method was capable of supporting extended data acquisition when suitable values were chosen for the transmission size, and its success provided a foundation for the design and implementation of the fourth and final method.

3-7) Slow-varying Transmission Size

The final method developed for the system combines the advantages of the previous methods by employing a transmission size that changes slowly and is properly synchronized between the client and server. As with the previous method, transmissions are padded with null data when the size is larger than the amount of data to be sent. This transmission scheme proved effective when implemented and was chosen as the transmission scheme for the final system. A detailed description of the scheme utilizing a slow-varying transmission size follows in Chapter 4.

Chapter 4: Final System Description

4-1) Overview

The final implementation for the wireless system uses the slow-varying transmission size described in the previous chapter. In this scheme, each transmission is defined as a variable-length packet from the client followed by a 3-Byte response packet from the server. A description of the flow of communication follows and can be seen in Figure 4.1, and the full definitions for all data structures can be found in the Appendix. The client initiates communication by establishing a connection to the server and sending the Startup data structure. Once the connection is established the Update data structure for client to server transmission is used. The server responds to each transmission with a short response packet consisting of flags and optional data which can include a command to the client or a request for special data. This interaction continues until the client sends the Shutdown data structure, at which point both hosts close the connection. While the client stops at this time, the server resumes waiting for a new connection to be established.

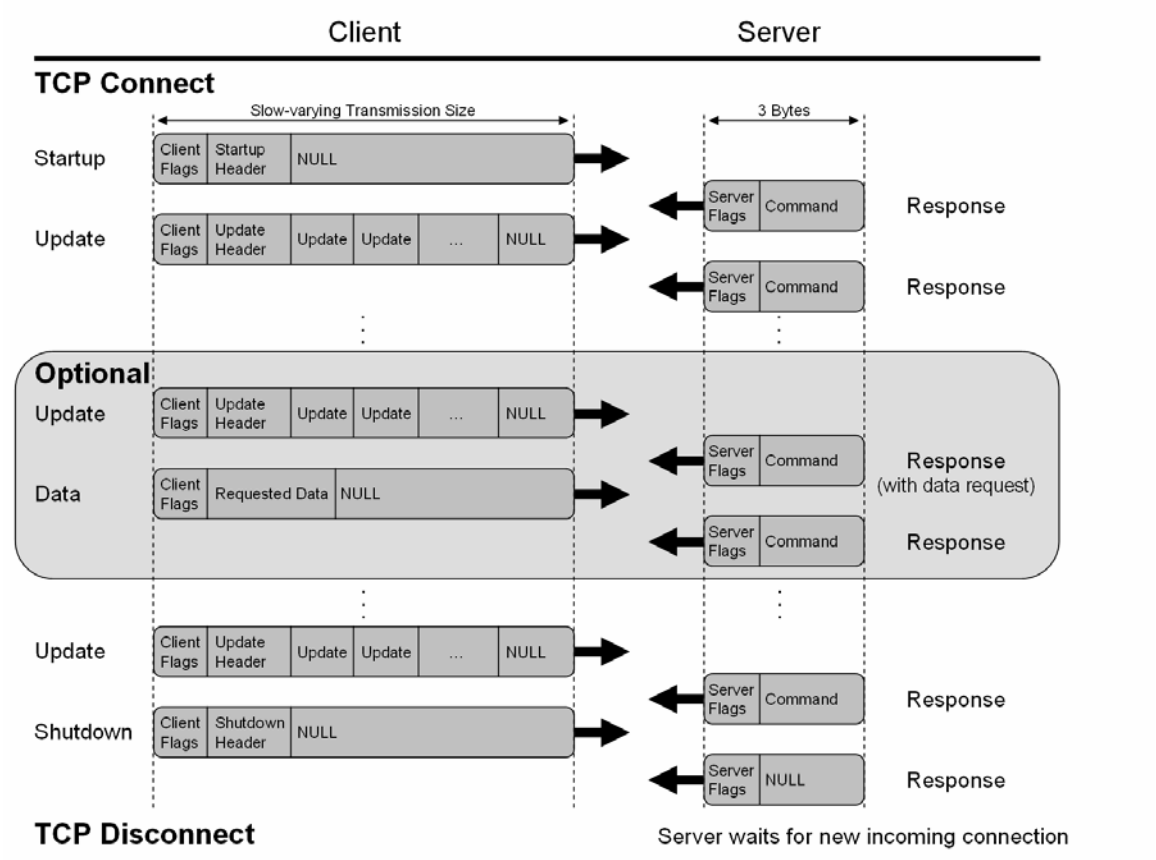


Figure 4.1: General flow of communication. The client sends a transmission to the server using the synchronized transmission size. The server uses a short response packet for acknowledging or requesting data and sending commands. Optional second transmissions are initiated by a server request and are used to transfer larger amounts of data which are only occasionally required. All definitions for data structures such as headers and flags can be found in the Appendix.

The server need not distinguish between new connections, reconnects, shutdowns, and normal operation while receiving because the client's transmission size is determined independently of the client state. This increases the robustness of the system as the server can receive the same amount of data regardless of the state of the client, then interpret it appropriately after it has been successfully received. Although the data contained in the headers varies depending on the state of the client, the first eight bits of all of the client header structures are encoded as eight true/false flags. The first two flags are used to indicate whether the client is starting, continuing, or shutting down a connection. The server uses these two flags to determine how to interpret the rest of the incoming transmission.

4-2) Client-Generated Data Structures

The client initiates a connection with the server when acquisition begins and after each dropped connection. In preparation for multiple simultaneous clients in the future, each client maintains a map display of the geophysical sensor measurements for its local subset of the entire survey area. The subset is specified by a map definition and allows the client to conserve memory, but the definition must be synchronized with the server in order for the server to place updates in their correct positions within the full survey area. Flags 3 and 4 are used in relation to the map definition to indicate when the display area should be reset or when the client is using a new definition. For example, a client will set both flags to true the first time it connects to the server in order to start a new map and specify the definition that the client will use, but will not set either flag when reconnecting to the server after a dropped connection. The server requests the map definition from the client when the corresponding client flag is set to true. Flags 5 and 6 are used to acknowledge commands from the server and indicate the overall status of the geophysical sensing system, respectively. Flag 7 is reserved for future use. Aside from the flags, the Startup header contains information about the current dataset including the dataset identification number, number of the first survey line to be acquired, and spacing between acquired lines. To ensure the correct network settings are used, two header values are used to specify the transfer timeout that the client will use and verify the initial transmission size. Two more header values are used for the client to indicate its initial LED brightness and speaker volume settings so the server can determine whether it is necessary for the values to be updated. Finally, a timestamp is included in the header to maintain synchronization between the system clocks of the client and server. Synchronization of the system clocks is necessary to ensure that data recorded on the server such as log files use timestamps that can be matched with the data recorded by the client.

The header of the update data structure contains diagnostic information such as the amount of time the client has been running, number of radar traces fused with laser coordinates, and current line number for acquisition. In addition, several header values are used for monitoring the status of data acquisition. The current walking speed of the user, distance from the target line, and a cumulative accuracy score for the user based on previous line deviations are all transmitted in order to ensure accurate data acquisition. Another header value is used to monitor the lasers from the positioning system that are visible to the positioning sensor, allowing the status of the lasers to be tracked and an audio alarm to be played if a laser battery fails. The next value of the header indicates how many position updates are encoded in the transmission so the server can extract the intended information without misinterpreting the null data used for padding the transmission. Position updates consist of three 2-Byte values for the X and Y bins (within the defined map) and the radar amplitude at a selected depth. When more data can be transmitted across the wireless network in the future, full radar traces will be transferred along with each position update.

The simplest header structure accompanies the shutdown procedure. The dataset identification number is included to ensure proper logging, and the number of the acquisition line at the time the client stopped recording is included to assist with restarting the remaining acquisition at the current location. As with the startup header, another timestamp is included to assist with synchronization. The shutdown structure is important because it allows both the client and the server to close the TCP connection cleanly and reuse the port immediately.

4-3) Transmission Size Synchronization

The synchronization of transmission size is performed via a three-way handshake between the mobile client and stationary server as illustrated in Figure 4.2. The process uses recent successes or failures of transmissions to continually determine whether the transmission size should be increased, decreased, or kept the same. This monitoring is performed by the server to minimize extra resources used by the client. Once the decision to change the size is made, the server sends a command to the client indicating the new size to be used. The client responds to the command in the next transmission by setting flag 5 to indicate that the new value has been received, and the server then responds with the similar flag 7 for a final confirmation. If transmission of the command and handshakes are successful, the new transmission size will be used by both hosts starting with the next transmission. Any error in the procedure will prevent the size from changing in order to maintain synchronization. In addition, the client includes the size that is currently being used in the header of each transmission for error-checking by the server. Finally, additional error-checking is performed by the server to detect when synchronization has been lost. If an error is detected, the server aborts the connection and attempts to correct the transmission size appropriately when the client reconnects.

