UNIVERSITY OF CALIFORNIA, SAN DIEGO

Antarctica: Exploring the Capabilities of Phased Array Antennas

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

in

Computer Science

by

Shaan Suresh Mahbubani

Committee in charge:

Professor Alex C. Snoeren, Chair
Professor Stefan Savage
Professor Geoffrey M. Voelker

2008

The Thesis of Shaan Suresh Mahbubani is approved and it is acceptable in quality and form for publication on microfilm:

Chair

University of California, San Diego

2008

DEDICATION

If you can dream–and not make dreams your master,

If you can think–and not make thoughts your aim;

If you can meet with Triumph and Disaster

And treat those two impostors just the same;

If you can bear to hear the truth you've spoken

Twisted by knaves to make a trap for fools,

Or watch the things you gave your life to, broken,

And stoop and build 'em up with worn-out tools:

If you can fill the unforgiving minute

With sixty seconds' worth of distance run,

Yours is the Earth and everything that's in it,

And–which is more–you'll be a Man, my son!

–Rudyard Kipling

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

First and foremost, I'd like to thank my advisor, Alex Snoeren. It has been both a pleasure and an honor to have had the opportunity to work with him on this project. I'd like to also thank Stefan Savage and Geoff Voelker for their undending support and ideas. My thanks to Mikhail Afansyev for helping modify the Atheros Linux kernel module, the gaussian fit algorithm, and (along with Brian Kantor) building much of the hardware needed to run these experiments. Finally, I'd like to thank Patrick Verkaik for helping to maintain the client node firmware, as well as always being available for ideas and support.

ABSTRACT OF THE THESIS

Antarctica: Exploring the Capabilities of Phased Array Antennas

by

Shaan Suresh Mahbubani

Master of Science in Computer Science

University of California San Diego, 2008

Professor Alex C. Snoeren, Chair

Beam-forming antennas provide the potential for spatial reuse when deployed over large areas. Phased array antennas are a particular type of beam-forming antennas that are electronically steerable. We present an exploration of the abilities of phased array antennas to form directional beam patterns. The phased array antennas we use are Phocus Array systems, from Fidelity Comtech [3]. We show that while the beam patterns achieved by the Phocus Arrays conform are

directional, they do not conform to specification very accurately. We hypothesize that the Phocus Arrays may be designed to be used primarily for line of bearing determination, and show evidence of this ability.

1

Introduction

In recent years, wireless Internet (802.11 WiFi) has experienced an explosion in availability; a large number of businesses and large organizations have deployed enterprise scale wireless networks. The rapid growth, both in terms of wide-scale deployment and user devices utilizing WiFi for Internet connectivity, has led to demand for enhanced service from users. The expected improvement in service is not only in the demand for more bandwidth and faster data rates, but also higher-layer requirements such as better security. It is these demands that have prompted a large body of work at all layers of the protocol stack aimed at achieving such improvements.

One area of the networking stack that can be modified to solve these problems is in the area of antenna design. The vast majority of mobile devices that are capable of accessing wireless networks use simple omnidirectional antennas. However,

higher layer protocols (such as 802.11) must take into account the possibility that wireless devices with omnidirectional antennas may interfere with each other when transmitting or receiving, and implement mechanisms aimed at avoiding such interference. In contrast, directional antennas provide an opportunity to mitigate the effects of interference more effectively at the physical layer.

Recently a sub-genre of directional antennas, beam-forming antennas, have become viable as enterprise-grade devices. Beam-forming antennas have the ability to transmit radio waves in a (possibly complex) beam pattern. Phased array antennas are a type of beam-forming antennas. Phased array antennas are groups of individual antennas that collectively emit radio waves in a beam pattern; they achieve this by controlling the phase and amplitude of the radio waves emitted from each individual component antenna, thereby controlling the resultant wave (as a product of constructive and destructive interference of the component waves).

Directional (and more specifically beam-forming) antennas offer several advantages over traditional omnidirectional antennas. Beam-forming antennas often have the ability to transmit over larger distances (since the emitted wave is the result of constructive interference from multiple component waves); furthermore, the beam can be made to form only over a specific area (since the beam can be complex, and therefore closely resemble a directional beam in any given direction). These two properties make directional antennas ideal for deployment over large open areas such as sports venues, concert venues, and other similar locations. In-

stituting wireless coverage of such areas with omnidirectional antennas may be more complicated. One option is to use many omndirectional devices that transmit with low power, resulting in more complexity (and possibly cost) at times of deployment and maintenance. Another option is to use fewer devices that transmit with much stronger power, causing interference to neighboring areas. Directional antennas, however, present an opportunity for ease of deployment (since fewer devices need to be deployed, as each device can transmit for longer distances), as well as interference avoidance to neighboring areas (since the beam can be shaped to accurately cover only the intended area).

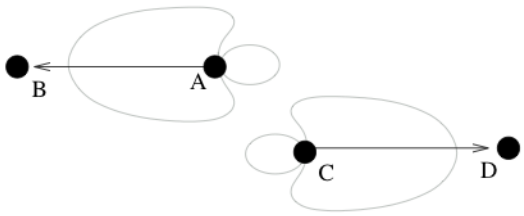


Figure 1.1: How spatial reuse can be achieved with the use of directional antennas. From Choudhury et al. [24].

Directional antennas can be used to achieve spatial reuse in a given geographic area. Spatial reuse is when nodes that are co-located within a given area are able to simulaneously transmit (or receive) at the same time with the use of directional antennas. Consider the nodes A, B, C and D in a network that is layed out as in Figure 1.1. If A and C transmit with omnidirectional antenna, they will interfere with each other, and thus will not be able to transmit at the same time. However,

as the diagram shows, if they both transmit with directional antenna, then they do not interfere with each other and can transmit at the same time. This behaviour has the net effect of increasing the total network throughput.

Since directional antennas achieve a directional beam pattern in both their transmit and receive patterns, these antennas can also be used to improve the security of a client's connection. An attacker's task is made much more difficult if they cannot hear the transmissions from a directional nodes that have their beam pointed in a different direction; similarly an attacker's abilities are limited if they cannot send packets to a node (because the node has its directional receive pattern pointed somewhere else). Note that attackers may also use directional antennas for malicious intent: An attacker may use directional antennas to increase the range over which packets can be heard.

Phased array antennas are not only directional, but electronically steerable; the phase and amplitude being emitted out of the individual antenna elements can be controlled and changed dynamically, resulting in the emitted beam pattern being reconfigured very quickly at run-time. This ability means that the emitted beam can be repointed at different areas, making the antennas useful for applications such as tracking movement of client devices (for example in search-and-rescue missions). The antennas can be reconfigured depending on a predetermined requirement (such as coverage of only one side of a conference hall), or dynamically during a frequently changing situation (such as mobile clients commuting across an area).

1.1 Focus of the thesis

We explore the capabilities of enterprise-grade phased array antennas with the use of Phocus Array systems from Fidelity Comtech [3]. Phocus Arrays are electronically steerable directional antennas that have the ability to transmit over large distances (several miles), while forming complex beam patterns. Specifically, we attempt to show how effective phased array antennas are at achieving spatial reuse; spatial reuse is achieved by maintaining connectivity to nodes in certain directions, while ignoring interference from nodes in other directions. To quantify the ability of phased array antennas to achieve spatial reuse, we explore their ability to effectively form complex directional beam patterns. The experiments conducted follow a logical progression through a set of experiments that aim to verify the beam pattern. We are able to demonstrate the that Phocus Arrays produce a receive pattern that is somewhat similar to what they should be theoretically producing; however the differences between the observed and expected patterns are non-trivial. Furthermore, we are unable to show that the Phocus Arrays transmit a beam pattern that conforms to specification. While we attempted to prove that the Phocus Arrays can be used for spatial reuse, to date we have been unable to demonstrate this behaviour. As an epilogue, we show that the Phocus Arrays are able to determine directionality for associated clients with relatively high accuracy; we theorize that this application is the one at which the Phocus Arrays are designed to be primarily used.

1.2 Overview of the thesis

The remainder of the thesis is organized as follows. Section two presents some background concepts fundamental to the understanding of the rest of the thesis; the section also elaborates on some related work in the areas of directional and beam-forming antennas. Section three presents the methodology used in the thesis, including the Phocus Array systems as well as other tools used in the experiments. Section four discusses the experiments conducted in pursuit of the goal of showing how effective phased array antennas are at achieving spatial reuse. Finally, section five summarizes some of the conclusions gained from the experiments performed.

2

Background and Related Work

This chapter contains the background and related work sections of the thesis. The background section begins by introducing some basic concepts about wireless networking, as well as directional antennas. The concept of wireless networking is introduced against the backdrop of general computer networks. The lower four layers of the OSI model are described, with focus placed on the physical layer and the 802.11 MAC. The concepts of radio waves and interference are introduced, within the context of omnidirectional and directional antennas. The chapter continues to examine some of the literature related to this work. Previous works showing the benefits of directional antennas are examined, including how using directional antennas can allow for greater network throughput and spatial reuse. The related work section continues to describe previous works that show how directional can be used in networks with the existing 802.11 MAC. Also examined are works that

build new systems (including new MAC protocols) that attempt to fully utilize the potential of directional antennas. The section ends with an examination of some of the other benefits that can be gained with the use of directional antennas.

2.1 Background

This section explains some of the underlying concepts related to the thesis. The section begins by laying out the fundamentals of wireless networking. General and wireless computer networks are examined. The seven layer OSI model of networking is introduced, with emphasis being placed on the physical, data link, network and transport layers. Some important aspects of 802.11 are discussed. The fundamentals of radio wave transmissions are explained, including interference. The section continues to introduce some of the different classes of antennas, including omnidirectional, directional, and beam forming antennas.

2.1.1 Wireless networking

This section elaborates on the concepts of wireless networking. It begins by introducing computer networks, and expounds on wireless networks. Networking protocols are explained in the lower four layers of the OSI model. The physical layer is discussed, with emphasis on radio waves; properties of radio waves such as the inverse power law and interference are explained. The link layer is introduced, with expansion on the 802.11 MAC. Some fundamentals of 802.11 are discussed,

including data rates, carrier sense, and retransmissions. The network and transport layers are explained, with focus on TCP, UDP and IP.

Computer networks

Computers today are connected to each other for the purposes of communication and information sharing. Such a collection of computers connected together is known as a computer network. One way of classifying the type of network formed by the computers is to refer to the underlying physical medium which is used to connect the computers in the network. Some of the most common types of connection media are physical cables, such as optical fiber or Ethernet. Such networks, because they are connected via a physical cable, are known as wired networks. Conversely, there exists a class of networks known as wireless networks. As the name denotes, computers in a wireless network communicate with each other without the use of wires. Instead, they communicate by transmitting information over the air with the use of some form of electromagnetic radiation. Another way of classifying computer networks is by their relative size (in terms of number of computers connected to them). A popular variant is the Local Area Network (or LAN). Such a network is used to connect computers in a relatively small geographic area, such as a school or office building. We are interested in the area of networking that lies in the intersection of wireless networks and local area networks - wireless LANs (or WLANs). A protocol is a standard set of messages and communication mechanisms that are agreed upon by multiple parties. Devices in a network often

adhere to a standardized protocol in order to be able to communicate with each other. The protocols used by computers that communicated in a network are divided into layers. Each layer is described in terms of the functionality performed by protocols that operate in that layer. The standard layer model used to describe computer networks is the Open Systems Interconnection Basic Reference Mode, or OSI seven layer model.

Wireless networks

A WLAN is a network of computers in a relatively small geographic area that connect to each other wirelessly. The physical signals transmitted by computers in such a network are some form of electromagnetic radiation. Electromagnetic radiation is energy that moves through space in a wave form. These waves have a wavelength (the physical distance between two adjacent peaks in the wave) and a frequency. Most computers connected to a WLAN at present communicate with the use of radio waves. Radio waves are any electromagnetic waves with a frequency that is between roughly 3 kHz and 300 GHz.

Physical layer

The lowest layer in the OSI seven layer model is the physical layer. The physical layer describes the physical medium of communication. The physical medium of communication used by computers in the wireless networks we are working with are radio waves. The most common types of radio waves used by computers in

a wireless network have a frequency of roughly 2.4 GHz. Devices that transmit radio waves at roughly 2.4 GHz are said to be transmitting in the 2.4GHz Industrial Science and Medical (ISM) band. The ISM bands are a set frequency ranges that are known as the unlicensed spectrum. In the United States, any device may transmit in the unlicensed spectrum, as long as the emitted radio wave adheres to certain rules (such as maximum allowed transmit power) specified by the Federal Communications Commission (FCC).

Interference

When two radio waves of the same frequency and amplitude are combined, the net resultant wave depends on the phase difference between the two. The phenomenon of a wave being produced as the result of the combination of multiple component waves is known as wave interference. If the two waves are completely in phase, the peaks and troughs of both waves pass by the same point in space at the same time (as if the waves are superimposed upon each other). The resultant wave has double the amplitude of each component wave. If two waves are exactly 180° out of phase, the peak of one wave occurs at the same time as the trough of the other wave. The peak of one wave cancels out the trough of the other wave, and the resultant wave is nullified. At any given time, the amplitude of the resultant wave is the scalar addition of the amplitude at that time of components of each individual wave.

Inverse-square law

Radio waves (and in fact all electromagnetic radiation) adhere to the inverse-square law. This law states that the power or strength of a physical phenomenon degrades with inverse proportionality to the square of the distance from the source of that phenomenon. This law can be expressed as $\rho \propto \frac{1}{r^2}$, where ρ is the power of the phenomenon, and r is the distance from the source. The strength of a radio wave at a given point is measured by the power of the wave at that point. The power of a radio wave is measured in decibels (dB). In terms of radio waves, this equation means that the strength of a radio wave (in dB) degrades in proportion to the inverse of the square of the distance from the source. Thus an increase in the strength of a radio wave at the source results in the wave propagating for a much longer distance.

Link layer

The physical layer describes the physical medium of communication between nodes in a network. However, there needs to be a standard set of messages passed between computers via the radio waves in order for them to be able to talk to each other. The layer above the physical layer is known as the data link layer. A link layer protocol is a standard set of messages and timing intervals that all computers who use the protocol agree to adhere to. Link layer messages are transmitted between computers via the physical medium of choice. The Institute of Electrical

Engineers (IEEE) is an organization that deals with the standardization (among other things) of link layer protocols. One such set of protocols is known as IEEE 802.11. 802.11 protocols describe the link layer messages that are passed between computers, the time intervals that need to be respected between transmissions, and the modulation schemes used. A modulation scheme is the way in which messages are encoded into physical signals for transmission. 802.11 compliant devices transmit in the 2.4 GHz ISM band. Two important variants of 802.11 that we are dealing with are 802.11b and 802.11g.

802.11

The differences of 802.11b and 802.11g are many. One of the important differences to understanding this work pertains to the supported transmission rates of the two protocols. Data is transmitted in discrete quantums known as packets. 802.11b supports the transmission of a packet at certain rates, measured in Mbps. 802.11b supports rates of 1Mbps to 11Mbps. 802.11g supports all the rates supported by 802.11b, as well as higher rates up to a maximum of 54Mbps. The different rates are achieved by changing the physical encoding of the packet into a radio wave. Since the higher rates are achieved through more dense encoding of the physical packet, packets sent at a higher rate are also more difficult to decode. Therefore, packets are only transmitted at higher rates when the link quality is perceived to be good. Over links that are perceived to be of poor quality (many packets are either dropped or can not be decoded correctly), packets are sent a a

lower rate. Thus it can be said that packets transmitted at a higher rate are harder to 'hear'. Over the air, packets are transmitted as radio waves. The noise floor is the signal level of the background noise that can be heard in a given environment. The strength (in dB) of the noise floor is also the strength at which radio waves can not be successfully decoded. The strength or power of the radio wave above the noise floor (in dB) is called the Received Signal Strength Indication, or RSSI. The strengths of the noise floor and the radio wave are measured by the wireless card. One indication of the quality of a link may be the RSSI, under the assumption a stronger radio wave allows packets to be encoded at higher rate (since higher rates are harder to decode and thus required a stronger signal), increasing the throughput over the link.

802.11 allows the specification of packets to be retried a certain number of times. Every packet that is transmitted in 802.11 must be acknowledged with an ACK packet from the recipient to the sender. If this ACK packet is not received by the sender, the sender may try to send the packet again a certain number of times. In general, data packets that are sent from a sender to a receiver are retried a number of times if there is no ACK received for them. Packets that are sent to multiple recipients (multicast), or to anyone within listening range (broadcast) are only transmitted once.

In wireless networking, interference occurs when multiple devices emit radio waves of the same frequency at the same time. When multiple nodes that are

nearby each other try and transmit at the same frequency (or channel) at the same time, the waves destructively interfere, and the component waves may not be able to be decoded by a receiver. To avoid destructive interference, the 802.11 protocol includes the specification of Carrier Sense Multiple Access with Collision Avoidance, or CSMA/CA. CSMA/CA specifies how each node should listen to the medium to detect if anyone else is transmitting before transmitting. The implication of CSMA/CA means that nearby nodes who can hear each other can not transmit at the same time, thereby reducing the possible total network throughput.

The 802.11 specification includes a feature called Request To Send (RTS)/Clear to Send (CTS). RTS/CTS is a mechanism which nodes in the network use to avoid having their transmissions collide with the transmissions of other nodes. The mechanism is iniated when a node who wishes to transmit, upon observing a clear channel (one in which there are no transmissions in progress) sends an RTS packet with a duration of the intended transmission. The intended recipient node should reply with a CTS packet, indicating that it will not transmit anything for the duration requested in the RTS. Any well behaved node who hears this exchange will also respect the duration specified, and avoid transmission for that time period.

802.11 nodes can be configured as either clients or Access Points (APs). An AP is a node to which client nodes can associate. AP nodes broadcast beacon packets every 100ms. These beacon packets let client nodes know that there is an AP node in the area. The beacon packets may contain a Service Set Identifier, or

SSID. The SSID is a name used to identify which network the AP provides access to. Nodes that are configured as clients do not broadcast beaon packets; they may associate with AP nodes (if associated is permitted by the AP's access privileges).

Consider the case when a node in an 802.11 network sends a data packet to another node. The receipient of the packet must send an acknowledgement (ACK) packet back to the sending node, signifying receipt of the packet. If the sending node does not receive this ACK (either because the destination node did not get the packet and therefore did not send an ACK, or the ACK was lost), the sending node will attempt to retransmit the packet.

Link quality

There may be some confusion regarding the definition of link quality. In layman's terms, the quality of a link between two noces is (obviously) how good the link is at allowing data to be transmitted across it. In more specific terms, there are several measures of the quality of a link. One measure may be the Round-Trip Time, or RTT. Keeping in mind the fact that if a packet is not correctly decoded at the receiver (and thus no ACK is sent), or the sender does not receive (either because one is not sent or it is unable to decode one) an ACK, the sender will attempt to retransmit the link layer packet. The RTT at the sender is defined as the time in between the first packet is transmited and the time that an ACK is received. If the packet must be retransmitted, or the ACK must be retransmitted, the RTT will increase. If either the original packet or the ACK was not able to

be correctly decoded (or was lost along the way), the link is obviously not as good as it could be. It is based upon this observation that the RTT may be used as an indication of link quality. Another possible candidate indicator of the link quality is the RSSI of the packets received on that link. The use of RSSI as an indicator of link quality is based on the assumption that the stronger a packet is (i.e. the higher its signal to noise ratio), the higher the probability is that it will be decoded correctly.

Network and Transport layers

The layer above the link layer is the network layer. Protocols implemented at this layer are responsible for the delivery of data arriving from higher layers. One of the most important aspects of this layer is the method in which nodes are addressed in the network. This addressing scheme determines how packets are routed and forwarded between source and destination nodes in the network.

The most common protocol in use at the Network layer is the Internet Protocol, or IP. In IP, hosts are addressed by globally unique integer addresses. Data from higher layers is passed to IP in discrete units called packets. The IP adds some information to the beginning of the packet, called a header. The IP header contains information such as the source host's IP address, and the destination host's IP address. If the packet is forwarded via intermediate nodes in between the source and destination, the intermediate nodes can look at the destination IP address in the IP header to determine where to forward the packet to next.

Above the network layer is the transport layer. The transport layer is responsible for allowing applications on different hosts in the higher application layer to communicate with each other. While the network layer is responsible for delivery of packets between hosts, the transport layer provides an intermediary between higher level applications and the network layer.

The two most commonly used Transport layer protocols are the Transmission Control Protocol (TCP), and the User Datagram Protocol (UDP). TCP is used by applications that required guaranteed in-order delivery of data. It is a connection oriented protocol, which means that hosts must set up a TCP session between them before TCP packets can be sent from one to the other. TCP guarantees in order reliable delivery by specifying that the recipient must transmit an acknowledgement to the sender to indicate which packets have been received. In contrast, UDP is a stateless protocol that does not guarantee delivery or ordering of packets. UDP is useful when the sending application needs to transmit data to any number of recipients, but does not need to know how many recipients received the transmission. For example, a server that broadcasts the current time to any number of recipients (such as a Network Time Protocol, or NTP server). There is no acknowledgement sent back from the recipient to indicate receipt of a UDP packet.

2.1.2 Antennas

This section discusses some of the different types of antennas that are used in wireless networks. The section begins by introducing the most common omnidirectional isotropic antenna. The section continues to discuss directional antennas and their benefits over omnidirectional antennas. Finally, beamforming antennas are explained, with further expansion of phased arrays.

Omnidirectional antennas

Computers in a wireless network communicate by sending data in the form of radio waves. To send radio waves, they need to have a physical component attached to them that is capable of transmitting and receiving the radio waves. Such a component is known as an antenna. An antenna is transducer: a device capable of converting energy from one form to another. Radio antennas convert data in the form of electrical signals transmitted to them over a wire from the computer into radio waves (when transmitting); they also convert radio waves they 'overhear' from the air back into electrical signals (when receiving).

An antenna that emits radio waves with uniform power in all directions is known as an omnidirectional antenna. Figure 2.1 is a graph of the power of a radio wave emitted by an omnidirectional antenna at a fixed radius in all directions.

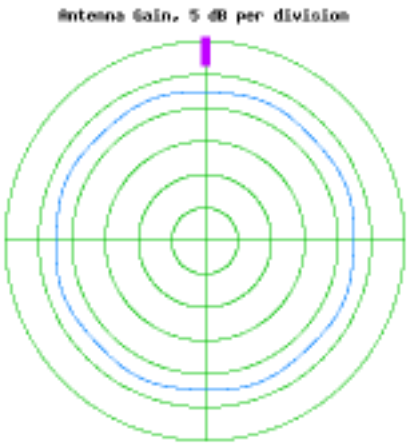


Figure 2.1: The beam pattern for an omnidirectional antenna. The line shows the relative beam strength at various angles to the Phocus Array. The distance the line is at any given point from the center of the graph indicates the magnitude of the beam at that angle. From Fidelity Comtech [3].

Directional antennas

Not all radio antennas emit radio waves of the same power in all directions (as omnidirectional antennas do). An antenna that emits a radio wave with greater power in a certain direction than in other directions is known as a directional antenna. The radio wave emitted from the antenna is said to have a higher gain in certain directions, i.e. the signal strength (measured in dB) of the radio wave in those directions is stronger than in other directions. The most famous example of a directional antenna is a Yagi-Uda antenna. Directional antennas apply a signal strength gain to the radio wave that is transmitted in certain directions. They also receive radio waves in those same directions with a gain over radio waves that

come from other direction. Put another way, directional antennas can 'hear' radio waves better in certain directions.

The properties of directional antennas give them some benefits over traditional omnidirectional antennas. One benefit is that since a directional antenna transmits in only one direction, it can focus all of its energy into transmitting a radio wave in that direction, resulting in a much stronger wave. Since radio waves adhere to the inverse-square law, a much stronger wave at the source results in the wave propagating for much longer distances. Thus a directional antenna provides the ability to transmit a radio wave over a much longer distance, allowing communication between nodes in a network that are much further away. Another benefit is that since a directional antenna can transmit a radio wave in only one direction, this means that co-located directional antennas can transmit simultaneously, achieving spatial reuse and increasing network throughput (this will be further explained below in the related work).

Beamforming antennas

Beamforming antennas are directional antennas that have a transmit/receive beam pattern. Whereas a simple directional antenna has one particular direction in which it achieves transmit and receive gain, a beamforming antenna transmits radio waves with varying transmit/receive gains several directions, forming a beam pattern. An example beam pattern, known as a co-phase beam pattern, is shown in Figure 2.2. This diagram shows that the beam pattern has a main lobe at 0°,

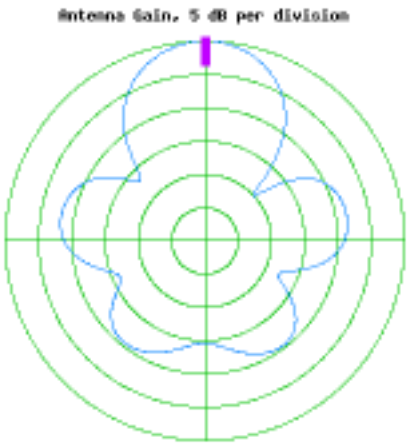


Figure 2.2: A example of a co-phase beam pattern. The line shows the relative beam strength at various angles to the Phocus Array. The distance the line is at any given point from the center of the graph indicates the magnitude of the beam at that angle. From Fidelity Comtech [3].

two smaller side lobes and two back lobes. The pattern also has local minima, known as 'nulls', for example at roughly 45°. These are areas in the beam pattern where the transmit/receive gain is minimal for that pattern.

Beamforming devices are devices that contain multiple transmitters/receivers. When transmitting, the phase of the wave emitted from each transmitter is controlled in such a way as to cause constructive and destructive interference in specific directions. It is this combination of multiple component waves that produces the beam pattern.

Phased array antennas

A phased array is a group of antennas that collectively act as one large radio antenna. A phased array antenna is a beamforming antenna. Radio waves from the component antennas are combined and controlled to produce a specific beam. One example of the beam pattern that can be created from a phased array antenna is Figure 2.2. The beam pattern emitted by a phased array antenna can be a static fixed pattern pointing in a specific direction, or can be changed very quickly. One common method of operation is to use multiple beam patterns, which are identical to each other except that they are rotations of each other. If the phased array antenna is made to quickly iterate through these patterns, it has the effect of scanning the azimuthal plane with the main lobe, allowing the direction of other radio entities to be established (this will be described in more detail below in the related work). The beam emitted by phased array antennas can also be made to be omnidirectional, as in Figure 2.1. Thus phased array antennas provide an advantage over static directional antennas in that they can very quickly be reconfigured to act as normal omnidirectional antenna, directional antenna that statically point in a fixed direction, or directional antenna that scan the azimuthal plane.

2.2 Related Work

This section discusses literature related to the thesis. Works defining the benefits of directional antennas are examined. The section continues to look at literature on how directional and beamforming antennas can be used. These works include how directional antennas can be used to achieve spatial reuse and build systems; further works examine how to adapt existing and develop new MAC protocols. Numerous other works also show how directional antennas can be used for many other uses, including improving network performance, gain better security, and attempt to perform node localization.

