1 Introduction

1.1 Project Goals

We propose to study the design of an interoperable family of wireless protocols for portable devices. These new protocol build on recent advances in physical layer design and software-configurable radios. This new family of wireless protocols has the following goals:

1. Universal connectivity among heterogeneous devices within this family;
2. Flexible tradeoff among communication range, data rate, and transmission power at the application layer;
3. Efficient spectral utilization collectively by a network of nodes.

1.2 Motivation

We are witnessing an explosion of devices with wireless communication capabilities. Cordless phones have gradually replaced wired phones; laptop computers and PDAs have built-it wireless network interfaces for Internet connectivity; DVR, DVD, and MP3 players can now play music and movies stored on a PC through a home wireless network.

On the surface, our environment is progressing rapidly towards the vision of ubiquitous computing. Observing carefully, however, one notices that, while more and more devices are wireless enabled, they form islands of connectivity. Different classes of devices often use different medium access control (MAC) and physical layer (PHY) protocols. Accordingly, these devices cannot communicate unless each device has several radios speaking different PHY and MAC protocols.

A variety of PHY and MAC protocols exist because different classes of devices have different range, power, and data rate requirements. For example, Bluetooth radios that connect cellular phones and wireless headsets provide short-range connectivity at 1Mbps. The IEEE 802.11 family of wireless networks provides longer-range connectivity to portable computers at up to 54Mbps. Wireless sensor network nodes for environmental monitoring typically have low bit rate links at a very low power using a tailor-made MAC protocol. To accommodate the differences in power, range, and data rates requirements, specific PHY and MAC protocols were designed to suit each class of devices.

In recent years, we have seen a proliferation of wireless PHY and MAC layer standards such as Bluetooth, wireless LAN (IEEE 802.11a[2], 80211b[3], 802.11g[6]), personal area networks (IEEE 802.15.3[9], 802.15.4[7]), and broadband wireless metropolitan area network (IEEE 802.16[4], 802.16a[8], 802.16c[5]). To make matters worse, in the future, as more spectrum is allocated to accommodate the fast growing needs of wireless communications, one can imagine the emergence of even more PHY and MAC standards and their associated problems:

1. Different devices are in range but cannot interoperate;
2. One type of network interferes with or starves another;

3. Different classes of devices require separate spectrum allocations resulting in poor spectrum utilization;
4. More MAC protocols are invented, causing confusion and management problems;
5. New spectral allocation requires building new non-backward compatible devices.

Fortunately, advancements in software-configurable MAC and PHY layers have brought a new possibility for a solution. Specifically, radios that can adapt their frequency and waveforms and run on configurable computing platforms are already commercially available (e.g.,[10][11]). Moreover, OFDM has emerged as a modulation scheme that performs well in fading environments. OFDM is a wideband modulation scheme that can easily scale to use a variable bandwidth. Moreover, OFDM supports simultaneous multi-user access by assigning users orthogonal sequences. Because of its many good features, it has found widespread applications in IEEE 802.11a, the European HiperLAN/2 standards, 802.16 broadband wireless access systems, digital TV broadcast systems DVB [17], and even future generation IP-based cellular phone system [13].

We believe the dream of a unified PHY/MAC framework in which a future generation of devices with highly variable requirements can interoperate is indeed within reach.

2 Overview

In this section, we discuss the three major aspects of the problem we are tackling, namely universal connectivity, flexible range/rate/power tradeoff, and efficient spectrum utilization. In section 2.1, we present the current state-of-the-art solutions and explain why they are not satisfactory. In section 2.2, we then introduce the key features of our proposed architecture and explain the rationale behind our choice. The details of our proposed architecture are presented in section 2.2.

2.1 Current Systems

The current systems are optimized for specific classes of requirements. As a result, they have limited connectivity, are inflexible in the tradeoffs they make, and do not utilize the spectrum efficiently.

Connectivity

Today, different classes of devices usually speak different PHY and MAC protocols. To achieve cross-class communication, a node must be equipped with at least two different radios. For example, a laptop computer is usually equipped with a 802.11 network adapter. However, if it needs to use a cellular phone to access the Internet or send faxes, it must use another radio that is installed on a cellular phone, such as a Bluetooth radio. This approach to providing universal connectivity is clearly inefficient because it increases cost and weight.