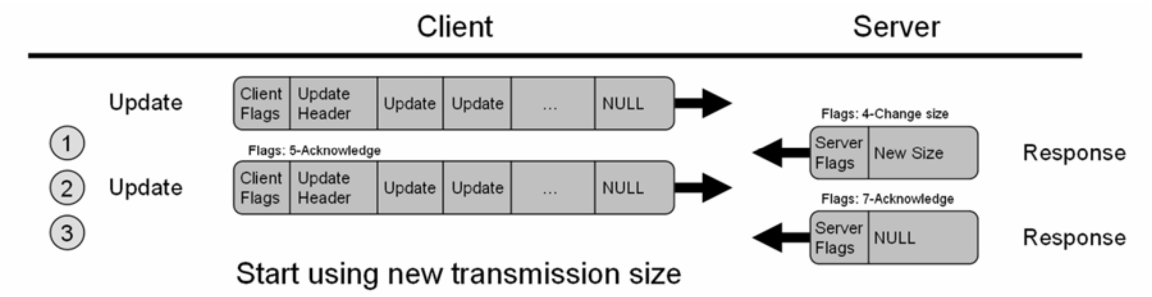


Figure 4.2: Communication flow for three-way handshake when changing transmission size. Step 1) The server initiates the change by sending a command to specify a new transmission size. Steps 2 and 3) Both hosts then acknowledge the new size before the final change is made. Flags are specified above the packets.

4-4) Server-Generated Data Structures

All client-to-server transmissions are followed by a server response. The server response contains only three Bytes consisting of eight flags and two optional data Bytes, and is used to send information requests or commands to the client. When no commands are sent to the client, the two data Bytes are set to null values. The information that can be requested is either the map definition that the client is using or a timestamp. The server must maintain the proper map definition for the client at all times in order to successfully interpret the position updates. However, as the definition rarely changes and requires 48 Bytes to transmit, a solution that does not require continuous sending of this information by the client is implemented. This is accomplished by enabling the client to specify when it is defining a new map and enabling the server to request the map definition if it does not already have it from previous transmissions. Similarly, the server may request a 16 Byte client timestamp at a regular interval (i.e. 60 seconds) instead of the client sending the extra data in the header of each update transmission. When either of these items are requested by the server, a second send and receive is performed immediately to transfer the data. The client sends the standard one Byte of flags followed by either 16 or 48 Bytes for the timestamp or map definition. The extra transmission is completed with another server response packet that acknowledges the received information.

The server data packet is also used to send commands to the client. There are four possible commands, each with a corresponding server flag. When a command is issued, the server sets the appropriate flag and uses the two data Bytes to encode the new desired value for the client to use. The client sets flag 5 to accept the command in the next transmission after the command is received, and the server then removes the command from its buffer to complete the process. In the event of an error during transmission, the server will resend the command until it is accepted by the client. Two of the commands allow the system volume and brightness on the client computer to be adjusted remotely, and two commands are used for adjusting the performance of the wireless network. The timeout for data transmissions can be adjusted in real-time to compromise between waiting too long for failed transmissions and timing out too quickly for successful transmissions. Most importantly, the size of data transmissions is adjusted using the last of the four commands. This command is used to implement the slow-varying transmission size described in Chapter 3.

The method for changing the transmission size and the method for sending the map definition or timestamp work together to provide a special case known as “fast switching” which is illustrated in Figure 4.3. A change in transmission size is initiated by the server in the server response packet. Under normal operating conditions, the client acknowledges the command in the next update transmission and the new size is used beginning with the third transmission. However, the transmission of extra data requested from the server allows this change to happen more quickly. In this case, the server response packet contains the command to change transmission size as well as a request for extra data. The request is most commonly for a timestamp as these are requested throughout data acquisition and not only during startup as with the map definition. The client accepts the command when it sends the requested data, and the result is that the new transmission size can be used starting with the second transmission instead of the third. This case occurs frequently during startup, but less often during normal operating conditions when size changes are less frequent.

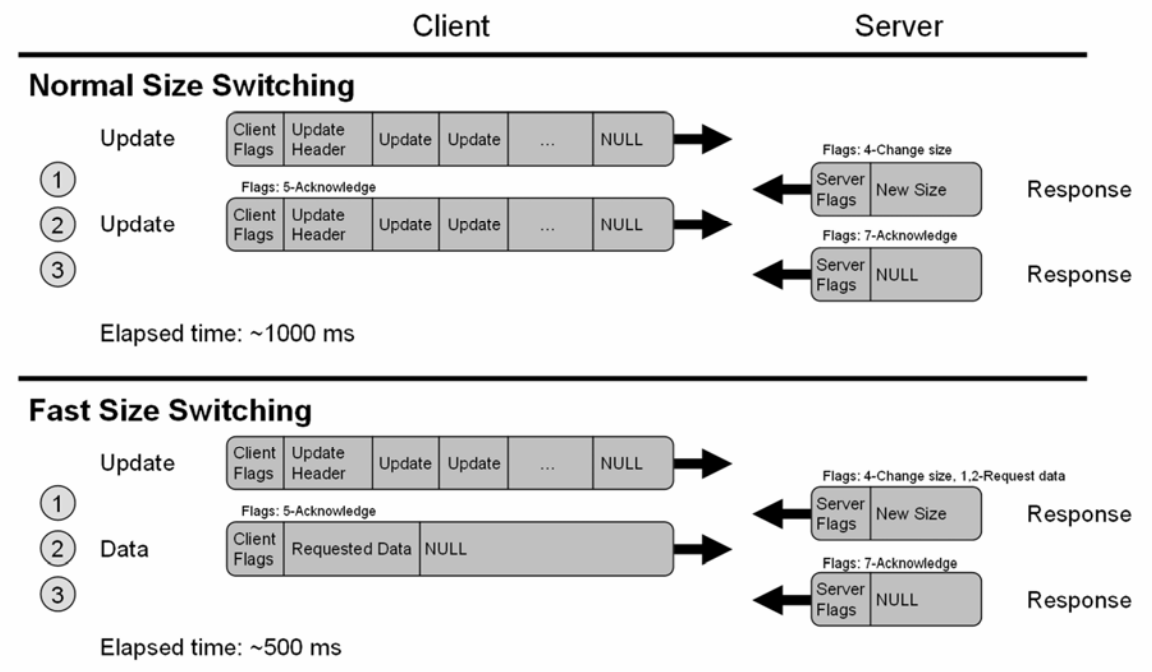


Figure 4.3: Overview of “fast size switching”. The instance of an optional data request along with a command to change the transmission size enables the three-way handshake to take place within one client-server interaction. Flag values are shown above packets.

4-5) Transmission Size Error-Avoidance

As mentioned before, the synchronization of the transmission size between the client and the server is essential in order to maintain reliable communication. Although the method for varying the transmission size is designed to be robust and prevent synchronization errors, some extra features are included so the system can avoid or recover from synchronization loss. First, incoming transmissions are scanned for validity and discarded if deemed invalid. For example, the received data is discarded if the client flags are conflicting, such as if the startup and shutdown flags are both set. Discarding invalid transmissions ensures that erroneous data will not be misinterpreted by the server, which is especially important in relation to the size of the next transmission. Once the server determines that a transmission is invalid, it attempts to respond to the client with a flag set to abort the connection. If the client receives the response it will abort the connection and attempt to reconnect, and if it does not it will abort the connection due to the lack of server response. In either case, the client will establish a new connection using the last good transmission size. If the client was acknowledging a change in transmission size at the time of the failure, it will attempt to acknowledge the change again before actually changing to the new size.

Two more features are included to help restore the synchronization of transmission size in the event that it is lost. One of these features makes use of the fact that when synchronization breaks down the client will send either more or less data than the server is expecting. If less data is sent than expected, the server will timeout waiting for more data and will abort the connection. However, because some of the data is successfully received the server can check the transmission size header to see if it is less than the current expected value and change to the lower value for the next connection. If more data is sent than expected the server will not get an error receiving, but will still detect the synchronization error due to the transmission size specified in the update header being larger than the current expected value. In this case the server will again abort the connection and attempt to correct the expected size for the new connection. The final method for regaining synchronization is used when all other methods fail. After several aborted connections, both the client and server return to a default transmission size in order to ensure that synchronization is restored.

4-6) Wireless Parameter Optimization

a) Connection and Transfer Timeouts

With the implementation of the final system, there are several parameters related to the network connection that need to be optimized. The simplest of these parameters are the timeouts related to both establishing a connection and transferring data. The *connection* timeout applies to the client only because the server waits indefinitely for a connection to be established. However, both the client and the server must use the same *transfer* timeout in order for communication to be successful. The result is that only two timeouts need to be optimized. A one second timeout is used when the client attempts to establish a connection to the server. The client aborts the connection attempt and tries again when the timeout is reached. This allows ample time for the connect routine to succeed under normal operating conditions, but prevents the client from wasting too much time when the wireless connection is weak or not present. Because the data transmissions involve significantly more data than the connect routine, a two second timeout was found to be the optimal transfer timeout. When the transfer timeout is exceeded, the connection is aborted and a new connection is established.

b) Transmission Size Control Settings

The other system parameters to optimize concern the six control settings for the slow-varying transmission size. These values control the rate at which the transmission size is increased or decreased based on the number of consecutive successful or failed transmissions and set bounds on the determined size. Testing of these values is performed by running the system and monitoring data throughput and stability. A good solution is capable of transmitting all desired data under strong network conditions while reducing transmissions enough to prevent system failure under weak network conditions. The best solution found through field tests increases the transmission size by one update (6 Bytes) for every three successful transmissions and decreases the size by one update for every two failed transmissions. The transmission size is bounded between 31 and 1259 Bytes. The minimum size is chosen to allow for the transmission header and one update to be sent at all times. The maximum size ensures that the largest transmission will only use one packet at the TCP layer. For data sizes greater than 1259 Bytes the TCP layer must segment the data into multiple packets for transmission, causing increased transfer times and packet overhead. Testing of these settings with the system showed stable performance over extended periods with no buffer overflows on the client and the ability to overcome transmission failures. A summary of the wireless transmission settings is shown in Figure 4.4.

