A Cooperative Communication Protocol for Wireless Ad-hoc Networks

by

Jeremy I. Silber

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degrees of

Bachelor of Science in Computer Science and Engineering

and

Master of Engineering in Electrical Engineering and Computer Science

at the Massachusetts Institute of Technology

June 2002

© Jeremy I. Silber, MMII. All rights reserved.

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
JUL 2 0 2004
LIBRARIES

The author hereby grants to MIT permission to reproduce and distribute publicly paper and electronic copies of this thesis document in whole or in part.

Author ....
Department of Electrical Engineering and Computer Science
May 10, 2002

Certified by ....
Andrew Lippman
Senior Research Scientist, MIT Media Laboratory
Thesis Supervisor

Accepted by ....
Arthur C. Smith
Chairman, Department Committee on Graduate Theses

ARCHIVES

A Cooperative Communication Protocol for Wireless Ad-hoc Networks

by

Jeremy I. Silber

Submitted to the Department of Electrical Engineering and Computer Science
on May 23, 2002, in partial fulfillment of the
requirements for the degrees of
Bachelor of Science in Computer Science and Engineering
and
Master of Engineering in Electrical Engineering and Computer Science

Abstract

This thesis presents the design and implementation of a communication protocol that utilizes local cooperation among nodes to efficiently transfer data. Multi-hop routing in ad-hoc wireless networks realizes some scalability benefits over direct transmission by utilizing cooperation in the network layer, where all nodes act as routers to relay messages. Cooperative transmission takes this idea a step further, moving cooperation to the link layer, where nodes actually broadcast signals simultaneously to increase signal strength. Using network topology information derived from propagation delay measurement, nodes dynamically establish and update membership in rebroadcasting cells. Rebroadcast cells use constructively interfering modulation schemes to broadcast radio signals together, directing an amplified signal toward the intended recipient. This results in a link-layer routing system well suited to real-time data streaming in mobile, ad-hoc, wireless networks.

Thesis Supervisor: Andrew Lippman
Title: Senior Research Scientist, Program in Media Arts and Sciences

This work was supported by the MIT Media Lab Digital Life Consortium and Intel Corporation.

Acknowledgments

I thank my thesis advisor, Andy Lippman, for offering me direction and guidance. He suggested the topic of a link-layer routing scheme, and gave me the necessary background in information theory and radio communication to approach the problem. He was quick to offer solutions when I worked my way into a corner, and constantly prompted me to focus my efforts on "The real problem."

I also owe many thanks to Joshua Lifton, who not only designed the Pushpin hardware architecture used in the infrared network, but provided assembled boards, taught me how to start programming them, and answered every question I came up with along the way.

Ilia Mirkin was largely responsible for the development of the serial gateways to the infrared cooperative network. He handled most of the hardware and the software modifications necessary for the gateway nodes, and was a great help in debugging the infrared network. Casey Muller assisted by creating visualization software for demonstrations of the infrared network's functionality.

Aggelos Bletsas was kind enough to discuss network time synchronization and radio transmission details with me, and Dimitris Vyzovitis provided some insights on network routing.

Contents

List of Figures

Chapter 1

Introduction

The value of a communication network derives from the connectivity it provides between nodes. The cost of supporting networked connectivity is also a function of the number of nodes, with availability of communication bandwidth constraining the total communication capacity achievable. Modern communication networks have sub-linear scalability: the capacity of the network may rise with the number of nodes, but the average communication capacity *per node* decreases. Thus network capacity must be viewed as a scarce resource; one node's communication decreases the capacity available for the others. However, no concrete evidence exists to prove that this must be the case - it may be possible to construct networks which scale linearly, where adding an additional node creates as much (or more) capacity than that node consumes.

If such a network is possible, it will likely require that we rethink the simplifying assumptions on which modern networks are based. For example, we should question the notion that different signals must be separated and sent from one transmitter to one receiver (via link-layer non-interference protocols). Some evidence seems to indicate that utilizing local interactions between nearby nodes may be the key to realizing higher-order scalability. This beneficial local interaction is the *cooperation* upon which the presented networking concepts are based.

A second questionable assumption of modern networking systems is that functionality should be separated into layers (i.e. the OSI seven-layer model for networking). These layers are separated by abstraction barriers designed to simplify the overall system design by minimizing interactions, allowing individual layers to act as independent components. Unfortunately, modular layers with standard interfaces often force assumptions on the high-level layers, precluding innovative protocols customized to the nature of the underlying physical network. Reliance on link-level noninterference protocols is one example; information hiding between layers is another. Signal strength or timing information from the link layer may be usefully employed in the network layer, but abstraction barriers typically hide this information from outside access.

1.1 Communication Network Scalability

Similar scaling problems arise in various types of communication networks. This section presents several examples, and how cooperative interactions between nearby nodes can improve network scalability.

In the case of Internet multicast, scaling problems arise as a result of error correction dialog. The more recipients of a multicasted data stream, the greater the likelihood of a reception error. Reception errors force the receiver to request a repeat of the missing data, and error correction traffic can quickly consume all available bandwidth, forcing actual data traffic to a halt. There have been a number of attempts to ameliorate this situation, creating scalable, reliable multicast (SRM)[9]. Floyd et al. showed that local error repair – information sharing among receivers to identify and correct errors – alleviates the scaling problem. By removing error correction responsibility from the single sender and moving it to the large set of receivers, they assure that adding nodes to the network (which increases the probable number of reception errors) simultaneously adds error correction capacity.

Radio networks also utilize one-to-many broadcast techniques, and have similar scalability problems. In this case the bottleneck is available radio bandwidth. Shannon's early work on information theory [10] established a limit on the data capacity of the wireless channel between two radios as a function of bandwidth and signal-to-noise ratio. In networks built on these point-to-point links, this bound becomes a bottleneck. Gupta and Kumar [11] show that network capacity scales with the square root of the number of nodes if all communications are simple point-to-point connections under a non-interference protocol. Thus the throughput per node decreases as the network scales. Again, cooperative interactions may offer a solution: no tight bound has been found for the communication capacity of a group of n radios communicating among each other (with multiple senders and/or multiple receivers) in a geographic region. Efforts to use multipath channels to multiplex signals suggest that the square root scaling may not be optimal. If a receiver can correctly receive signals from more than one node at a time, networks may be able to exceed a square root scaling rule[19].

The cooperative approach to the scaling problem is gaining ground in the guise of peer-to-peer overlay networks such as Napster[5], Gnutella[1], Morpheus[3], Kazaa[2], and MFTP[4]. The key to the scalability of these networks is that every node provides bandwidth and storage resources, and local interactions (file transfer directly from one peer to another) replace the conventional client-server architecture.

In the case of wireless ad-hoc networks, a similar scaling problem arises due to the scarcity of bandwidth. Radio transmissions are (generally) omnidirectional, so a signal will often be received by many nodes other than the intended receiver. All nodes receiving this signal are effectively unable to receive other signals (in the same bandwidth), and this limits the communication capacity of the network. The more nodes added to the network, the further the average message must travel, using more bandwidth resources. Additionally, each node needs to know more about the network to correctly route messages destined for an increasingly large set of intended receivers. When ad-hoc wireless networks grow large, both the scalability of the routing scheme (e.g. size of routing tables) and the scalability of bandwidth use become problematic.

1.2 Contribution of this Work

This work attempts to bring the concept of cooperation to bear on the problem of scaling in wireless ad-hoc networks. Presented is a method for cooperative routing of data without decoding signals in transit. The goal of this approach is to produce a scalable network which implements routing based solely on signal propagation information measured in the link layer. The result is a network which dynamically routes individual bits without excessive overhead and with minimal latency. Secondary goals include reduction of transmission power and efficient utilization of bandwidth resources.

This work is motivated by the technological and economic gains possible when network nodes serve to increase network capacity. Replacing existing networks that rely on fixed wired stations (e.g. cellular telephony) with cooperative networks could allow systems where connectivity and capacity are created by the mobile nodes of the network. Rather than approximating future peak capacity and constructing fixed based stations to provide it, the infrastructure for capacity would be resident in the mobile nodes themselves, providing capacity exactly when and where it is needed.

1.3 Thesis Overview

Chapter 2 presents the current state of wireless networks, ad-hoc networking, and theoretical scalability bounds. The references presented provide context for this work.

Chapter 3 presents a design for a cooperative communication protocol, built explicitly for scalable real-time data transfer in mobile ad-hoc wireless networks. The protocol emphasizes cooperation at the lowest level, using a collection of nearby nodes (a *rebroadcast cell*) to broadcast a signal simultaneously. To determine membership of the rebroadcast cell, the protocol uses link-layer measures of signal propagation delay as estimates of network topology.

Chapter 4 validates the components of the presented cooperative communication protocol (multiple-node broadcast and propagation delay estimations of network topology) with a network implemented in hardware using infrared as a communication medium. The design of this network's modulation scheme, channel access, time synchronization, and topology measurement are presented.