2.2.1 Benefits of directional antennas

The de facto standard antennas used by consumer devices in wireless networks are omnidirectional antennas. These antennas allow the network to function normally without having to pre-configure the physical orientation of nodes with respect to each other; furthermore, nodes may change change their orientation without significantly impacting the network topology. However, the use of omnidirectional antennas hinders the network designer's ability to lay out the network in such a way as to maximize spatial reuse without introducing interference between neighboring nodes. The interference observed by neighboring nodes has the effect of reducing network capacity.

Gupta and Kumar [17] observe that the per node throughput in an ad hoc

network with omnidirectional antennas is $\Theta(\frac{1}{\sqrt{n \log n}})$ bits/sec. This model can be extended to show that the total end to end capacity is $O(\frac{1}{\sqrt{n}})$ bits/sec. This equation shows that the capacity of an ad hoc network scales sub-linearly with the number of nodes. The main reasons for this are two fold. One reason is that all nodes that neighbor a transmitting or receiving node are prevented from transmitting or receiving any data due to interference. Another reason is that with the increase in the number of nodes in the network comes an increase in the number of hops through the network; thus as the number of nodes increase, so does the amount of traffic each node forwards for other nodes, i.e. traffic for which it is not the original transmitter or intended recipient. Yi et al. [31] extend this model to determine the capacity of ad hoc networks with directional antennas. They show that in an arbitrary network, the lower bound on achievable improvement in capacity with the use of directional antennas is a factor of $\frac{2\pi}{\sqrt{\alpha\beta}}$, where α and β are the widths of the main beam for a transmitting and receiving node. In a random network, throughput capacity can be improved by a factor of $\frac{4\pi^2}{\alpha\beta}$. The authors claim that these gains are achieved as a result of decreasing local interference and the global overhead of relaying data over multiple hops.

Network capacity can be improved with the use of beamforming antennas. Ramanathan [26] examines the performance of ad hoc networks when using beamforming antennas. He notes that the maximum throughput achievable by each node as observed by [17] is due in part to the fact that transmissions in the net-

work are omni-directional; such transmissions cause interference to neighbor nodes in directions other than that in which the intended recipient lies, as well as reducing the possible range of transmissions due to lower signal strength. Specifically, Ramanathan examines what advantages can be afforded to the network and MAC layers by two properties of beamforming antennas: reduced interference to non recipient nodes due to a narrow steerable beam, and better signal to noise ratio arising from higher signal gain in a specific direction. He measures the effectiveness of several modifications to the various areas of the traditional networking protocol stack. His results show that beamforming antennas can increase throughput across the network while reducing end to end delay.

2.2.2 Utilizing directional antennas

This section discusses literature that explores the various ways in which directional antennas can be used, and the advantages they provide over omnidirectional antennas. Works are examined that show how directional antennas can be used to achieve spatial reuse, build better MACs (as well as multiple layers of the protocol stack), improve network throughput, and achieve greater security with the use of directional and beamforming antennas.

Achieving spatial reuse

In addition to exploring the advantages of using directional antennas over omnidirectional antennas, previous works have sought to build networks with direc-

tional antennas in order to exploit those advantages. Neufeld and Grunwald [21] claim that steerable phase array antennas will soon be widely available, and cheap enough to be considered for wide-scale deployment in commercial wireless networks. They observe that much work has been done on providing a new MAC protocol to effectively utilize the capabilities of directional antennas in such networks, and note that moving to a new backwards incompatible MAC layer would result in the abandonment of a large number of wireless devices based on 802.11. The authors claim that the main problems with using directional antennas with an 802.11 MAC arise from the assumptions that 802.11 makes about the physical orientation and capabilities of each entity in a wireless network. For example, the process of listening before transmitting to see if the medium is busy is only effective if all possible interfering nodes can be heard. For example, the problem of deafness [15] occurs when a node C tries to transmit to a node B which is receiving data from node A; when A uses a directional antenna pointed at B, C can not hear the transmission and thus does not know that if it tries to transmit to B, B will not be able to hear the transmission due to interference. The 802.11 MAC assumes that all transmissions from all possible interfering neighbors can be observed. For this assumption to hold in a network with directional antennas, the authors propose that the antenna sacrifice some spatial reuse by widening their beam angles. If the antenna beams are as narrow as possible, interference at nodes other than the intended recipient of a transmission is minimized, and spatial reuse (and deafness)

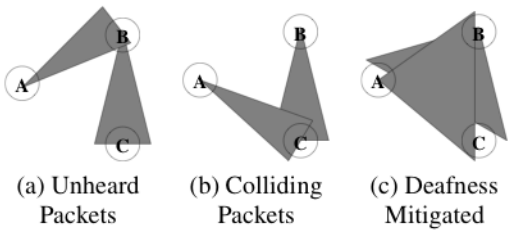


Figure 2.3: Using an 802.11 MAC effectively with directional antenna by widening the beam angle. From Neufeld and Grunwald [21].

is maximized. By widening the directional beam such that neighbors in the area are able to hear the transmission from one node to another, spatial reuse is still achieved to a certain degree, and nearby nodes who may want to transmit can still rely on carrier sense to decide when to transmit without causing interference; an example of interference mitigation by widening a directional node's beam is shown in Figure 2.3.

Choudhury et al. [24] explore how build an ad-hoc network using beamforming antennas. They attached electronically steerable passive array (ESPAR) antennas to the wireless cards of several laptops. Their nodes select the best beam to transmit to a given neighbor by first performing a sweep. A sweep entails sending a beacon packet with a unique identifier using each of 12 beams. Each neighbor node remembers and notifies the original node of which beam it should use to send data to it. Figure 1.1 shows an example of spatial reuse that the authors used their testbed to examine. They show that the aggregate throughput of two links when using directional antenna for both transmitter and receiver is almost

double the throughput attained when all nodes are using omnidirectional antenna. They attribute this throughput to the lower amount of interference caused by neighboring links.

Building systems

In contrast with Neufeld and Grunwald, Choudhury et al. [16] propose a new MAC protocol to fully exploit the capabilities of directional antennas. They propose that there are two important properties of directional antennas that provide an advantage over omnidirectional antennas. One property of directional antennas is the ability to form a narrow beam and direct it from a transmitter to a receiver. This ability allows more than one pair of nodes that are nearby each other to simultaneously communicate, thereby achieving spatial reuse. Another property is the ability of directional antennas to focus their transmit energy into one direction, achieving a long range beam. This ability allows communication between nodes that would not be able to transmit data to each other if their antennas were omnidirectional. The authors propose the Basic Directional Medium Access Control (DMAC) protocol, where a node listens in omnidirectional mode, but reserves the channel with a directional RTS/CTS handshake. They note that that this protocol presents several issues, including: hidden terminal problems that are introduced by the directionality of both sender and receiver; the existence of null regions in a node's beam pattern, coupled with network topology may have unintended consequences on link availability; and deafness (as explained in the

next paragraph). The authors further propose a second protocol, the Multi-Hop RTS MAC (MMAC). They note that while deafness and hidden terminals are still possible with the use of MMAC, performance can be improved with the use of finer-grained control over each node's directionality.

Going one step further than Choudhury et al., Ramanathan et al. [27] describe a complete system solution for ad hoc networking with directional antennas. Recognizing the various advantages directional antennas have in ad hoc networking, the authors propose that simply replacing omnidirectional antennas with directional antennas is not sufficient to maximize those advantages. They claim that in order to maximize the full potential offered by directional antennas, the network and MAC layers of the protocol stack need to be redesigned to interact with each other and control the antenna system. This redesign must include mechanisms that were developed with omnidirectional antennas in mind, such as neighbor discovery and routing. The interacting components of various layers of the network protocol stack should be able to control the state of the antenna, including the beam direction and transmit power. The authors describe the system they developed to achieve this goal, called Utilizing Directional Antennas for Ad hoc Networking (UDAAN). The UDAAN system includes a new CSMA/CA based MAC, a directional neighbor discovery mechanism, a new module which takes information from the link layer and makes it available to other layers, and a new routing and forwarding protocol.

Gaining security

In addition to increasing per node throughput and network capacity, using directional antennas in a wireless network also provides the opportunity for increased security in the network. Carey and Grunwald [13] propose a network design that incorporates phased array antennas with the goal of increasing security in mind. The phased array antennas they use have four important properties that allow them to be used to enhance security. The antennas are directional, allowing the ability to avoid transmitting in unintended directions. The signal transmit power from each antenna can be controlled, thereby allowing the range of the transmitted signal to be controlled. Multiple antennas in different physical locations can be used, providing the opportunity for the dynamic switching of which direction a transmission for a single recipient can come from. The authors combine these capabilities to form a Multiple Input, Single Output (MISO) network layout, where one client can be reached by several antennas in different physical orientations with respect to the client. They propose an algorithm called "spatial data striping" 2.4. In this algorithm, a client is associated with an access point, W. When there is data to be sent from the wired backbone to the client, the access point (AP) associated with the client will look up which other APs can be heard by the client in a "station table"; the packet is then forwarded to those APs, who transmit them to the client. Because the 802.11 MAC only accepts packets based on the the AP to which it is associated, the other APs must spoof the MAC ad-

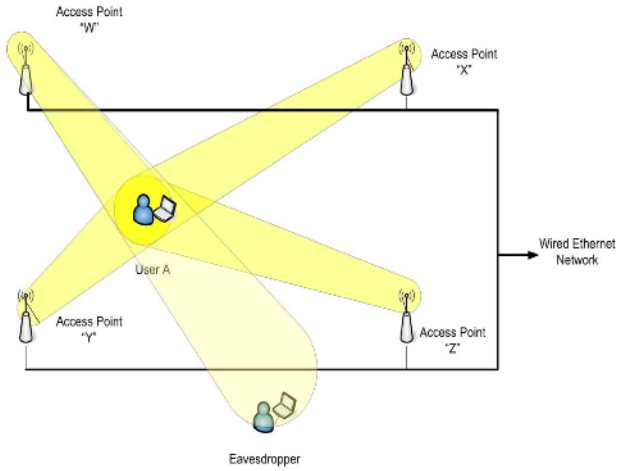


Figure 2.4: Achieving better security with Spatial Data Striping. From Carey and Grunwald [13].

dress of W when transmitting data to the client. In this design, packets can only be overheard by adversaries who are in the same direction of the beam from the transmitting antenna, and relatively close to the client (since the physical distance the beam extends to is limited by a reduction in the beam's transmit power). The authors admit that several issues may affect their proposed algorithm, including the determination of direction from the APs to the client, the need for spoofing mac addresses of the associated AP by other APs, and the effect of multi-path reflection and fading on the determination of direction to the client by each AP; however they propose that outdoor environments such as a baseball field would be an appropriate application of their design.

Other uses

Several other works have investigated the advantages of beamforming and antennas in terms of various aspects of network performance and communication between nodes. Navda et al. [20] use phased array antennas to improve communication between a moving vehicle and a fixed roadside AP. Balamis [11] shows how antenna arrays can be used with digital signal processing algorithms to build adaptive beamforming antennas; the impact on network throughput from using these antennas is examined. Vilzmann et al. [29] show how beamforming antennas can be used to reduce the number of hops between nodes in a network; this reduction leads to a speed up in the time taken to flood messages through the network. Winters et al. [30] examine the achievable increase in range with the use of phased array antennas.

Other uses for directional antennas include location determination and positioning systems. Blanco et al. [12] examine the effectiveness of directionality in indoor environments. They further explore the impact of this directionality on spatial reuse and the ability to perform localization. Niculescu et al. [22] use directional antenna for indoor positioning. Sayraan-Pour and Kaspar [28] perform direction estimation inside buildings with beamforming antennas.

Yet another benefit gained from the use of antenna arrays is spatial reuse. Park et al. [23] show that with multiple transmitters and receivers, spatial reuse can be achieved.

There are a number of works that deal with developing new MAC protocols for use with beamforming and directional antennas. Choudhury et al. [14] design a new MAC for use with directional antennas, as do Zhu et al. [32]. Raman et al. [25] design a new MAC for use with long distance mesh networks with directional antennas. Ko et al. [19] design multiple new MAC schemes to best utilize directional antennas. Huangand [18] compares, contrasts, and evaluates the relative performances of several directional and omnidirectional MAC protocols.

3

Methodology

This chapter discusses the hardware tools available for testing; these tools include the Phocus Array system and its modes of operation, and the Soekris box embedded platform. The chapter continues to discuss the locations in which experiments take place. The chapter also explains what software tools are available for testing, including those that generate packets, as well as those tools that can measure and capture packets.

3.1 Hardware tools

This section introduces the physical devices that the experiments use. This section describes the Phocus Array system, the phased array antenna on which the experiments are performed. This section introduces the device, the ways in which it can be confiugured, and its modes of operation. The tests use Soekris box

embedded devices as client side network node/monitoring nodes; this section also discusses these devices.

3.1.1 Phocus Arrays

The phased array antennas that the tests use are Phocus Array systems from Fidelity Comtech [3]. The Phocus Array systems are IEEE 802.11b/g compliant beamforming antennas operating in the 2.4 GHz ISM band. Figure 3.1 is a block diagam of the main functional components of a Phocus Array System. The antennas are powered by a Power Over Ethernet (POE) injector that takes a 24V 2.5A input. The Ethernet cable providing power is plugged into the base of the antennas, and fed internally into a single board computer. The computer is controlled by an Intel XScale [5] 425 processor. Attached to the board is a wireless LAN (WLAN) card. The WLAN card transmits one single radio wave, which is carried by a radio frequency cable to an eight-way radio frequency splitter. Each output from the splitter is fed into a transmit receive (T/R) module. These modules are devices that modify the phase and transmit power of the RF signals that are fed into them. The output of each T/R module is sent to an antenna, which transmits the radio wave. The radio wave transmitted by the Phocus Array can be as strong as 15dB.

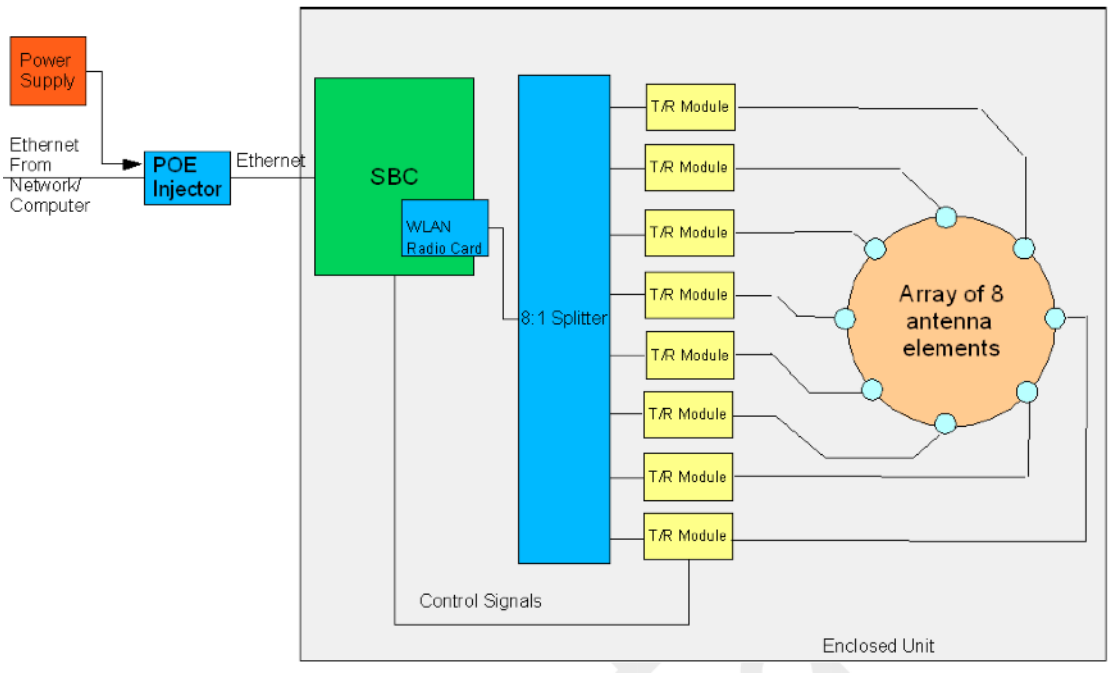


Figure 3.1: A block diagram of the main logical units in a Phocus Array System. From Fidelity Comtech [3].

The embedded computer in the Phocus Arrays runs OpenWRT [8], a Linux based operating system for embedded computers. The device driver for the wireless lan card is an Atheros MadWifi driver.

Usage

There are two interfaces that allow a user to configure the Phocus Arrays. The main interface is a set of web pages that allow a user to configure, administer and monitor the system. The pages allow many aspects of the Phocus Arrays to be configured, including the mode in which to operate and the beam pattern to

use. In the Phocus Array systems, each of the eight antennas is controlled by its own electronic transmit receive module. Each module controls the phase and amplitude of each antenna element, thereby allowing the creation complex resultant beam patterns. The phase and amplitude needed by all eight antennae to form one beam pattern can be pre-computed and stored as one state configuration; many such configurations can be stored in flash on the Phocus Array system. In addition to controlling the antennas to transmit a certain beam pattern, the transmit modules can also be made to change the antenna configuration very quickly at run time. The Phocus Array is initially loaded with 17 beam patterns: one omnidirectional (as shown in Figure 2.1), and 16 directional patterns. When the omnidirectional pattern is loaded, the Phocus Array acts as a normal omnidirectional antenna, transmitting a radio wave with equal strength in all directions. All of the directional patterns are either very similar or identical. The main feature of the directional patterns is the fact that there is a maximal area of the pattern, known as the main lobe. The angle of the main lobe of each of the patterns with respect to the Phocus Array occurs at a multiple of 22.5°. There are 16 different multiples of 22.5° in 360°, which yields 16 directional patterns. The different types of directional patterns differ by the magnitude and location of the side and back lobes (for example, see the difference between the side and back lobes of Figures 3.2 and 2.2).

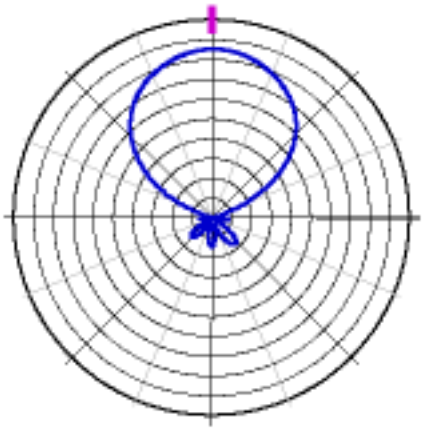


Figure 3.2: The beam pattern for an directional antenna. The line shows the relative beam strength at various angles to the Phocus Array. The distance the line is at any given point from the center of the graph indicates the magnitude of the beam at that angle. From Fidelity Comtech [3].

Modes

The Phocus Array operates in one of two modes, either static or dynamic. In static mode, one of several different beam patterns is loaded, and the system does not change the pattern. The beam pattern can be omnidirectional, such as Figure 2.1, or a more complex directional pattern. Two examples of a directional beam pattern are: a co-phase pattern, as shown in Figure 2.2; and a low-sidelobe pattern, as shown in Figure 3.2. The figures show the transmit/receive strength of the antenna for every angle. For example, in Figure 3.2, packets transmitted from the Phocus Array to another node who is at 0° (relative to the Phocus Array) will have much stronger RSSI than packets transmitted from the Phocus Array to

a node who is at 270°; this observation assumes that both clients are the same distance away from the Phocus Array, and have identical wireless radios/antennas (and thus identical receive sensitivities). Ignoring client-side differences such as receive sensitivity, the RSSI of packets received at a client to the Phocus Array is a function of the angle at which a client lies to the Phocus Array and the beam pattern that is loaded. Note that the receive pattern of the Phocus Array is identical to the transmit pattern; i.e., the antenna exhibits receive sensitivity according to the pattern that is loaded and the angle of the transmitting node. Thus if nodes with identical distances and transmit strengths are assumed, the RSSI of packets received at the Phocus Array becomes a function of the angle at which the transmitting node lies relative to the Phocus Array.

The Phocus Array system can also operate in dynamic mode. In dynamic mode, the system maintains which of its static patterns provides the best link to each associated client, and configures itself into that optimal pattern when transmitting to the client. The optimal beam pattern for each client is ascertained in the following way. The Phocus Array will configure itself into the first of the 16 directional beam patterns. While in that pattern, it will send a random packet containing dummy data to each client. Each client, upon reception of that packet, will send back a link-layer acknowledgement packet to the Phocus Array. The Phocus Array records a mapping from the client's MAC address and the beam pattern currently loaded to the signal strength of the acknowledgement into local

XML file. The Phocus Array then reconfigures itself into the next static beam pattern, and repeats the process. At the end of a full sweep, the Phocus Array has a signal strength measurement for every pair of associated clients and beam patterns. The optimal pattern to use with each client is simply the pattern that yields the strongest signal strength measurement for that client. This sweep is performed at a regular interval; the duration of that interval can be configured on the web page interface (with a minimum interval of five seconds). In static mode, this interval is ignored and the chosen beam pattern is not changed. In dynamic mode, a single beam pattern cannot be chosen.

3.1.2 Clients

The experiments use external devices to test the functionality of the Phocus Arrays; these devices act either as associated clients or as monitoring devices. As clients, the nodes associate with a Phocus Array, thus allowing transmission (and measurement) of data in either direction. As monitoring nodes, the devices promiscuously listen to every packet that can be decoded; packets from a particular source can be filtered out and analyzed. The tests use modified Soekris Engineering based devices as external network nodes for both of these purposes. This section explains these devices.

Devices

The external monitoring and client nodes are modified Soekris Engineering [10] net4826 based devices. Soekris boxes are small, portable computers that can be powered by a DC battery. The boxes have two physical interfaces: an RS-232 serial port and an Ethernet port. The boxes also provide two external RF interfaces, into which radio antennas can be plugged. Each Soekris box has 128MB of RAM and 64MB of flash, controlled by a 266-MHz AMD Geode CPU. Attached are two mini PCI based Wistron CM9 802.11a/b/g wireless cards, which are each wired to one of the external RF interfaces. Connected to each of the Soekris box's external RF interfaces is an omnidirectional "rubber duck" antenna that provides a gain of two to three decibels. The boxes are running BusyBox [2], a Linux distribution designed for embedded devices.

Usage

A network interface can be created and associated with either of the two wireless cards. Both wireless cards can be used simultaneously, if there is an interface created and associated with each (resulting in two wireless network interfaces). Each interface can be configured to associate with an access point with a specific SSID. Other aspects of the interface can be specified, such as the channel to be on, and the power (in dB) to transmit packets at. The interface may or may not allow specification of the rate at which to send packets out; this depends on whether

the underlying driver allows this property to be specified by the creator of the interface.

The Soekris boxes run Linux, allowing any software (memory permitting) to be compiled and run on the devices. The wireless cards used are supported by the Atheros MadWifi driver; modifications to the driver can be implemented, compiled for the boxes, and loaded at run-time.

3.2 Locations

The testing is performed in several different locations. Each location offers several advantages and disadvantages, and the location used in each particular test depends on the requirements of the test.

One of the locations is indoors. This location is the third floor lobby in the CSE building at UCSD. A diagram depicting the floor layout is shown as Figure 3.3. Experiments are conducted in the lobby area (the open space at the intersection of the two wings of the building). This location offers the advantage of easy access; hardware devices can be plugged in to AC outlets in the walls, allowing testing over long periods of time. However, this location offers several disadvantages. There are a large number of people who walk through the lobby area at all times of the day, acting as sources of interference. Furthermore, the lobby has several flat surfaces that cause reflections and observational irregularities. Finally, the lobby is small, allowing a maximum test distance of 30 ft.

Figure 3.3: Floor plan of the indoor testing area

Figure 3.4: Arial photo of Warren Mall on the UCSD campus. From Google Maps [4].

The second location is an outdoor location on the UCSD campus known as Warren Mall; this location shall hereafter be referred to as the mall. The mall is located adjacent to the UCSD CSE building. It provides a testing location that is roughly 80 ft in width, and roughly 450 ft in length. The mall is surrounded by buildings; one side of the mall the buildings are over four stories high, whereas on the other side of the mall the buildings are only two stories high. There are some trees that are located in the mall, however there is also a large grass patch in the middle of the mall that provides a relatively unobstructed testing area. The mall provides a larger testing area than the indoor location, with less interference from people and theoretically less reflection from flat surfaces at close range. An overhead picture of the mall is shown as Figure 3.4.

Figure 3.5: Arial photo of Warren Field on the UCSD campus. From Google Maps [4].

The third location is also an outdoor location, known as Warren Field (hereafter referred to as the field). The field provides a large, flat testing area; the width of the field is roughly 300 ft, and the length is well over 500 ft. There are no trees or other objects in the field, providing a relatively unobstructed testing area. The field is bordered by a chain link fence; two story buildings are located several feet from the outside of the fence along one side of the field. There are few structures within 50 feet of the field on any other side. An overhead image of the field is shown as Figure 3.5. When testing in the indoor location, devices may be plugged into the AC outlets in the walls. Outdoors, devices must be powered by batteries.

3.3 Software tools

The experiments require the use of software tools on the hardware devices; these tools are broadly categorized by their main functionality within the test. One set of tools generates packets and injects them into the network; another set allows the observation and recording of information at a particular node about packets in the network. More succinctly, most of the software tools either generate or capture packets. Other software helps to control various aspects of the testing, for example the Atheros driver. This section discusses the software tools that are running in the experiments.

3.3.1 Atheros driver

The wireless LAN cards in the Phocus Arrays and Soekris boxes are based on chipsets developed by Atheros [1]. The Linux driver for Atheros based chipsets is known as the Multiband Atheros Driver for Wireless Fidelity (MadWifi) [7]. The MadWifi driver is open source, and hence can be modified to perform as precisely as required.

Since the MadWifi driver is the driver for the WLAN card, the driver specifies some characteristics of how a packet should be transmitted over the air. When a packet is passed to the MadWifi driver to be transmitted, the driver creates a packet descriptor associated with that packet. The packet descriptor is a specification for how that packet is to be sent. This packet descriptor is passed to the lower

layers of the driver, which transmit the packet as specified in the packet descriptor. One of the fields of the packet descriptor is the transmit rate, corresponding to the 802.11 rate at which the packet should be transmitted at. Another field of the packet descriptor is the maximum number of times the packet should be retried. For data packets, the maximum number of times a packet should be retried is specified by a constant, which is set to 11 by default. For multicast and broadcast packets, the maximum number of times they should be transmitted is set to one.

3.3.2 Virtual Access Points

Several of the tests monitor packets at a node that is simultaneously sending and/or receiving data. Measuring the RSSI of a packet requires the observation of packets through a monitor interface. To associate with another node, a node needs a client interface. Since the Soekries boxes and Phocus Arrays only have one physical wireless card, multiple interfaces are created that use the same underlying physical device. This set-up can be achieved with the use of Virtual Access Points.

A Virtual Access Point (VAP) is a software device that is associated with a network interface. Multiple VAPs can be associated with a single network interface. VAPs act as an interface between the operating system and the MadWifi driver. VAPs allow the specification of how packets sent by the MadWifi driver should be transmitted, including (but not limited to) the source IP address to be use, the destination host, and the transmit power of the physical packet. VAP

configurations are modified with *iwconfig*.