Parameter	**Minimum**	**Maximum**	**Typical**
Connection Timeout	500 ms	10000 ms	1000 ms
Transfer Timeout	500 ms	10000 ms	2000 ms
Consecutive Successes	1	-	3
Amount of Increase	1	-	1
Consecutive Failures	1	-	2
Amount of Decrease	1	-	1
Minimum Transmission Size	31	-	31 Bytes
Maximum Transmission Size	31	-	1259 Bytes

Figure 4.4: Summary of parameters for wireless data transmission system. The two timeouts specify how long the client and server wait during a connection or transfer attempt at the TCP level before failing. The next four settings determine how frequently and how much to increase or decrease the transmission size based on recent successes or failures. The last two parameters specify an absolute minimum and maximum transmission size.

Chapter 5: Application and Results

5-1) Water Infiltration Experiment

The wireless data transmission system was tested during a field experiment designed to measure the infiltration of water into the subsurface. The 3D GPR system described in Chapter 3 was used as the geophysical sensor for the experiment because of the amount of data generated by the system. The goal of the experiment was to track the infiltration of water into the subsurface over time by repeatedly surveying a specified area before and after artificially injecting water and using the resulting 3D data volumes to construct a "4D GPR" dataset. Because GPR is sensitive to changes in water content [35], the volumes acquired after the injection show a visible effect from the infiltrated water. Advanced processing of these data volumes along with the use of petrophysical transfer functions [36] enables the extraction of changes in water content between two input volumes. The ability to non-destructively track the propagation of fluids in the subsurface has applications in the field of hydrology.

The infiltration experiment was completed over a period of two weeks from July 9-23, 2007 at Ingraham Park in Coral Gables, FL (Figure 5.1). The survey area was chosen to be 18 x 20 meters and could be acquired in ~90 minutes with the 3D GPR system. After two preliminary volumes were acquired for reference, a 4 x 4 meter square artificial pond was installed in the center of the survey area and 3200 Liters of water were injected over a period of four hours (Figure 5.2). The artificial pond was then removed, and 14 post-infiltration surveys were acquired over the next two weeks to track the propagation of the water within the survey area.

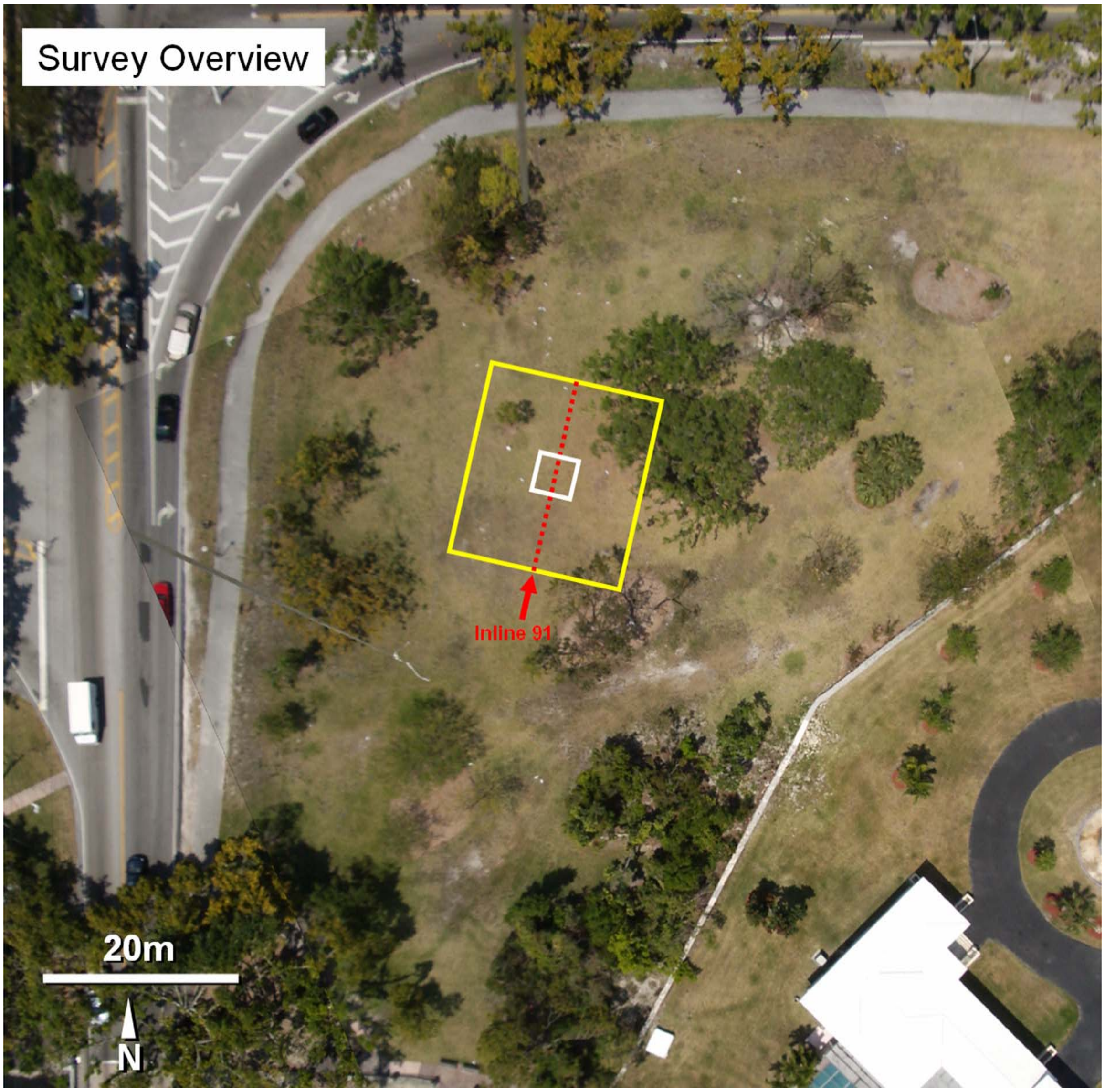


Figure 5.1: Aerial photo of Ingraham Park in Coral Gables, FL with overview of 4D GPR experiment. The large box outlines the 18 x 20 m survey area, while the small box outlines the 4 x 4 m pond area where 3200 L of water were injected. The dotted line indicates the location of an extracted line of the GPR data.

5-2) Role of Wireless System in Experiment

The wireless data transmission system designed for this thesis was integrated with the 3D GPR system and used in the acquisition of all 16 surveys. The surveys were acquired under varying conditions during both day and night with the aid of five users. The server computer was kept stationary in the back of a vehicle, and the wireless router was placed roughly between the server and the closest edge of the survey area. This pattern was repeated successfully through five setups over the two week period.

Figure 5.2: Photo of the 4 x 4 m artificial pond and the five participants of the 4D GPR experiment. 3200 L of water were injected into the pond over a period of 4 hours so the infiltration of the water into the shallow subsurface could be tracked using repeated 3D GPR surveys.

The most significant role for the system in this experiment was to provide statistics about the accuracy of the user positioning the sensor during acquisition. In order for changes in water content to be reliably extracted from the data volumes, the repeated surveys needed to be acquired with very high accuracy. At the same time, the amount of data to be acquired for the experiment necessitated frequent user changes due to physical exhaustion. The wireless transmission system aided in the decision to change users by providing real-time statistics about the accuracy of the user such as walking speed or distance from the target acquisition line (Figure 5.3). Examination of these statistics allowed the decision to change users to be made before there was a noticeable effect on the quality of the acquired data.