After showing the feasibility of a network built around the concept of cooperation, Chapter 5 presents analysis of such a networks' performance. Qualitative findings on the infrared simulation network are presented, along with derivations of latency and bandwidth efficiency of cooperative networks (as compared to packet-switched multihop networks). Finally, a software simulation package is presented for the calculation of transmission power. The quantitative simulation results form an estimate of the power use of various networking schemes and how this power use scales with network size.

Finally, Chapter 6 reviews the findings of this work and suggests future avenues of research.

Chapter 2

Background

This thesis draws from past research in radio, network routing algorithms, and theoretical work in scalability. This chapter presents relevant work in radio (cellular phone networks, wireless data, television), ad-hoc network routing, and the scalability of networks.

2.1 Wireless Networks

Cellular phone networks are a good canonical example of the single-hop network upon which this system hopes to improve. Single-hop networks transfer data direction from a sender to a receiver, without utilizing any other nodes of the network for routing purposes. Most cellular calls today use second generation ("2G") cellular technology, employing digital circuit-switched routing. Bandwidth is shared among phones in an area using one of several access protocols: typically Time Division Multiple Access (TDMA), Frequency Division Multiple Access(FDMA), and Code Division Multiple Access (CDMA)[14]. A common assumption between these protocols is that individual nodes are adversaries, competing for an allocation of bandwidth.

Wireless local area data networking is currently dominated by the 802.11 family of standards (802.11a, 802.11b, 802.11g)[13]. In the developing field of Personal Area Networking, Bluetooth[12] has achieved the greatest amount of attention. Neither currently provides any multi-hop routing capabilities: they do not utilize other network nodes to assist in the transmission of data from sender to receiver. 802.11 network configurations are typified by a number of mobile devices and a set of fixed base stations. Unlike cellular phone networks, a mobile node does not keep a static identifier as it moves within range of different base stations. While 802.11 does specify an "ad-hoc mode" in which two mobile nodes can communicate directly, it does not provide any support for larger ad-hoc networks (which would necessitate multi-hop routing and non-hierarchical addressing). Bluetooth specializes in creating a small local network ("piconet") of devices with one master and up to seven slave devices. Piconets can theoretically be hierarchically connected into a larger "scatternet" which would use multi-hop routing, but many technical details of scatternets remain unresolved and scatternets have not yet been implemented.

Television broadcasting requires careful attention to multipath delayed signal propagation. Analog television broadcasts are subject to 'ghosting', a visible artifact of multipath signal distortion. When a television signal reaches a receiver via several different paths (e.g. reflections off city buildings or reception from two different transmitters), the two signals can arrive at different times and interfere with each other. Coded Orthogonal Frequency Domain Multiplexing (COFDM)[22] mitigates this problem by multiplexing a signal over many independent frequencies, with the symbol time (the duration of an individual signal) in each frequency long enough to assure that multipath effects reinforce the current symbol rather than interfere with subsequent ones.

BLAST[23], a research project at Bell Labs, utilizes the nonlinear properties of radio propagation and multiple transmitting and receiving antennas to increase the amount of information that can be broadcast in a given frequency bandwidth. It utilizes the multipath effect – the result of a radio signal arriving both directly from the transmitter and after reflecting off distant surfaces – to send multiple channels of information at the same carrier frequency.

2.2 Ad-hoc Mobile Routing

Drastic decreases in price of wireless networking hardware has prompted considerable research into ad-hoc mobile routing (directing network traffic through a non-hierarchical network of mobile and potentially unreliable nodes). Targeted uses range from eliminating wires between electronic devices to large-scale sensor networks. The various proposed algorithms differ in their weighting of bandwidth, reliability, node mobility, and scalability as design goals.

The routing algorithms most commonly used on the wired Internet are based on the periodic propagation of local link-state or distance-vector information. Such schemes do not translate well to the wireless world, where nodes enter and exit the network without warning, and move constantly, making and breaking connections too frequently for global information to be updated. Mobile ad-hoc networks often use on-demand route discovery [15, 6] to avoid the overhead of maintaining unused routes.

Hyphos[18] is an ad-hoc routing protocol that employs "contour routing" to move messages successively closer to the recipient (where distance is measured in number of hops). Each Hyphos node maintains a table of distance estimates to other nodes and relays a message only if its distance to the intended recipient is less than the sender's. Every relayed message carries a hop counter that is used to update relaying nodes' distance estimate to the sender. The Hyphos system is designed to inexpensively connect tens or hundreds of nodes with very simple hardware. However, the protocols are inefficient when nodes are in rapid motion or the membership of the network changes rapidly. Like all packet-switched network protocols, each hop over which a message is relayed introduces latency (delay between data transmission and reception) that is undesirable in real-time communications.

Some other ad-hoc routing systems have used information from the link layer to inform network-layer routing decisions. The Signal Stability based Adaptive Routing (SSA)[6] system uses signal strength and consistency information from the link layer to choose routes which will likely require less maintenance as nodes move, enter, or exit the network. SSA does not attempt to derive any physical or topological distance information from the link layer. [7] uses end-to-end propagation delay calculated in the network layer to derive the topology of a multicast network.

2.3 Scalability of Wireless Networks

The question of the total communication capacity of a wireless network is currently open. Cellularization using wired base stations linearizes the communication capacity of a network, but only by setting up an expensive, static network of base stations connected to each other via an external wired network. Gupta [11] and Shepard [21] show that mobile wireless networks with variable-power transceivers operating under a non-interference link-layer protocol can achieve a total capacity that scales with the square root of the number of network nodes. Space-time coding techniques like BLAST [23] may be capable of achieving total network capacity scaling linear with the number of nodes [19].

Recent work [21] also suggests that the mobility of the network nodes can be utilized to improve the network's scalability, but this effect only comes into play when acceptable delivery latencies are large in comparison to the relative velocities of network nodes.

Chapter 3

Cooperative RF

This chapter presents a framework for utilizing cooperation among radio transceiver nodes to propagate data across a wireless ad-hoc network. Simple cooperation rules create emergent network properties that are used to design a full-featured dynamic routing algorithm entirely in the radio frequency domain.

3.1 Goals

The proposed protocol is designed to operate in a dynamic network of mobile wireless nodes, replacing point-to-point protocols and multi-hop networking protocols built on point-to-point link layers. The protocol described is designed to support real-time, bandwidth-intensive applications (such as telephony). Several properties are desirable for such applications:

1. Latency - Telephony, video-conferencing, and other highly interactive applications require minimal *latency* (the delay between the sending of a bit and its reception at the other end of the channel). The Internet's use of packet-queueing has created problems for this class of application.

2. Availability - The system should be capable of routing data around any nodes that have moved or failed, and should ideally be capable of routing data through

multiple paths simultaneously to ensure reliable delivery or to increase signal strength.

3. Duplex Connectivity - Telephony data flows in both directions between the two endpoints, and requires high bandwidth in both directions.

4. Scalability - The system should be designed to handle increased network traffic as the number of nodes in an area increases. Routing should distribute load when possible to avoid bottlenecks constraining network capacity.

5. Power Efficiency - The system should attempt to route data in such a way as to minimize necessary aggregate and per-node broadcast power.

6. Bandwidth Efficiency - The system should use as little bandwidth as possible for as little time as possible in as small an area as possible under normal use cases.

Conventional multi-hop systems work reasonably well in achieving availability, scalability, and power efficiency with duplex connections. Latency and bandwidth efficiency, however, are problematic. Moreover, existing multi-hop systems require additional (transmission and computation) overhead for packet address headers, and many routing schemes require prohibitively large tables of network state information that must be updated whenever nodes move.

3.2 Design Overview

The proposed protocol attempts to meet the stated design goals through cooperation between network nodes, merging the availability, scalability, and power efficiency of a multi-hop network with the low latency and computation overhead of a direct point-to-point link. The protocol moves cooperation from the network-layer (where nodes relay data packets for other nodes) to the link layer, so that nodes repeat amplified signals for other nodes. The protocol selects a minimal set of nodes between a sender and a receiver as members of a "repeater cell," forming a pathway for data transfer.

To do repeater cell membership selection without any externally provided network state information, the protocol extrapolates network topology from observed signal propagation delays.

The proposed protocol also demonstrates a reason to break the conventional abstraction barrier between the network's physical propagation (link) layer and its network routing system. Removing the abstraction barrier allows us to tailor the routing system to the physical medium, and to use physical propagation measurements (e.g. timed delays) in the routing algorithms. The many-to-many aspect of radio transmission is utilized by the protocol's multi-path routing instead of being hidden under a one-to-one non-interference access layer. The simultaneous reception of two signals should not create a "collision" that destroys the data in transit (as in the one-to-one abstraction). Simultaneous reception should instead be used to increase signal strength. In addition to customizing the routing for the physical medium, the protocol should share information across the conventional abstraction barrier between link and network layers. This allows the network layer to select rebroadcast cells using topology derived from link layer propagation delay measurements.