Packaged with the MadWifi driver is a tool that allows the creation and configuration of VAPs. This tool is called wlanconfig. At the time of creation, a VAP is specified to be in one of several modes. wlanconfig can be used to create a VAP, specify which network interface the VAP should be bound to, and the mode that the VAP should be in. Three important modes are *ap*, *sta* and *monitor*. In *ap* mode, a device acts as an access point. Other 802.11 client devices may attempt to associate with it In *sta* mode, a device acts as a client station. In this mode, a device may associate with devices that are access points In *monitor* mode, the device passively listens and attempts to decode every packet it can; it neither sends out beacon packets, nor associates with other nodes.

3.3.3 Packet generation

The experiments require controlled mechanisms for sending data packets from one node to another. A controlled mechanism implies that certain factors must be fixed, such as the size of the data packets and the number of packets per second. Using a transport layer protocol such as TCP is inappropriate for such controlled experiments, because TCP varies the number of packets that are transmitted per second according to the perceived link quality. For example, TCP includes features such as slow start, and multiplicative increase and linear decrease of the data rate; these features do not allow consistent measurements between experiments that are

trying to objectively measure the link characteristics. A protocol such as UDP is therefore more appropriate, since UDP allows the transmission of data with fixed packet sizes and a constant number of packets per second.

Ping

Ping is a simple utility that tests end to end connection between hosts. A source node sends an ICMP echo request packet to another node in the network (the particular destination node is specified with its IP address). Upon reception, the destination node will send an ICMP response back to the source node. The source node records receipt of the response, and the round-trip time (RTT) taken for the response to return. If the destination node does not receive the original echo request, or the source node does not hear the ICMP response, a timeout for that particular ping packet will occur. During a ping session, the source node will display the RTT for any packets it receives. At the end of a ping session, ping will display the loss rate (calculated from the number of responses it gets and the number of packets it sends out), as well as the average RTT.

Iperf

Ping only generates one ICMP packet per second. Some tests require a node to transmit a larger amount of data to other nodes in the network. Iperf [6] is a commonly used software tool that generates data traffic. Iperf is often used for testing bandwidth between nodes in a network. Iperf allows the generation of TCP

and UDP data streams. Iperf is run in one of two modes: a client mode, and a server mode. While in client mode, iperf establishes a connection with an iperf server. The client then generates data packets, and sends them via either a TCP connection or a UDP stream. With UDP streams, iperf allows the specification of the size of a datagram and the bandwidth to try and send. Iperf is also used to measure the bandwidth achieved over a link. In server mode, iperf waits for incoming connections from a client. Upon receiving a session initiation request, an iperf server starts a timer for that session. It measures (among other factors) the total amount of data transferred from the client for the duration of the session. When the client stops sending, it notifies the server that the session should be closed. The server then prints out a report about the session, including the total number of bytes transferred and the average bandwidth achieved during the session. Iperf does not give an indication of the physical or MAC layer characteristics of the packets received; it is a transport layer measurement tool.

Ttcp

Ttcp is a tool that is similar to iperf. However, ttcp is much more reliable over low quality links. Like iperf, ttcp operates in either client or server mode; the client sends data to the server. Unlike iperf, the session open and close behaviour of ttcp is much more reliable, even over low quality links; this fact makes ttcp much more useful in tests where the link quality is low. Ttcp allows a stream of UDP packets to be generated; the number and size of the packets generated can be

specified. Like iperf, ttcp runs in either client or server mode. The server runs at the recipient node (the Soekris box), while the client runs at the transmitting node (the PA). The client sends the UDP stream to the server; when the client closes the session, the server reports statistics about the session (including the total number of bytes transferred).

RUDE

Tttcp requires a server to be running on a destination node; the ttcp client runs on the source node and sends data to the server. However, ttcp does not allow the specification of the size and number of packets to be sent. Real-time UDP Data Emitter (RUDE) [9] is a lightweight software tool that generates packets. In contrast with ttcp, RUDE allows the specification of the number and size of packets to be sent. This provides a controlled source of data into the network from a source node to a destination node.

RUDE functions as follows. The executable runs with a command line argument, which is the name of a script file. Lines in the script file are structured as follows.

starttime flowid ON srcport destip:destport CONSTANT pktrate pktsize

or

stoptime flowid OFF

These fields can be explained as follows.

starttime - the time in microseconds to start transmitting

stoptime - the time in microseconds to stop transmitting

flowid - a unique integer identifier to refer to a particular transmission flow

srcport - the port on the source host to send the packets out of

destip:destport - the IP address of the destination host, and the port on that host to send to

pktrate - the number of packets to generate per second

pktsize - the size of the generated packets

3.3.4 Packet Measurement

Software running on a source node generates packets and injects them into the network. Conversely, the tests require a mechanism for observing the packets that are received at the destination node. Characteristics of the received packets that need to be observed include (but are not limited to) the number of packets received per second, and the RSSI of the packets. If the number of packets received per second is much lower than the number of packets transmitted, it may be an indication of poor link quality. The RSSI of received packets is compared in different test scenarios to compare link qualities, beam patterns, and other properties of the transmission.

Iwconfig

The Linux tool iwconfig displays the current state of a wireless interface. Included in the output of iwconfig is a signal strength measurement; this value is the

signal strength of a recent packet received at that interface. It is unclear whether or not iwconfig updates this value with every single packet received. Iwconfig can be run in a loop, and the output stored and parsed to produce a set of such signal strength measurements. This sample of measurements can be used as an indication of the signal strength of packets received at that interface. There are several drawbacks to this method of measurement. Firstly, the average taken from a sample of measurements can often be quite different from the actual average, since the RSSI of different packets between two nodes may vary significantly. Secondly, the signal strength value as reported by iwconfig is the RSSI of some recent packet received at that interface. Packets received from different sources are not differentiated between when this value is updated; therefore using iwconfig does not allow the differentiation of packets from multiple sources. While no differentiation is needed when there is only one source node, when multiple nodes send data to the same destination iwconfig is often inadequate.

Tcpdump

Tcpdump is a software tool that analyzes the packets received at a destination node. Tcpdump is a much more sophisticated measurement tool than iwconfig. Iwconfig only prints out the signal level of some packet received at an interface at some recent point in time; tcpdump actually parses every piece of information included in the packet headers. Tcpdump analyzes all of the packets that arrive through a network interface or VAP. It provides many command line options,

including options to filter the packets analyzed by any one of a number of fields, including the transport layer protocol used to transmit the packet and the IP address of the source host. The packets can then be printed out to the screen (with varying levels of detail), or saved to a file for post-processing. If tcpdump dumps the packets arriving at a *monitor* interface, the 802.11 header (which includes the RSSI of the packet) can also be displayed (or saved).

4

Experiments

This chapter presents the experiments that are performed. The experiments explore the capabilities of phased array antenna; more specifically, the experiments aim to determine whether phased array antennas can be used to achieve spatial reuse. To do this, the beam forming abilities of such antennas must first be understood. The experiments fall into three main categories: static tests, pattern rotation tests, and physical rotation tests. Static tests are performed to compare aspects of the beam pattern achieved by a Phocus Array from different angles. In static tests, all nodes in the network remain physically stationary; they do not move, rotate, or change state during the test. Pattern rotation tests are performed to determine the differences between the pattern at different angles, with a coarse granularity. In pattern rotation tests, the pattern on the Phocus Array is changed during the test. The Phocus Array is initially configured with the first directional

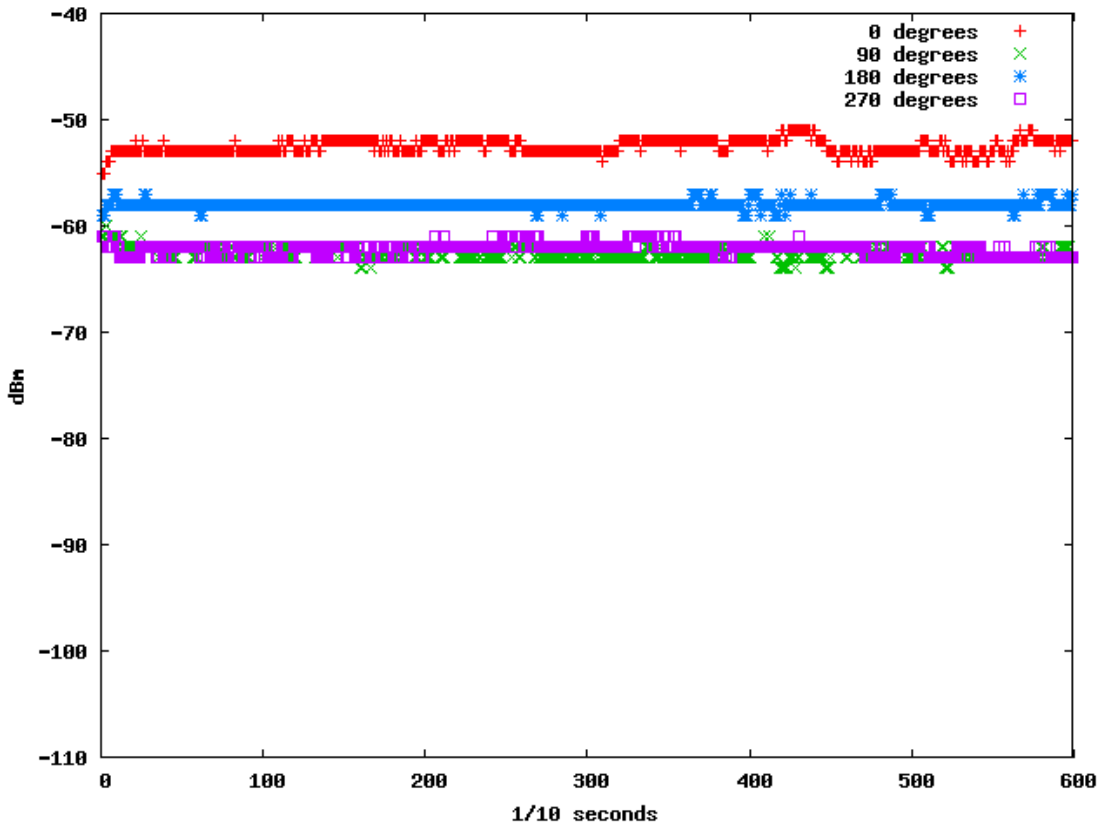


Figure 4.1: Beacon strengths from the Phocus Array at an associated client 45 ft away, indoors

pattern it has been pre-loaded with; as the test progresses, the Phocus Array is reconfigured to use subsequent patterns (hence the pattern is 'rotated' during the test). Physical tests are performed to accurately (and with fine granularity) measure the Phocus Array's ability to emit a single beam pattern. In physical rotation tests, the pattern used by the Phocus Array remains constant during the test; instead, the Phocus Array is physically rotated through 360° during the test.

4.1 Static

This section describes the static tests that are performed, which attempt to validate the effectiveness of the beam emitted form the Phocus Array. Tests are categorized as static when all nodes in the network are physically stationary and immobile; nodes do not move or rotate. In addition to the lack of physical movement, nodes do not change their configuration either. Client nodes remain either associated or unassociated for the duration of the test; the Phocus Arrays do not change their configured beam pattern. The Soekris boxes will be referred to as Sx, where x is an integer used to uniquely identify and differe‘ntiate between the boxes in the test.

4.1.1 Beacon packets

If the Phocus Arrays are effective beam-forming antennas, the signal strength of packets being transmitted out of the antenna should vary according to the angle at which a recipient node lies to it. Static tests aim to verify that the signal strength of the radio wave emitted by the Phocus Array at various angles conforms to the beam pattern that the Phocus Array has been configured with. To test this, client nodes can be placed at various angles to the Phocus Array, and the signal strength of beacon packets from the Phocus Array at those angles measured. These tests involve setting the Phocus Array up as an AP, and Soekris boxes as clients associated with the Phocus Array. In AP mode, the Phocus Array

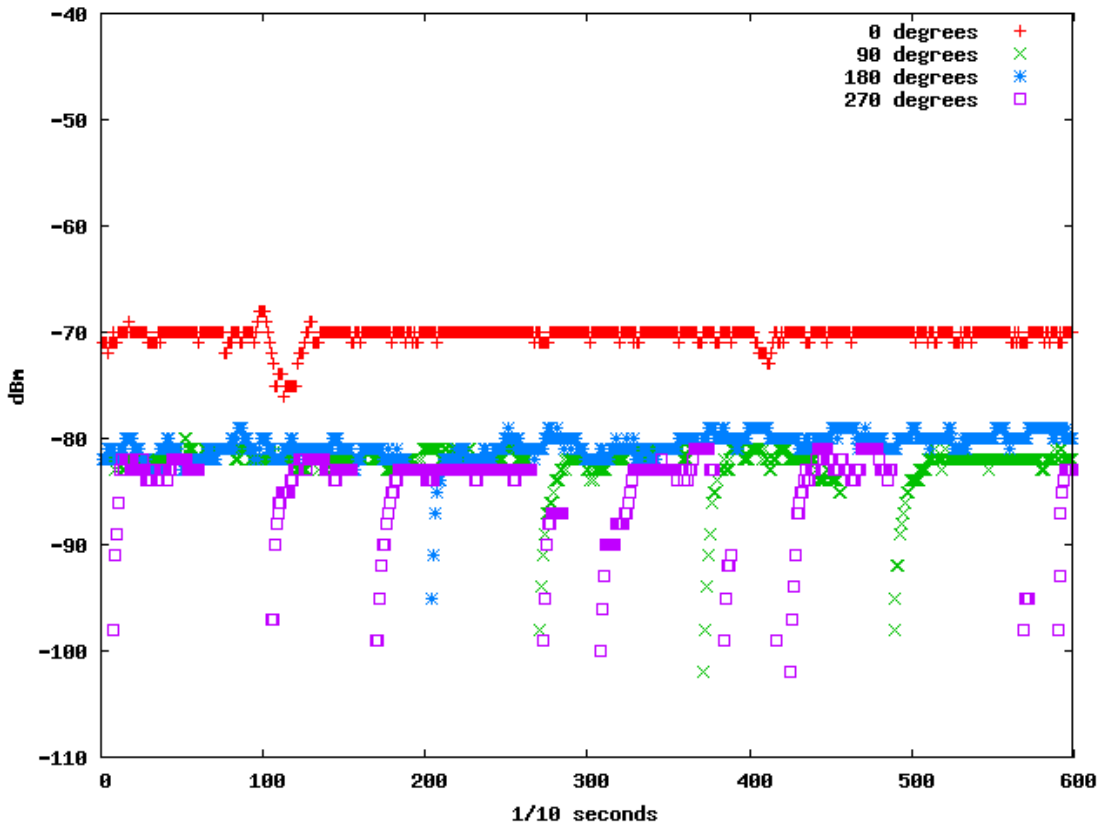


Figure 4.2: Beacon strengths from the Phocus Array at an associated client 45 ft away in the mall

transmits beacons; the RSSI of the beacon packets is observed at the Soekris box via the monitor interface, and recorded.The signal strength at various angles can be measured and compared, using diagrams of the beam pattern as models for what the relative signal strengths should be.

This test aims to verify that the radio wave emitted by the Phocus Array at certain angles conforms to the beam pattern diagram. The Phocus Array is loaded with the co-phase (Figure 2.2) pattern, and placed at one end of the indoor testing

area. A Soekris box S1 is configured as a client, associated with the Phocus Array, and placed at the other end of the testing area (45 ft away). The Phocus Array is initially oriented facing S1, i.e. with the S1 at 0° to the Phocus Array. On S1, iwconfig is run in a loop every 100 ms for 60 seconds, and the signal strength reported is recorded. The test is then repeated, after the Phocus Array has been physically rotated 90°. The test is run with S1 at 0°, 90°, 180° and 270° to the Phocus Array. The results for all four tests are plotted in one graph, shown as Figure 4.1. As expected, the signal strength is relatively constant at each angle. Furthermore, the signal strength 0° is stronger than at any other angle; the RSSI at 90° and 270° are roughly equivalent. However, the RSSI at 180° is stronger than at 90° or 270°; from the diagram of the beam pattern, the RSSI at 180° should be the weakest. Furthermore, the RSSI at all angles does not differ by a large number; the diagram seems to indicate a much larger difference between the signal strengths at those angles. This test is performed indoors, with many hard surfaces at close range that might cause reflection; such reflection may explain why the relative signal strengths reported differ from those that are expected.

In an attempt to in an attempt to mitigate the effects of reflection from large flat surfaces at close range to the test, further tests take place in an outdoor environment. Other aspects of the previous test need not be changed, apart from conducting the test outdoors. The Phocus Array is placed at one end of the mall testing area. S1 is placed at a distance of 60 m away, at 0° to the Phocus Array.

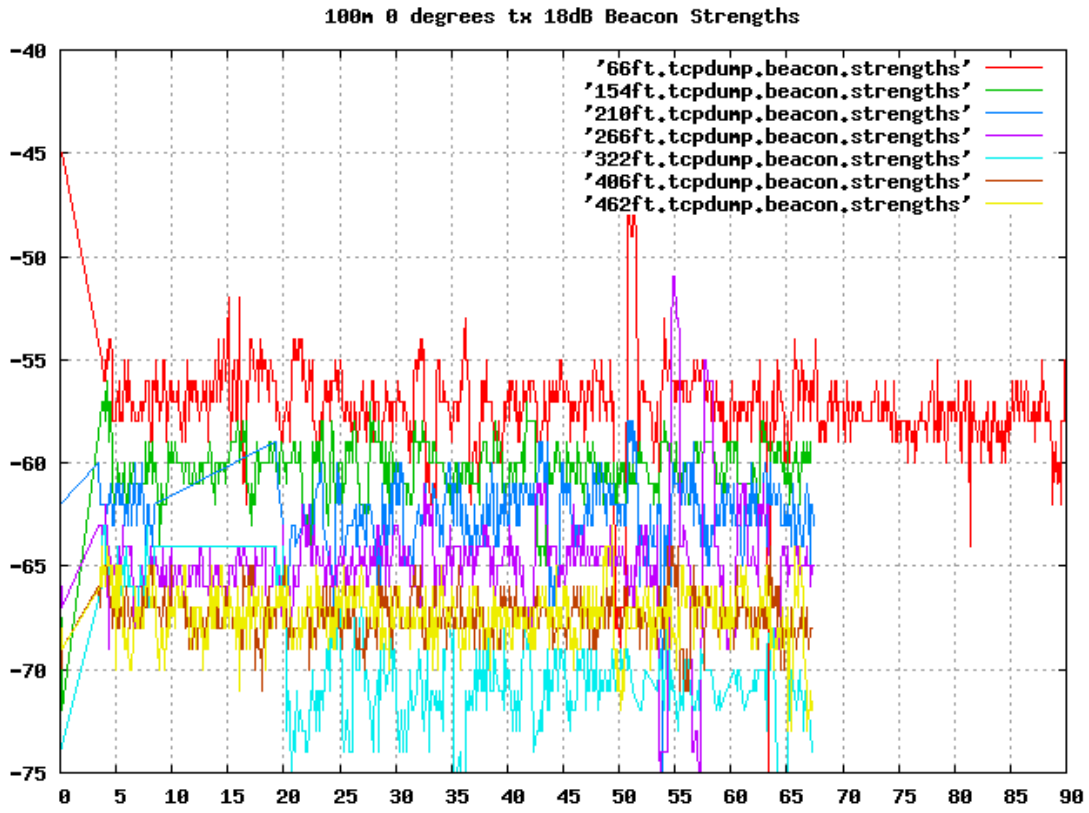


Figure 4.3: Beacon strengths from the Phocus Array during a ping at an associated client various distances away in the mall

The signal strength as reported by iwconfig every 100 ms is recorded. The test is repeated after rotating the Phocus Array such that S1 is 0°, 90°, 180° and 270° to the Phocus Array. The results are shown as Figure 4.2. Once again, as expected, the RSSI at 0° is stronger than at any other angle. However, the graph shows that the RSSI for all other angles is roughly equivalent; this observation again contradicts the diagram of the beam pattern. Furthermore, the signal strengths do not seem to be constant at other angles; for example, at 90° and 270° there is

wide (and frequent) variation between -80 dB and -100 dB in the observed RSSI. These results indicate that reflections in the indoor environment are not the main cause of differences between the expected and observed beam pattern.

4.1.2 Data

The emitted beam pattern should not only be apparent when monitoring beacon packets, but should also be observable when data packets are being transmitted. If this is so, there must exist two different angles on the Phocus Array such that the emitted beam at one angle is relatively strong, and the beam at the other angle is relatively weak. Tests can be performed similar to the tests during which no data is transmitted, in which one of the nodes transmits data to another node in the network. Tests in which data is not transmitted do not show the expected differences in signal strength at various angles to the Phocus Array; the signal strength at a client node of beacons from the Phocus Array at 90° is roughly the same as at 180° and 270°. Tests with data, however, may result in observable differences at these angles.

Infrequent data

When sending data form oe node to another, the simplest test that can be performed is one in which one data packet per second is transmitted from a source to a destination node. A software tool that generates exactly one data packet per second is ping. The signal strength of the ICMP response can be measured at the

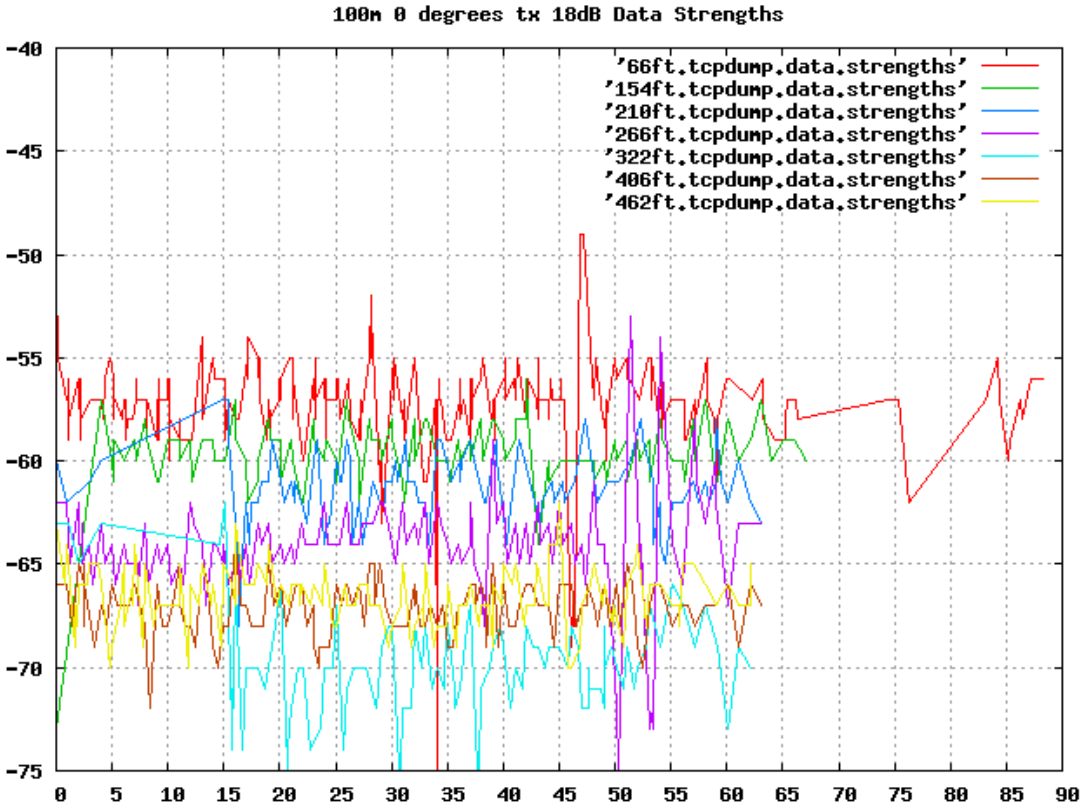


Figure 4.4: Data packet strengths from the Phocus Array during a ping at an associated client various distances away in the mall

sending node, as well as the RTT (both may be good measures of the link quality).