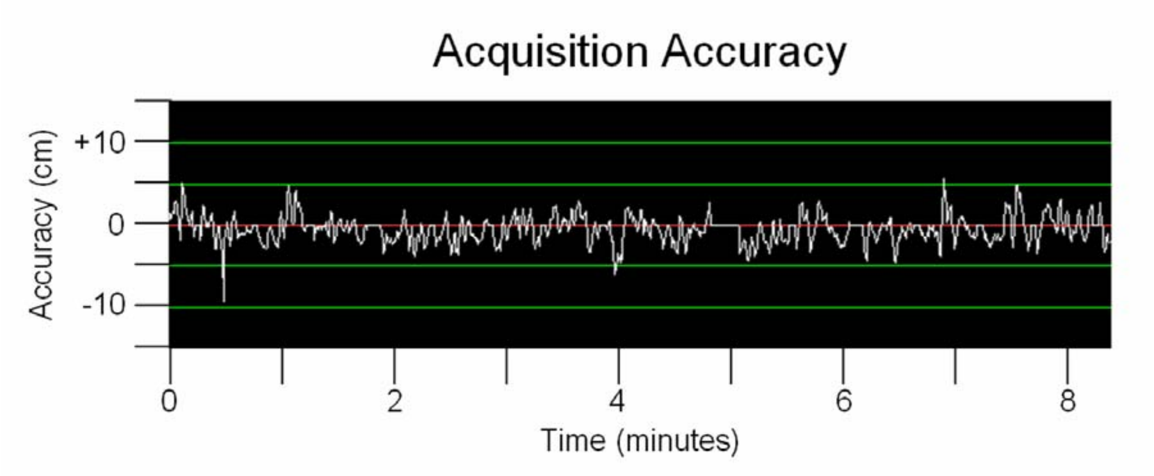
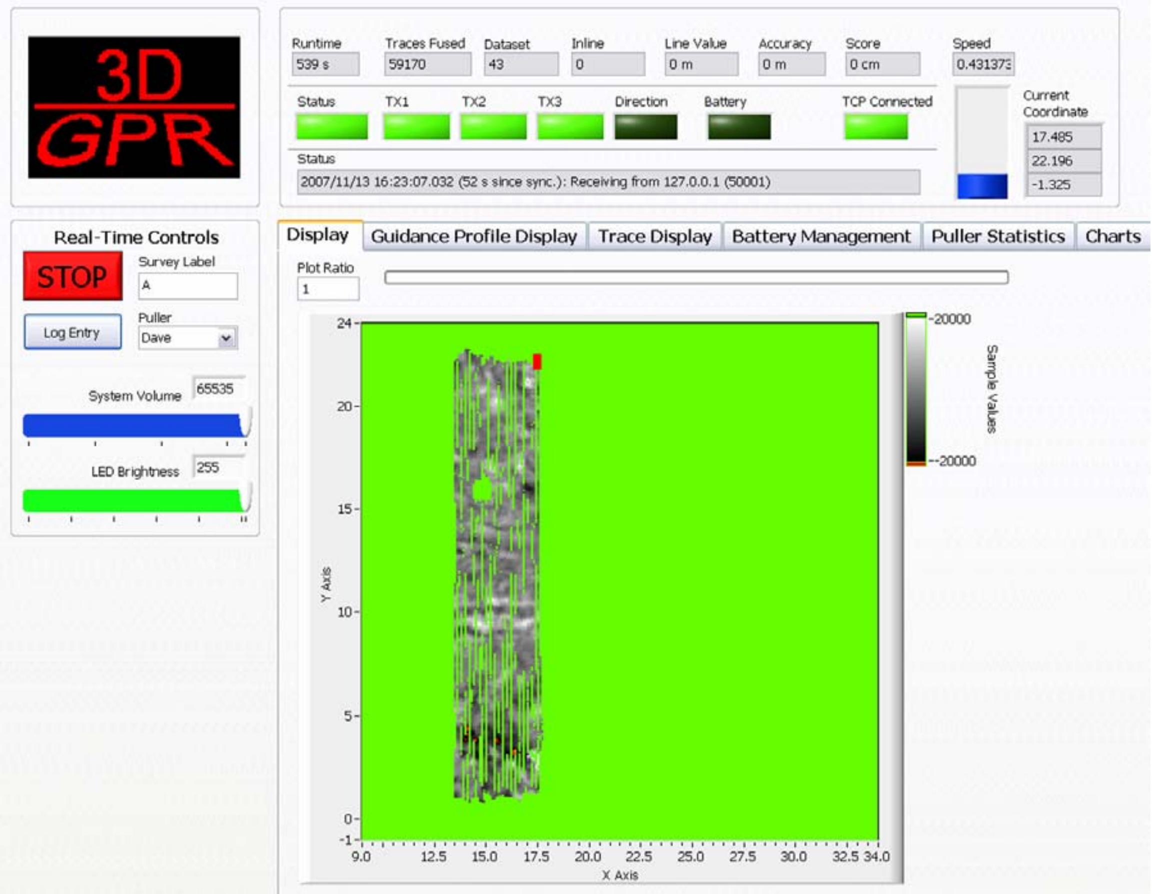


Figure 5.3: Real-time chart used for monitoring user accuracy during acquisition of the survey. The plot shows how far the geophysical sensor is from the target line at the time of each transmission. Increasing deviations over time are an indication that the user is getting tired and should be replaced.

5-3) Results from 4D GPR Experiment

The system performed as desired throughout the experiment and facilitated the error-free acquisition of the 16 3D GPR surveys. The progress of each survey was monitored from the server during acquisition to verify stable performance (Figure 5.4). At the beginning of the night, commands were successfully issued from the server to reduce the volume of the audio indicators and the brightness of the guidance LEDs. On two occasions, failing batteries were detected early by the system and the user was alerted so the batteries could be changed before the quality of the acquired data suffered.



Runtime: 539 s
Traces Fused: 59170
Dataset: 43
Inline: 0
Line Value: 0 m
Accuracy: 0 m
Score: 0 cm
Speed: 0.431373
Current Coordinate: 17.485, 22.196, -1.325
Status: 2007/11/13 16:23:07.032 (52 s since sync.): Receiving from 127.0.0.1 (50001)
Survey Label: A
Puller: Dave
System Volume: 65535
LED Brightness: 255
Plot Ratio: 1

Figure 5.4: Screenshot of server during data acquisition. The display with the green background shows a horizontal depth slice corresponding to approximately 1 m. depth, and the red rectangle indicates the current location of the mobile client. Usage statistics such as positioning accuracy and walking speed are monitored on this screen along with the survey display to ensure successful data acquisition and catch problems quickly.

The survey area defined for this experiment was well within the operating range of the wireless network, with both host computers within 30 meters of the wireless router at all times. Furthermore, there were no other wireless networks detected at the survey site. The result of this was that the wireless transmission system performed very well for this experiment. Updates were sent from the client to the server every 500 ms with only five connection failures over the 16 surveys and between zero and three sensor failures per survey. The optimal transmission size during these surveys was found to be 1259

Bytes, or the maximum allowable size. This means that the network was capable of transferring more data for these surveys due to the close proximity of the wireless hosts and router. However, 5-10 updates on average were sent for each transmission, meaning less than 100 Bytes of data needed to be transferred per transmission. Therefore, the amount of data to be transferred by the current system was much less than the available data rate of the wireless network. While the full transmission scheme sent 19 kbps, only 2 kbps consisted of real data. This extra bandwidth will be necessary when more data needs to be transferred in the future system.

The combination of the laser positioning system and the wireless network permits spatial attributes of the wireless network to be examined. This is done by measuring metrics from the wireless network and plotting them by location on a map. Figure 5.5 shows the received signal-to-noise ratio detected at the Data Link Layer of the network stack. This attribute was measured using third-party software, and the effect of averaging over time is noticeable as vertical stripes which show inconsistencies in neighboring acquisition lines. For comparison, stationary measurements were acquired at several points around the survey area over a period of three minutes and averaged for a long-term result. However, the stationary measurements are consistently weaker than the mobile measurements because the client computer was kept on the ground during the stationary measurements. The mobile signal strength is consistently weaker at the southern end of the survey, which is most likely due to the line-of-sight between the mobile client and the wireless router being blocked by the user. This data suggests that the strength of the wireless network could be improved by mounting the client antenna so that it always maintains a line-of-sight with the wireless router and is at least 1.5 m. from the ground.

Note that there was no line-of-sight blockage in the stationary tests. To measure the maximum attainable throughput of the wireless hardware configuration at the Application Layer, a method was devised to transfer data chunks as quickly as possible and use the elapsed time for each transfer to calculate the throughput of the network. Because this experiment required significant resources from the client computer, 3D GPR data was not acquired. Figure 5.6 shows the results of the experiment. Although there are significant variations in the calculated throughput, areas can be identified with a visible trend that agrees with the measured signal strength in Figure 5.5.