Worse, radios running different PHY/MAC protocols that share the same spectrum often interfere with each other, especially in close proximity. For example, Bluetooth radios, many home cordless phones, and 802.11b/g devices all share the same unlicensed ISM 2.4-GHz band. Experimental studies have confirmed the mutual interference of 802.11b and Bluetooth [16], [18]. When we use our 2.4GHz digital cordless phone, our 802.11b client loses its connection with its access point until we manually change the channel on the cordless phone. [1] Due to the interference between different classes of radios, even devices with multiple radios might not connect to two networks simultaneously.

Tradeoffs

Requirements of the user change over time and systems should adapt to these changes. Consider a user with a laptop computer who is working in her office. While in her office, her computer is plugged in to the wall socket, and so she is not concerned about the power consumption of her wireless network interface. Instead, she is more concerned about the data rate that her laptop computer provides. However, when she leaves her office and goes to an outdoor coffee shop where there is no power supply, she becomes more concerned about the lifetime of the battery of her computer than the speed of her downloads. Similarly, a mobile phone is only useful if it is connected. In areas where connectivity is poor, it is better to trade data rate (i.e. voice quality) for a longer range, than to have a fixed range and a fixed data rate.

Spectrum Utilization

Existing PHYs are designed with a specific amount of bandwidth in mind. They cannot use more bandwidth even if more is available. A good example is 802.11b. In the US, according to 802.11b standard [3], one can have three independent channels (each of 22MHz). However, each ad hoc network can only use 1 channel for the entire network. In an area where there are no other networks nearby, this is extremely wasteful. There are two ways one can more effectively use the available spectrum. In the first method, one can define a new PHY that uses more bandwidth (e.g., use all 3 independent channels) to achieve higher throughput. This is basically the approach taken by some radio vendors [12] who use proprietary methods to bundle two channels together to form a higher speed link as a workaround to use more of the precious spectrum. There are two problems with this approach. First, such a PHY does not work in countries where the spectrum allocation is less than that of the US. More importantly, the amount of spectrum that can be used effectively is limited by the amount of bandwidth that individual nodes can modulate.

2.2 Our Approach

We propose a flexible MAC/PHY architecture that improves the connectivity, tradeoffs, and spectrum utilization. We first provide an overview of the architecture, then we explain how it addresses our objectives.

Architecture Overview

The protocols we are designing are for multiple devices in an ad-hoc network. At any given time, a number of devices may be communicating simultaneously. Moreover, the different transmissions may have different bit rates, powers, and ranges. The devices are compatible in the sense that any two devices may communicate by agreeing on a commonly supported protocol. Finally,

[1] The test was conducted using a Linksys 802.11b Access Point (model no. BEFW11S4), a Lucent Orinoco Gold 802.11b PC card, and a Uniden DSSS cordless phone (model no. TRU-448). All devices are in the same room.

some transmissions may be bursty and use a contention avoidance protocol while others may be constant bit rate and use a TDMA protocol.

Thus, our system combines a family of PHY and MAC schemes. The PHY family is designed for flexibility and simplicity of implementation. The MAC protocols build on existing schemes.

The PHY of our family of protocols uses OFDM. In a standard OFDM system, the transmitter and the receiver agree on a hoping sequence. The hoping sequence specifies the OFDM sub-carriers to be used in successive time slots. The transmitter sends one OFDM symbol in each time slot in the sub-carrier specified by the sequence and the receiver listens to the sub-carriers according to the same sequence. Moreover, in each time slot, the transmitter selects a modulation scheme (e.g., N-QAM) based on the signal-to-interference ratio. The sub-carriers are arranged into groups of adjacent sub-carriers that we call *bands*.

In our protocols, we extend this standard OFDM system by enabling one radio to modulate multiple OFDM sub-carriers in each time slot. This capability permits the radio to transmit at a higher bit rate by using more spectrum and more power.