In the first iteration of the test, the Phocus Array is placed in the middle of the mall testing area. S1 and S2 are set up with two interfaces: a client interface that is associated with the Phocus Array, and a monitor interface. S1 is placed 66 ft away at 0° to the Phocus Array, and associates with it. S2 is placed 154 ft away, at 180° to the Phocus Array, and associates with it. Both S1 and S2 ping the Phocus Array for the duration of the test. Using tcpdump on both interfaces,

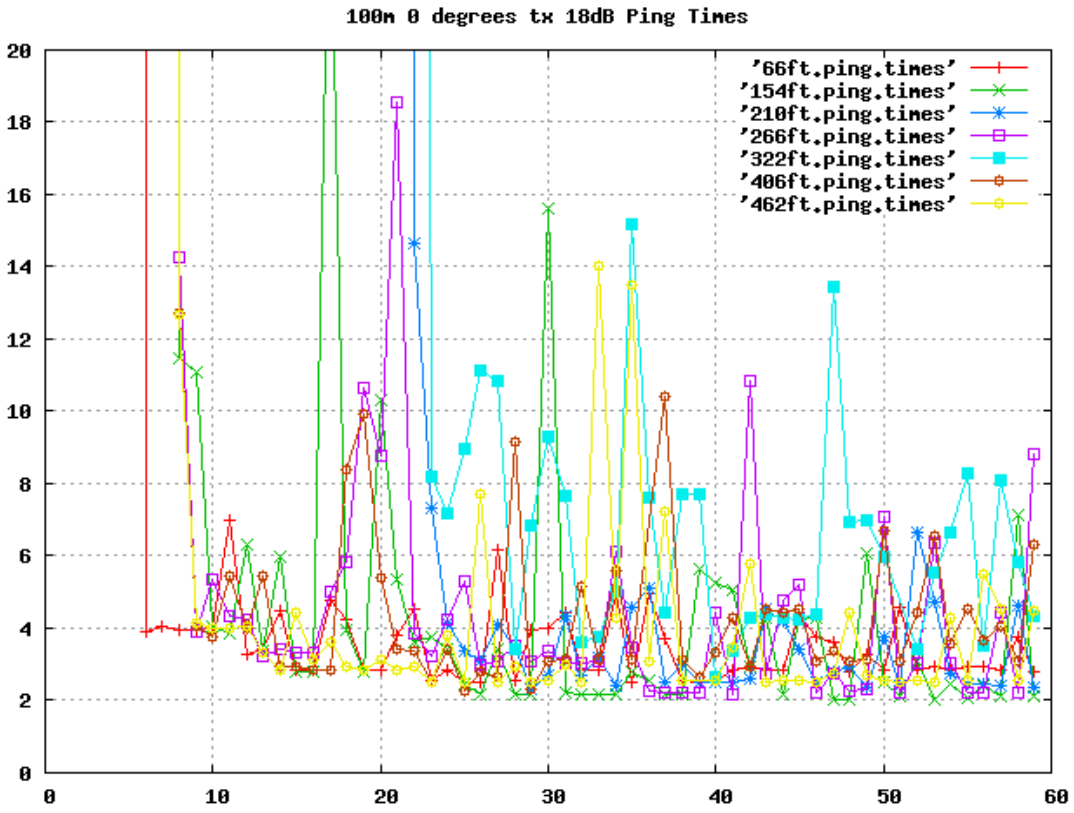


Figure 4.5: RTTs from the Phocus Array during a ping at an associated client various distances away in the mall

the strengths of beacon and data packets are measured at both S1 and S2, as well as the RTT of the pings. The test is repeated with S2 at distances of 210 ft, 266 ft, 322 ft, 406 ft, and 462 ft. Figure 4.3 shows the RSSI of the beacon packets from these tests, while Figures 4.4 and 4.5 show the RSSI of the data (ICMP) packets and the RTTs, respectively. As the distance between the Phocus Array and S2 increases, the RSSIs of both the beacon and the data packets generally drop. However, there is no direct correlation between the distance and the RSSI;

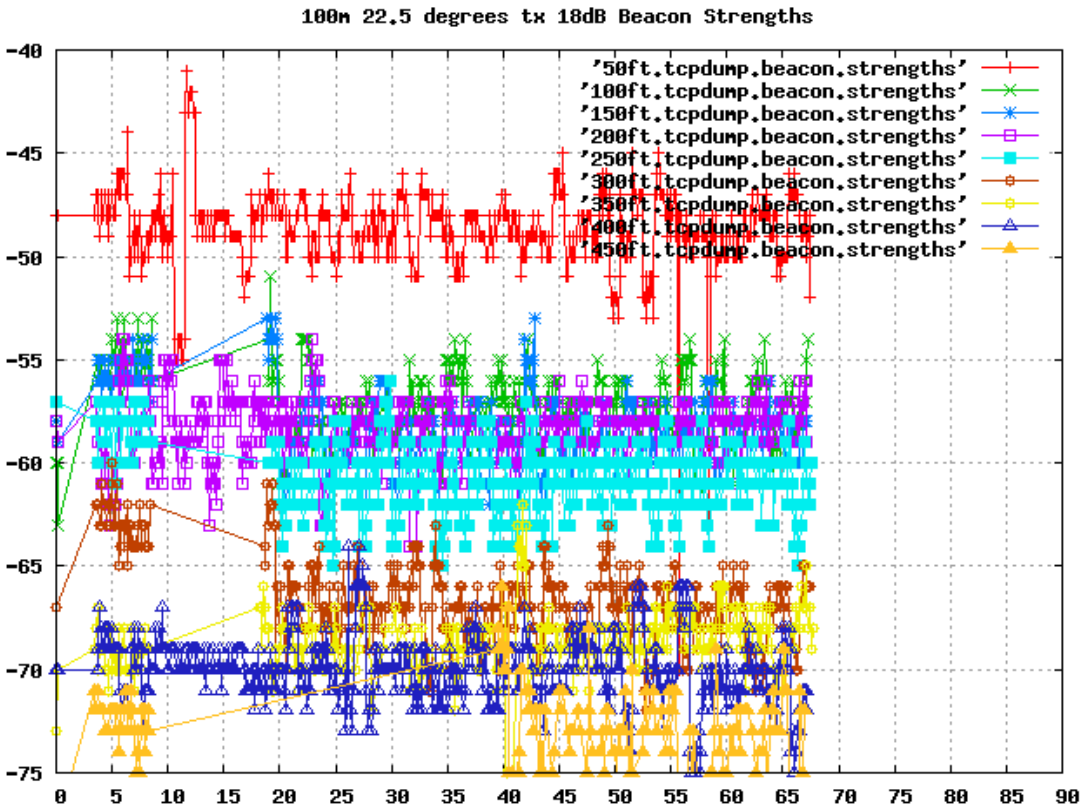


Figure 4.6: Beacon strengths from the Phocus Array during a ping at an associated client various distances away in the mall; Phocus Array pattern is 22.5°

for example, the RSSI for (beacons and data) packets at 322 ft is lower than at 406 ft and 462 ft. Even at 462 ft, connectivity is maintained, and S2 can still ping the Phocus Array (and vice versa). The RTT graph is inconclusive: the RTT for all distances is similar, with large variations., leading to the conclusion that RTT is not a good metric to use for measurement in these tests. Thus the following possible explanations can be drawn from these graphs. The transmit power at the Soekris boxes and/or the Phocus Array may be too high (yielding connectivity at

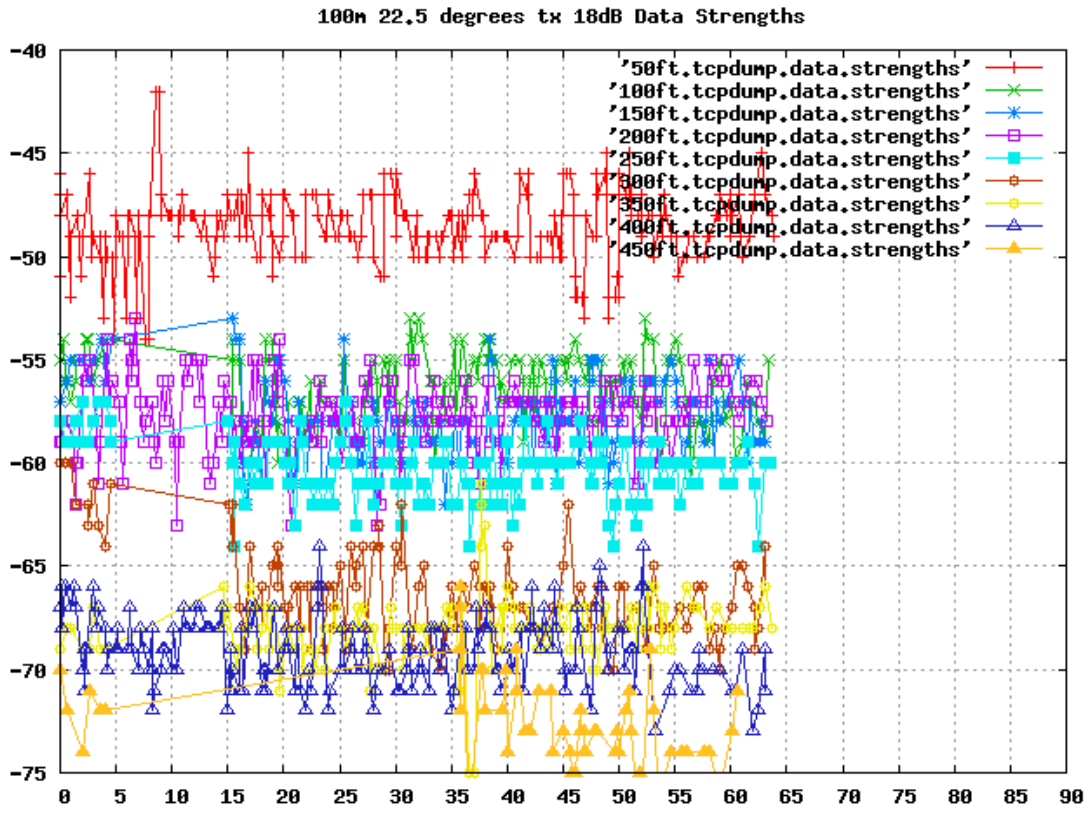


Figure 4.7: Data packet strengths from the Phocus Array during a ping at an associated client various distances away in the mall; Phocus Array pattern is 22.5°

long distances, even at the supposed weak angle). The beam strength may not be as weak as assumed at the weak angle; alternatively, there may be an angle other than 180° that yields a loss in connectivity at a distance in the range of these tests.

A metric better than RTT that can be used to measure the quality of a link is RSSI. Further tests attempt to determine if the angle at which a client lies with respect to the Phocus Array makes a difference to the RSSI of packets received at the client. A small change that can be made to the previous test to check if the

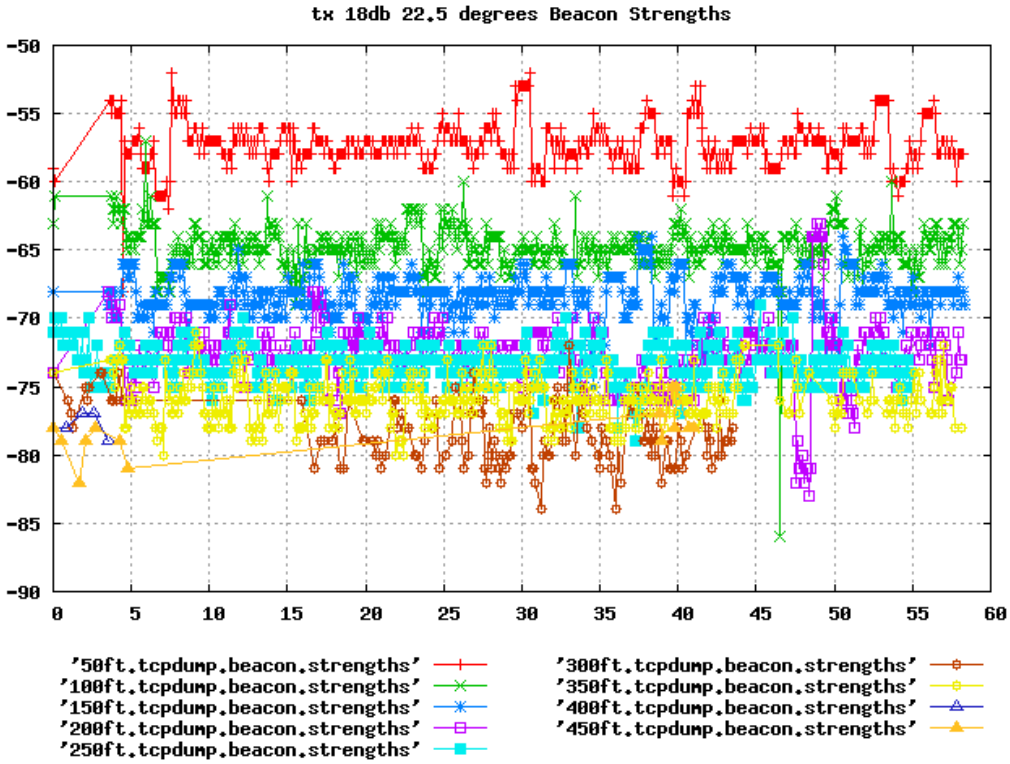


Figure 4.8: Beacon strengths from the Phocus Array during a ping at an associated client various distances away in the mall; Phocus Array pattern is 22.5°, transmit power is damped to 10 dB

angle that the Soekris box is to the Phocus Array makes a large difference is to change that angle. To achieve this, the Phocus Array can be the same physical orientation, but the pattern changed before the test. This version of the test is identical to the previous version, other than the fact that the pattern on the Phocus Array is rotated 22.5°; thus S1 and S2 now lie at 337.5° and 157.5° to the main lobe of the Phocus Array, respectively. Figures 4.6 and 4.7 show the RSSI of beacon and data packets during these tests. These graphs are similar to the previous

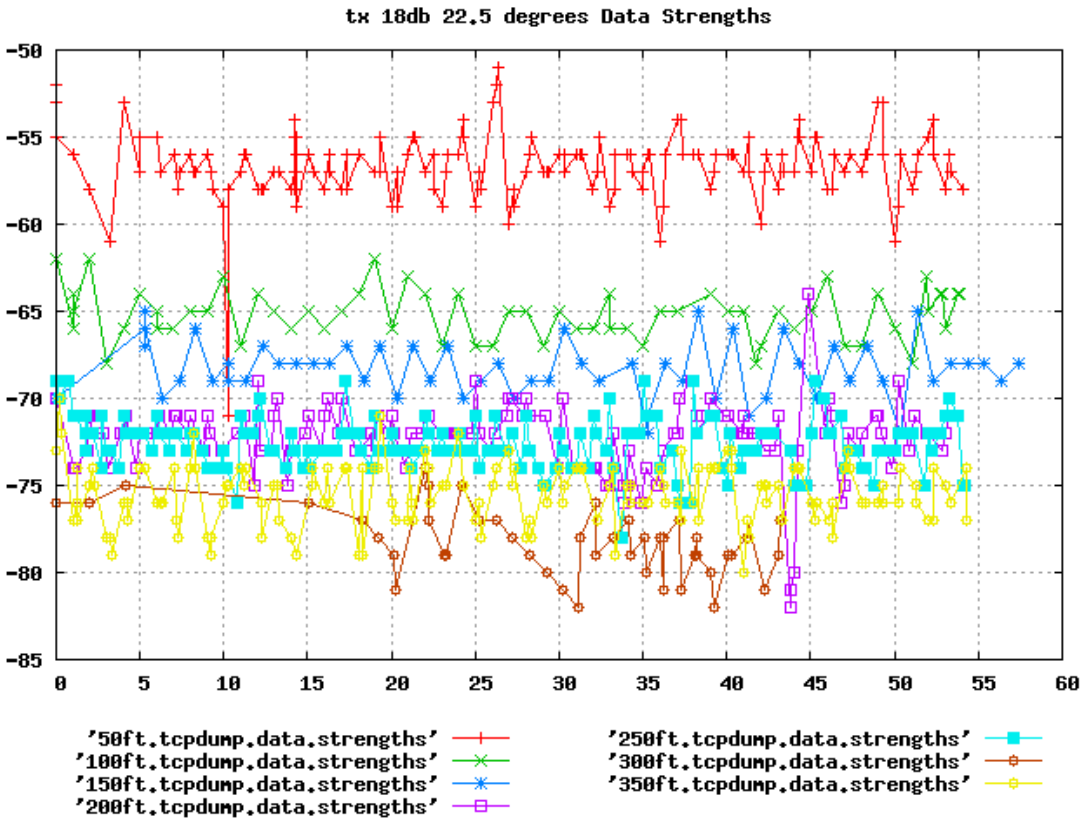


Figure 4.9: Data packet strengths from the Phocus Array during a ping at an associated client various distances away in the mall; Phocus Array pattern is 22.5°, transmit power is damped to 10 dB

graphs (where the pattern on the Phocus Array is set to 0°), indicating that there is not a large correlation between the variability seen at different distances and the particular angle at which the Soekris box lies to the Phocus Array.

Further tests attempt to break connectivity at certain angles. If connectivity can be broken at one angle, yet maintained at another angle, the testing will have effectively shown a difference in the beam pattern at those two angles. One change

that can be made to the previous test to to see if connectivity can be broken at long distances is to damp the transmit power of the Phocus Array. The default transmit power on the Phocus Array is 18 dB. This test is identical to the previous test (with the Phocus Array pattern set to 22.5°), with the transmit power at the Phocus Array damped to 10 dB. Figures 4.8 and 4.9 show the RSSI of beacon and data packets in this test. Once again, the graphs look similar to previous results, indicating that the transmit power of the Phocus Array does not account for the similarities (and large range) of RSSI at the client at different (long) distances.

Frequent data

The RSSI between a particular source and destination has been shown to exhibit a large range of variation. One possible improvement to the testing method is to increase the number of samples measured. Previous tests indicate that a more systematic method must be used to find a pair of angles to the Phocus Array with the greatest difference in signal strength. These next set of tests attempt to achieve a more systematic method of testing, in which more data samples are available for analysis. The tests use iperf to generate more than one data packet per second.

The tests aim to show a pair of angles such that there is connectivity at one angle, and no connectivity at another. This behavior will show that differences exist in the antenna's beam pattern. To observe packets with the maximum RSSI, a Soekris box S1 is placed at 0° to the Phocus Array (the center of the main lobe

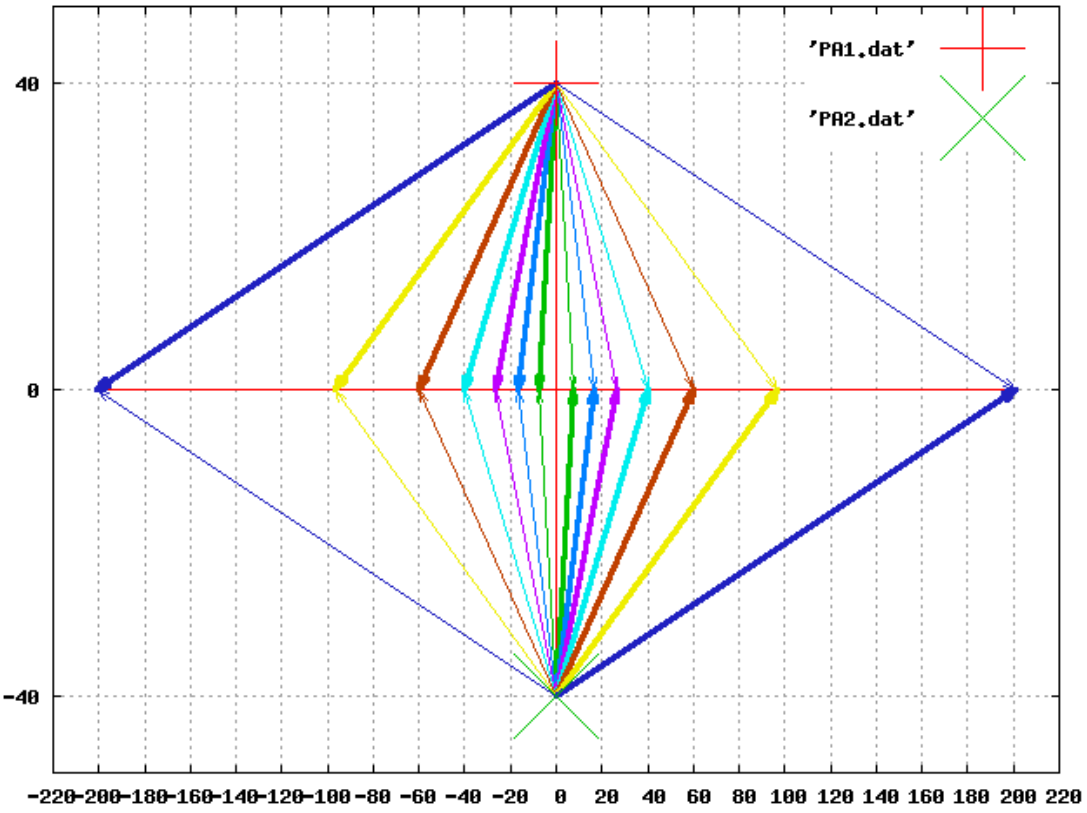


Figure 4.10: Diagram showing possible angles that can be tested within the mall

of the beam pattern). Figure 4.10 is a diagram showing possible placement of S2. The dimensions of the diagram correspond to the physical dimensions of the mall testing area. Consider the lower half of the diagram. Assume a Phocus Array PA1 is placed at the middle of the lower boundary of the mall, and a Soekris box S1 is placed at the middle of the right boundary of the mall. Every vertical line occurs at an offset of 22.5° from the neighboring vertical lines. Every point where a vertical lines extending from PA1 meets the central red line is a possible placement position for S2. The goal is to place S2 along the red line at one of

Table 4.1: Possible angles and distances of a Soekris box to a Phocus Array in the mall

Angle	112.5°	135°	157.5°	202.5°	225°	247.5
Distance (ft)	72	104.5	205	205	104.5	72

those intersection points such that packets received from PA1 at S1 and S2 have a maximal difference in RSSI. Obviously, S2 must be placed in the half of the mall that is away from S1 (where the main lobe will be pointing); furthermore, given that radio waves obey the inverse-square law, the greater the distance that S2 is from PA1 the better. Thus the best possible candidates for placement of S2 is the leftmost intersection three points on the red line; these points correspond to angles of 112.5°, 135° and 157.5° to the Phocus Array. Of course, instead being at the bottom of the mall, the Phocus Array could be at the top. Considering the top half of the graph, the possible positions for S2 are at 202.5°, 225° and 247.5°. The possible angles and distances from the Phocus Array are summarized in Table 4.1.

To find a pair of angles as described, the first step is to find an angle and distance such that there is little or no connectivity from a node at that angle to the Phocus Array. The following test runs with the calculated distances for each angle in Table 4.1. The Phocus Array is placed at one end of the mall. S1 is placed at a distance away from the Phocus Array, with the distance determined by Table 4.1. S1 is at 0° physically to the Phocus Array; the Phocus Array is loaded with the beam pattern that makes S1 lie at the the angle in the beam pattern that is

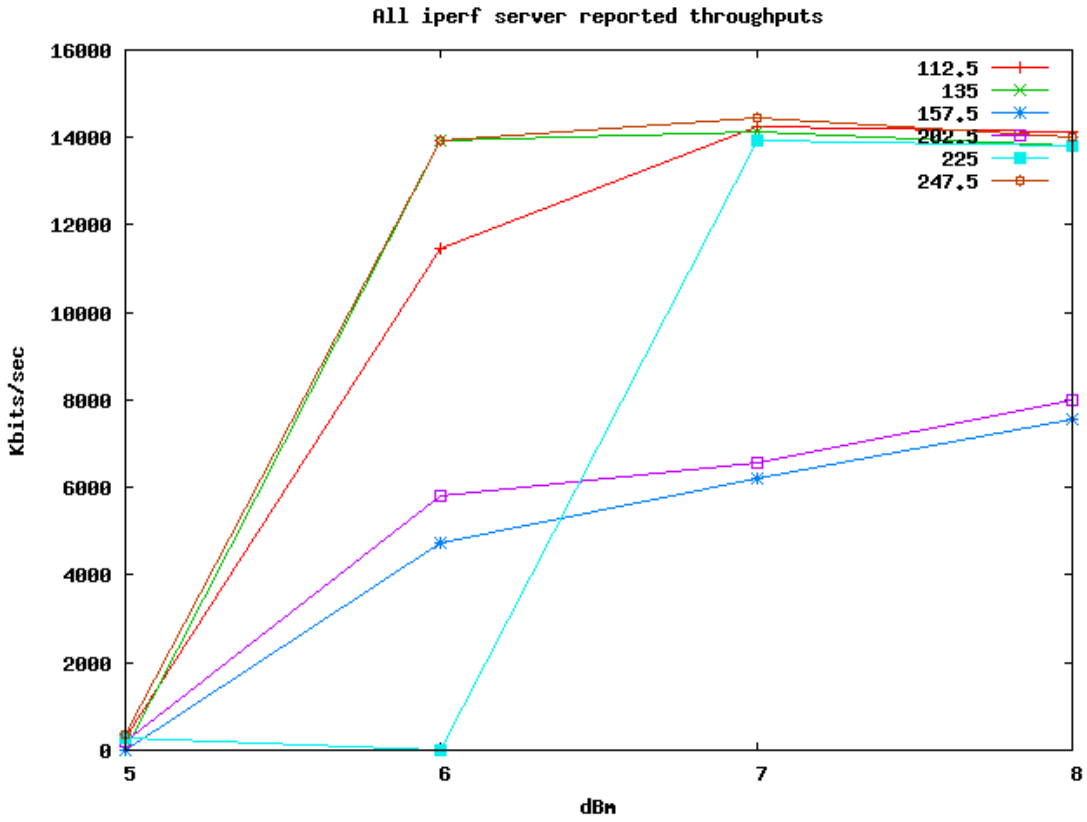


Figure 4.11: Server reported average throughputs for various angle/distance, power combinations

specified by the test. For example, in one version of the test S1 is placed 205 ft away from the Phocus Array, with the Phocus Array physically facing the S1. The Phocus Array is loaded with the pattern that makes S1 to be at 157.5° in the beam pattern. An iperf server is run on the Phocus Array, while an iperf client on the Soekris box connects to the server. For each angle and distance in the table, the iperf client sends UDP data to the server as fast as possible for 60 seconds. This is repeated with the Phocus Array transmitting at 5 dB, 6 dB, 7 dB and 8 dB.

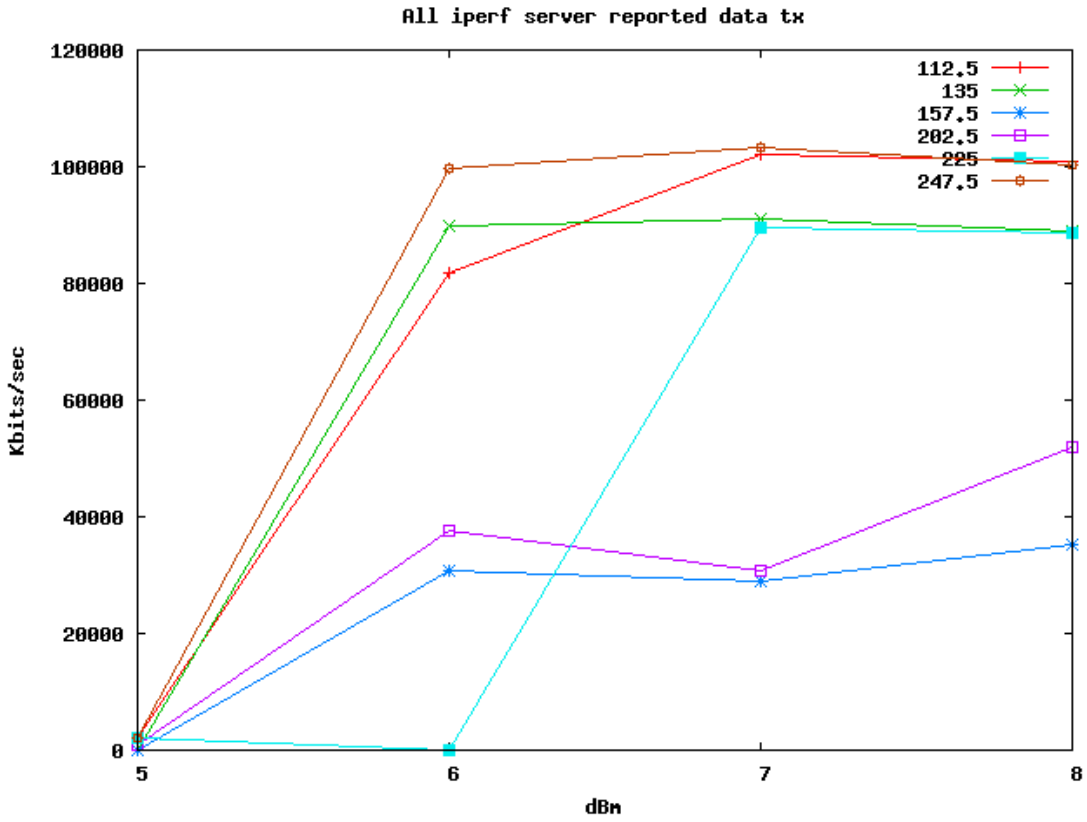


Figure 4.12: Server reported total bytes transferred for various angle/distance, power combinations

Figures 4.11 and 4.12 show the average throughput and amount of data transferred for each of the tests (as reported by the iperf server). As expected, most of the graphs appear in pairs: the angles at which the distances are the same yield lines on the graph that are similar (except for 225°). The graphs increase very slightly (but are generally relatively flat) for power values of 6 dB to 8 dB. At 5 dB the throughput and data transferred is roughly 0. These graphs show that there does exist an angle, distance and transmit power combination such that connectivity is

lost for a client (in fact all angles and distances lose connection when the transmit power is set to 5 dB). The next step is show that there exists a distance and power combination such that at 0° there is connectivity and at another angle there is no connectivity.