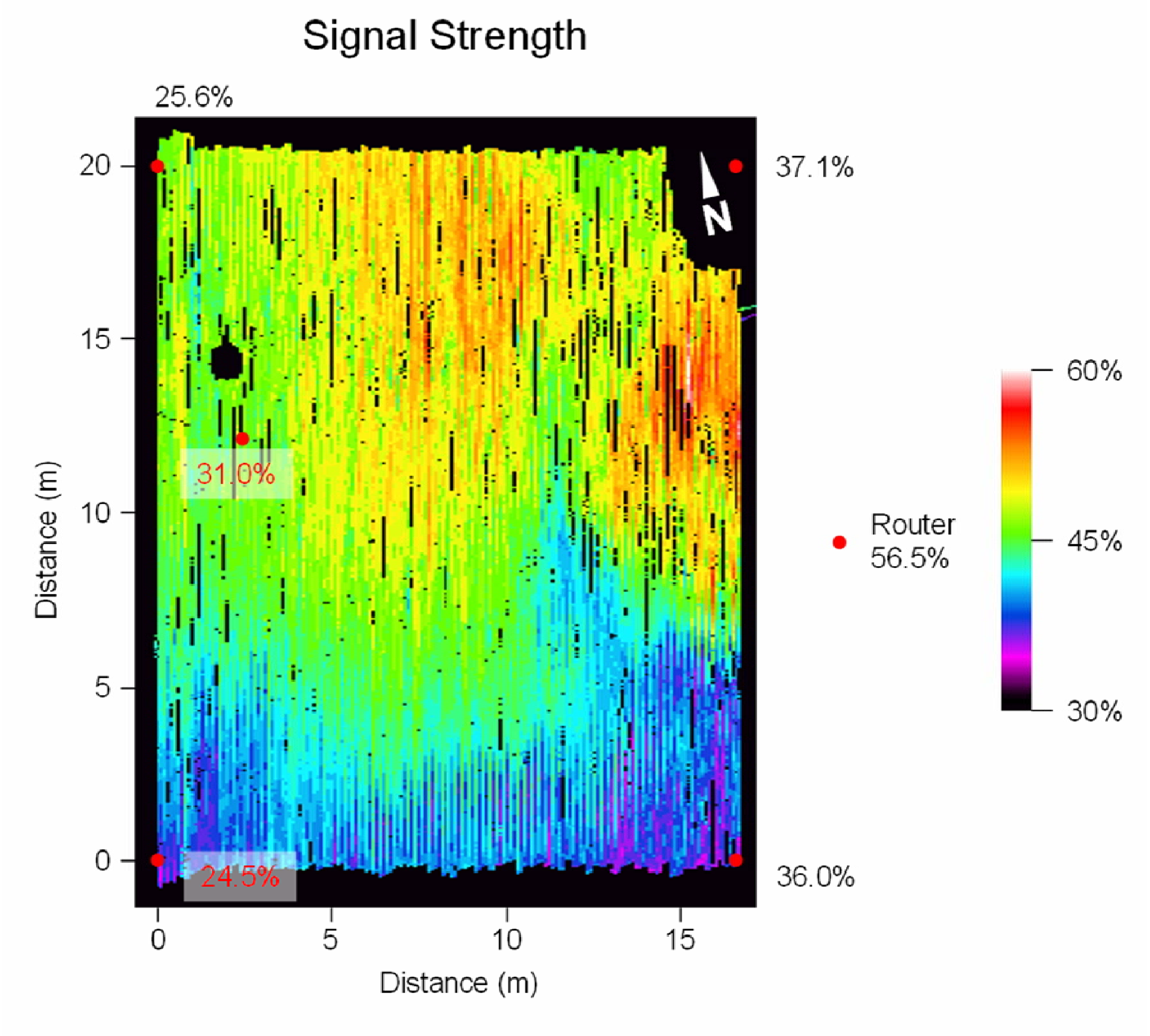


Figure 5.5: Surface map with color-coding based on received signal strength from third-party software. Vertical striping is a result of time averaging which yields delayed results. Lower signal strength values in the southern part of the survey are thought to be caused by loss of line-of-sight with the wireless router due to blockage by the user. Average values for several stationary measurements are displayed for comparison along with a measurement taken next to the wireless router on the eastern side of the survey area. These values are suspected to be lower than the mobile measurements because the client host was on the ground during the stationary measurements and ~1.5 m above the ground while attached to the user's back during the mobile measurements.

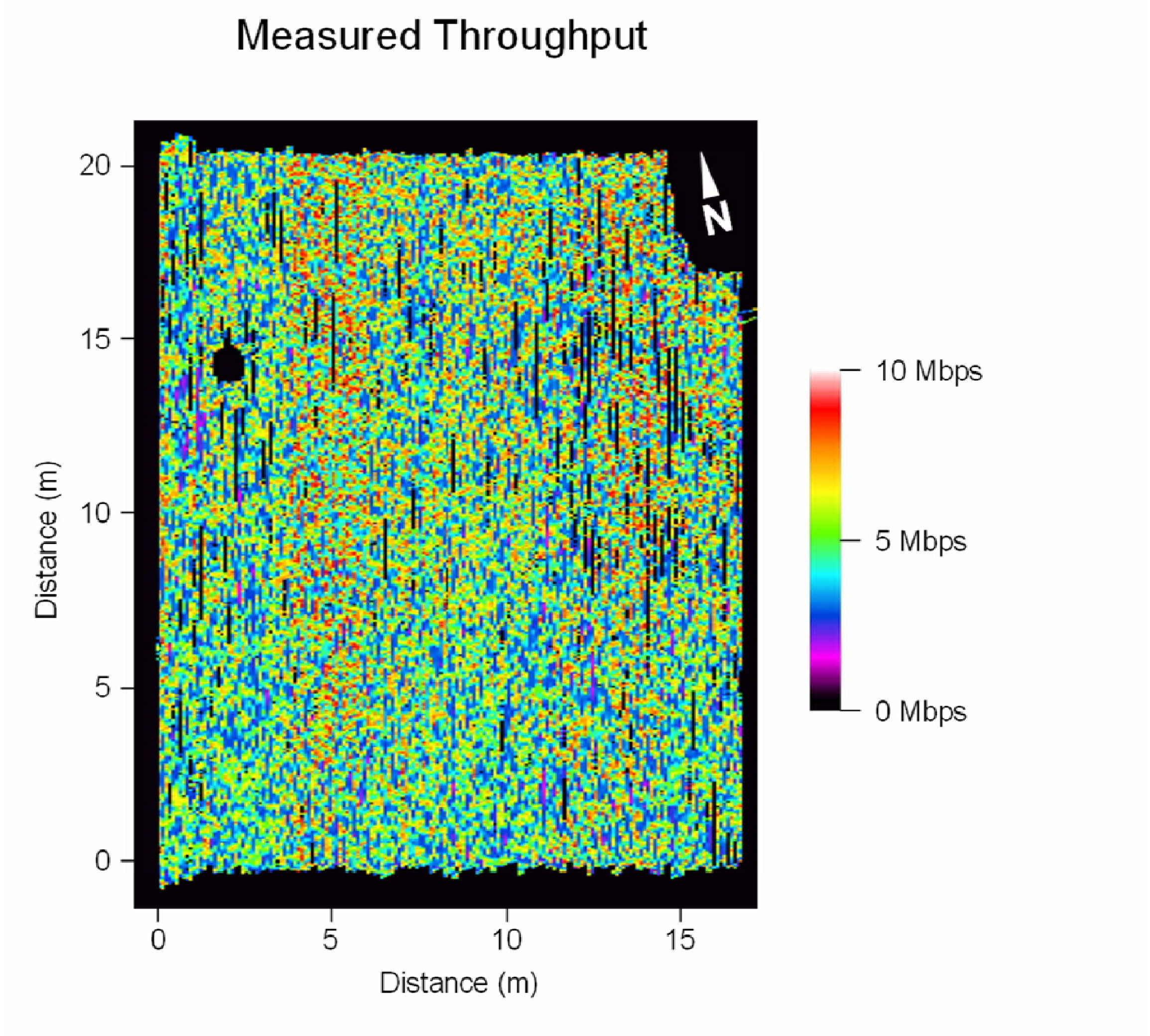


Figure 5.6: Surface map with color-coding based on measured throughput from a designed experiment. Areas of higher throughput can be identified which coincide with areas of higher signal strength seen in Figure 5.5. The average measured throughput over the entire survey area is 4.23 Mbps.

The success of the wireless transmission system contributed to the success of the 4D GPR experiment. Figure 5.7 shows the repeatability of the acquired data by comparing one line from each of the two surveys acquired before the water infiltration as well as the difference between these two lines. The wireless system helped to maintain high repeatability by indicating when to change antenna puller, and the data can be seen to have only minor differences in the repeated acquisitions. Comparisons between surveys acquired after the water infiltration can be used to study the propagation of the injected water into the subsurface over short time intervals. For example, Figure 5.8 shows a sample line acquired both 6 hours and 9 hours after the end of infiltration. Although the differences between the acquisitions related to changes in water content are small, the high accuracy with which the data was acquired permits the extraction of these changes. The result is that the wetting front of the injected water can be extracted and examined in three dimensions as shown in Figure 5.9.