3.2.1 Communication Channel Properties

The system is designed to use radio as the transmission medium, but is readily adaptable to any medium that can be efficiently multiplexed in a dimension other than time (like space or frequency). This includes light, radio, ultrasound, sonar, and capacitive coupling. Each node is assumed to be capable of localized omnidirectional broadcasting and omnidirectional reception.

3.2.2 Modulation

Rebroadcasting of radio symbols by a set of repeater nodes introduces many copies of the original signal, which may arrive at a receiver at different times. These signals also may travel through different transmission channels with different fading, multipath, noise, and distortion properties. The modulation technique must be robust to these effects, which cause recombination of potentially distorted copies of a signal with an arbitrary (but bounded) phase offset. Section 3.3 presents one suitable modulation scheme and explains how it achieves these properties.

3.2.3 Control Channel

Most on-demand wireless communication protocols feature one or more control channels for the allocation of bandwidth to network nodes. The proposed system should feature such a control channel for bandwidth allocation, but proposes no special advances in control channel protocol. The control channel is assumed to assign two unoccupied communication channels (e.g. frequency bands or time slots) to two network nodes seeking to exchange data. One of these channels will be used for data traveling in each direction. The algorithms presented specify the communications which occur in these channels after allocation.

3.2.4 Network Time Synchronization

The presented cooperative communication protocol utilizes a synchronized clock at each node to extrapolate network topology from received signals. Achieving a common synchronized clock, though difficult on the Internet due to packet queueing at each stop on the routing path, is much simpler given nodes communicating directly with each other over radio. [17, 8] discuss possible time synchronization methods and their accuracy. Clock synchronization must be maintained with an error less than the length of the guard period (see Section 3.3), and synchronization accuracy affects the efficiency of repeater cell membership selection.

3.3 Constructive Interference

The cooperative communication protocol relies on the reliable reception of a radio symbol when copies arrive from multiple transmitters at slightly different times. The signal-to-noise ratio measured at a receiver should increase with the number of nearby transmitting nodes. This property of *constructive interference* is one of the key underpinnings of the protocol, allowing nodes to cooperate in the transmission of radio signals.

Constructive interference can be achieved with signals designed such that several copies combine into a single, more easily demodulated signal. While the simplest of modulation techniques might seem to exhibit this property (e.g. an amplitude modulated sinusoidal carrier wave), the possibility of fading and compositional phase delay complicate matters. Fading occurs when a transmission channel (the environment between transmitter and receiver) attenuates some signals more than others. Differential propagation delay occurs when a signal travels over multiple paths of different lengths from transmitter to receiver, such that each copy arrives at a slightly different time. Most types of signal modulation are not robust to recombination with fading and delay: if two simple AM signals are combined with a relative delay (i.e. phase difference) of half their frequency, the carriers cancel each other out and the signal is lost.

Coded Orthogonal Frequency Domain Multiplexing (COFDM) is an example modulation technique that exhibits constructive interference. COFDM multiplexes a data stream over many modulated carriers at different (orthogonal) frequencies. This frequency diversity and the use of diversity coding makes COFDM resistant to phase cancellation and fading: signal losses due to these effects occur only on some of the carrier frequencies, and diversity coding allows the data stream to be correctly received despite several missing components.

COFDM is able to correctly decode a symbol even when many copies of it arrive at slightly different times. Each symbol is broadcast for t_{symbol} seconds, and symbols are separated by a quiescent guard period of t_{guard} seconds. The guard period provides a buffer, preventing late-arriving copies of a symbol from interfering with the next symbol in the stream. As long as all copies of a symbol arrive within t_{guard} seconds of the symbol start time, they will finish before the next symbol period begins. (t_{symbol}) is relatively long, so that delayed copies of the symbol will reinforce the copies that arrive earlier, increasing the signal to noise ratio. In television broadcasting, this constructive interference property causes multipath delays from signal reflection to improve reception rather than causing interference (spatially displaced "ghost" images). In this cooperative communication protocol, the multi-path delays are caused by rebroadcasts, and long signal times allows rebroadcasted copies of a signal to strengthen the signal rather than destructively interfere.

The bandwidth necessary for each carrier in a COFDM system is inversely proportional to the choice of t_{symbol}, so the theoretical capacity of a frequency band is constant whether it be used by a single carrier modulated by a signal of d symbols per second, or by n different orthogonal carriers modulated by multiplexed signals of d/n symbols per second. Introducing a guard period prevents interference from delayed copies of earlier symbols, but reduces the data rate of each carrier. Since the cooperative system propagates signals over a much larger distance than most COFDM implementations, and suffers some retransmission delay at repeater nodes, guard periods between signals should be increased. Using large numbers of orthogonal frequency carriers (and correspondingly long symbol times) will minimize the bandwidth lost to extending guard delays. Section 5.4.1 analyzes the bandwidth efficiency of the cooperative protocol.

3.4 Routing

3.4.1 Extrapolating Network Topology

Efficient routing in a network requires detailed knowledge of the network's topology. With the wired Internet, this information is typically held in hierarchical routing tables that are updated regularly. Internet node addresses can encode information useful for routing (e.g. all 18.*.*.* addresses should be routed toward MIT) because nodes are stationary and their addresses reflect their positions in the network topology. Routing table updates occur relatively rarely, and changes in these tables may be slow to propagate through the entire network. Mobile ad-hoc networks present a different class of problem, since nodes are rearranged frequently (if not constantly), making static hierarchical addressing impossible. Whatever information is necessary for routing must propagate quickly to all nodes responsible for forwarding data, but routing table updates should use as little network capacity as possible.

The cooperative protocol does not explicitly propagate information on network topology through any type of routing or distance table. Instead, the nodes measure the time of arrival of incoming symbols, and compare this to the globally-synchronized clock. There are specific legal start times for symbols, and the receiving node can calculate the delay between the last legal start time and the arrival of a symbol. This delay is taken as the network propagation time of the symbol from its source to the receiving node. The network propagation time is the combination of the time taken for propagation of the signal across the physical medium and the accumulated rebroadcast delay incurred in receiving and retransmitting the signal at any intermediate nodes. The protocol uses different logical channels (e.g. different frequencies or time slots) for data originating at the different endpoints, so any intermediate node can distinguish the symbols sent from one endpoint from symbols sent by the other. Given the network propagation delays measured from each endpoint, the node can dynamically determine whether it should be a member of a rebroadcasting cell.

3.4.2 Dynamic Rebroadcast Cells

In wired packet networking, a specific node in the network receives an incoming packet, inspects the header to discover the packet's intended destination, then passes the packet along to a network node closer to the destination. For example, a switch on an Ethernet LAN often has an uplink connection to the Internet and several wired downlink connections to local computers. It analyzes incoming packets from the Internet to discover the destination address, then forwards that packet only along the downlink wire connecting to the appropriate computer.

Wireless communication changes the nature of the routing decision. A wireless node typically uses an omnidirectional local broadcast transmitter, and so the wireless routing decision is not which neighbor to pass the signal to, but simply whether to retransmit the signal or not. Conventional wireless networking builds a point-to-point layer over the broadcast radio medium by transmitting a "to:" header with the data packet. All nearby nodes receive and analyze the packet header, discarding the packet if the "to:" address does not match the node's address. Unlike the case of the Ethernet switch, all neighboring nodes receive the packet; even if they choose to ignore the incoming signal, they are unable to transmit or receive other signals on that channel due to interference. This has repercussions for the efficiency and scalability of the network: nearby nodes could be used to amplify and rebroadcast the signals they overhear, or could at least learn about network topology from measuring propagation delay.

In the cooperative system, there are no packets, and thus no "to:" addresses. Each transmission is a many-to-one broadcast of information, and the sender does not actually know which of its neighbors may be on a routing path toward the final destination. Instead, the system relies on dynamically selected "rebroadcast cells" of nodes which will amplify and rebroadcast signals on a specific channel (i.e. between two specific endpoints). Using network propagation delay measurement, nodes constantly update their distance estimates from the endpoints, joining and leaving the rebroadcast cell as appropriate.

3.4.3 Rebroadcast Cell Membership Selection

The rebroadcast cell membership selection algorithm is distributed among the intermediate nodes, and attempts to select a rebroadcast cell of nodes which are on or near a shortest path route between the two endpoints. The algorithm relies only on delay measurement (as a distance estimate), with each node using its apparent distance from the two endpoints to determine its own membership in the cell. No explicit transfer of routing tables, distance vectors, or any other network topology information is necessary.

Figure 3.4.3 lists pseudocode for the cell membership selection algorithm that

```
while (!clock_synced) {
  sync_clock()
}
delayA = 0;              // network propagation delay from A
delayB = 0;              // network propagation delay from B
while ( channel_active ) {
  if ( receiveA() )
    delayA = current_time() - last_start_time();
  else if ( receiveB() )
    delayB = current_time() - last_start_time();
  if ( delayA + delayB < t_guard - epsilon )
    rebroadcast();
}
```

Figure 3-1: Rebroadcast Cell Membership Selection Algorithm (pseudocode)

runs at each intermediate node. The effect of this code is to rebroadcast signals when the sum of the apparent network propagation delay from both endpoints is less than t_{guard}, the (fixed) length of the guard period. This ensures that, if the node rebroadcasts the symbol, there is enough time remaining for the signal to reach its destination with an accumulated delay less than t_{guard}. The constant epsilon allows a small buffer to protect against clock skew and other sources of error.