The protocols have two features: 1) Band-Hopping, and 2) In-Band Channels.

1. Band-Hopping

The devices use different bands to enable full use of the spectrum. For instance, devices A and B may be using band 1 while devices C and D use band 2. At any given time, the radios modulate and demodulate only the sub-carriers of one band. This design choice simplifies the radios and reduces their cost.

Assigning the receivers to fixed bands might result in an imbalanced network and would limit the achievable throughput. Accordingly, to adapt to the variable number of active devices, they hop across bands.

2. In-Band Channels

In any given band, a transmitter follows a sequence of sub-carriers. Recall that the main benefit of this hoping among sub-carriers is to make the system less sensitive to fading. For instance, suppose band 1 consists of channels 1 through 10. Imagine transmitter A is in band 1 and uses two sub-carriers at a time. In successive time slots, the transmitter may use the sub-carriers $(1,2),(2,3),(3,4),\ldots,(9,10),(10,1),(1,2),\ldots$, and so on. In each time slot, the radio chooses the modulation scheme. For instance, if the signal-to-interference ratio is good, it may use 64-QAM to modulate each sub-carrier.

Different transmitters that are in the same band use orthogonal sequences in that band. Thus, while transmitter A is using the sequence $\mathbf{s}_1 := \{(1,2),(2,3),(3,4),\ldots\}$, transmitter B may be using the sequence $\mathbf{s}_2 := \{3,4,5,\ldots\}$. Note that the sequences $\mathbf{s}_1$ and $\mathbf{s}_2$ are orthogonal. We call such sequences of sub-carriers within a band a *channel*.

Thus, a low-power radio may use one sub-carrier at a time with a low-rate modulation such as BPSK. A high-power radio may use two sub-carriers at a time with a fast modulation such as 64-QAM and achieve a transmission rate that is 12 times faster than the low-power radio. When the high-power radio communicates with the low power radio, it uses the single-sub-carrier BPSK protocol.

The Medium Access Protocol has three tasks: 1) Synchronizing the nodes; 2) Selecting the band-hopping sequence; 3) Selecting the channels within each band.

Benefits of Architecture

In the previous section, we have presented an OFDM framework that can provide a common platform for radios to inter-operate while accommodating different data rate, communication range (or SNR), and transmission power requirements.

Using our proposed physical layer, we can control the data rate of a physical channel by either changing the number of the modulation levels or by using a different number of OFDM sequences. To control the amount of power used, one can change the transmit power per sequence or use a different number of sequences. Similarly, to tune the effective communication range of a logical channel, the transmit power per sequence and the modulation/channel code can be adjusted.

To gain a sense of the level of control one has on a channel, consider the following example. The 802.11a standard defines that each radio modulates 53 sub-carriers at a time. Only 48 out of the 53 sub-carriers are used for data. In addition, there are 8 modulation/channel code combination that encodes between 0.5 to 4 data bits per OFDM symbol according to our definition of an OFDM symbol. Therefore, one can create a channel with up to $48 \times (4/0.5) = 384$ times difference in data rate using practically the same hardware. Such flexibility in the control of the resulting logical channel provides the foundation for building a family of compatible MAC protocols.

3 Research Issues

The architecture we propose reuses many known schemes. For instance, its PHY is a small extension of familiar OFDM schemes. The architecture uses techniques from known channel-sharing MAC protocols such as 802.11[1]. The new aspect of the architecture is the channel selection protocol.

3.1 Problem Formulation

The design problem can be formulated as follows. One is given a number of OFDM subcarriers and a time slot duration. Consider a set of devices and assume, for simplicity, that they are all in communication reach of each other (no hidden terminal). The problem is to allocate sequences of one or multiple sub-carriers to the different radios to meet application requirements while satisfying the limitations of the radios.

By choice, we have restricted this problem based on practical implementation considerations. As we explained earlier, the system uses disjoint bands of adjacent sub-carriers and a combination of band-hopping and in-band channels. Thus restricted, the design problem consists of the following tasks: 1) Design of Channels; 2) Design of Band-Hopping Sequences; 3) Synchronization; 4) Band Selection; 5) Channel Selection. We comment briefly on each of these tasks.