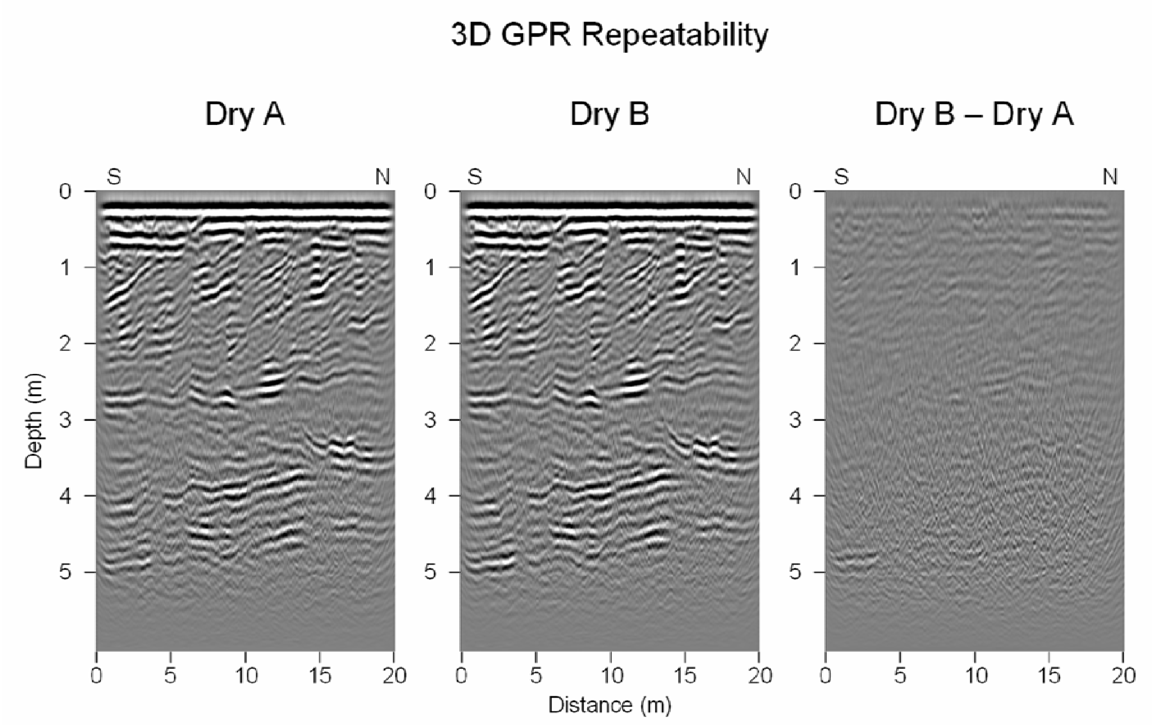


Figure 5.7: Comparison of Line 91 for two “dry” 3D GPR surveys acquired before the water injection. The data for the “Dry A” and “Dry B” surveys is shown along with the difference between the two lines. Difference amplitudes are small in relation to the input amplitudes indicating a high degree of repeatability.

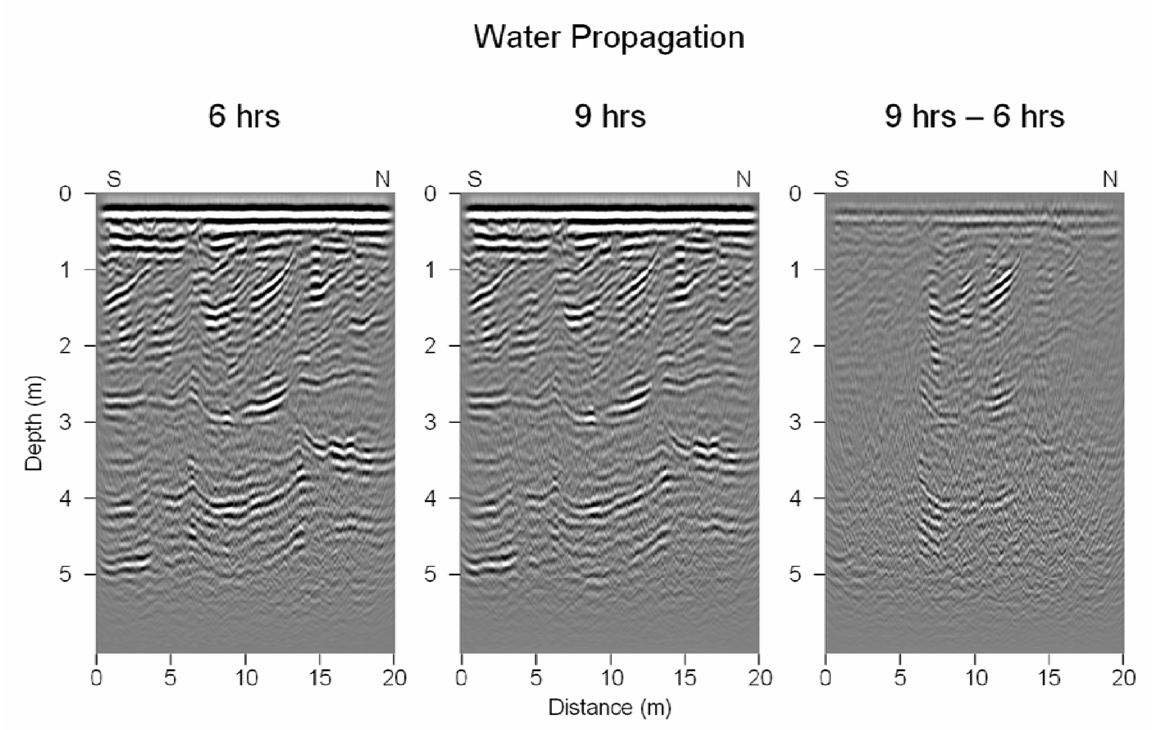


Figure 5.8: Comparison of Line 91 for 3D GPR surveys acquired 6 hours and 9 hours after the end of the water injection. Small shifts in the data related to changes in water content result in large differences in amplitude between the two input volumes.

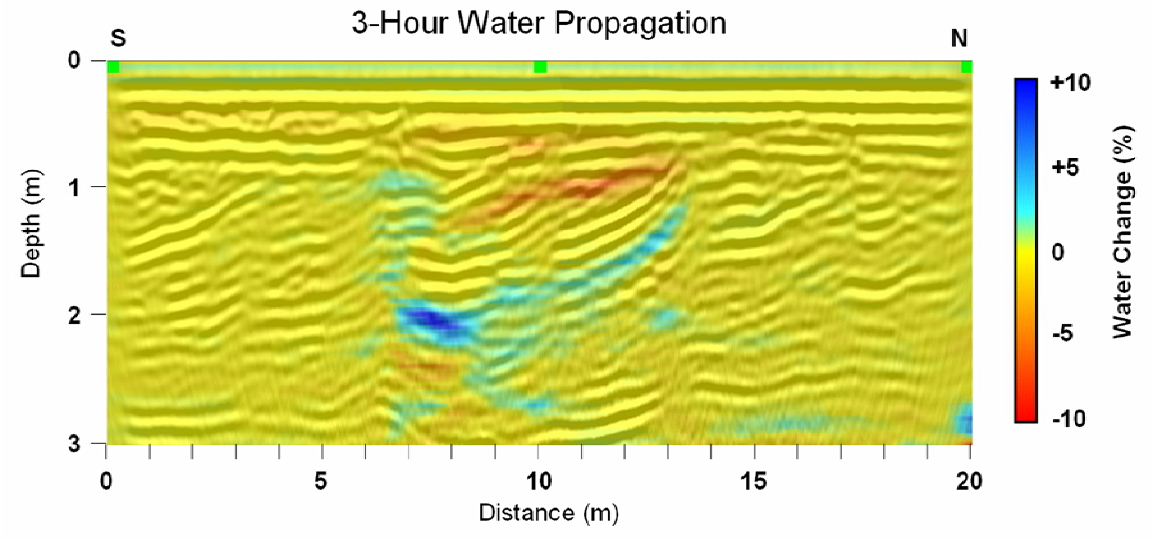


Figure 5.9: Semi-transparent change in water content over three hours overlaid on input data acquired 6 hours after the end of water injection. Local increases in water content are visible in blue at ~2 m depth as the injected water propagates down with gravity. Decreases in water content are visible in red as the top meter of the subsurface begins to drain.

Chapter 6: Discussion

6-1) Significance of System

The wireless data transmission system designed in this thesis is significant because it improves the efficiency at which geophysical data can be acquired. The real-time data display provides a map view of the acquired data for quality control, which allows the user to verify the success of acquisition as well as make decisions based on the data while still on the survey site. Also, the use of audible alarms to indicate immediate problems such as failing batteries or blocked lasers ensures that such problems will be detected early regardless of whether a user is monitoring the real-time display on the server. This system is also a first step toward a system that supports multiple automated clients acquiring a survey simultaneously. For such a system, centralized control of the clients will be necessary in order to ensure that all of the desired data is acquired and that the clients do not collide while moving around the survey area. The server designed in this project will eventually serve as the centralized hub for scheduling and managing the multiple clients.

6-2) System Limitations

The current system maintained low network throughput rates in order to conserve system resources for the GPR acquisition software. Figure 6.1 compares the achieved throughput rate with rates that would be required to transfer all of the acquired data for the current GPR sensor as well as the next-generation GPR sensor which will be released in the near future.

Data Acquisition Rates				
System	**GPR Rate**	**Position Rate**	**Packet Header**	**Total**
Current GPR	200 kbps	121 kbps	400 bps	322 kbps
Future GPR	1.6 Mbps	242 kbps	400 bps	1.84 Mbps

Network Throughput Rates			
System	**Transmission Size**	**Transmissions/Second**	**Total**
Implemented	1259 Bytes	2	20.1 kbps
Max Observed	26.4 kBytes	2	4.23 Mbps

Figure 6.1: Comparison of geophysical acquisition and wireless network data rates. The average observed data rate of 4.23 Mbps over the wireless network is capable of supporting several mobile clients using the future GPR system with a total acquisition rate of 1.84 Mbps per geophysical sensor.