This cell membership code uses only apparent propagation delay from the endpoints, and has no explicit provision for changing broadcast cell membership according to the shortest-path distance between the two endpoints. This distance can be measured by either endpoint, and could be communicated explicitly to intermediate nodes. Such an approach would, however, require both decoding of signals at intermediate nodes and the overhead of transferring network topology information to a large set of intermediate nodes. Instead, the endpoints (which can directly measure the total propagation delay) manipulate their outgoing signals to "fake" accumulated propagation delay, achieving the desired result.

The node initiating communication acquires an empty channel, then transmits data symbols at the beginning of each symbol period (see Figure 3.4.3). The other endpoint (referred to as the "receiving" node, though it is also sending data back to

```
while (!clock_synced) {
  sync_clock()
}
acquire_channel();
while ( data_to_send ) {
  while ( !legal_start_time( current_time() ) );
  broadcast();
}
release_channel();
```

Figure 3-2: Initiating Node Transmission Algorithm

the initiating node) uses a more complex algorithm to time its responses, manipulating the intermediate nodes into creating a rebroadcast cell of appropriate size and shape. The receiving node measures the propagation delay from the initiating node, and sends the symbols in its reply with an added delay, such that intermediate nodes not near the shortest path will decline to rebroadcast the signal. This added delay serves as a time-to-live limit on symbol propagation, since intermediate nodes compare summed delays to a fixed threshold. The delay is the fixed delay threshold $t_{guard} - \epsilon$ minus the measured propagation delay from the initiating node, minus a spread constant. Subtracting the measured propagation delay gives the reply enough time to travel back to the initiating node over the reverse path (here the protocol assumes symmetric connectivity). Subtracting a spread constant allows nodes near, but not quite on, the shortest path to join the rebroadcast cell, providing multi-path routing for increased stability and signal strength.

Immediately after channel allocation, when no nodes have any information about their distance from any other nodes, this version of the algorithm will flood the local network with the first symbol, giving all nodes a delay measurement from the initiating node. The reply from the receiving node, and all subsequent symbols from either node, will be repeated only by the nodes in the rebroadcast cell. A slight modification of the initiating node's transmission algorithm could limit the flooding effect by initially inserting a delay in symbol transmission (setting a small time-to-

```
delayA = 0;
while ( channel_active && delayA != 0 ) {
  if ( receiveA() )
    delayA = current_time() - last_start_time();
  while ( !legal_start_time( current_time() ) );
  delay( t_guard - delayA - epsilon - spread_constant );
  broadcast();
}
```

Figure 3-3: "Receiving" Node Transmission Algorithm

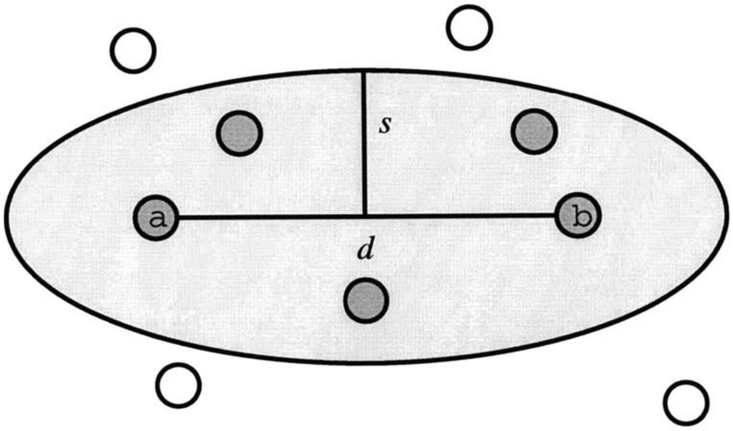

Figure 3-4: Rebroadcast Cell Shape. The shaded nodes form the rebroadcast cell for communication between nodes *a* and *b*.

live limit), then progressively reducing this delay until the signal reaches the receiving node. More efficient behavior would require an external source of network topology information, which could be implemented via side-band communication.

3.4.4 Cell Characteristics

Once data has been successfully sent from the initiating node to the receiver and vice versa, membership in the repeater cell is established. Depending on the choice of the spread constant, the repeater cell can resemble any shape from a single-path route straight between the two endpoints to a wide ellipse with the endpoints as foci.

If we assume that the propagation delay is dominated by the speed of the physical signal traveling from node to node (and not by the performance of node hardware), the rebroadcast cell will consist of the set of nodes which belong to paths near the shortest path length. For instance, if the shortest path from node A to node B is a single hop with delay d and the spread constant is s, the set of nodes on paths of delay less than $d + s$ fall in the interior of the ellipse with A and B as foci, major axis d, and eccentricity $\frac{d}{s+d}$ (See Figure 3-4).

The ellipsoid rebroadcast cell shape provides multi-path diversity, but the wider the ellipse the greater the total power use and the more nodes are in the reception area and unable to reuse the frequency band. The selection of an ideal spread constant will vary with the nature of the physical medium, transceiver hardware, and required quality of service.

Chapter 4

Simulation Network

This chapter introduces the simulation network developed in hardware to test the theories behind cooperative communication. The simulation uses both constructive interference and network topology measurements to form a cooperative communication network, using infrared as the communication medium. It demonstrates dynamic rebroadcast cell formation, and serves as a simple testbed for experimentation and modification.

4.1 Simulation Network Overview

The simulation network was designed to illustrate the concepts of constructive interference and network topology measurements, and to facilitate experimentation. While real-world networks are most useful when they connect points miles away from each other, the simulation network was built to facilitate easy visualization and manipulation of the entire network by a single observer. This requires a physically embodied simulation scaled down to table-top size. The small scale and need for a high degree of control over the environment led to several design choices, documented below.

One goal of the simulation network was to test the behavior of a cooperative communication protocol given high rates of movement of individual nodes. To allow rapid node movement on a table-top scale, the simulated network nodes are situated on custom-designed air hockey pucks which levitate just above the surface of an air-hockey table. This nearly eliminates friction, allowing the pucks to float across the table at high speeds. An experimenter can set the pucks in motion and observe how the network reacts to rapid and continuous reconfiguration.

Radio frequency (RF) communications are difficult to scale down. Radio waves act differently on a sub-meter scale than on a kilometer scale, and it is difficult to use radio at such low power. RF transceivers are bulky, complicated, and power-hungry, making individual nodes complex, expensive, and large. RF waves penetrate solid objects, making it difficult to exclude outside sources of RF interference or control the propagation of signals. For these reasons, infrared (IR) light was chosen as a communication medium rather than RF. IR transceiver hardware is simple, inexpensive, and requires little power. IR signals do not penetrate opaque objects, so outside noise can be easily excluded from the simulation environment, and barriers can be used to constrain network connectivity. IR transceivers are made to work on a scale of centimeters rather than kilometers, and prove quite adequate for table-top networking demonstrations. The purpose of the simulation network is not to validate RF-domain cooperative networking, but to demonstrate the viability of the basic concepts of constructive interference and derived network topology. Implementation of a full-featured cooperative network using radio is left as a future research topic.

4.2 Properties of Infrared Communication

Typical RF transmitters modulate data onto a carrier signal at a specific frequency, allowing the available frequency bandwidth to be split into several different communication channels modulated at different frequencies (frequency division multiplexing). The IR transmitters used, in contrast, do not use modulated carriers. They simply use switched emission of uncorrelated infrared light to send signals. The infrared receivers do not produce an analog signal strength measure; they are triggered by incident infrared light, producing a brief voltage spike, and remain saturated until the incident light ceases. These receivers serve only to detect the rising edge of an infrared pulse, providing no information on the pulse's intensity or length.

The digital nature of the IR transceivers prevents the simulation network from using frequency multiplexing schemes like COFDM, as an RF implementation might. The simulation network instead uses time division multiplexing to split the IR channel into two logical channels (one for data traveling in each direction). The behavior of these IR channels in the simulation network could be compared to that of individual RF carrier channels in a COFDM implementation.

The infrared transceivers used also differ from RF transceivers in terms of the directionality of signal propagation. While a full-scale RF system would likely use omnidirectional RF antennas, the IR transceivers used by the simulation network focus emitted infrared in a relatively narrow cone. To minimize this effect, each node has a set of 4 transceivers oriented at right angles. Even so, there are intermediate angles for which the signal propagation is significantly weaker, complicating the task of routing. With a true omnidirectional signal, signal strength is (at least to a first order approximation) simply a function of distance from the transmitter. With the directional IR transmitters used, angular position with respect to the transmitter is also a factor in the signal strength at any given point.