Design of Channels

As we stated, within each band, the channels are orthogonal. This choice simplifies the design of the radios. Each radio can demodulate in parallel the sub-carriers in the band. At any given time, a number of radios in one band may be using some channels. Another radio that wants to transmit in that band must discover easily the free channels. We propose a simple set of cyclic permutations for the channels.

Design of Band-Hopping Sequences

The different band-hopping sequences cannot all be orthogonal in general because there may be more devices than bands. A simple scheme is designed to identify sequences that are almost orthogonal. We propose a scheme of pseudo-random sequences identified by the seed of the random number generator.

Synchronization

The different nodes must be synchronized to limit their interference. Note that we only require local synchronization. We propose to use the approach used by other OFDM systems such as 802.11a[2] or those techniques summarized in [14] and to extend them to a ad-hoc network environment.

Band Selection

We propose to investigate a number of band-selection algorithm. The first algorithm we will try is "meet-and-stay." We describe it in section 4.2.

Channel Selection

We propose to study a simple algorithm for the channel selection. The basic idea is to discover a free channel through carrier sense and use it.

3.2 Related Work

The band and channel selection problem is not simple for two reasons. First, the algorithm must be on-line. That is, the transmission requests arrive at random times. Future requests are not known when a new request is accommodated. Second, it is distributed. That is, the devices have an incomplete knowledge of the state of the network when they make their allocations.

A multi-channel band protocol must coordinate the communicating nodes so that a sender and its receiver tune their radios to the same band during data transfer. In this section, we survey existing multi-band protocol proposals and discuss their limitations.

The core issue in the design of a multi-band MAC is that nodes are equipped with fewer radios than the number of bands. Since a node cannot monitor all band at all times, a node has difficulty keeping track of which bands/channels are free (channel uncertainty) and whether a particular neighbor is busy (neighbor uncertainty).

The existing approaches to solving the channel and neighbor uncertainty problems can be classified into three generalized approaches: (i) control radio, (ii) common band, and (iii) common band hopping sequence.

In the control radio approach, each node is equipped with one control radio in addition to one or more data radios. The control radio is always tuned to the control band where all nodes contend for access to any of the data bands. Since each node always listens to the control band, it can overhear control messages from its neighboring nodes so that it can keep track of the status of all the bands and neighbors with little difficulty. This approach is used in by DCA[23], DPC[15], and DCA-PC[24].

In the common band approach, the time is divided into fixed periods. In the first phase of each period, all the nodes meet on a default band to exchange control messages to decide how to assign themselves to the various available bands during the second phase of the period. Data transfers occur only during the second phase of each period. This approach is used in MMAC[19].

In the common hopping sequence approach, all nodes that are not actively sending or receiving follows a common hopping sequence through all the bands. A common hopping sequence guarantees senders can use RTS/CTS to find their prospective receivers if they are not currently busy. Nodes overhear the control messages destined for other nodes in order to keep track of band and node status. This common hopping sequence technique is used in HRMA[20], CHMA[22], and CHAT[21].

All three approaches have a common feature: all nodes waiting to transmit always converge on the same band at the same time. The medium access requests of all the senders are serialized on 1 band. The difference is that in both the control radio and the common band approaches, all medium access requests are always resolved in the same band. In the common hopping sequence approach, the common band changes over time. But in all three approaches, the common band can become a bottleneck when the data packets are not much longer than the control packets or when man bands are available.

4 Preliminary Results

We have designed a preliminary MB-MAC to test our ideas. In this section, we use a high-level discrete-time simulator to evaluate the potential of our design. We describe this approach here as it illustrates the methodology that we plan to use to evaluate the more realistic protocols that we will develop.

The simulator is not meant to be as precise as a packet-level event driven simulation. Instead, it serves to quickly validate that our design has the potential to achieve good performance in a variety of operating conditions. In particular, we hope to use the high-level simulator to find out the answers to the following questions:

1. Given a load exceeding the normal operating region of a MAC protocol, will the throughput drops to zero (i.e. congestion collapse)?