There are several characteristics of the implemented system that warrant improvement. The first limitations relate to the bounds on the mobile client transmission size. The minimum size is specified such that the header and at least one position update are always transmitted, however in some situations this is still more traffic than the network can support. A possible solution would be to implement a “sleep mode” for the client in which attempts to connect to the server are performed less frequently after several consecutive failed attempts. This would reduce the resources used by the client attempting to establish a connection when the network is temporarily inoperable. The maximum transmission size is also a limitation of the current system. For this thesis the maximum size was specified as 1259 Bytes to prevent one transmission from using more than one packet at the TCP layer of the network stack. Although this solution worked well for the implemented system, there are certainly times when the network could support more transferred data. Better optimization of the transmission size to maximize the amount of transferable data would be the first step toward a system in which all acquired data could be transferred to the server for centralized processing and faster results.

There are several limitations of the system that are related to the TCP protocol. The most important of these is the overhead of handshaking to acknowledge transmissions in either direction. Because this overhead is required for all interactions at the TCP layer, a minimum of two extra transmissions are required for each update and response between the client and server. In addition, the extra TCP packet required for the client to acknowledge the server response nearly doubles the amount of data that must be transferred for the response due to the small amount of high level data to be sent. Another limitation associated with the use of the TCP protocol concerns port usage. When a TCP connection fails, the port occupied by the connection becomes temporarily unavailable. The implemented system attempts to circumvent this problem by cycling through a range of ports and restarting at the minimum value when all allotted ports have been used. However, this solution is undesirable because it requires more than one available port and in some cases could be misinterpreted as “port scanning”, a security risk in which an intruder attempts to connect to each port sequentially on a host computer in an attempt to discover an insecure port.

The third limitation of the system is the latency associated with changing the transmission size due to the use of a three-way handshake. This limitation is tolerable because the transmission size is permitted to change only slowly, but the system would benefit from being able to apply the change more quickly once the decision to change is made.

6-3) Future Research

The future research associated with this system focuses mainly on preparing for an automated wireless geophysical sensor network. The new system will require that the server monitor and coordinate several sensors driven by robotic clients as well as receive data from the clients for centralized accumulation and data processing. Completion of the new system will require improvements in the efficiency of data transmission as well as several significant enhancements to the functionality of the server.

The most important step in improving the efficiency of data transmission will be to use the UDP protocol instead of TCP. Recall from Chapter 2 that UDP is a connectionless protocol, and that much less is done to ensure the integrity of received data than in the TCP protocol. However, the goal of implementing UDP in this system will be to mimic the behavior of TCP with some modifications. Features such as packet sequence numbers and checksums will still be necessary to ensure that data is received in the right order and without errors, but the method for acknowledging packets will change in order to combine the server acknowledgment and the server response packets. This will solve the limitation discussed in the previous section concerning the extra overhead of acknowledgments in the TCP protocol. Furthermore, the use of UDP will circumvent the problem of ports being temporarily unusable after a failed connection. Because no connection is established under the UDP protocol, the port will not undergo any changes in the event of a packet failure and will therefore be instantly available after a failure.

There are two ways in which future research regarding the wireless networking hardware can improve the functionality of the system. The first is to implement a hardware scheme that uses more than one wireless router. This step will be important for the implementation of the wireless geophysical sensor network because the multiple clients will acquire data multiplicatively faster, warranting larger survey areas. This will also allow the system to support more clients because each client will communicate with its closest respective router instead of all clients communicating through one router. However, this improvement might also introduce several problems. The first is that there will be some increase in network traffic due to the necessity for messages to be relayed between routers en route to the end hosts. Second, handovers will be required for the clients to change to the nearest router, and the effect of this on system resources will need to be tested in order to ensure that acquisition by the geophysical sensor does not fail.

The second improvement to the hardware scheme is to use hardware that transmits using the 802.11n standard. This new standard, which is officially scheduled for public release in November 2007, provides data rates up to four times that of the current networking technologies such as 802.11g at similar ranges. The increased data rate is made possible by combining four frequency channels as defined by the 802.11g standard into one larger channel, quadrupling the bandwidth of the channel and permitting more data to be sent with each low-level transmission.

The future system will also be expected to transmit all acquired data from the clients to the centralized server. This will require that 1.6 Mbps be transferred per client instead of the currently observed 108 kbps. Some previously mentioned improvements such as changing from TCP to UDP and upgrading to the 802.11n protocol will contribute to the success of this improvement, but it is likely that a more innovative solution will be required. Such a solution may be attained by performing data reduction at the client instead of transmitting all raw data. Another possibility is to implement lossless compression of the data before it is transmitted over the wireless channel.

Another improvement to be made through future research is to upgrade the server for simultaneous communication with multiple clients. In order to do this, the server must handle each client individually through the use of multiple threads or multiple processes. The best method for implementing this multi-client server will need to be determined based on the requirements of the system. This improvement will introduce more hosts that need to share the wireless network and will therefore increase the risk of interference between hosts, so development may be required in order to avoid such interference.

Finally, future research should be applied to integrating the system to geophysical sensors other than Ground Penetrating Radar. The GPR sensor was used for the development of this system because it represents the highest data acquisition rate and sensitivity to computer resources, therefore integration with other geophysical sensors should require relatively few modifications. However, at least one modification will be required in order to distinguish the different kinds of data updates provided by the different sensors. New geophysical sensors should include large enough data buffers so they can survive short communication interruptions. The current GPR system had no data buffer, which forced the wireless system to have an extremely low footprint on system resources. With such buffered instruments the full wireless bandwidth can be better utilized. Also, the sensitivity of the other sensors to computer load will need to be observed and may warrant improvements in the overall stability of the system.

Chapter 7: Conclusion

7-1) Content Review

In this thesis, a wireless transmission system was designed and implemented to assist with the acquisition of geophysical data. Two applications were developed to run in tandem on a mobile client computer attached to the geophysical sensor and a stationary server computer adjacent to the survey area. The applications support the transfer of the acquired geophysical data from the client to the server for a real-time map display of the results during acquisition, as well as commands to control the mobile system from the stationary server without interrupting the geophysical data acquisition. Data transmission from the client to server is optimized through the use of a slow-varying transmission size which attempts to send more data when the wireless connection is strong and less data when the connection is weak. Because the client is always moving and wireless signal quality changes rapidly with location, adjustments are made slowly in an attempt to compensate only for large-scale fading of the wireless signal related primarily to the distance between the transmitter and receiver.

The system was integrated with a 3D Ground Penetrating Radar system for development and testing purposes. GPR was an ideal geophysical sensor for the development of this system because it is sensitive to changes in system load on the client computer and represents the highest data acquisition rate of the available sensors. The final system was successfully tested during a 4D GPR experiment that spanned two weeks and involved the acquisition of 16 data volumes. The system accomplished its tasks without degrading the performance of the 3D GPR system, and improved the efficiency and precision with which the large amount of data was acquired.

7-2) Summary of Contributions

The most significant contribution of the implemented system is the ability to monitor and map the progress of geophysical data acquisition from a stationary server computer without stopping the acquisition on the mobile client computer. This contribution enables errors during acquisition to be detected and corrected more quickly than previously possible, which results in data with better quality being acquired in less time than before the development of the system. Another contribution of the system is the ability to control settings on the client computer from the stationary server through the use of command messages. This allows for certain aspects of the geophysical sensor to be adjusted in real-time without having to stop acquisition on the client computer, improving both the flexibility and efficiency of the geophysical system.

7-3) Significance

The system developed in this thesis is significant to geophysical imaging because it is one of the necessary steps toward implementing an automated wireless geophysical sensor network. Such a system will require centralized data collection as well as coordination of several robotic client rovers, and the implemented transmission system provides a basic foundation for both of these tasks. The currently implemented system also improves the efficiency with which high-resolution geophysical data can be acquired, permitting larger and more frequent surveys and therefore more advanced studies of the shallow subsurface.

The major significance of the system in relation to wireless networking comes from a byproduct of the system. Acquisition of high-resolution geophysical data requires that a given survey area be scanned in a series of closely-spaced parallel lines, resulting in a dense collection of spatial points within the survey area. Attribute maps such as topography are currently displayed by color-coding the given attribute on the spatial data. Similar attribute maps can be made using metrics from the wireless network connection to gain new insight into the variability of wireless Local Area Networks (LANs) within a given area. Such high-resolution attribute maps of wireless networks do not yet exist, but can be acquired easily with this system by measuring the desired metric and linking it accordingly with the position data. Several wireless attributes were measured in this thesis, and future research may provide new understandings of the spatial variations of wireless networks.