Table 4.2: Data transmitted using iperf at different angles, varying transmit power, 802.11 rate and distance

dB	802.11 rate	distance(ft)	pattern	pkts received
4	auto	0	0	0
4	11	0	0	0
5	auto	200	0	> 0
5	auto	200	135	> 0
5	11	200	0	4.9MB
5	11	200	135	2.42MB
5	12	200	135	>0
5	18	200	135	0
5	18	200	0	0
5	18	100	0	>0
5	18	100	135	0
5	18	60	135	>0
5	18	40	135	>0
5	18	20	135	>0
6	auto	200	0	>>0
6	auto	200	135	>>0

Subsequent tests attempt to find a 'weak' angle such that a lack of connectivity can be induced under certain conditions; in addition, a finer granularity of control over the link's connectivity should be achieved than has been thus far. From 4.11, the most dramatic drop-off from connectivity at 6 dB to no connectivity at 5 dB is at 104.5 ft and an angle of 135°. Thus 135° seems the most likely candidate

to be the 'weak' angle. Since damping the transmit power seems to have such a dramatic effect, the next step is to vary a factor that elicits more fine-grained control over the connectivity of the link. The most most obvious candidates are the 802.11 rate at which data packets are transmitted at, and the distance at which the client lies from the Phocus Array. The next test changes these variables in an attempt to find a transmit power, rate and distance such that at 0° there is some connectivity, and at 135° there is none. To vary the 802.11 rate, the Atheros driver on the Phocus Array is modified and compiled. One version of the driver is compiled for each rate to be transmitted at (since each version of the driver fixed the rate to one particular rate). The transmit power is considered to be the coarse-grained control, the 802.11 rate the finer-grained control, and the distance the most fine-grained control. The Phocus Array is placed at one end of the mall. S1 is placed at various distances to the Phocus Array. The transmit power on the Phocus Array is varied, as is the version of the driver that is used at runtime. An iperf server runs on the Phocus Array; the iperf client runs on S1 and sends UDP data as fast as possible to the server for 60 seconds. Table 4.2 shows the results of this test. The rightmost column displays the amount of data transmitted during the test. A measurement of >0 means that the server reported a connection being initiated, but not a connection being closed (and thus could not calculate the total amount of data transferred, average bandwidth, etc.); a measurement of >>0 means that the amount of data transferred is large. The best candidate for

an appropriate combination of transmit power, 802.11 rate and distance is 5 dB, 18 Mbps, and 100 ft. In this combination, there is no data transferred at 135°, but some data transferred at 0°. Unfortunately, the iperf server does not provide an accurate reading for how much data. If the iperf server does not receive an explicit disconnect message from the client, it will not close the session. This behavior is more prevalent in low quality links (such as those which this test tries to induce); thus iperf is an inadequate tool for this test.

Further tests mandate that a tool that behaves more consistently over low quality links be used. One such tool is ttcp; ttcp works better than iperf over low quality links, because the establishment and disestablishment of a session between a client and a server is very simple, resulting in fewer errors. Therefore the previous test can be repeated using ttcp instead of iperf. An angle other than 135° that is a weak angle as per the beam pattern diagram is 157°; further testing uses this angle as the weak ange (either angle is appropriate for use as the weak angle). The transmit power and 802.11 rate at the Phocus Array is varied, in an attempt to discover a combination at which there is connectivity (data transferred) when the pattern is set to 0° but no connectivity when the pattern is set to 157°. Since the previous test indicated that changing the distance did not seem to have a significant effect on the results, in this test the distance is fixed at 150 ft. Table 4.4 shows the results of this test. This test is also inconclusive, for several reasons. Consider iteration zero and one of the test at 9 dB, 18Mbps, with pattern 157°:

these two iterations show great variation between tests. Also consider the tests at 9 dB, 24Mbps and pattern 0°. Iteration zero resulted in a much lower amount of data transferred than iteration one; furthermore, both iterations when the pattern is 157° yield more data transferred than iteration zero when the 0° pattern.

Table 4.3: Data transmitted using RUDE at different angles, varying transmit power and 802.11 rate

dB	rate	pattern	iter	pkts	-	dB	rate	pattern	iter	pkts
8	24	0	0	22	-	-	-	-	-	-
8	36	0	0	0	-	-	-	-	-	-
8	36	0	1	0	-	-	-	-	-	-
8	48	0	0	0	-	-	-	-	-	-
9	24	0	0	7226	-	-	-	-	-	-
9	36	0	0	2485	-	-	-	-	-	-
9	48	0	0	7	-	9	48	157	0	26
9	48	0	1	104	-	9	48	157	1	58
9	48	0	2	15	-	9	48	157	2	21
9	48	0	3	27	-	-	-	-	-	-
9	48	0	4	242	-	-	-	-	-	-
9	48	0	5	143	-	-	-	-	-	-
9	54	0	0	5	-	9	54	157	0	8
9	54	0	1	1	-	9	54	157	1	17
9	54	0	2	14	-	9	54	157	2	44
10	48	0	0	896	-	10	48	157	0	240
10	48	0	1	177	-	10	48	157	1	771
10	48	0	2	483	-	10	48	157	2	451
10	48	0	3	133	-	-	-	-	-	-
10	48	0	4	109	-	-	-	-	-	-
10	48	0	5	44	-	-	-	-	-	-
10	54	0	0	23	-	10	54	157	0	284
10	54	0	1	17	-	10	54	157	1	239
10	54	0	2	416	-	10	54	157	2	78
10	54	0	3	25	-	-	-	-	-	-
11	54	0	0	193	-	11	54	157	0	864
11	54	0	1	504	-	11	54	157	1	415
11	54	0	2	205	-	11	54	157	2	949

The large amount of variability in these results shows that using ttcp may not be adequate. Using ttcp allows control over the size of data packets, but not the number of packets that are sent per second.

Table 4.4: Data transmitted using ttcp at different angles, varying transmit power and 802.11 rate

dB	802.11 rate	pattern	iteration	pkts received
18	a	0	0	8647
18	5	0	0	2350
12	5	0	0	712
8	5	0	0	724
7	5	0	0	401
6	5	0	0	0
7	5	0	1	459
7	5	157	0	526
7	5	157	1	258
8	5	157	0	1213
8	5	157	1	798
18	12	0	0	23103
12	12	0	0	11955
8	12	0	0	159
10	12	0	0	11620
9	12	0	0	6244
9	12	157	0	5208
17	18	157	0	101
9	18	157	0	3585
17	18	157	1	174
9	18	157	1	1195
9	18	0	0	8388
17	24	0	0	10
9	24	0	0	3766
9	24	157	0	5376
9	24	157	1	5513
9	24	0	1	13235
8	24	0	0	1898

Controlled frequent data

Subsequent tests require precise calculation of the link quality by measuring the number of packets received per second at the destination node. To achieve this accuracy, two factors must be controlled. The rate at which the packet generation application sends data to the wireless card must be controlled; furthermore, the number of packets being sent out by the wireless card needs to be controlled. The first factor can be controlled with the use of a packet generation software that allows the specification of the number of packets per second. The second factor can be controlled by ensuring the packet generation software does not send more packets down to the wireless card than the wireless card can transmit to the air, as well as turning off retransmissions at the wireless card. Turning off retransmissions at the wireless card ensures that each packet is only transmitted once, and thus allows the precise calculation of how many packets were sent out of a transmitting node in a given test. RUDE is a software tool that allows the specification of how many packets are transmitted per second. The previous test can be repeated, with a few changes. RUDE is used instead ttcp. The application is configured to transmit 350 packets per second, and each packet is 1000 bytes; hence the bit rate from the application to the driver is 350KB/sec (well below the 802.11 rate, hence ensuring that no packets are dropped at the wireless card). The Atheros driver is further modified to ensure that packets are transmitted only once (retransmissions are turned off). These modifications allow the precise calculation

of how many packets are being sent out of the Phocus Array during a test of a given duration. The results of this test are shown as Table 4.3. This table exhibits several anomalies. The main anomaly is that the number of packets received with the 157° pattern is almost always greater than the number received with the 0° pattern (in otherwise identical configurations). Furthermore, the amount of variation is huge in different iterations of identical configurations. One possible reason is that the 802.11 rates used are too high. Recall that the higher the 802.11 rate used to transmit a packet, the more difficult it is for a receiver to decode the packet. Using higher 802.11 rates may therefore result in more susceptibility to random interference, producing such anomalous results.

In an attempt to reduce the susceptibility of testing to random interference, and produce more deterministic results, further testing can be performed using lower 802.11 rates The results of testing with lower rates can be seen in Table 4.5. As before, the test is inconclusive. The same anomalies exist; the number of packets received using the 157° pattern is greater than the number received in the 0° pattern, and there is a large degree of variation in subsequent iterations of the same configuration.

Table 4.5: Data transmitted using RUDE at different angles, varying transmit power and 802.11 rate (lower rates)

dB	rate	pattern	iter	pkts	-	dB	rate	pattern	iter	pkts
5	1	0	0	88	-	5	1	157	0	1
5	1	0	1	139	-	5	1	157	1	3053
5	1	0	2	164	-	5	1	157	2	2947
-	-	-	-	-	-	5	1	157	3	2627
6	1	0	0	15147	-	6	1	157	0	282
6	1	0	1	16709	-	6	1	157	1	269
6	1	0	2	16138	-	6	1	157	2	242
4	2	0	0	0	-	4	2	157	0	0
5	2	0	0	7	-	5	2	157	0	101
5	2	0	1	2	-	5	2	157	1	138
5	2	0	2	2	-	5	2	157	2	142
6	2	0	0	15472	-	6	2	157	0	14817
5	5.5	0	0	0		5	5.5	157	0	0
5	5.5	0	1	0		5	5.5	157	1	0
6	5.5	0	0	17394	-	6	5.5	157	0	12674
5	6	0	0	0	-	5	6	157	0	0
6	6	0	0	2963	-	6	6	157	0	1455
-	-	-	-	-	-	6	6	157	1	1543
-	-	-	-	-	-	6	6	157	2	1111
-	-	-	-	-	-	5	9	157	0	0
-	-	-	-	-	-	6	9	157	0	522
-	-	-	-	-	-	6	9	157	1	716
-	-	-	-	-	-	6	9	157	2	501
-	-	-	-	-	-	5	11	157	0	0
-	-	-	-	-	-	6	11	157	0	13427
-	-	-	-	-	-	6	11	157	1	13375
-	-	-	-	-	-	6	11	157	2	14404

Overall, the static tests performed are inconclusive. The tests do not yield a pair of angles such that connectivity between a Soekris box and the Phocus Array is achieved at one but not at the other; more broadly, the tests fail to prove that the antenna effectively emits a beam pattern than is consistent with the diagram.

4.2 Pattern rotation

The uncertainty surrounding the relative beam strengths at various angles around the Phocus Array calls into question whether or not the emitted beam conforms to the pattern described in the diagram (Figure 2.2). One way to effectively verify the emitted beam pattern is by observing the beam strength at every angle around the Phocus Array from a fixed distance. This set-up can be achieved by keeping the Phocus Array and associated client node physically stationary, and rotating the pattern on the Phocus Array. In these tests, the Phocus Array is configured in the first directional beam pattern. After a short period of observation (to measure the properties of the packets in that configuration), the pattern is then changed to the next pattern. The pattern is gradually changed through all of the patterns, yielding a rotation through all of the patterns. Packets are aggregated according to the pattern that is loaded during their transmission; properties of the packets within these aggregate classes can then be compared. This allows the comparison of the beam at various positions in the pattern, as well as measurement of the Phocus Array's to change patterns (and the effectiveness of those patterns). This test yields a granularity of 22.5°, since each pattern is offset from the other patterns by that angle.

4.2.1 Beacons only

This set of tests attempts to measure the emitted beam pattern by rotating the pattern, keeping all nodes physically stationary. The simplest test to run while rotating the pattern is a test where no data is being transmitted. If the beam pattern is being emitted effectively from the Phocus Array, and beacon packets are transmitted according to this beam pattern, then a test without any data should suffice. The Phocus Array is configured as an AP, and placed at one end of the mall testing area. Four Soekris boxes are placed in line with the Phocus Array, at distances of 66 ft (S4), 110 ft (S3), 154 ft (S2) and 198 ft (S1). Each Soekris box is configured as a client, and associated with the Phocus Array. The Phocus Array is oriented such that the Soekris boxes are at the Phocus Array's physical 0°, and loaded with the first co-phase beam pattern. The signal strength as reported by iwconfig is recorded periodically for 60 seconds. After 60 seconds, the pattern on the Phocus Array is rotated to the next pattern (offset 22.5° from the current pattern), and the 60 seconds of monitoring is repeated. This process of configuring a pattern and sniffing for 60 seconds is repeated through all of the directional beam patterns. The results are shown as Figure 4.13.

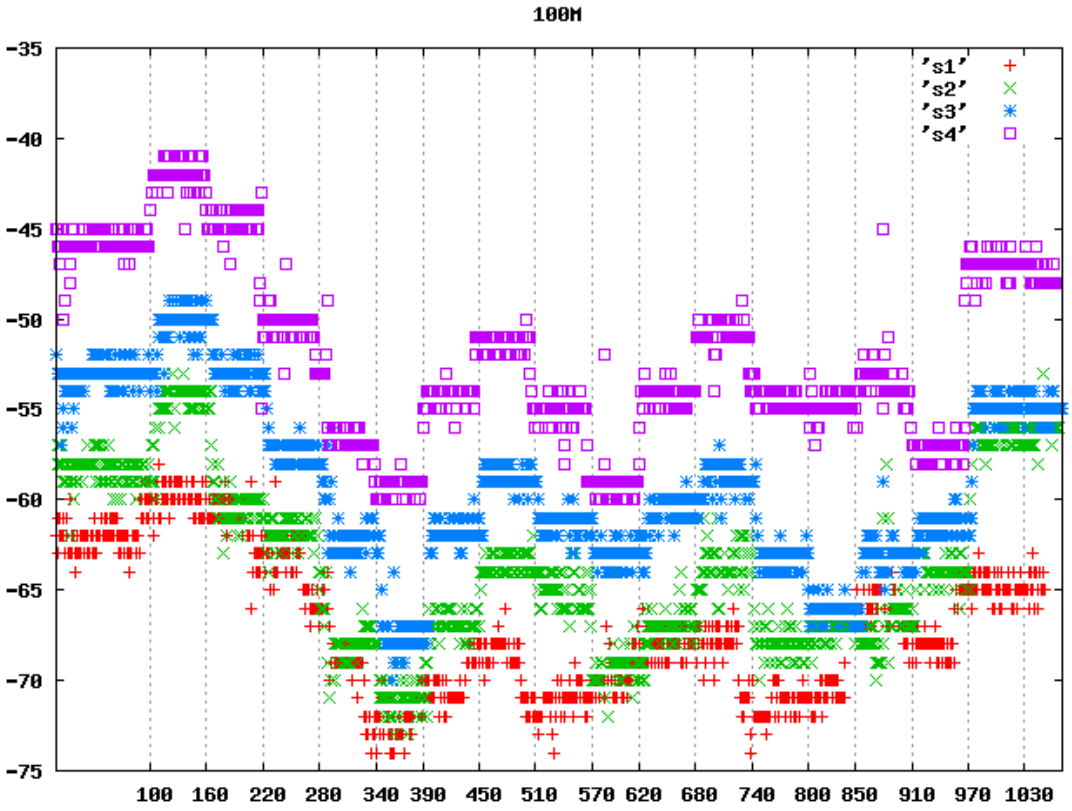


Figure 4.13: RSSIs reported by iwconfig on four clients while the Phocus Array pattern is rotated

Figure 4.13 can be plotted as a polar plot, shown as Figure 4.14. The x-axis from the linear plot corresponds to the rotational axis in the polar plot; measurements taken during the 60 seconds that the Phocus Array is in a single configuration have been aggregated into a single polar point. The angle from 0° on which a point is plotted is the angle of the main lobe while those measurements were being taken. The distance of a point from the center of the graph is the average of the RSSI values for those measurements, scaled by a linear constant.

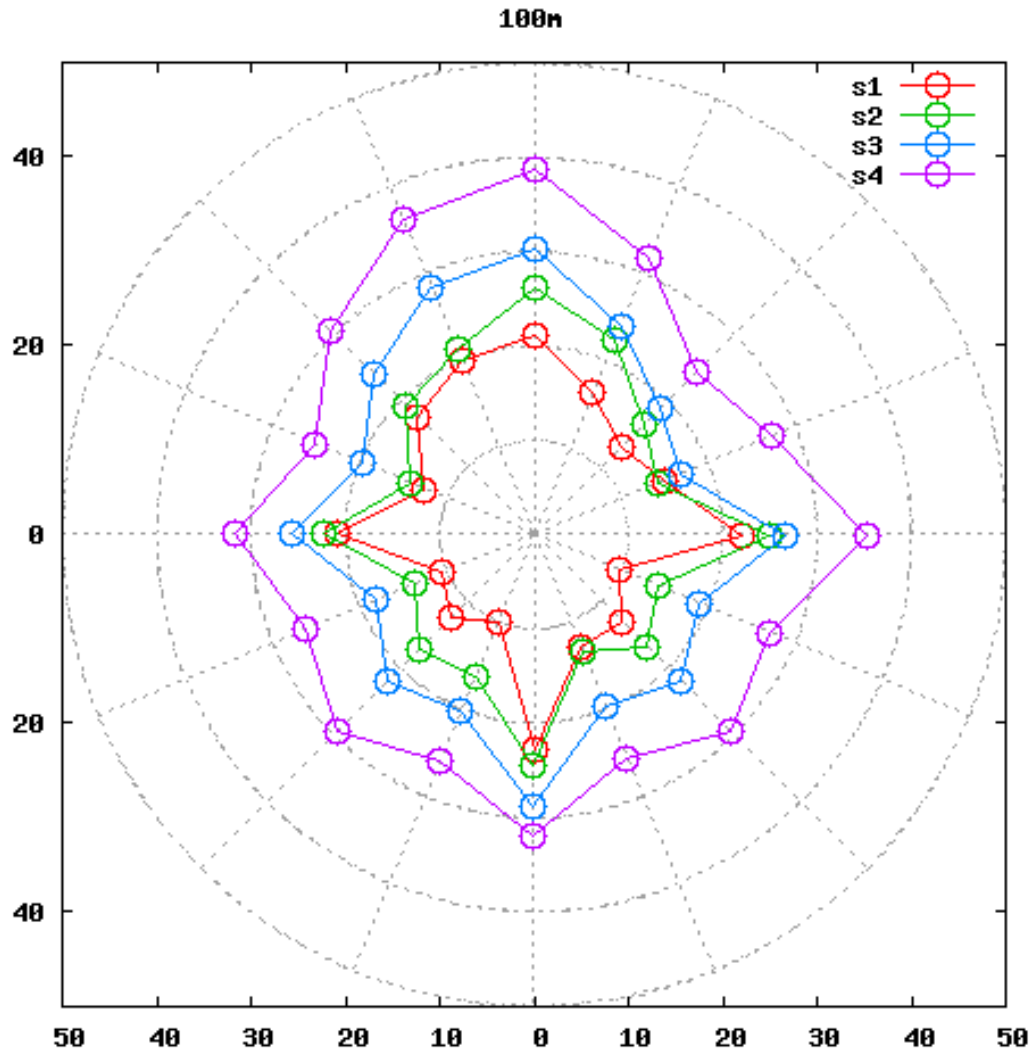


Figure 4.14: Polar plot of RSSIs reported by iwconfig on four clients while the Phocus Array pattern is rotated

Figure 4.14 allows a visual display of the measured beam pattern. As expected, the closer a Soekris box is to the Phocus Array, the greater the RSSI values (for the most part) are, for every pattern tested. However, the observed pattern looks very different from the diagram in Figure 2.2.

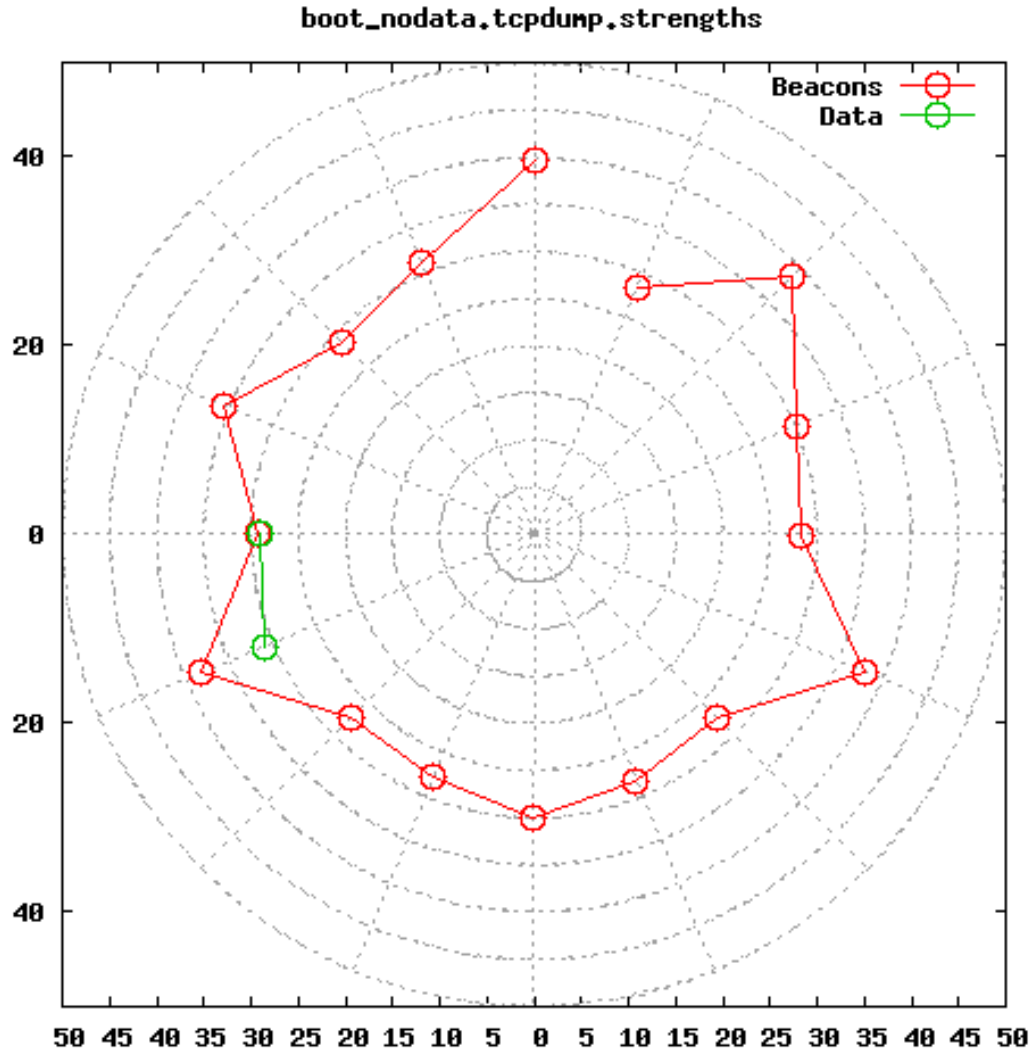


Figure 4.15: Polar plot of RSSIs reported by iwconfig on one client while the Phocus Array pattern is rotated

One possible explanation for the unexpected pattern observed in Figure 4.14 is that the measurement granularity is not fine enough. More specifically, using sample data from iwconfig is not as accurate as measuring the RSSI of every packet received during a test.

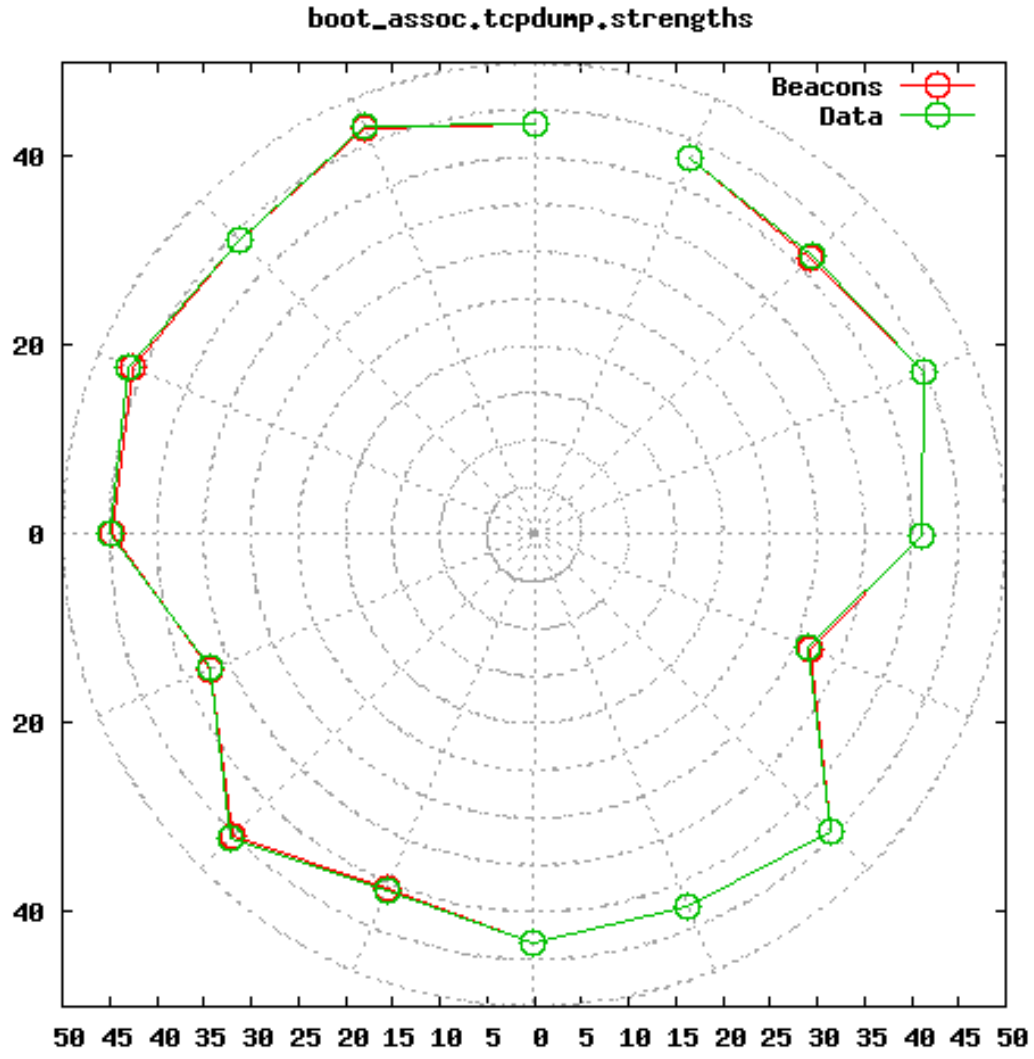


Figure 4.16: Polar plot of RSSIs reported by tcpdump on one client while the Phocus Array pattern is rotated and data is being transmitted

Further tests measure every single packet received, allowing for more accurate analysis. The previous test can be repeated using tcpdump to record every packet that is received. Only one Soekris box is used, since the RSSI fades with distance as expected (and hence the pattern for all four Soekris boxes in the previous result

is roughly concentric). Figure 4.15 shows the results of this test. The observed pattern, while clearly variable in different directions, is once again not similar to the expected pattern depicted in the diagram.