2. Given the data packets are very short, will the MAC perform reasonably well?

4.1 Simulation Model

The simulator is a discrete-time simulator implemented in MATLAB. For our preliminary study presented here, we assume there are N stationary nodes in the network. A total of B OFDM bands are available, each with M sub-carriers. Each node has one radio that can tune to any of the B bands and can modulate all M sub-carriers at a time. All nodes can hear each other. Time is slotted. Each slot is 500μs which is enough for a short data packet on the order of 256 bytes plus an acknowledgment (ACK). Data packet longer than 256 bytes require additional time slots for transmission. For simplicity, we assume that one can send exactly 256 bytes in every slot. Nodes are synchronized to the slot boundaries.

4.2 MAC Algorithm

The MAC protocol that we have tested can be described roughly as follows. Recall that the devices must select the band and then the channel.

Each node has a "home" pseudo-random sequence of bands. In each band, a node uses all the sub-carriers in parallel. To transmit, node A jumps to the home band of one of its receiver, say B, and listens to that band. If the band is idle, node A transmits a "hello B"message. If B is idle, it should be in its home band and able to hear the "hello B" message and reply to it. The two nodes A and B stay together on the band where they met until the transmission is complete. Once the transmission is over, the two nodes A and B return to their home band as if the packet exchange had never occurred.

It is possible that node A finds the band idle even though it is in use. For instance, A may have arrived in the band during a gap between transmissions. To reduce the chances of collision, our protocol transmits in the band with a probability that is inversely proportional to the number of nodes that should be in that band, as estimated from the known pseudo random sequences.

This MAC protocol assumes that the nodes are perfectly synchronized and that there are no hidden terminals.

Here is a more detailed description of the protocol:

1. A node i follows a pseudo-random band hopping sequence $X(S_i, t)$ that is i.i.d. and uniformly distributed over $1, 2, \ldots, B$. S_i is the seed that node i uses to generate random numbers. t is the step of the discrete-time clock. $X()$ is a well known function shared among all the nodes in the network. Fig.1 illustrates the pseudo-random band hopping patterns.

2. Unless changed otherwise, node i listens on its default band $X(S_i, t)$ during slot t.

3. Each packet, including beacons, that node i sends contains the seed S_i. So simply by overhearing a packet from node i, any node j can predict the future hopping sequence of node i. We will therefore assume that each node eventually knows the seeds of its neighbors.

4. Right before the beginning of each slot t, a sender node first checks if any receiver j will be in the same band $C = X(S_i, t)$ as itself in slot t. If so, the sender transmits the packet to one such receiver j with probability $p_{tx} = 1/\hat{N}(C, t)$. $\hat{N}(C, t)$ is the estimated number of nodes in band C during time slot t. Each noce can calculate $\hat{N}(C, t)$ by counting the nodes in $\{m : X(S_m, t) = X(S_j, t)\}$.

5. If none of the receivers of i will be in band C in slot t, node i chooses randomly one of the bands for which it has at least 1 receiver, C'. Then with probability $1/\hat{N}(C', t)$, it changes its band for slot t to C' and transmits a packet to a receiver on channel C'.

6. Before sending any packet, each node is required to detect carrier for a brief amount of time to avoid collision with an ongoing packet trains to be explained in step 11

7. If i decides not to transmit, it repeats step 4 in the next time slot whenever it is not busy sending or receiving and it has some data queued.

8. The packet the sender sends includes a bit that indicates whether more packets are queued at the sender for this receiver.