Development of this system will continue in the future in order to support the automated wireless geophysical sensor network. The most significant advancements will focus on automation of the mobile client and the addition of multiple simultaneous clients. These advancements relate to the transmission system in that the server will be required to handle simultaneous connections to multiple clients as well as coordinate the tasks of the clients to maximize efficiency of data acquisition and avoid collisions between the mobile rovers. The final proposed system will be capable of imaging large areas with minimal human involvement and several kinds of geophysical sensors simultaneously acquiring data.

Bibliography

[1] National Instruments, Corp. (2005). Product Literature. [Online].
Available: http://www.ni.com

[2] D. Goodman, Y. Nishimura, and J.D. Rogers, "GPR Time Slices in Archaeological Prospection," Archaeological Prospection, 2, 85-89, 1995.

[3] R. Birken, D. E. Miller, M. Burns, P. Albats, R. Casadonte, R. Deming, T. Derubeis, T. Hansen, and M. Oristaglio, "Efficient large-scale underground utility mapping in New York City using a multi-channel groundpenetrating imaging radar system," in Proc. SPIE, 2002, vol. 4758, pp. 186–191.

[4] A.P. Anna, S.W. Cosway, and T. De Souza, "Application of GPR to Map Concrete to Delineate Embedded Structural Elements and Defects," *Ninth Int'l Conference on GPR*, Santa Barbara, CA, pp. 359-364, Apr. 2002

[5] R. Roberts and D. Cist, "Enhanced Target Imaging in 3-D Using GPR Data from Orthogonal Profile Lines," *Ninth Int'l Conference on GPR*, Santa Barbara, CA, pp. 256-261, Apr. 2002.

[6] D.K. Butler, H.H. Bennett, and J.H. Ballard, "Overview of Multimethod Geophysical System Development for Enhanced Near-surface Target Detection, Discrimination, and Characterization," The Leading Edge, 352-356, March 2006.

[7] A. Neal, M. Grasmueck, D.F. McNeill, D.A. Viggiano, and G.P. Eberli, "Full-resolution 3D Radar Stratigraphy of Complex Oolitic Sedimentary Architecture: Miami Limestone, Miami, Florida, USA," Running Ahead, 2007.

[8] A.L. Endres, W.P. Clement, and D.L. Rudolph, "Ground Penetrating Radar Imaging of an Aquifer During a Pumping Test," Ground Water, 38, 556-576, 2000.

[9] S. Truss, M. Grasmueck, S. Vega, and D.A. Viggiano, "Imaging Rainfall Drainage within the Miami Oolitic Limestone using High-resolution Time-lapse Groud-penetrating Radar," Water Resour. Res., 43, 2007.

[10] G. P. Tsoflias, T. Halihan, and J. M. Sharp, Jr., "Monitoring pumping test response in a fractured aquifer using ground penetrating radar," Water Resour. Res., vol. 37, no. 5, pp. 1221–1229, 2001.

[11] NAEVA Geophysics Inc., (2007). Product Literature. [Online]. Available: http://www.naevageophysics.com

[12] R. Young and N. Lord, "A hybrid laser-tracking/GPS location method allowing GPR acquisition in rugged terrain," Lead. Edge, vol. 21, no. 5, pp. 486–490, 2002.

[13] Mala Geoscience, AB. (2005). Product Literature. [Online]. Available: http://www.malags.se

[14] A. P. Annan, Ground Penetrating Radar Principles, Procedures and Applications. Mississauga, ON, Canada: Sensors and Software, 2003, p. 278.

[15] M. Grasmueck, R. Weger, and H. Horstmeyer, "Full-resolution 3-D GPR imaging," Geophysics, vol. 70, no. 1, pp. K12–K19, Jan./Feb. 2005.

[16] ISO/IEC 7498-1, Information technology – Open Systems Interconnection – Basic Reference Model: The Basic Model, June 1996.

[17] IEEE 802.3, Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications, December 2005.

[18] IEEE 802.11, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, June 2003.

[19] RFC 791, Internet Protocol, September 1981.

[20] RFC 793, Transmission Control Protocol, September 1981.

[21] RFC 768, User Datagram Protocol, August 1980.

[22] RFC 2616, Hypertext Transfer Protocol – HTTP/1.1, June 1999.

[23] RFC 2821, Simple Mail Transfer Protocol, April 2001.

[24] T.A. Maufer, *IP Fundamentals*, New Jersey: Prentice Hall, 1999.

[25] A. S. Tanenbaum, *Computer Networks*, 4th ed. New Jersey: Prentice Hall PTR, 2003, pp. 524-532.

[26] K. Fall and S. Floyd, "Simulation-based Comparisons of Tahoe, Reno and SACK TCP," ACM SIGCOMM Computer Communication Review, 26-3, 5-21, 1996.

[27] J.W. Mark and W. Zhuang, *Wireless Communications and Networking*, New Jersey: Prentice Hall, 2003.

[28] M. Garcia, J. Choque, L. Sanchez, L. Munoz, “An Experimental Study of Snoop TCP Performance over the IEEE 802.11b WLAN,” IEEE Wireless Personal Multimedia Communications, 3, 1068-1072, 2002.

[29] A. Bakne and B.R. Badrinath, “I-TCP: Indirect TCP for Mobile Hosts,” *Proc. 15th Int’l Conf. on Distr. Computer Systems*, IEEE, pp. 136-143, 1995.

[30] RFC 1001, Protocol Standard for a NetBIOS Service on a TCP/UDP Transport: Concepts and Methods, March 1987.

[31] M. Grasmueck and D.A. Viggiano, “Integration of Ground-penetrating Radar and Laser Position Sensors for Real Time 3D Data Fusion,” IEEE Transactions on Geoscience and Remote Sensing, 45, 130-137, 2006.

[32] M. Grasmueck, “3-D Ground-penetrating radar applied to fracture imaging in gneiss,” Geophysics, vol. 61, no. 4, pp. 1050–1064, Jul./Aug. 1996.

[33] T. M. Hedges, S. Casteel, T. Cuff, T. Pratt, R. Slater, and D. Todd, “Low cost transmitter with calibration means for use in position measurement systems,” U.S. Pat. Off., Washington, DC, U.S. Patent 6 618 133, Sep. 9, 2003.

[34] Metris (formerly Arc Second). (2006). Product Literature. [Online]. Available: http://www.indoorgps.com.

[35] J.A. Huisman, S.S. Hubbard, J.D. Redman, and A.P. Annan, “Measuring Soil Water Content with Ground Penetrating Radar: A Review,” Vadose Zone Journal, 2, 476-491, 2003.

[36] G.C. Topp, J.L. Davis, and A.P. Annan, “Electromagnetic Determination of Soil Water Content: Measurements in Coaxial Transmission Lines,” Water Resour. Res., 16, 574-582, 1980.

Appendix: Data Structures

Client Data Structures

Client Flags: Each bit interpreted as True/False

Bit	Description
0	Startup
1	Shutdown
2	Abort
3	Reset Map
4	New Map Definition
5	Command Accepted (ACK)
6	System Status Flag
7	(unused)

Startup Header:

Bytes	Format	Description
0	U8	Client Flags
1-2	I16	Dataset Number
3-4	I16	Start Line Number
5-6	U16	Transfer Timeout (ms)
7-8	U16	Bytes/Transmission
9-10	U16	Rotation Angle
11-12	U16	Line Spacing (mm)
13-14	U16	Volume
15	U8	Brightness
16-31	UNIX Timestamp	Timestamp

Update Header:

Bytes	Format	Description
0	U8	Client Flags
1-2	U16	Traces Fused
3-4	U16	Runtime (s)
5-6	I16	Current Line Number
7	U8	Walking Speed
8	I8	User Accuracy
9	U8	Visible Lasers
10-11	U16	Score
12	U8	Number of Updates
13-16	I32	Current X-Value
17-20	I32	Current Y-Value
21-24	I32	Current Z-Value

Display Update:

Bytes	Format	Description
0-1	I16	X Bin
2-3	I16	Y Bin
4-5	I16	Sensor Amplitude

Map Definition:

Bytes	Format	Description
0-7	DBL	Minimum X Value
8-15	DBL	Maximum X Value
16-23	DBL	Minimum Y Value
24-31	DBL	Maximum Y Value
32-39	DBL	X Spacing
40-47	DBL	Y Spacing

Timestamp:

Bytes	Format	Description
0-16	UNIX Timestamp	Timestamp

Shutdown Header:

Bytes	Format	Description
0	U8	Client Flags
1-2	I16	Dataset Number
3-4	I16	Final Line Number
5-20	UNIX Timestamp	Timestamp

Server Data Structures

Server Flags: Each bit interpreted as True/False

Bit	Description
0	Abort
1	Map Request
2	Timestamp Request
3	Timeout Command
4	Transmission Size Command
5	Brightness Command
6	Volume Command
7	Acknowledge Size Change

Server Response:

Bytes	**Format**	**Description**
0	U8	Server Flags
1-2	U16	Command (or NULL)

Server Commands:

Command	**Format**	**Range**
Timeout	U16, in ms	0 – 65.5 seconds
Transmission Size	U16, in Bytes	0 – 65535 Bytes
LED Brightness	U16, no value	0 (off) – 255 (full bright)
Volume Command	U16, no value	0 (mute) – 65535 (loud)