4.2.2 Data

It is conceivable that only data packets are transmitted in in a the beam pattern, whereas beacon packets are transmitted in an omnidirectional pattern. This hypothesis can be verified with a variation on the pattern rotation test, in which data is being transmitted. This test is the same as the test in which data is not being transmitted, with the only difference being that RUDE is used to transmit data from the Phocus Array to the client. At the client, tcpdump can be run to measure the RSSI values of data packets and beacon packets. The results of this test can be see as Figure 4.16. In the graph, the beacon packets and data packets exhibit similar RSSI values, invalidating the hypothesis that beacon packets may be sent out with a different beam pattern than data packets. Not only does the pattern not resemble the pattern in the diagram, but the the observed pattern is also very different from the previously observed beam pattern in Figure 4.15.

The pattern rotation tests should yield the pattern diagram given in Figure 2.2 under the assumption that all patterns are identical. While all directional patterns are very similar, they are in fact not identical. Figure 4.17 shows the directional pattern that points the main lobe at 22.5°. This diagram is not a direct rotation

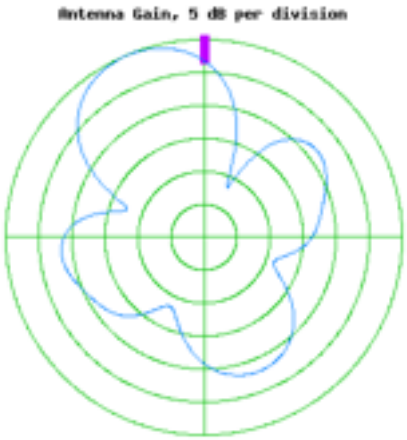


Figure 4.17: The co-phase pattern at 22.5°. The line shows the relative beam strength at various angles to the Phocus Array. The distance the line is at any given point from the center of the graph indicates the magnitude of the beam at that angle. From Fidelity Comtech [3].

from Figure 2.2; for example, Figure 2.2 has two back lobes, whereas Figure 4.17 only has one. This observation may explain why the pattern observed in the pattern rotation tests differs from Figure Figure 2.2.

4.3 Physical rotation

To more accurately measure the beam pattern emitted by the Phocus Array, the signal strength of the radio wave from the Phocus Array at all angles should be measured. This measurement can be achieved by keeping the pattern constant, and either rotating the client around the Phocus Array, or physically rotating

the Phocus Array while keeping the client statinary. Physically rotating the unit keeps the 'state' of the unit constant (without changing any internal configuration), thereby allowing a more concrete measurement of the pattern emitted by the Phocus Array.

The physical rotation experiments can be broadly categorized into two types, according to the beam pattern they attempt to test. The beam pattern from the Phocus Array is not only a transmit pattern, but a receive pattern as well. The physical rotation tests attempt to test both the transmit as well as the receive patterns; we can thus divide the tests into transmit beam and receive beam tests.

4.3.1 Transmit beam

The transmit beam pattern experiments attempt to measure the beam pattern emitted from the Phocus Array. In these tests, packets are transmitted from the Phocus Array; data packets are received and measured at a client node, while beacon packets are either received and measured at a client node or at a monitor node.

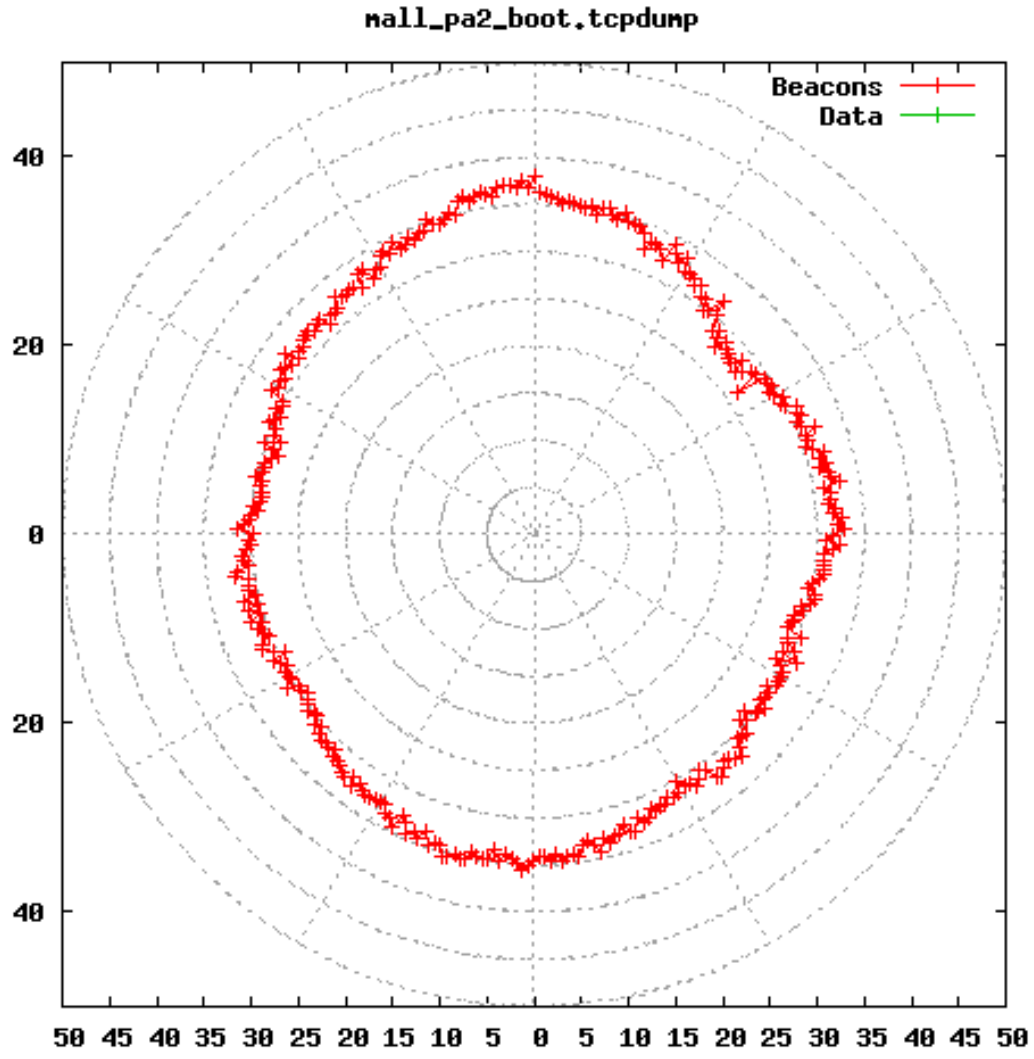


Figure 4.18: Polar plot of RSSIs reported by tcpdump on one client while the Phocus Array is rotated in the mall

Beacons only

These tests attempt to accurately measure the beam pattern emitted by the Phocus Array, by physically rotating it. The simplest experiment to run when

rotating the Phocus Array is without any data being transmitted. In this test, the Phocus Array is set up as an AP, loaded with the co-phase beam pattern, and placed at one end of the mall testing area with its physical 0° facing the other end of the mall. A Soekris box S1 is placed 150ft away, at the other end of the mall. S1 is configured as a monitor, with tcpdump running on it. The Phocus Array is in the default boot state, without any modifications to the Atheros driver. The Phocus Array is slowly rotated through 360° (1° every three seconds, thus taking roughly 18 minutes for a full rotation). The RSSI of the beacon packets is captured by tcpdump on S1, and recorded. The results of this test are shown as Figure 4.18. The graph clearly looks nothing like the beam pattern diagram in Figure 2.2; in fact, the observed pattern seems almost omnidirectional.

One possible explanation for the observed pattern being almost omnidirectional is that the packets received at the Soekris box may have been reflected off the buildings that surround the mall. Possibly multiple reflections may have an unknown effect on the RSSI of the packets; this effect when averaged out may have caused packets of all angles from the Phocus Array to appear to have a similar RSSI. Data from a previous test can be analyzed to determine if reflections are occuring. If so, then there will exist multiple clearly defined distributions (with the use of a gaussian fit function) within the range of RSSIs of packets received at any particular angle. Points can be plotted that are the average of these multiple distributions; if reflections are occuring, multiple distinct lines will appear on a polar plot.

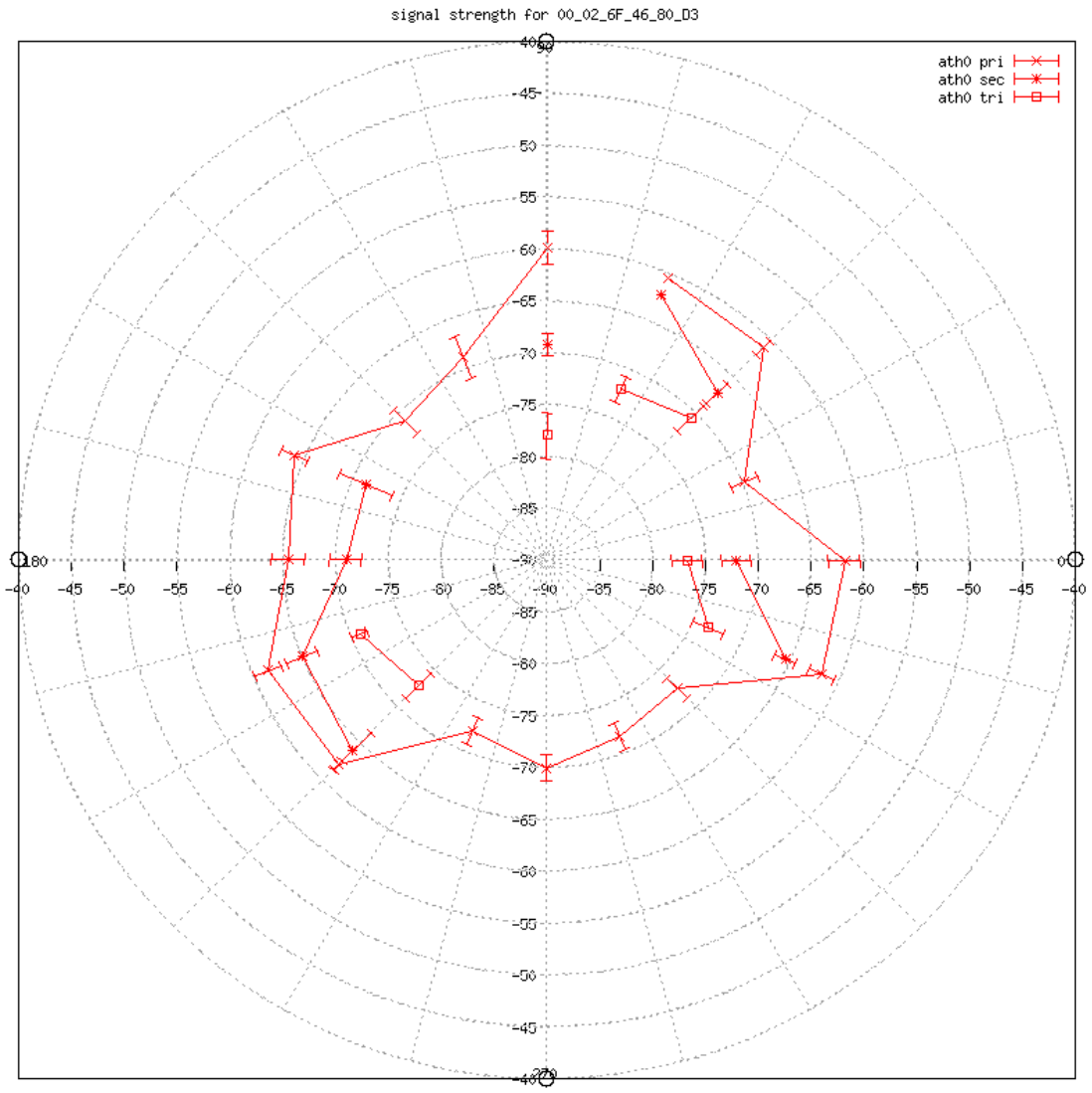


Figure 4.19: Polar plot of RSSIs reported by tcpdump on one client while the pattern is rotated. The multiple lines are multiple distributions that emerge after filtering the data through a gaussian fit function.

Data from a previous pattern rotation test can be analyzed with a gaussian fit function, resulting in Figure 4.19. As the graph shows, there are clearly multiple

points at several angles; lines can be drawn between the average of the different distributions at different angles, resulting in multiple lines around the graph. These multiple lines indicate where reflections are occuring and causing noise in the observed pattern.

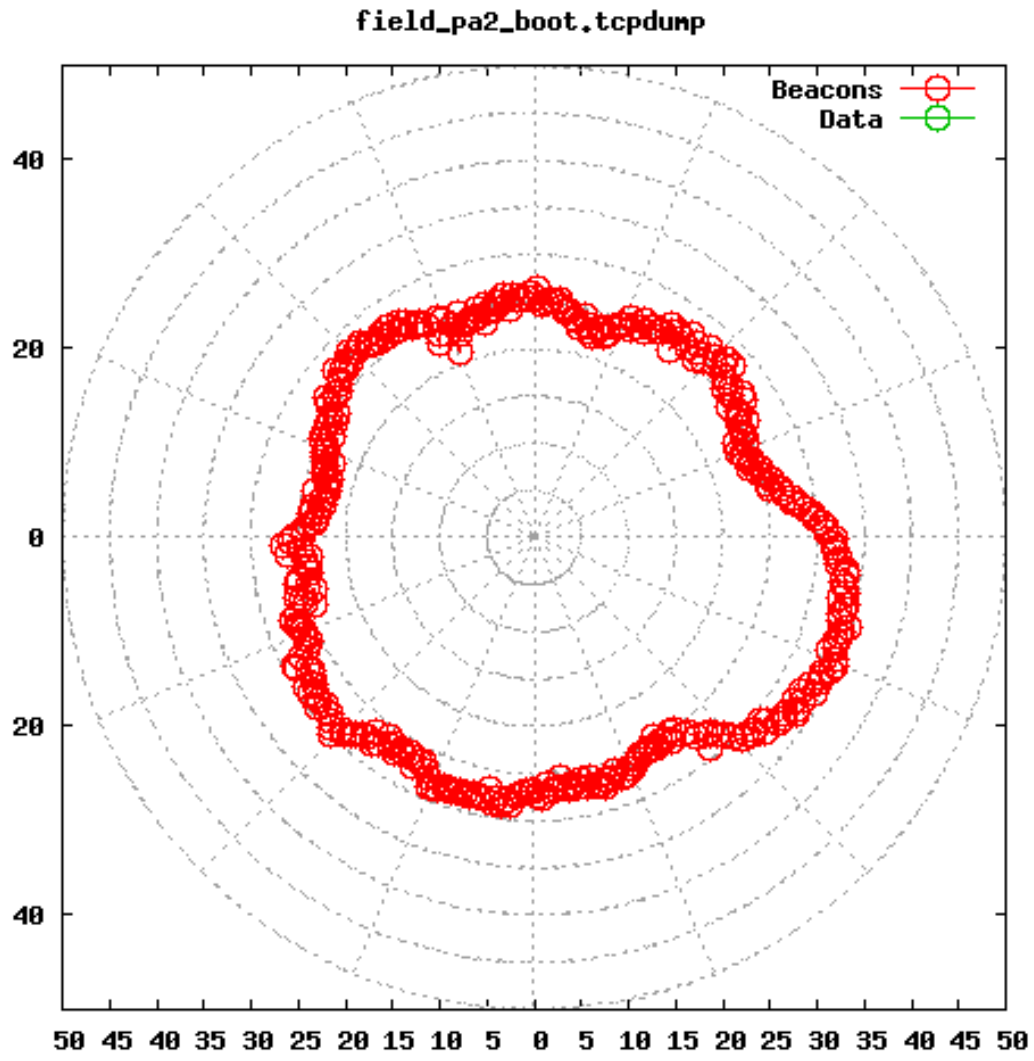


Figure 4.20: Polar plot of RSSIs reported by tcpdump on one client while the Phocus Array is rotated in the field

One way to mitigate the effect of multiple reflections is to change the testing location to an area where there are no surfaces to cause packet reflection. The previous test can be repeated in exactly the same manner, in the field testing location. The field provides a flat, open area with large distances between the testing area and any buildings; testing in the field should ideally provide a relatively reflection free environment. The results of this test can be seen as Figure 4.20. The observed beam pattern in the field is more directional than as observed in the mall (there is a greater difference between the highest and lowest point in the graph); however, the beam pattern still does not resemble the pattern diagram.

The progression of testing thus far has focussed on improving the test set-up, instrumentation, and methodology; however, the observed beam pattern is still not as expected. One way to verify the test methodology is to replicate the test using a known directional antenna; if the observed pattern is as expected, the test methodology will have been proven to be correct. The previous test can be repeated using a known directional antenna instead of the Phocus Array. The directional antenna only transmits packets in one direction; the emitted beam pattern should have a large main lobe in the direction of transmission, and null in all other directions. The results of this test are shown as Figure 4.21. The observed pattern is as expected. In the direction of transmission (the top half of the graph), the pattern displays a large main lobe. The main lobe has its edges at roughly 30° and 210°. There are almost no packets received in between the angles of 30°

and 270° (going clockwise). This experiment shows that the testing methodology is correct, and that any results observed with this testing methodology can be believed as the pattern that is emitted from the antenna being used (as opposed to a possible error in the test set-up).

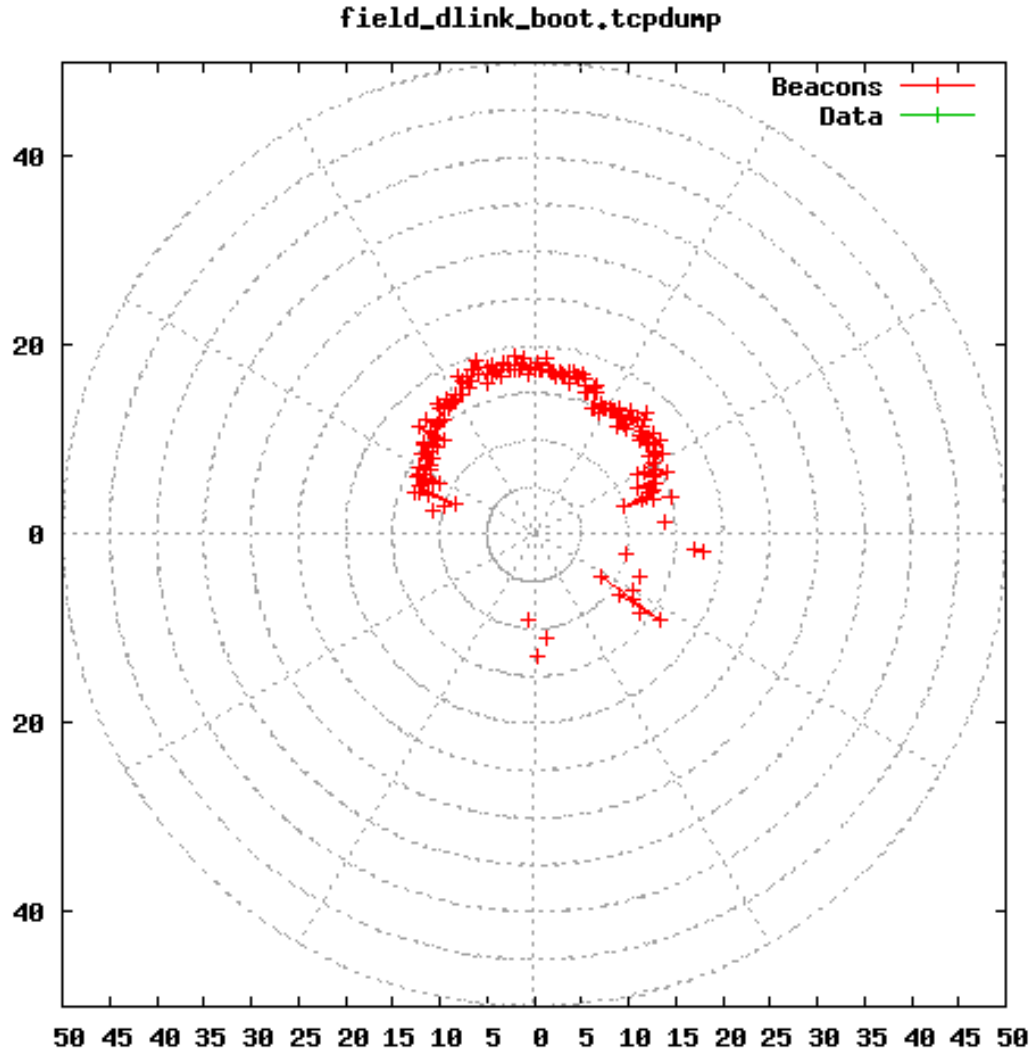
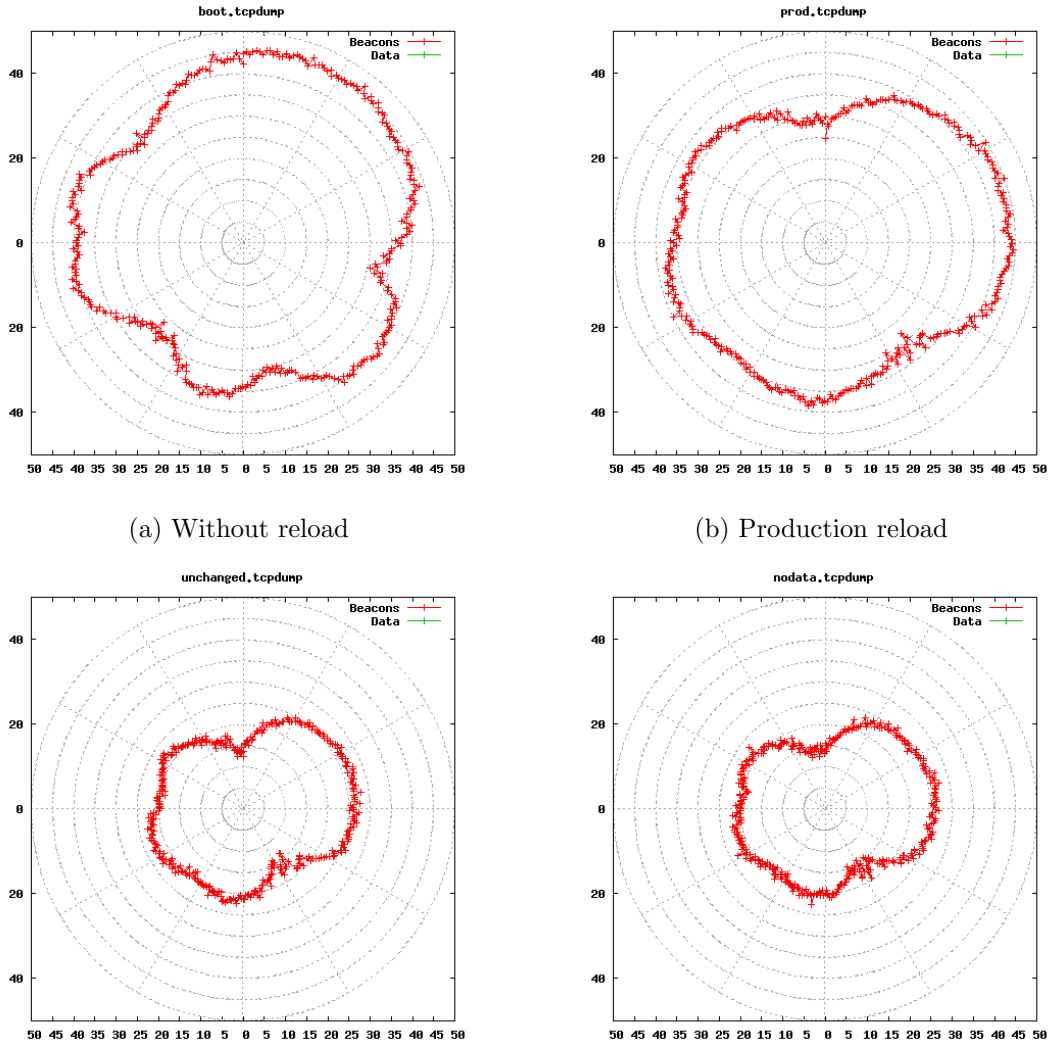


Figure 4.21: Polar plot of RSSIs reported by tcpdump on one client while a directional antenna is rotated in the field

Having verified the test set-up, the next step is to investigate why the beam pattern emitted by the Phocus Array does not resemble the diagram. One possible explanation regards how the Phocus Array interacts with the Atheros driver. At boot time, the Phocus Array loads the Atheros driver, and initializes each antenna's T/R module with a phase an magnitude to use when transmitting; the combination of the waves emitted by each antenna (given the differences in phase and magnitude) is what produces the beam pattern. One possible way to test whether the Phocus Array is loading the Atheros driver correctly is to compare the beam pattern emitted when the Phocus Array boots in its default state with the beam pattern emitted when various versions of the driver are loaded. The Atheros driver can be reloaded at runtime, and the beam pattern tested. The different modifications that can be made to the Atheros driver that can be reloaded are infinite; three main versions seem the best candidates for testing. The Atheros driver that is already loaded on the Phocus Array (the version of the driver in production) can be reloaded at run time, to see if the boot sequence contains an error. A version of the driver that is unchanged, but compiled directly from the source code can be loaded, to test if the production version of the Atheros driver is incorrect. Yet another version of the driver, with the 802.11 rate fixed and the retransmissions turned off can also be used, to test whether these factors cause the incorrect beam pattern. This test is set-up in exactly the same way as the previous test; the only difference is how the Atheros driver is treated at runtime.



(a) Without reload

(b) Production reload

(c) Compiled unmodified reload

(d) Compiled modified reload

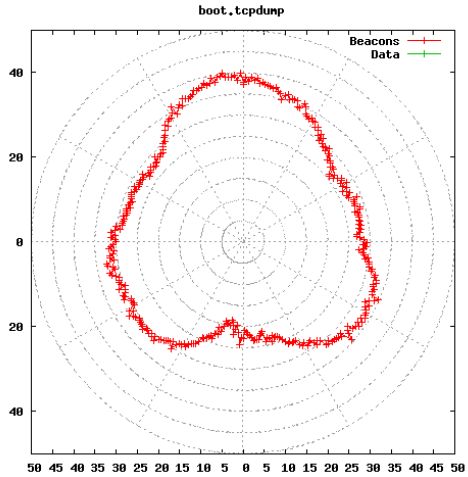
Figure 4.22: Polar plot of RSSIs reported by tcpdump on one client while the Phocus Array is rotated in the field

The pattern emitted without loading the Atheros driver can be seen as Figure 4.22a. The pattern emitted when the production, unchanged, and controlled versions of the driver are reloaded at runtime can be seen as Figures 4.22b, 4.22c and

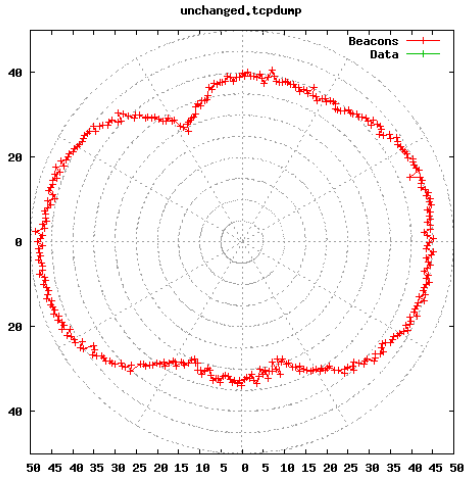
4.22d. The unchanged and changed versions of the driver produce similar beam patterns, indicating that the compilation and reload procedures are consistent. The production beam pattern is also similar to the compiled versions, except with a larger magnitude at all angles. This fact supports the idea that the reload procedure is consistent; however for some reason any compiled code results in a lower beam strength at all angles. The pattern observed without reloading the driver is vastly different from when the same code is reloaded a runtime; this fact indicates that the reload process (while consistent) does not initialize the driver in the same way as the Phocus Array does when it boots.