Additionally, the diminutive scale of the simulation network in relation to the speed of light (the propagation velocity of both IR and RF signals) changes the nature of distance measurement in the simulated network. In a full-scale system, physical distance between nodes will result in measurable propagation latencies. At table-top scale, these latencies are not easily measurable. As such, measurements of propagation time in the simulation network are dominated by the retransmit delay of intermediate nodes, not by the signal propagation time. This changes the results of the cooperative cell membership algorithm somewhat, with propagation delay representing a measure of hop count rather than physical distance.

4.3 Node Hardware and Simulation Environment

Each node in the IR simulation network is a battery-powered microprocessor with four infrared transceivers. The microprocessor is Cygnal's C8051F016 running at 22MHz, the transceivers are Agilent's HSDL-1001, and the PCB was designed by Joshua Lifton[16]. The processor and transceiver boards are attached via a connector and both are mounted on custom-designed air hockey pucks laser-cut from sheets of acrylic. The pucks are designed to float above an an air-hockey table, which pumps air through regularly spaced holes in its smooth surface. Levitated above the table, the pucks experience very little friction, allowing high-mobility testing of the network.

Complementing the mobile nodes are a pair of gateway nodes which provide RS-232 serial connections the cooperative network. Setting the gateway nodes as the endpoints of a connection allows data routing from a PC through a serial cable to the network, across the network of mobile wireless nodes, and then back through the other gateway's serial connection to a target PC. This simplifies quality of service analysis.

4.4 Bit Representation

The IR receivers used in the simulation network trigger (outputting a voltage pulse) when incident infrared light intensity crosses a threshold, and remain triggered until the intensity falls below a threshold level. As such, they can only be used to send and receive pulses, and carrier-modulated communication (in the traditional sense) is not possible.

Without carrier-modulation, the IR simulation cannot use COFDM (or any other frequency multiplexing method). Instead, it simulates the behavior of just one of the channels (others would operate at orthogonal, non-interfering frequencies, and therefore any one can be simulated independently). Nevertheless, the simulation models the most important features of the cooperative protocol: constructive interference and propagation delay as a measure of network topology.

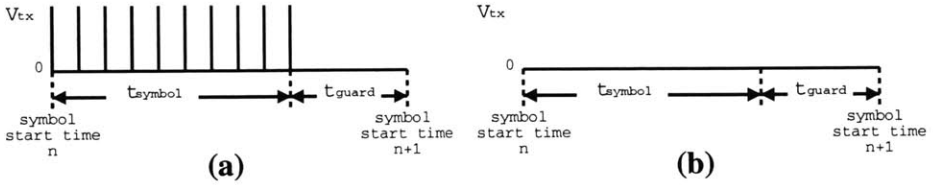

Figure 4-1: IR Bit Representations of (a) logical one, (b) logical zero.

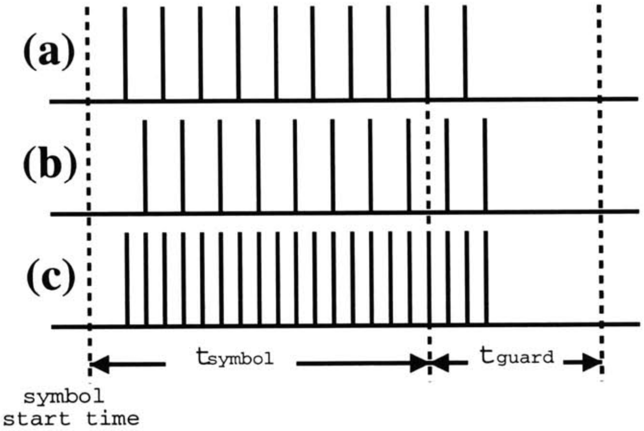

Figure 4-2: Constructive Interference of IR Bit Representations. Logical one symbols (a) and (b) arrive together (with different delays) at a receiver (c) to form a stronger logical one symbol.

The modulation scheme used, like IRDA, encodes a bit's digital value by a pulse or the absence of a pulse. Unlike IRDA, the simulation network represents each bit by more than one pulse. A logical zero is represented by no pulses, and a logical one is represented with a series of pulses (see Figure 4-1). The pulses occur for a period t_{symbol}, and are followed by a quiescent guard period lasting a duration t_{guard}. A receiving node counts incoming pulses, and considers it a logical one if the count is great than a noise threshold value. This allows nodes in the network to differentiate between a received bit and spurious IR noise, and allows delayed copies of bits to strengthen the received signal.

This bit representation parallels the symbol extension used by COFDM to enable constructive interference. The infrared receivers register only the rising edge of infrared intensity, however, so the simulation network uses a series of rapid IR pulses (extending the symbol by repetition) where COFDM simply extends the period of transmission of each carrier-modulated symbol. This bit representation results in a kind of constructive interference, since two logical one signals received concurrently (with phase delay) form a set of twice as many pulses (see Figure 4-2. Even if the number of pulses successfully received from each of the two transmitters is below the reception threshold, the number of pulses received from both transmitters together can be above the threshold. This bit representation does not exhibit constructive interference between logical zero symbols, however: two zero symbols are indistinguishable (at the receiver) from one zero symbol. This modulation technique is far from optimal, but is reasonably resistant to noise and is simple to implement and demonstrate.

The sum $t_{symbol} + t_{guard}$ determines the data rate possible over this single channel. The achieved data rate is $\frac{1}{t_{symbol}+t_{guard}}$. Longer t_{guard} settings allow longer routing paths, and longer t_{symbol} and higher noise threshold increases resistance to environmental noise. The frequency of IR pulses can be modified for a rough simulation of transmission power control.

4.5 Time Synchronization

A production cooperative communication network should use a special channel to achieve and maintain synchronous clocks. Since the IR hardware precludes the use of non-interfering sideband channels, the simulation network uses a simpler method.

The first endpoint node to access the channel (of the two assigned endpoints) operates as a master clock. Other nodes synchronize to any nearby node with a synchronized clock, propagating the clock synchronization across the network. Rather than any explicit synchronization signal or packet, nodes synchronize to a modified logical one symbol. All synchronized nodes are programmed to cease pulses at the beginning of the guard period (so relaying nodes transmit pulses for less than t_{symbol} seconds). Unsynchronized nodes listen for the passage of a logical one symbol and

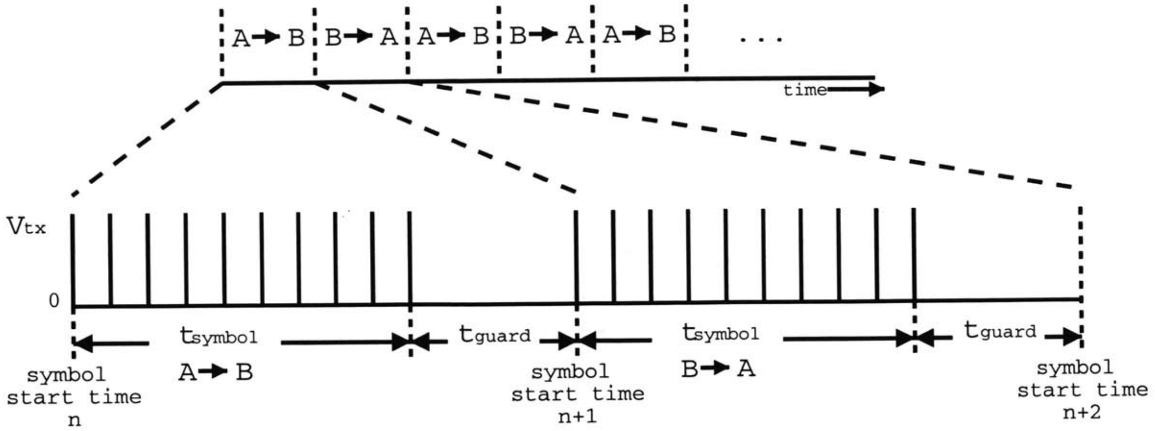

Figure 4-4: Time division of the IR channel. Symbols traveling from endpoint A to endpoint B are interleaved with symbols traveling in the opposite direction.

4.6 Time Division of Communication Channel

The dynamic cell membership algorithm requires that intermediate nodes be able to measure and differentiate the (apparent) propagation delay of signals originating from each of the two endpoints. Since the simulation uses unmodulated IR, signals traveling in opposite directions cannot use different frequencies. Instead, the IR simulation uses time division multiplexing to interleave signals traveling in opposite directions.

Intermediate nodes need not identify which of the two time slots represents endpoint A sending and which represents B sending, they need only know that any two consecutive time slots contain symbols traveling from alternate endpoints. All nodes in the network are synchronized to the start of a symbol period, which is also the start of a time slot, so the origin of an incoming symbol is apparent from its arrival time.

4.7 Network Layer and Applications

The simulation network has a simple network layer implemented entirely at the endpoints of a communication. Data is sent in frames of fixed size, beginning with a number of start bits which serve only to synchronize all clocks with that of the sender. Frames can include an arbitrary number of flags or headers. One such flag is currently used to identify frames that are sent without any data simply to maintain clock synchronization. A trivial modification would add a frame length header, enabling variable-length frames.