Figures 4.22a and 4.22b indicate that the process of reloading the same driver at runtime results in a different beam pattern. To mitigate this, the previous test (in all four configurations) can be repeated, but by forcing the Phocus Array to use the different versions of the driver at boot time. Loading the production version of the code does not need to be performed as a separate test from booting the Phocus Array in its default configuration, yielding only three tests: using the default boot version, the compiled but unchanged version, and the modified and compiled version. The results of these tests can be seen as Figures 4.23a, 4.23b and 4.23c. These three graphs look vastly different; possible explanations are that either the version of the driver in production is different from the compiled versions (and in fact the changes between the compiled versions also cause a change in the beam pattern), or that the Phocus Array is inconsistent in the initialization of the

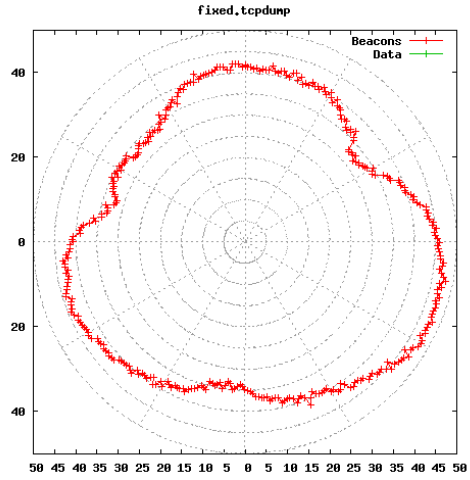
driver no matter what version is used. The fact that Figure 4.23a looks different from 4.22a, even though the tests were identical, would seem to indicate that the latter explanation is more likely.



(a) Without reload



(b) Compiled unmodified reload

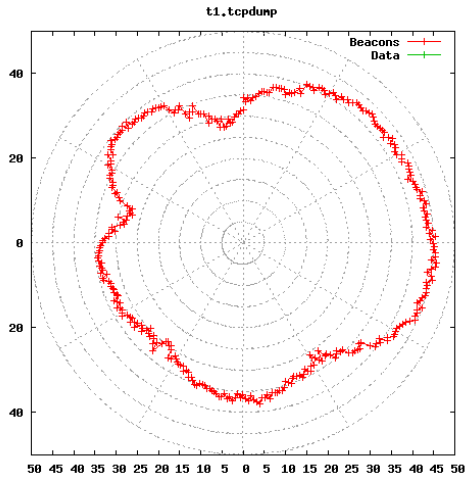


(c) Compiled modified reload

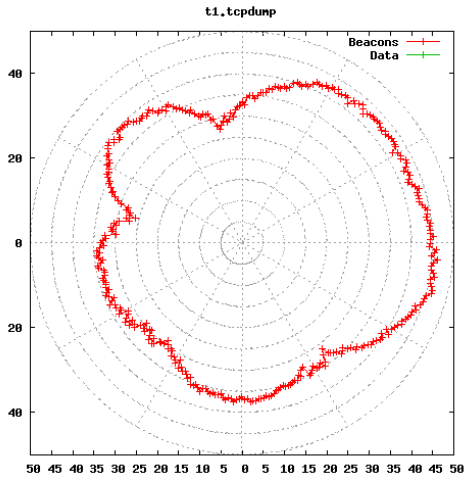
Figure 4.23: Polar plot of RSSIs reported by tcpdump on one client while the Phocus Array is rotated in the field, repeated

Previous testing has shown that the beam pattern emitted by the Phocus Array varies. The question then becomes one of discovering what cause the Phocus Array to change its beam pattern; does the pattern change over time, or only across reboots? To determine this, the following test may be used. The Phocus Array is rebooted, and rotated twice. The Phocus Array is then rebooted, and rotated once more. Using the pattern in the first rotation of the first reboot as a baseline, the other two patterns will determine if the pattern changes over time (the second rotation of the first reboot), or only across reboots. The results for this test are shown as Figures 4.24a, 4.24b, and 4.24c (the two rotations after the first reboot, and then the rotation after the second reboot). As the graphs show, there is no variation over time; the first and second rotations after the first reboot yield a similar beam pattern. However, after a reboot the beam pattern changes.

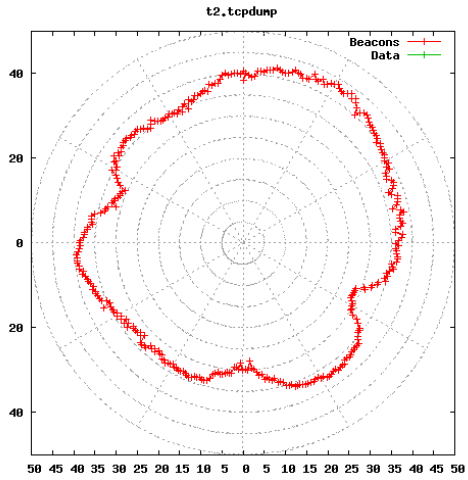
One possible explanation for the inconsistency across reboots is that the pattern is not being correctly loaded at boot time. A test that could verify this would be to repeat the previous test, but try to force the setting of a pattern on the Phocus Array before every single rotation (as opposed to assuming the pattern is being correctly loaded as per prior configurations). The results of this test are show similar properties as the previous three: that the pattern is preserved over time, but not across reboots.



(a) Reboot one, rotation one



(b) Reboot one, rotation two



(c) Reboot two, rotation one

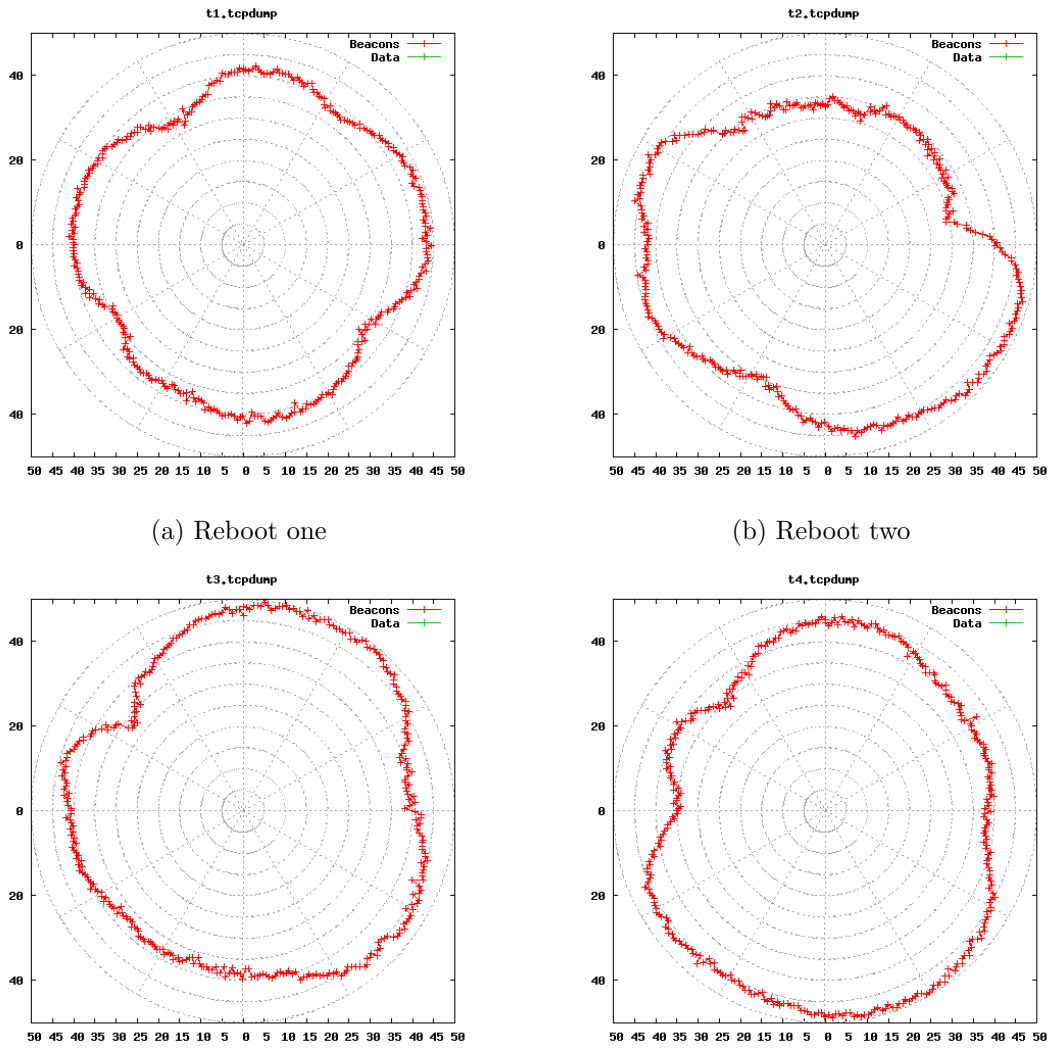
Figure 4.24: Polar plot of RSSIs reported by tcpdump on one client while the Phocus Array is rotated, with multiple rotations and reboots

To confirm that the pattern is not preserved across reboots, the Phocus Array can be rebooted and the pattern measured several times in a row. In this test, the Phocus Array is rebooted and the pattern is measured. The Phocus Array

is then rebooted and the pattern is measured again several times, four in total. The results of these four reboots and measurements are shown as Figures 4.25a, 4.25b, 4.25c, 4.25d. The graphs clearly show that the pattern is different after every reboot; this behavior indicates that the Phocus Array does not function in a consistent manner.

Data

The Phocus Array has been shown to be inconsistent across reboots. However, the patterns observed are only the patterns for beacons; it is possible that the emitted pattern conforms to the pattern diagram with data packets only. To test this, an experiment can be run in which data is transmitted from the Phocus Array to a client. The Phocus Array is set up as an AP, loaded with a beam pattern, and placed at one end of the field. The modified and compiled Atheros driver is loaded at runtime, to allow the specification of how many packets are being sent out from the Phocus Array during the test. A Soekris box S1 is placed 150ft away, set up as a client, and associated with the Phocus Array. The Phocus Array undergoes one full rotation, during which RUDE is used to transmit data to S1. Figure 4.26 shows the result of this test. As the graph shows, the observed pattern for data packets and for beacon packets is very similar; furthermore, the observed beam pattern is not only nothing like the pattern diagram, but appears to be almost omnidirectional.



(a) Reboot one

(b) Reboot two

(c) Reboot three

(d) Reboot four

Figure 4.25: Polar plot of RSSIs reported by tcpdump on one client while the Phocus Array is rotated after multiple reboots

4.3.2 Receive beam

The beam pattern observed in the transmit tests does not resemble the pattern diagram. Theoretically, the receive beam pattern is the same as the transmit beam

pattern. If the receive pattern is effectively achieved, packets arriving from a node in a certain direction will achieve a gain (relative to packets from other directions) that is correlated with the relative strength of the beam pattern in that direction. For example, in our low-sidelobe pattern (Figure 3.2), packets from nodes at 0° to the Phocus Array should have much higher signal strength than packets from nodes at 270° (assuming the nodes are transmitting packets from equal distances and with equal power). In receive tests, data is transmitted from a client node. A monitor interface is set up at the Phocus Array; the RSSI of the data packets is observed and recorded. If a client node transmits to the Phocus Array while the Phocus Array completes a full rotation, the graph of RSSI against time can be interpreted as a polar plot, which produces a visual representation of the beam pattern. Receive tests can be categorized by the mode in which the Phocus Array is configured during the test: dynamic, or static.

Dynamic

In dynamic tests, the antenna is configured in dynamic mode. In dynamic mode, the antenna rotates through each of the 16 loaded directional patterns, eliciting and measuring the RSSI of a response from each client. After a full sweep through all the patterns, for each client the Phocus Array has an RSSI reading for every pattern. The mapping is stored in an XML file called station_stats.xml.

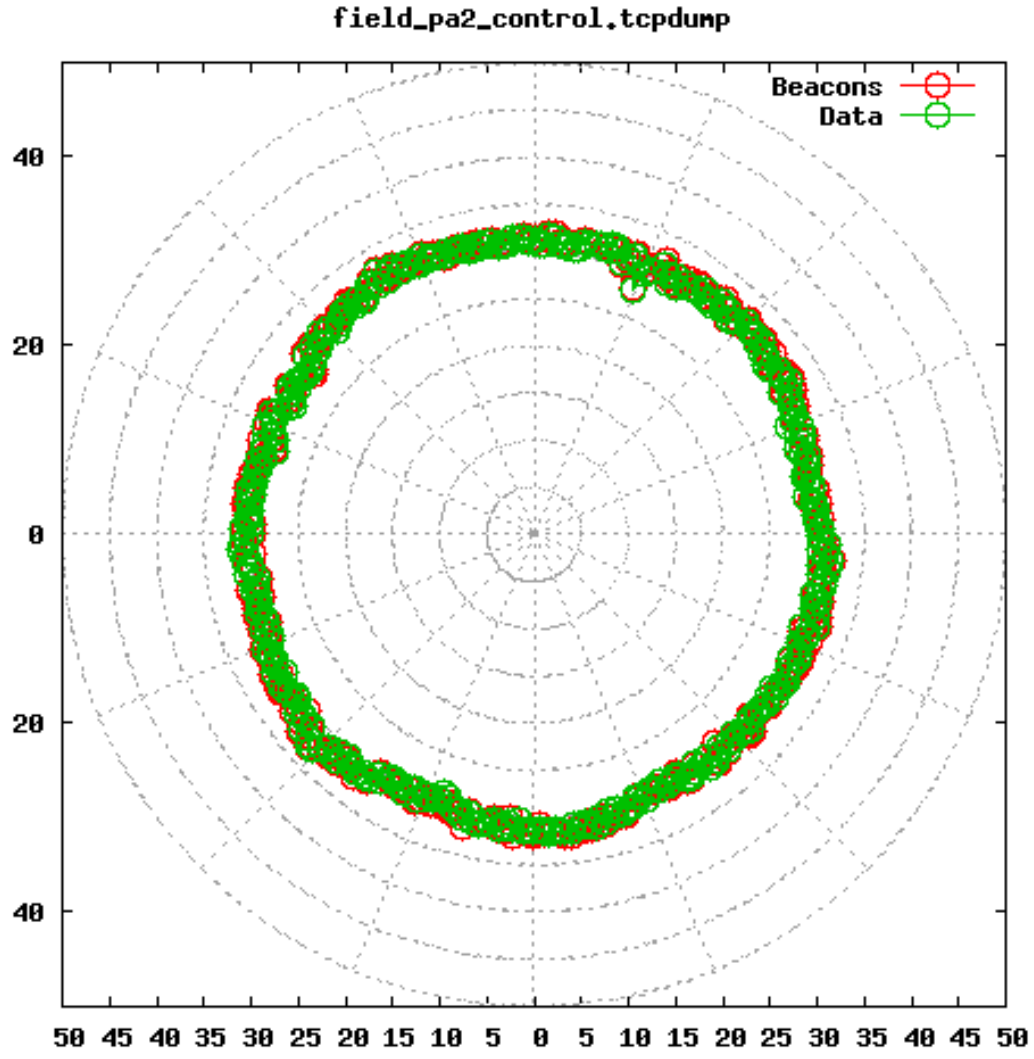


Figure 4.26: Polar plot of RSSIs reported by tcpdump on one client while the Phocus Array is rotated once with data being transmitted

If the Phocus Array is gradually physically rotated every few degrees (for example every 5°) through one full rotation, and at each 5° angle the contents if the XML file are recorded, by the time the Phocus Array completes one full rotation a polar plot of the beam pattern for all 16 patterns can be generated (with the

granularity of the points on the polar plot bring 5°). If the Phocus Array is allowed to complete multiple sweeps at each physical angle, the readings from those multiple sweeps can be combined and averaged to provide statistical significance beyond only one reading. In this test, the Phocus Array is configured as an AP in dynamic mode. The Phocus Array is located at one end of the field. A Soekris box S1 is placed at a distance (150ft in this test) away from the Phocus Array, at a physical angle of 0° to the Phocus Array. The XML file is polled every second; upon the XML file being updated three times, the Phocus Array is rotated through 5°. This process is repeated until the Phocus Array completes a full 360° rotation. At the end of the test, 16 polar plots can be generated. The test is repeated when the Phocus Array is loaded with the cophase (Figure 2.2) patterns, as well as with the low-sidelobe patterns (Figure 3.2). For each version of the test, 16 graphs can be generated. Figure 4.27 shows the graph for the co-phase 0° pattern. Figure 4.28 is the graph for the low-sidelobe 22.5° pattern.

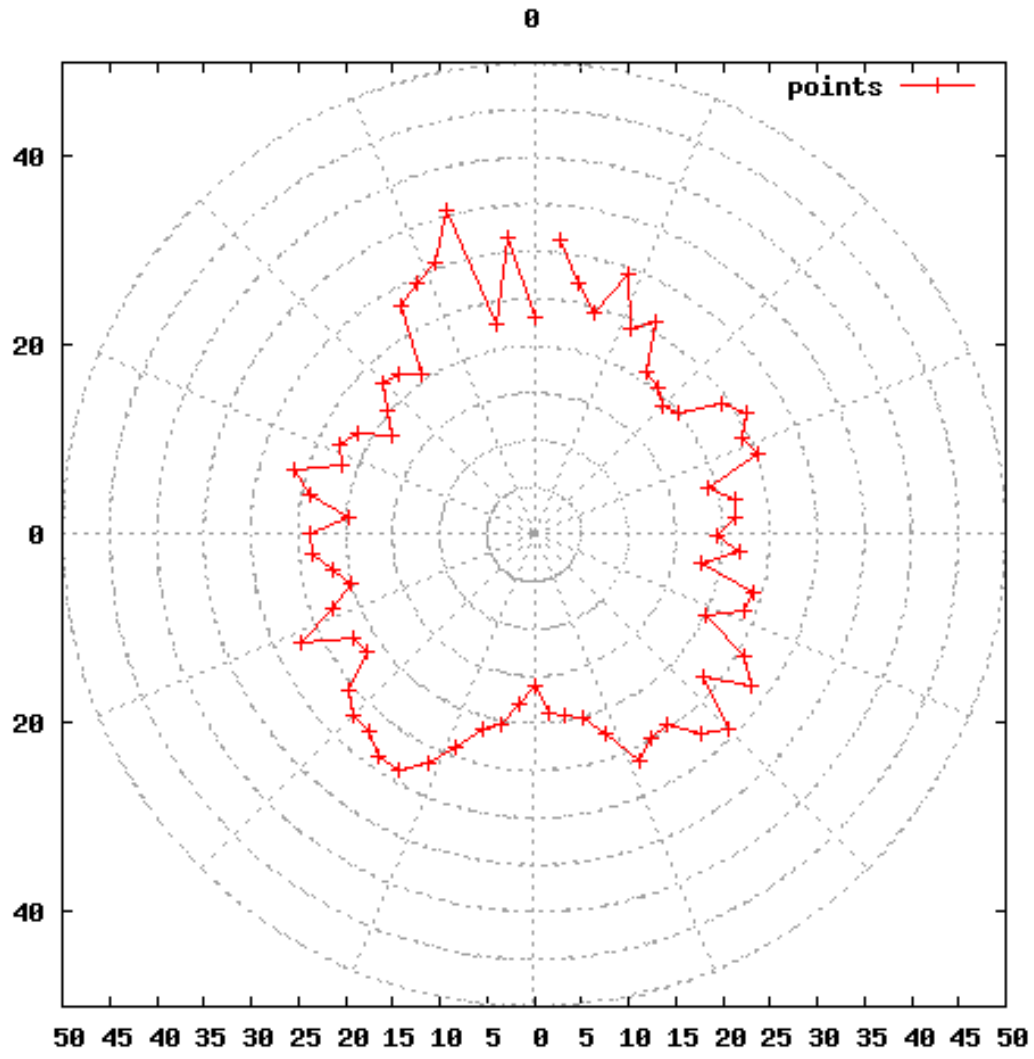


Figure 4.27: Polar plot of RSSIs reported in the XML file on the Phocus Array while in dynamic mode; pattern is co-phase 0°

The graphs look more like the pattern diagrams than the transmit tests seemed to produce. The low-sidelobe graph in particular looks much more like the pattern diagram than the co-phase pattern. One possible explanation is that the low-sidelobe pattern is simpler, and thus easier to effectively achieve. The co-phase

pattern is more complicated, with multiple non-primary nodes and several nulls; this pattern may be to complex to achieve in practice.

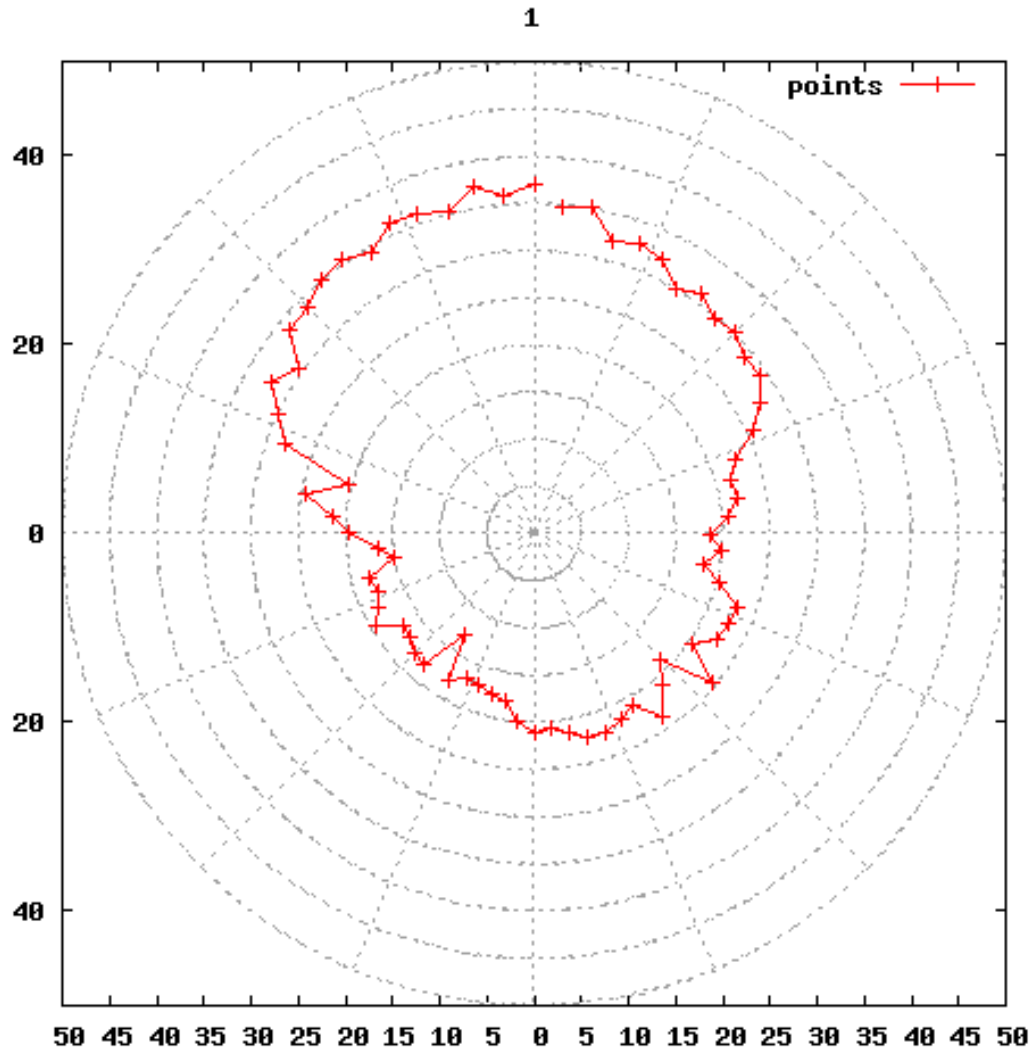


Figure 4.28: Polar plot of RSSIs reported in the XML file on the Phocus Array while in dynamic mode; pattern is low-sidelobe 22.5°

Static

The dynamic tests have verified that the receive pattern is more likely to resemble the pattern diagram than the transmit tests. However, the dynamic tests have several drawbacks. For example, the granularity of the dynamic tests is only every 5°. While the test may be modified to achieve the granularity of every degree, running the test with that granularity would take a long time to collect enough samples to achieve statistical significance (given that the minimum sweep time is every five seconds). Furthermore, the dynamic test yields a graph for each of 16 patterns; the Phocus Array rotates through all 16 patterns very quickly, every sweep interval. This frequent reconfiguration of the Phocus Array may introduce some unforeseen variability in the Phocus Array's ability to achieve an effective receive beam in a steady state. For these reasons, experimenting with the Phocus Array in static mode may yield more accurate results. In these tests, the Phocus Array is put into static mode, and configured with a beam pattern. The Phocus Array is configured as an AP, with a monitor interface running, and placed at one end of the field. A Soekris box S1 is configured as a client, associated with the Phocus Array, and placed a short distance away (30ft) at a physical angle of 0° to the Phocus Array. S1 sends data with RUDE at a constant rate to the Phocus Array; tcpdump is run on the Phocus Array on the monitor interface. Data packets from S1 are filtered out, and recorded at the Phocus Array. The Phocus Array is gradually rotated through 360°. The RSSI of packets from S1 can be graphed

and displayed as a polar plot, which shows the achieved receive beam pattern. The results of this test are shown as Figure 4.29. This graph clearly shows a well defined receive beam pattern that closely resembles the pattern diagram.