Two applications were designed for demonstration of the cooperative network. Both exist entirely in the endpoint nodes; intermediate relaying nodes have no knowledge of the nature of the data they are routing, and can route data for either application with impunity.

The first application features reliable delivery of small packets, with the sender periodically repeating packets until a correct confirmation is received from the recipient. Each packet has an error-correction code to facilitate recovery of partially-received packets.

The second application tunnels a simple serial connection through the cooperative network. The application code runs on the serial-to-IR gateway nodes. Both nodes receive characters from the RS-232 serial ports using their processor's UART hardware. The characters received at each gateway's serial port are transmitted bit-by-bit to the other gateway over the simulation network. At the receiving gateway, the bits are reassembled and sent out the serial interface, again using the hardware UART. This application is an example use of a real-time duplex data stream without delivery guarantees. The achieved data rate across the cooperative network was approximately 1200bps in each direction, for 2400bps total.

Chapter 5

Results and Analysis

The IR simulation network demonstrates the viability of routing data entirely in the physical link layer, and validates some of the techniques underlying the proposed cooperative communication protocol design. This chapter addresses the protocol's characteristics, its performance, and its efficiency.

5.1 Hardware-level Routing Protocol

The cooperative protocol was successfully implemented as a multi-hop transport protocol operating entirely in the network link layer of each node, without requiring any explicit network state information. A link-layer routing protocol is an accomplishment in and of itself, showing a possible alternative to complicated software layer routing implemented over a simple point to point link layer. More importantly, implementing routing in the link layer allows use of signal propagation information that normally would not cross the abstraction layer between link and network layers.

The implemented simulation succeeds in measuring signal propagation times and using these measurements as the sole input to a routing algorithm. In effect, the routing information is extracted from the data stream, allowing dynamic (per-bit) routing without excessive overhead. Where most link layers simply categorize a bit as a zero or one and throw away any other property of the received signal as unwanted noise, nodes in this cooperative network looks for both the data and hints as to the signal's propagation. This technique could be extended to make routing or application-level decisions based on received signal properties like delay or signal strength.

5.2 Constructive Interference

Modulation schemes which enable constructive interference are presented in Chapter 3, and a specific (simple) example is developed in Chapter 4. Constructive interference has great potential for power savings and even for improving the scalability of wireless networks, and also proved to be very effective in the IR simulation network.

The IR modulation scheme shows some of the promise of constructively interfering modulation, though it was less than ideal. It features constructive interference on "one" data bits, but "zero" bits did not constructively interfere (nor did they destructively interfere). Given the restrictions of the available IR transceiver hardware, this was an acceptable compromise. The modulation scheme did capably accomplish a synchronized time base and prevented collisions between nearby stations sending simultaneously (without any standard non-interference protocols). Unfortunately, this non-interference came at the cost of low data rate, which (without carrier-modulated IR) could not be offset by the use of many orthogonally multiplexed communication channels.

5.3 Latency

This section compares latency properties of the cooperative network with that of a packet-based multi-hop network. The latency figures given are lower bounds; when bandwidth is scarce, one or more transmitting nodes may need to wait for the channel to become free. Also ignored is any time required to buffer a suitably large amount of data to create a packet (in the packet-based multi-hop case) or to create a set of coded symbols (in the cooperative network case using COFDM).

5.3.1 Latency in the Cooperative Network

Packet-based multi-hop routing requires that an entire packet be received, decoded, and retransmitted in sequence (if the packet is relayed on a different channel, only the packet header need be received before decoding begins). Using intrinsic network topology and constructive interference, the process is much faster: the node begins receiving, and as soon as the signal can be clearly discerned, the node stops receiving, uses the propagation delay to decide whether it is in the routing path (see Section 3.4.2), and begins transmitting if so. The total time from signal reception to rebroadcast at node i ($t_{r,i}$ below) is only the time necessary to receive a clear signal (t_{rx}) and compute the routing decision (t_{comp}). If the hardware has a large switching time from reception to transmission ($t_{rx \to tx}$) this may dominate the computation time t_{comp}.

$$t_{r,i} = t_{rx} + max(t_{comp}, t_{rx \to tx}) \tag{5.1}$$

It should be noted that this delay, incurred at each node along the routing path, *does not depend on the data rate* or on the number of bits in a code unit (i.e. the number of orthogonal carriers). If each node has similar hardware, we can replace each $t_{r,i}$ with that of an average node. If the signal is routed over a path of l hops, and travels a total distance d, the total propagation latency is the sum of the delays at each node and the time of propagation through the communication medium ($t_{propagation}$):

$$\begin{aligned} t_{delay} &= \sum_{k=1}^{l} t_{r,i} + t_{propagation} \\ &= l * [t_{rx} + max(t_{comp}, t_{rx \to tx})] + t_{propagation} \end{aligned} \tag{5.2}$$

This delay depends only on the transceiver hardware (specifically the time necessary to recognize an incoming data symbol and the time necessary to switch from receive to transmit modes), the time to compute the routing decision, and the total propagation distance of the signal. The routing decision is computationally simple (a few fixed point subtraction, addition, and comparison operations are sufficient), so t_{comp} should be easily minimized. The physical propagation distance is not the straight-line distance between the two communicating nodes, but the total distance the signal travels along the shortest path it takes from relay to relay. If the channel allocation by the control channel leaves sufficient space between different connections using the same frequency space (i.e. no crosstalk), the average latency should be very close to the minimum (Equation 5.2).

5.3.2 Latency in Packet-based Multi-hop Networks

Packet-based networks must transfer data a packet at a time. The choice of packet size is a tradeoff between latency and throughput: smaller packets move through the network faster, but larger packets amortize the cost of packet headers over more bits, and so achieve a greater throughput. If a network uses a fixed packet size of s_{data} bits, with a header of s_{header} bits, and a communication data rate of D bits per second, the time to receive a packet is:

$$t'_{packet} = \frac{s_{data} + s_{header}}{D} \tag{5.3}$$

The retransmission delay at each node is at least the packet reception time (assuming the retransmission uses the same communication channel as the received packet, we must wait for the channel to become free again). There is an additional computation delay as the node decides whether or not to retransmit, and how the header should be changed (if at all). For the purpose of argument, we will assume that this computation occurs in parallel with the reception of the packet data, and so adds no latency. The total end-to-end latency (assuming a path length of l') is then at least:

$$\begin{aligned} t'_{delay} &= \sum_{k=1}^{l'} t'_{r,i} + t'_{propagation} \\ &= l' * [\frac{s_{data} + s_{header}}{D}] + t'_{propagation} \end{aligned} \tag{5.4}$$

Unlike the cooperative network latency, the packet multi-hop latency depends heavily on the data rate of the channel and on the packet size. Minimizing latency requires either a high-bandwidth channel (even when less bandwidth would be sufficient for the desired throughput) or a small packet size (decreasing efficiency – see Section 5.4). If we assume, as seems reasonable, that the propagation distance for packet-based multihop is similar to that for the cooperative network, then $t'_{propagation} \approx t_{propagation}$. However, the multi-hop latency depends linearly on the size of the packet, making packet-based multi-hop typically result in much greater end-to-end communication latency than the cooperative network.

Furthermore, the latency of packet-based multi-hop routing increases drastically when the network approaches its capacity and packets begin to queue at intermediate nodes. Queueing delays, in this case, depend on the non-interference protocol's operation, and are potentially unbounded. In contrast, the cooperative protocol has a hard limit on latency, like most circuit-switched networks, and any successfully established channel will perform within this latency bound.

5.4 Bandwidth Efficiency

This section compares the bandwidth efficiency of the cooperative system to a packet-based multi-hop system. There are tradeoffs between the two approaches which make them preferable for different applications.

5.4.1 Cooperative Network

The bandwidth efficiency of the cooperative network depends mainly on the maximum allowed distance between the communicating parties. The guard period of quiescence after the symbol time is a constant for the duration of a communication (though it could be a variable set by the channel establishment algorithm), and should be set high enough to prevent a delayed signal from interfering with the beginning of the next symbol. The maximum delay on a signal depends on the distance (both physically and in terms of hops) between sender and receiver, and so the length of the guard period determines the maximum possible distance that a signal can travel over the cooperative network.

Setting the guard period (t_{guard}) lower than the minimum propagation time between the endpoints will prevent data from reaching its destination (the routing algorithm will deem the path too long). Setting the guard period too long will decrease the bandwidth efficiency of the system. The ideal setting therefore will vary with the size of the network, and could be different for different channels, or even dynamically set (though this would require a broadcast message in the control channel to all routers).