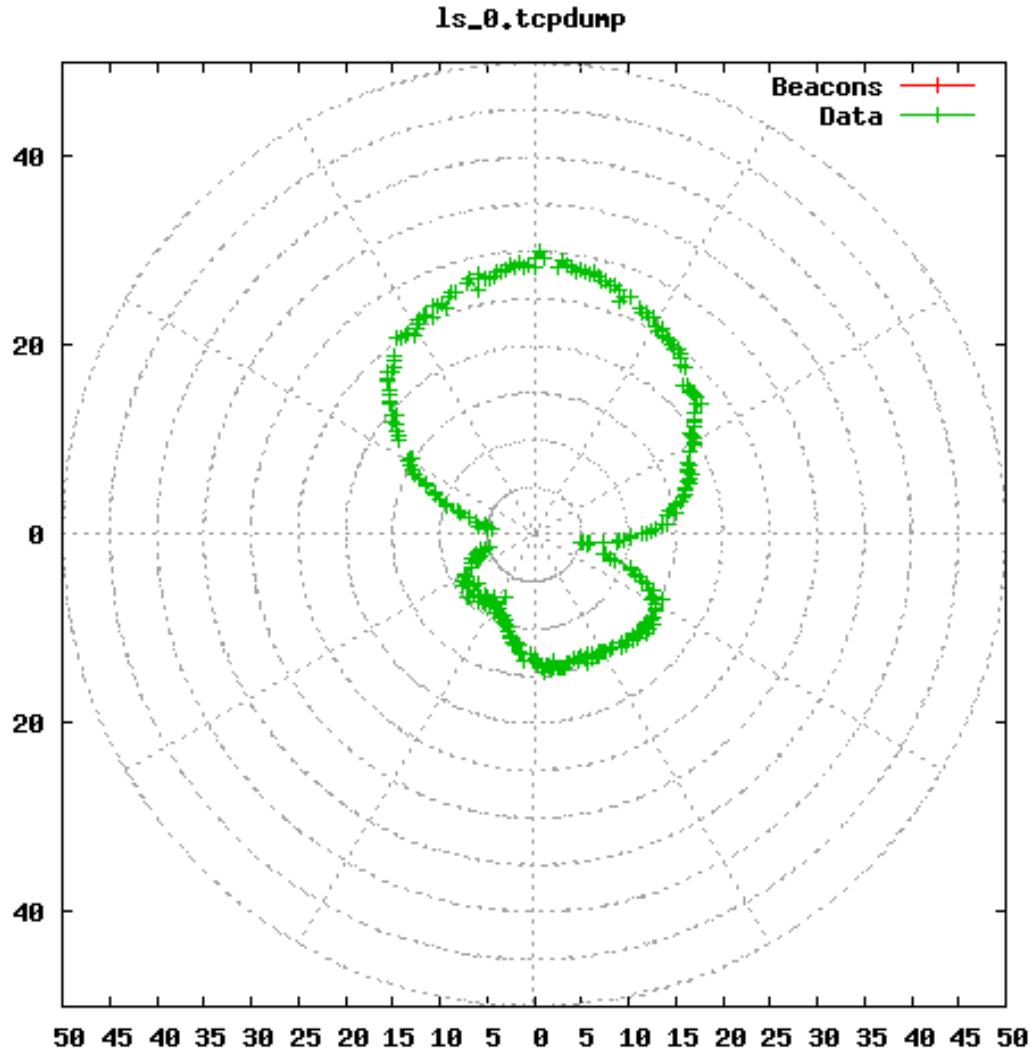


Figure 4.29: Polar plot of RSSIs reported by tcpdump on the Phocus Array; pattern is low-sidelobe 0°

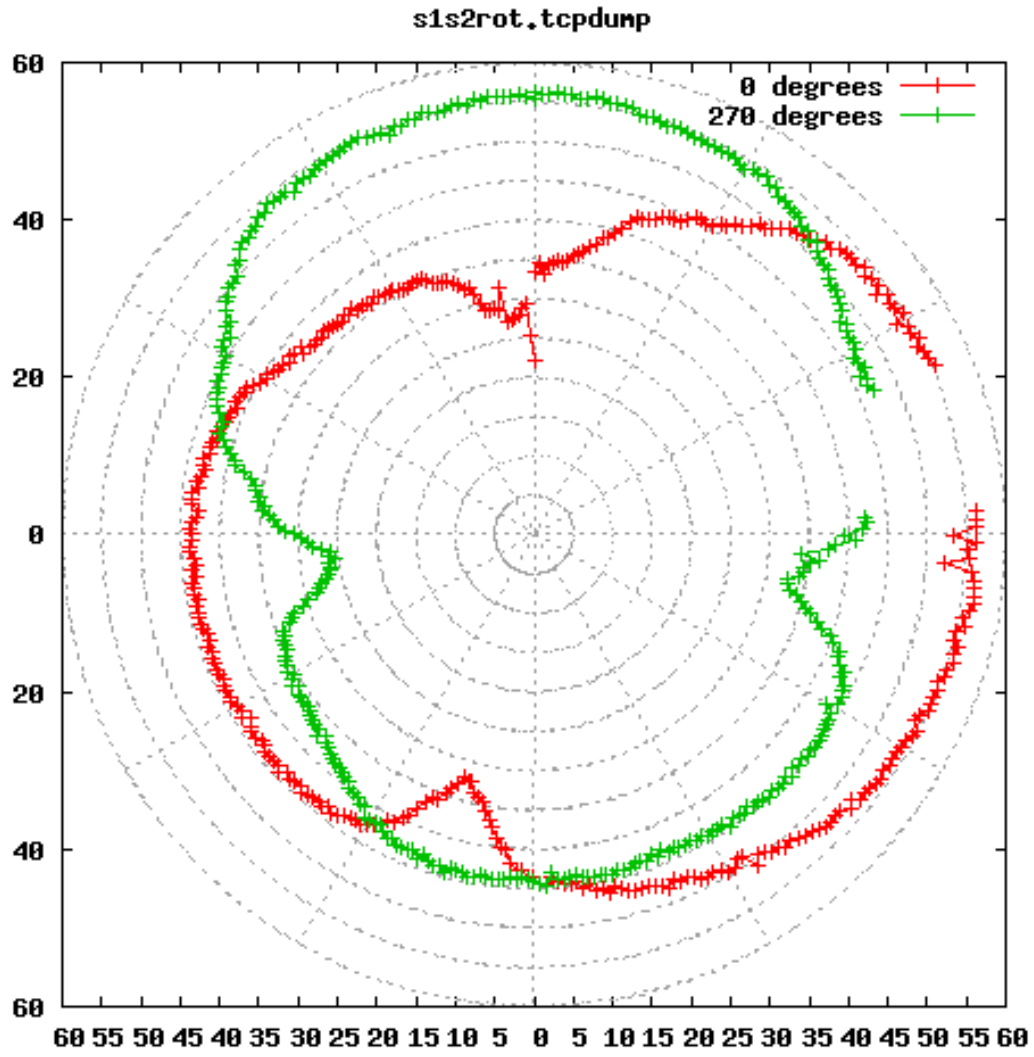


Figure 4.30: Polar plot of RSSIs reported by tcpdump on the Phocus Array; pattern is low-sidelobe 0°; clients at 0° and 270°

The previous test can be further extended by adding another data source. A second Soekris box S2 can be associated with the Phocus Array and placed on the field at a physical angle 270°; RUDE is run on S2, transmitted data packets to the Phocus Array. If the Phocus Array is then rotated through 360°, the resulting

graph should display the same receive pattern twice, with one of the patterns being rotated by 90°. The results of this test are shown as Figure 4.30. The graph is as expected; the receive pattern from the client at 0° is the same as Figure 4.29. The receive pattern from the client at 270° has the same magnitude as the pattern from the other client, and is rotated by 90°.

The experiments thus far have shown that the beam pattern emitted by the Phocus Array in static mode does indeed conform to the pattern diagram. Further testing can attempt to show that spatial reuse is achievable with the use of multiple client nodes. A client node located at the maximal (main lobe, 0°) area of the beam pattern should achieve much greater throughput than a node at the minimal area in the pattern (roughly at 270°). Moreover, throughput between the client at the main lobe and the Phocus Array should not be affected by traffic originating from the node at 270°. Tests that demonstrate this behavior have to date been unsuccessful.

5

Conclusions

The tests presented iterate through a logical progression, changing various control factors about the testing in an attempt to verify the effectiveness of the Phocus Array's beam forming ability. The factors that are changed vary from the state of the Phocus Array during the test (static, rotating its pattern, or physically rotating), to the type and amount of data transmitted between nodes (no data, infrequent data, controlled frequent data). Several other factors also vary between tests, such as the type of pattern being tested and the location. While each and every test yields information about either the testing methodology or the antenna itself, here we attempt to summarize the main conclusions that can be drawn.

5.1 Main conclusions

While the static tests did not yield much information about the beam pattern, they aided in refining the testing methodology. The beam pattern should not be assumed; the static tests showed that the only way to truly understand how to use the pattern to its fullest potential (whatever the higher level goal may be), the emitted pattern must first be quantified. This quantification requires not only accurate measurements of each and every packet that is transmitted, but accurate control over the transmission (both in terms of quantity and characteristics) of those packets. Achieving control over the packet measurement entails observing and measuring every packet that can be heard. Control over the packet transmission entails 'turning off' features of the network protocol stack, such as link-layer retransmissions. Our testing has shown that the effects of reflections on results in wireless networking can be significant; such reflections should be considered in the face of results that display large degrees of variation. Appropriate testing locations should be chosen in order to minimize the occurrence of such reflections.

Quantification of the beam pattern also requires the internal state of the antenna to remain static; the pattern rotation tests showed that even though the rotated patterns were very similar, the variation between them was large enough that it was unknown whether any unexpected results should be attributed to that variation or to some other unknown factor.

Finally, theoretical equalities may not be represented in reality. For example,

the receive beam pattern was shown to be much more accurate than the transmit beam pattern, even though both patterns should theoretically have been identical. Whether the difference in the observed patterns was in fact due to variation in the Phocus Array, or an artifact of the testing methodology is unknown.

By the end of the experiments shown, the receive beam pattern had been verified. Thus the high-level goal of using beam-forming antennas can be addressed. Theoretically, nodes at the main lobe in the beam pattern will achieve better throughput than nodes at the minimal areas of the beam pattern. In the extreme case, nodes at the main lobe will be able to achieve connectivity to the Phocus Array, while nodes at the minimal areas will not. This extreme case will result in spatial reuse being effectively demonstrated, under the assumption that destructive interference will not occur when connectivity can not be achieved. In an attempt to prove that spatial reuse is possible, we have conducted experiments that place clients nodes at the strong and weak lobes, and measured the throughput at each; we have varied the transmit powers of the clients, as well as their distances and angles to the Phocus Arrays. However, to date our experiments have not succeeded: the throughput of multiple co-located links undergo degradation when both data is transmitted along both links simultaneously. However, we remain optimistic that spatial reuse is attainable with the Phocus Arrays. Furthermore, the Phocus Arrays seem to perform well in other applications, such as line of bearing determination; this ability is described in the epilogue.

5.2 Epilogue: Directionality

Our current results show that the Phocus Arrays are able to effectively achieve a directional pattern. Specifically, they are able to emit a beam pattern that has a clearly observable main lobe. As discussed, the Phocus Arrays operate in one of two modes, static or dynamic. Dynamic mode allows a Phocus Array to determine the line of bearing to a client node; this line of bearing algorithm has been built in to the Phocus Arrays. We believe that there is a correlation between the fact that one of the Phocus Array's main abilities is to emit a well defined main lobe, and the fact that the Phocus Array includes an alogorithm for line of bearing determination; this correlation exists because the more pronounced the emitted main lobe is, the more effective the Phocus Array will be at determining the direction at which a client lies. We believe that since the line of bearing determination algorithm has been built into the Phocus Array system, and the most effective area of the beam that it forms is the main lobe, that this may be the main application for which the Phocus Array is intended to be used. With this in mind, it would be useful to observe how effective the Phocus Arrays are at determining the line of bearing at which a client node lies; here we describe experiments that show how effective this ability is.

In dynamic mode, the Phocus Array performs sweeps, measuring the RSSI of packets from each client in each directional pattern. On the Phocus Array filesystem exists a file called station_stats.xml. In dynamic mode, for each associ-

ated client the Phocus Array stores a mapping of the RSSI of packets from that client for each directional pattern that is loaded. This mapping is stored in the station_stats.xml file. In each directional pattern the main lobe is in a known direction. If the receive beam pattern behaves as assumed, then the packet from the client with the strongest RSSI will be heard when the main lobe is pointing closest to that client. This observation leads to an application whereby the direction of a client can be determined when the Phocus Array is operating in dynamic mode, within a 22.5° granularity. Experiments can be performed to test this ability by placing client nodes at known angles to the Phocus Array, and observing the angle at which the Phocus Array reports it determines the client to be. This test is performed in the outdoor mall environment (since the indoor environment is too small). Consider the mall as a cartesian graph, with (0,0) at the center of the mall, and units in feet. In this test, two Soekris boxes S1 and S2 are placed at (-37.5,0) and (37.5,0) respectively; i.e. they are equidistant (37.5 feet in either direction) from an imaginary linethat bisexts the mall vertically, and both lie on an imaginary line that bisects the mall horizontally (the orientations of horizontal and vertical are arbitrary, as long as there is enough space for all co-ordinates in the test). The Phocus Array is placed at the location (0,0), i.e. at the intersection between the horizontal and vertical lines of bisection. Thus the angle at which S1 lies is 270°, and the angle at which S2 lies is 90°. The Phocus Array is placed in dynamic mode, and performs the sweep around the directional patterns. The line

of bearing is determined, and recorded. The reported line of bearing is recorded 12 times (since the sweep takes five seconds, this yields a one minute test). Figure 5.1 shows the results of this test. The red lines are the reported lines of bearing; the length of the impulses from the center of the graph indicate how many times that angle is reported as the line of bearing. The thin blue lines are the actual physical angles at which the Soekris boxes lie to the Phocus Array. The purple lines are the vector sums of the different component lines for each Soekris box. For example there are two red lines, a longer one at 270°, and a much shorter one at 292.5°. The vector sums of these two lines (taken as vectors) is a long purple line that is almost (but not quite) at 270°. The purple lines can be thought of as the average angle reported. As the figure shows, the determined lines of bearing almost precisely coincide with the actual angle at which the Soekries boxes lie; the line of bearing algorithm in this case seems to perform very well.

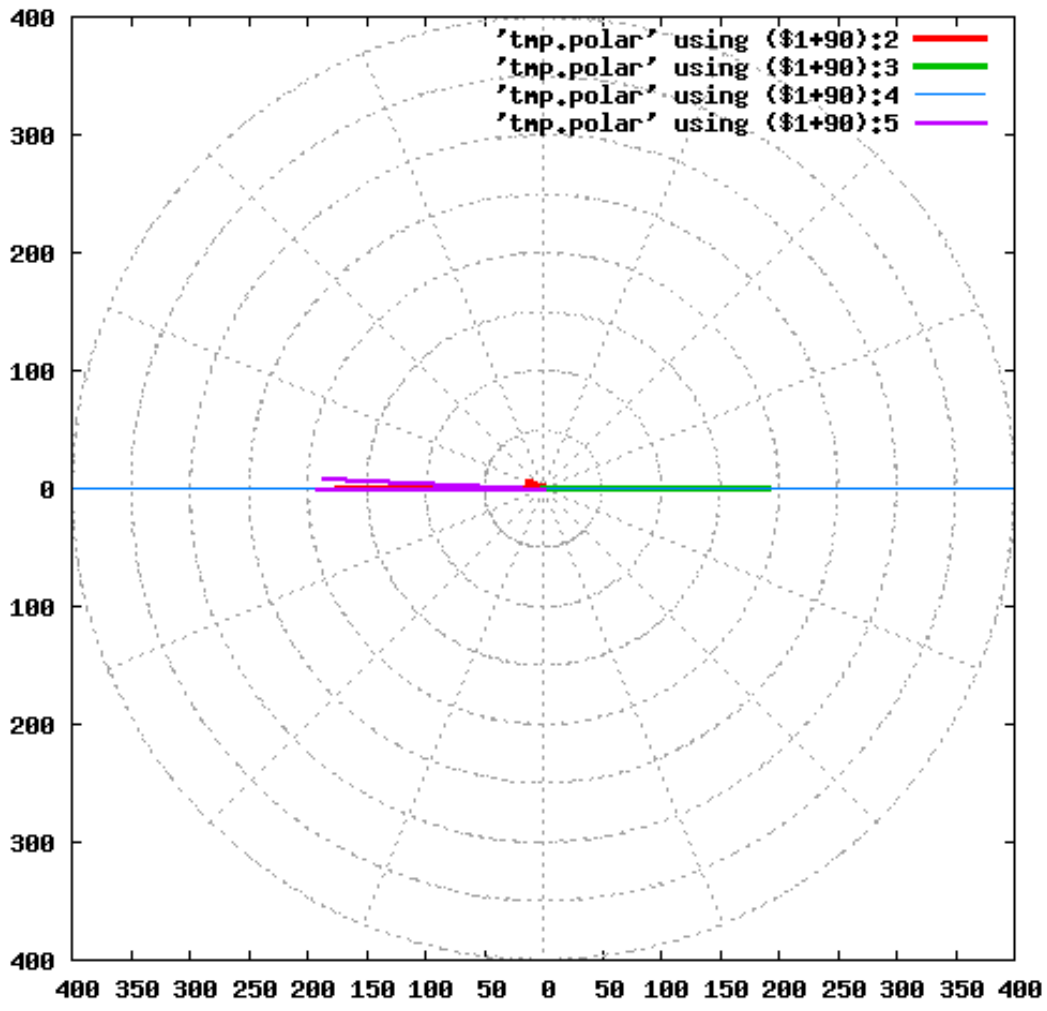
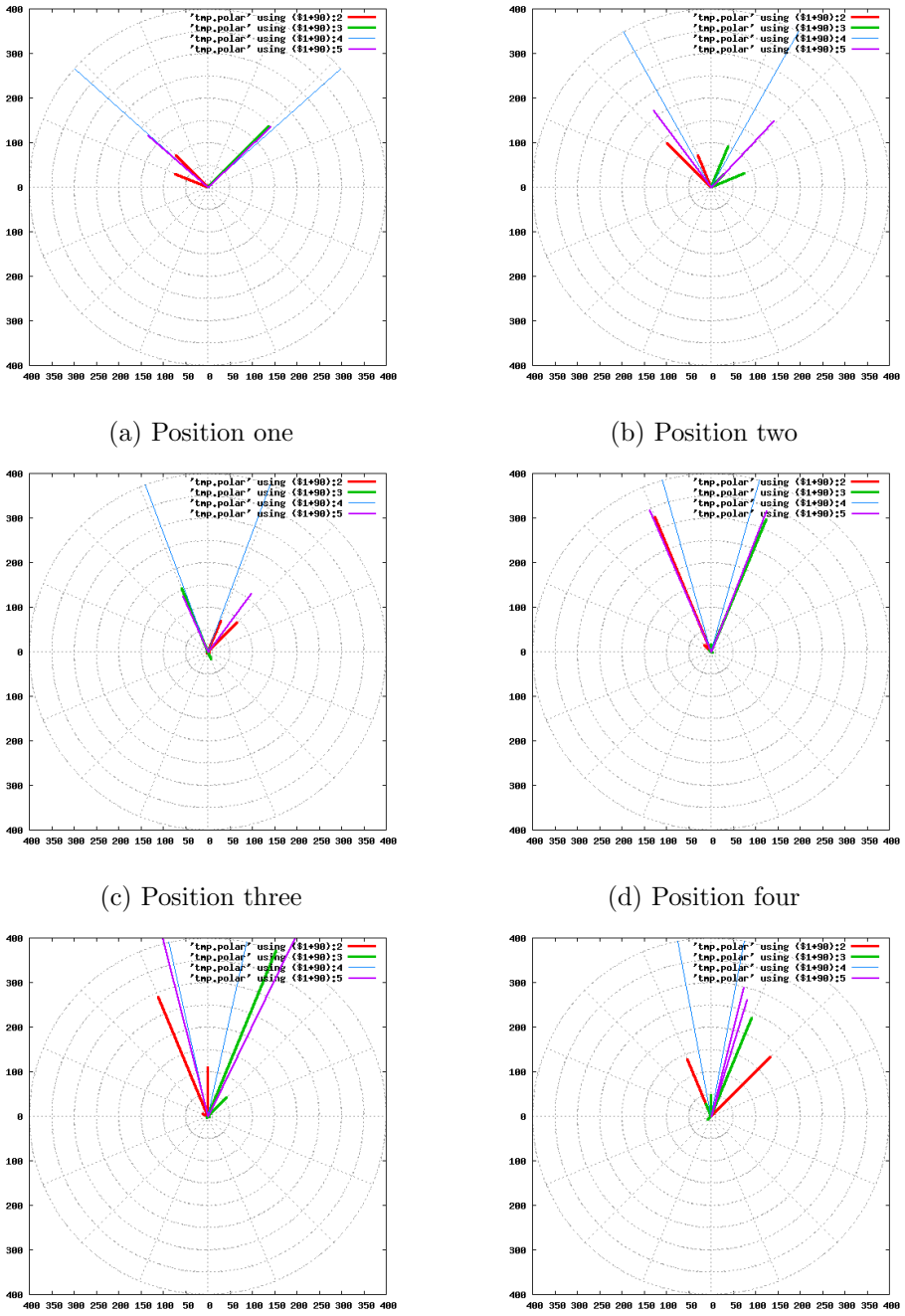


Figure 5.1: Polar plot during a directionality test showing the determined line of bearing with the Phocus Array at position zero

This test is repeated for various positions of the Phocus Array; each position lies on the vertical bisection of the mall, but at a different distance from the horizontal bisection. Each position therefore yields a different physical angle to the Soekris boxes. The Phocus Array is placed at (0,28), (0,56), (0,84), (0,112),

(0,140), (0,168) (in cartesian co-ordinates, with the units correlating to feet in physical distance). The results are shown as Figures 5.2a, 5.2b, 5.2c, 5.2d, 5.2e and 5.2f. In each case, the thin blue line represents the actual angle at which the Soekris boxes lie; the purple line is the average angle that is reported by the Phocus Array. The angles become more acute as the Phocus Array moves forward. The reported line of bearing is generally correct, however as shown by Figure 5.2f when the angle between the clients becomes very acute, the determined line of bearing may differ quite significantly from the actual angle (in contrast with Figure 5.1 in which the angle between the clients is very wide, and the determined line of bearing is very accurate). These results seem to support our theory that the Phocus Arrays have been designed to be used mainly in this dynamic mode, with their intended application being to be used to determine directionality.



(a) Position one

(b) Position two

(c) Position three

(d) Position four

(e) Position five

(f) Position six

Figure 5.2: Polar plot during a directionality test showing the determined line of bearing from the Phocus Array

Bibliography

[1] Atheros communications. http://www.atheros.com/.

[2] Busybox. http://www.busybox.net/.

[3] Fidelity comtech. http://www.fidelity-comtech.com.

[4] Google maps. http://maps.google.com/.

[5] Intel xscale. http://www.intel.com/design/intelxscale/.

[6] Iperf. http://sourceforge.net/projects/iperf.

[7] Madwifi. http://www.madwifi.org/.

[8] Openwrt. http://openwrt.org/.

[9] Rude and crude. http://rude.sourceforge.net/.

[10] Soekris engineering. http://soekris.com/.

[11] C. Balanis. Smart antennas for future reconfigurable wireless communication networks. *Wireless Communication Technology, 2003. IEEE Topical Conference on*, pages 181–182, Oct. 2003.

[12] M. Blanco, R. Kokku, K. Ramachandran, S. Rangarajan, and K. Sundaresan. On the effectiveness of switched beam antennas in indoor environments. In M. Claypool and S. Uhlig, editors, *PAM*, volume 4979 of *Lecture Notes in Computer Science*, pages 122–131. Springer, 2008.

[13] J. Carey and D. Grunwald. Enhancing wlan security with smart antennas: a physical layer response for information assurance. *Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th*, 1:318–320 Vol. 1, 26-29 Sept. 2004.

[14] R. Choudhury, X. Yang, R. Ramanathan, and N. Vaidya. On designing mac protocols for wireless networks using directional antennas. *Mobile Computing, IEEE Transactions on*, 5(5):477–491, May 2006.

[15] R. R. Choudhury and N. H. Vaidya. Deafness: A mac problem in ad hoc networks when using directional antennas. *icnp*, 00:283–292, 2004.

[16] R. R. Choudhury, X. Yang, R. Ramanathan, and N. H. Vaidya. Using directional antennas for medium access control in ad hoc networks. In *MobiCom '02: Proceedings of the 8th annual international conference on Mobile computing and networking*, pages 59–70, New York, NY, USA, 2002. ACM.

[17] P. Gupta and P. Kumar. The capacity of wireless networks. *Information Theory, IEEE Transactions on*, 46(2):388–404, Mar 2000.

[18] Z. Huang and C.-C. Shen. A comparison study of omnidirectional and directional mac protocols for ad hoc networks. *Global Telecommunications Conference, 2002. GLOBECOM '02. IEEE*, 1:57–61 vol.1, Nov. 2002.

[19] Y.-B. Ko, V. Shankarkumar, and N. Vaidya. Medium access control protocols using directional antennas in ad hoc networks. *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, 1:13–21 vol.1, 2000.

[20] V. Navda, A. P. Subramanian, K. Dhanasekaran, A. Timm-Giel, and S. Das. Mobisteer: using steerable beam directional antenna for vehicular network access. In *MobiSys '07: Proceedings of the 5th international conference on Mobile systems, applications and services*, pages 192–205, New York, NY, USA, 2007. ACM.

[21] M. Neufeld and D. Grunwald. Using phase array antennas with the 802.11 mac protocol. *Broadband Networks, 2004. BroadNets 2004. Proceedings. First International Conference on*, pages 733–735, 25-29 Oct. 2004.

[22] D. Niculescu and B. Nath. Vor base stations for indoor 802.11 positioning. In *MobiCom '04: Proceedings of the 10th annual international conference on Mobile computing and networking*, pages 58–69, New York, NY, USA, 2004. ACM.

[23] J.-S. Park, A. Nandan, M. Gerla, and H. Lee. Space-mac: enabling spatial reuse using mimo channel-aware mac. *Communications, 2005. ICC 2005. 2005 IEEE International Conference on*, 5:3642–3646 Vol. 5, May 2005.

[24] J. B. R. R. Choudhury, T. Ueda and N. Vaidya. Beamnet: Ad hoc networking testbed using beamforming antennas, 2005.

[25] B. Raman and K. Chebrolu. Design and evaluation of a new mac protocol for long-distance 802.11 mesh networks. In *MobiCom '05: Proceedings of the 11th annual international conference on Mobile computing and networking*, pages 156–169, New York, NY, USA, 2005. ACM.

[26] R. Ramanathan. On the performance of ad hoc networks with beamforming antennas, 2001.

[27] R. Ramanathan, J. Redi, C. Santivanez, D. Wiggins, and S. Polit. Ad hoc networking with directional antennas: a complete system solution. *Selected Areas in Communications, IEEE Journal on*, 23(3):496–506, March 2005.

[28] K. Sayrafian-Pour and D. Kaspar. Source-assisted direction estimation inside buildings. *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, pages 1–8, April 2006.

[29] R. Vilzmann, C. Bettstetter, D. Medina, and C. Hartmann. Hop distances and flooding in wireless multihop networks with randomized beamforming. In *MSWiM '05: Proceedings of the 8th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems*, pages 20–27, New York, NY, USA, 2005. ACM.

[30] J. Winters and M. Gans. The range increase of adaptive versus phased arrays in mobile radio systems. *Vehicular Technology, IEEE Transactions on*, 48(2):353–362, Mar 1999.

[31] S. Yi, Y. Pei, and S. Kalyanaraman. On the capacity improvement of ad hoc wireless networks using directional antennas. In *MobiHoc '03: Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*, pages 108–116, New York, NY, USA, 2003. ACM.

[32] C. Zhu, T. Nadeem, and J. Agre. Enhancing 802.11 wireless networks with directional antenna and multiple receivers. *Military Communications Conference, 2006. MILCOM 2006*, pages 1–8, Oct. 2006.