The bandwidth taken by each channel in an OFDM system (we ignore the coding for the time being) depends on the switching frequency of any symbols in that channel, which is the inverse of the symbol period, t_{symbol}. We will consider 100% efficiency to be the data rate achieved when symbols are sent for t_{symbol} and there is no delay between symbols. This scheme would approach the Shannon limit of the communication channel. The guard period, necessary to prevent destructive interference from delays, decreases the effective bandwidth by requiring a period of quiescence between symbols. The bandwidth efficiency of the cooperative system, then is degraded in proportion to the guard period:

$$E_{bw} = \frac{t_{symbol}}{t_{symbol} + t_{guard}} \tag{5.5}$$

This equation gives the bandwidth efficiency achieved in transporting the data across the network, from sender to receiver. The bandwidth efficiency of the system is not as simple as it may seem: decreasing the bandwidth of each orthogonal COFDM channel increases symbol time, increasing the efficiency for a fixed guard period. Radio hardware is likely to constrain the system to a minimum feasible bandwidth per channel, placing an upper bound on the symbol time.

5.4.2 Packet-based Multi-hop Efficiency

The maximum packetized data throughput on a channel approaches the Shannon limit when the packet size, s_{data}, is much larger than the header size, s_{header}. However, this leads to prohibitively high latencies. With smaller packet sizes, packet-based multi-hop system throughput suffers an inefficiency at each hop:

$$E'_{bw} = \frac{s_{data}}{s_{data} + s_{header}} \tag{5.6}$$

While the cooperative system transmits a bit all the way to the end receiver entirely during a single symbol time, multi-hop routing systems retransmit each bit at each of the hops on the routing path. Each such transmission need only have enough power to reach the next node in the path, however, so the network capacity used relative to cooperative routing is indeterminate. If we sum the bandwidth use over a routing path of l hops, the bandwidth efficiency of a multi-hop routing protocol can be stated as:

$$E'_{bw} = \frac{1}{l} * \frac{s_{data}}{s_{data} + s_{header}} \tag{5.7}$$

Detailed analysis of the time-bandwidth product (or the time-bandwidth-area product), which would be a useful metric for comparison of the cooperative network to a similar packet-based multi-hop network, is beyond the scope of this work.

5.5 Power Efficiency

A software simulator was developed to evaluate the transmission power usage of a network with and without cooperative radio transmission. In theory, cooperative transmission should allow some nodes to decrease their broadcast power (because the signal will be strengthened by other nodes).

The simulator randomly populates a space with wireless nodes and randomly chooses a sender and receiver for transmission of data. It assumes an inverse-squared signal falloff rule and a constant Gaussian noise level across all space and across all

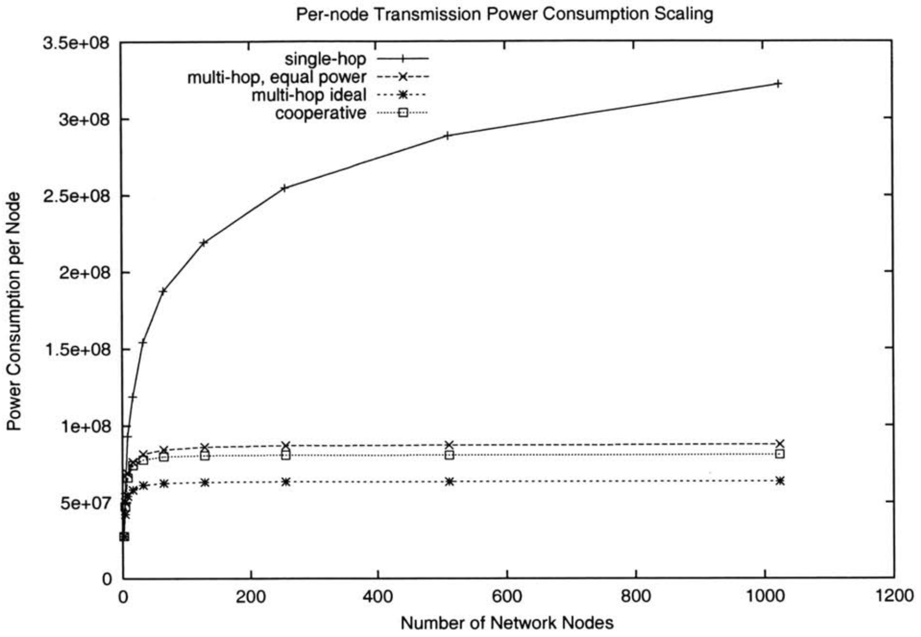

Figure 5-1: Transmission power per node vs. network size for various network routing schemes.

simulations. The simulator compares a single-hop path (direct from sender to receiver), a multi-hop path (a shortest path through the network where all edges are weighted by the square of their length), and a cooperative rebroadcast cell. Power expended during calculation of the shortest path or the rebroadcast cell is not included in the simulation. An optimal multi-hop path is computed to minimize global aggregate transmission power, and the cooperative rebroadcast cell is selected by the algorithm given in Section 3.4.2 with a spread factor of 1.1. Figures 5-1 and 5-2 show the resulting per-node and aggregate power scaling measured. Each data point is the average of 100 trials.

Figure 5-1 presents the per-node power scaling of various network routing schemes. Single-hop transmission directly from sender to receiver is the least scalable, far exceeding the other routing schemes. The most energy-efficient scheme is idealized multi-hop relaying of packets from sender to receiver via intermediate routers, each of which sets transmission power independently to a global optimum. Since the ide-

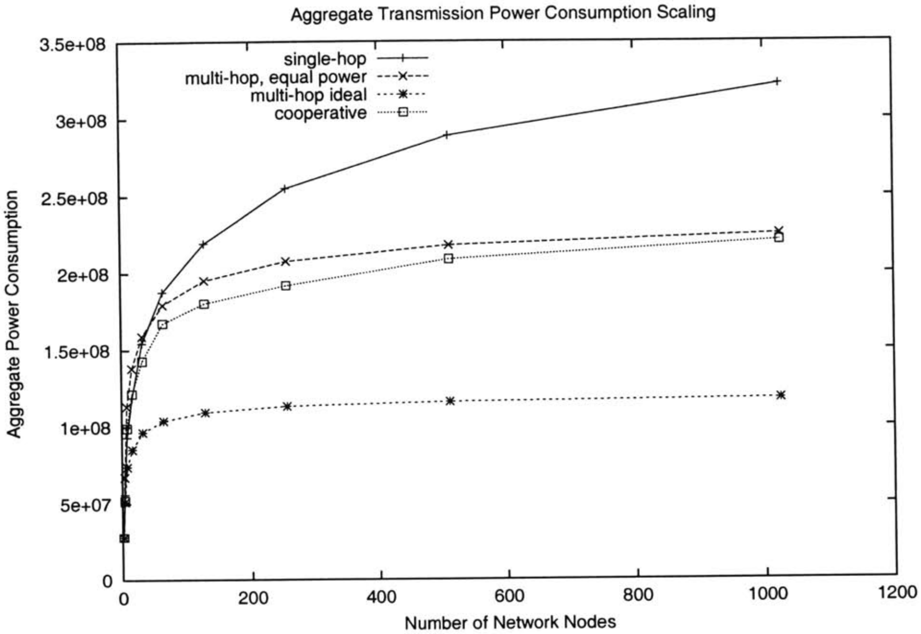

Figure 5-2: Aggregate transmission power vs. network size for various network routing schemes.

alized multi-hop power control requires global optimization, the "multi-hop, equal power" curve is more realistic, featuring a set of relaying nodes that all operate at the lowest transmission power necessary to achieve connectivity. The cooperative protocol, also using the same transmission power at every node, has a scaling function looks very much like that of the multi-hop schemes, but outperforms the multi-hop equal power scheme by a small margin. This graph demonstrates that a cooperative network should feature per-node transmission power scaling similar to equivalent multi-hop systems.

Figure 5-2 presents the aggregate transmission power use (summed over all transmitting nodes) necessary to transmit data using the schemes. Again, single-hop direct transmission is the least efficient, and ideal multi-hop (with transmission power set independently at each node for globally minimal aggregate power use) scales the best. Comparing the more feasible multi-hop with equal transmission power at each node to the cooperative scheme again shows similar scaling, with the cooperative power consumption slightly lower.

5.6 Noise Coverage

Decreased transmission power, all else remaining the same, reduces the amount of noise present at nodes that are not involved in the communication. Thus decreasing transmission power at a node decreases the "noise coverage" – the area for which nodes overhear too much noise to correctly receive another transmission. This gives a measure of the spatial efficiency of the protocol: smaller noise coverage area means fewer nodes unable to engage in other communication, and thus greater communication capacity of the network as a whole. The simulator results in Section 5.5 demonstrate that transmission power of a cooperative network is similar to that of a multi-hop network, so the noise coverage area should be similar in both cases. Detailed analysis of this effect is left to future research.

Chapter 6

Conclusions

This protocol demonstrates the possibility of a network with multi-hop routing implemented entirely in the radio frequency domain, exploiting interference between sending nodes instead of avoiding it. It further shows the potential benefits of breaking abstraction barriers that hide useful information or constrain the utilization of a scarce resource. The proposed protocol uses propagation delay information from the link layer to implement routing, and utilizes the broadcast nature of the radio medium rather than relying on a simplifying, but constraining, non-interference abstraction.

6.1 Cooperative Radio

Shannon's work on information theory sets a theoretical upper bound on the information capacity of a *point-to-point* communication channel, relating the communication capacity to the channel bandwidth and noise level. The relationship between capacity and the number of communicating nodes, on the other hand, is not yet known. Cooperation among radio nodes may allow new advances in scalability of wireless networks, with important technological, economic, and societal impact.

If linear or super-linear capacity scaling is possible, then new nodes may in fact bring more resources to the network than they use. Currently, communication bandwidth is seen as a scarce resource, and is closely guarded by governmental organizations (like the Federal Communications Commission in the US, which auctioned off 3G cellular bandwidth for hundreds of millions of dollars). If, however, each new node in a network does not consume net capacity, then capacity will not be a scarce resource, and it will not need to be strictly regulated. The total cost of operating a network will fall dramatically, allowing the growth of new uses like pervasive wireless computing.

6.2 Breaking the Abstraction Barrier

In many ways, the division of network information transmission into a link layer and a network layer is arbitrary, and puts artificial limits on the system. The link layer describes how to put physical representations of bits into a transmission channel and how to retrieve them from the signal received on the other end. The network layer logically organizes a network such that data can be passed reliably from one node to a specific destination, regardless of physical distance. Existing research is overwhelmingly focused (and constrained) to fit into one of these two categories, ignoring the potential for a systematic approach to solve both problems at once.

The concept of an abstraction barrier – a well-defined interface to a system that prevents all other interaction with the outside world – is a standby in system engineering. By reducing interactions between system components to a well-documented and tightly-controlled interface, designers can drastically simplify analysis of the system as a whole. Without abstraction barriers, designers must consider every possible interaction between components, so system complexity grows in complexity as the square of their size (if not faster). However, by constraining the interaction between components, abstractions preclude interactions that may be beneficial to system designs.

The following sections discuss two ways that breaking conventional abstraction barriers can lead to better system design: information sharing between layers, and avoiding abstraction constraints.

6.2.1 Information Sharing

Abstraction barriers simplify systems by preventing components from reading or altering the internal state of other components. In most cases, this is a sound principle. However, sometimes the interfaces between components hide too much, unnecessarily constraining the implementation of a system.

The cooperative protocol is able to implement a networking scheme very much like Hyphos [18] but without the need for various packet header information. This is possible primarily because the cooperative protocol allows what would conventionally be called the physical link layer (the radio system) to share information with what would be called the network layer.

Information hiding across abstraction barriers is a useful paradigm, but designers should take care to ensure that the benefit of system simplification does not prevent information sharing that would lead to a better system design. Perhaps other adhoc networking schemes would benefit from the availability of physical propagation measurements in their routing algorithms.

6.2.2 Avoiding Constraining Abstractions

In addition to hiding information, abstraction barriers can unnecessarily constrain functionality. One oft-cited example is Internet multicast (one-to-many communication). The misguided approach is to build multicast functionality over existing unicast protocols, sending one piece of data repeatedly to the list of many users. However, since many of the duplicate data packets will likely traverse the same path, it is far less expensive (in terms of network resources) to implement multicast as a special network function. This is especially true when multicast operates on (non-switched) Ethernet LAN, which is physically a one-to-many system. Since every station receives every packet, unicast incurs cost to simulate a one-to-one structure (using a "to:" header that all other nodes ignore). It is wasteful to implement a one-to-many transmission on a one-to-many network incurring the overhead of a one-to-one abstraction.

The cooperative protocol, similarly, utilizes one-to-many communication on a one-to-many physical medium (an analog to Internet multicast). Most ad-hoc network systems are built on top of non-interference protocols that incur significant cost to prevent simultaneous reception of two signals, forming a one-to-one abstraction like in Internet unicast. Given our concern with efficiently utilizing network resources, the cost of such an abstraction must be considered before layering the organization of an ad-hoc network above it. The "end-to-end argument" [20] states that "a function or service should be carried out within a network layer only if it is needed by all clients of that layer, and it can be completely implemented in that layer." Non-interference medium access control may, in fact, not be needed by all ad-hoc network systems.

6.3 Future Work

This work presents evidence that local cooperation in communication networks can lead to improvements in network scalability. Implementing more complex cooperative networks and examining the limits of scalability in cooperative ad-hoc networks remains a broad area for future research. This section suggests several potential directions for continuing the investigation of cooperation and cooperative networks.

Perhaps the most apparent continuation of this work would be the implementation of full-featured cooperative networks and evaluation against currently existing alternatives. For example, the implementation of a cooperative radio network using COFDM, including the development of necessary hardware and a control channel protocol. Ideally, a control channel would run distributed algorithms and result in an efficient allocation of bandwidth to communicating nodes, enabling dense frequency reuse.

Power control algorithms in a cooperative radio network is another worthy research avenue. Such algorithms could be developed in simulation or tested on a physical implementation of a cooperative network. Power control should include both the regulation of transmission power (to optimize some combination of per-node power usage and aggregate network power usage), and algorithms to control the power consumed by a radio receiver awaiting a signal in idle mode.

Further network services could also be implemented in a cooperative network. Multicast routing could be developed either in an overlay network or by modifying the rebroadcast cell membership algorithms, and may prove useful in media distribution or other information broadcasting. Differentiated quality of service may help a cooperative network handle large volumes of traffic efficiently by prioritizing real-time or high-bandwidth applications over less time-critical ones.

Finally, further work is necessary on the scalability limits of cooperative networks. It is clear that they do not fall into the category of network analyzed by Gupta, and so are not necessarily bound by the $sqrtn$ capacity scaling rule he derives for networks under non-interference protocols. Studies of cooperative networks should yield hints to the limits of network scalability, and provide clues to the construction of networks where every node creates more capacity than it consumes.

Bibliography

[1] Gnutella. http://gnutella.wego.com.

[2] Kazaa. http://www.kazaa.com.

[3] Morpheus. http://www.musiccity.com.

[4] Multisource file transfer protocol. http://www.edonkey2000.com/overview.html.

[5] Napster. http://www.napster.com.

[6] R. Dube, C. D. Rais, K. Wang, and S. K. Tripathi. Signal stability based adaptive routing (SSA) for ad hoc mobile networks. *IEEE Personal Communication*, February 1997.

[7] N. Duffield, J. Horowitz, and F. Presti. Adaptive multicast topology inference, 2001.

[8] J. Elson and D. Estrin. Time synchronization for wireless sensor networks, 2001.

[9] Sally Floyd, Van Jacobson, Ching-Gung Liu, Steven McCanne, and Lixia Zhang. A reliable multicast framework for light-weight sessions and application level framing. *IEEE/ACM Transactions on Networking*, 5(6):784–803, 1997.

[10] R. G. Gallager. *Information Theory and Reliable Communications.* John Wiley and Sons, Inc, New York, NY, 1968.

[11] P. Gupta and P. Kumar. Capacity of wireless networks, 1999.

[12] J. Haartsen, M. Naghshineh, and J. Inouye. Bluetooth: Vision, goals, and architecture. *Mobile Computing and Communications Review*, October 1998.

[13] IEEE. Wireless LAN medium access control (MAC) and physical layer (PHY) specifications. *IEEE Std. 802.11b-1999*, 1999.

[14] L. W. Couch II. *Digital and Analog Communication Systems.* Prentice Hall, Upper Saddle River, NJ, 2001.

[15] D. Johnson and D. Maltz. Dynamic source routing in ad-hoc wireless networks. *Mobile Computing*, 1996.

[16] J. Lifton, D. Seetharam, M. Broxton, and J. Paradiso. Pushpin computing system overview: a platform for distributed, embedded, ubiquitous sensor networks. In *Proceedings of the International Conference on Pervasive Computing*, August 2002.

[17] Mills. Internet time synchronization: The network time protocol. In *Zhonghua Yang and T. Anthony Marsland (Eds.), Global States and Time in Distributed Systems, IEEE Computer Society Press.* 1994.

[18] R. Poor. Hyphos: A self-organizing, wireless network. Master's thesis, Massachusetts Institute of Technology, June 1997.

[19] D. P. Reed. How wireless networks scale: The illusion of spectrum scarcity. Talk at International Symposium on Advanced Radio Technologies, March 2002.

[20] Jerome H. Saltzer, David P. Reed, and David D. Clark. End-to-end arguments in system design. *ACM Transactions on Computer Systems*, 2(4):277–288, November 1984.

[21] T. J. Shepard. A channel access scheme for large dense packet radio networks. *Computer Communication Review*, 26(4), October 1996.

[22] J.H. Stott. The how and why of COFDM. *EBU Technical Review*, (278):43–50, 1998.

[23] VP. W. Wolniansky, G. Foschini, G. Golden, and R. A. Valenzuela. V-BLAST: an architecture for realizing very high data rates over the rich-scattering wireless channel. pages 295–300, New York, NY, 1998. URSI International Symposium on Signals, Systems, and Electronics, IEEE.