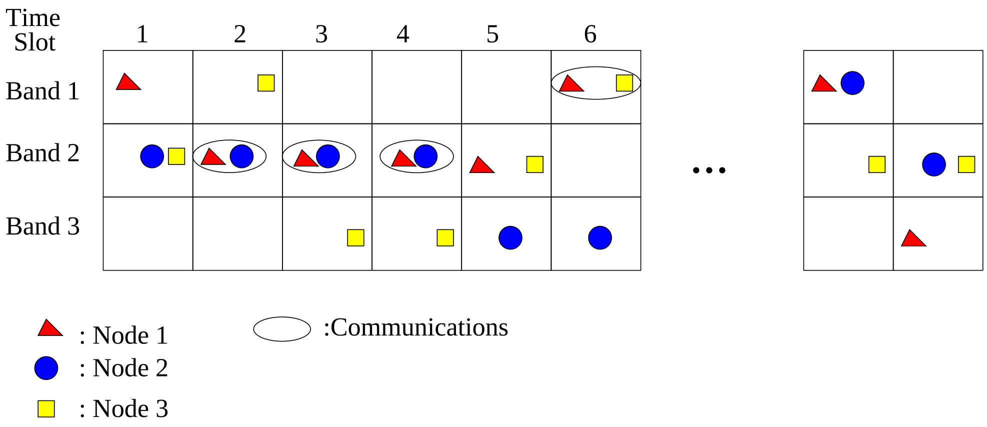

Figure 1: Illustration of the basic algorithm. Nodes follow a pseudo-random band hopping sequence. Nodes meet by chance. After two nodes meet, they can stay together to transfer more packets.

9. Upon receiving the data packet, the receiver returns an ACK. If the sender had indicated that more data are queued, the receiver temporarily suspends its band hopping and remains in the current band until it has finished communicating with the sender or until a timeout.

10. The ACK message also has a bit which indicates whether the receiver has any data for the sender in the reverse direction.

11. At this point, if i has more packets to send to j, or vice versa, they may continue to exchange packets on the same channel until all packets for i has been sent. [2]

12. After the data transfer between i and j is finished, say at slot t', both nodes return to their default bands $X(S_i, t')$ and $X(S_j, t')$ as if they had never exchanged packets.

4.3 Traffic Pattern

In the simulations, we assume a single hop ad hoc wireless LAN where all N nodes can hear each other. There are $N(N-1)$ possible unidirectional flows between every pair of distinct nodes. At the beginning of the simulation, each flow is inactive. During each time slot, a random number of inactive flows become active following a Poisson arrival model. While a flow is active, a packet might arrive at the flow during each time slot in an i.i.d. fashion with a fixed probability. Each flow becomes inactive after a fixed period of time. In the simulation, packet are lost due to collisions only, but not because of transmission corruption.

Here is a list of parameters we used for simulating two extremely high load situations:

[2] In practice, to avoid starvation, there should be a limit as to how many packets can be transferred in a train.

Parameters	Experiment 1	Experiment 2
Charateristics	Long-lived flows (better)	Singleton Packets (worse)
Number of nodes	50	50
Number of bands	5	5
Packet Length (256 bytes/slot)	5 slots/pkt	1 slot/pkt
Load	80%	80%
Flow Duration	50 slots	1 slot
i.i.d. Packet Arrival Prob Per Slot	0.2	1
Simulation Time	100,000 slots	100,000 slots

Load is scaled by the number of bands. It is calculated as the average number of packets arriving per time slot multiplied by the packet length (in slots) and then divided by the number of bands.

4.4 Simulation Results

We have simulated the system to measure its throughput, delays, and backlogs. We have considered two scenarios. In the first scenario, the packets are long, with a transmission time of five time slots. In the second scenario, the packets are short, with a transmission time of one packet. The load, however, remains the same.

Overview of Results

The simulation results show that the system is stable under heavy load and has acceptable delays. The stability is not surprising. Indeed, nodes stay together after they meet until the transmitter has completely exhausted its backlog for the receiver. As the backlog increases, the MAC protocol becomes more and more efficient. In the worst case, the delays are comparable to those of a round-robin system.

Behavior Under Heavy Load, Long Packets

The load on the system is 80%. That load corresponds to four of the five bands being busy, on average, in each time slot. There are many relatively long lived connections transferring an average of 10 long packets (1280 Bytes). We collected statistics on the average queue length, the medium access delay for the data packets, and the usage of time slots. Figure 2 (right) shows the average queue length at each node varies significantly over time but the average per source queue lengths have not grown unbounded during the simulation time of 100,000 steps. In fact, many queues are emptied from time to time. The mean delay is roughly 104 slots which translates to about 52 ms–a reasonable figure for a heavily loaded overloaded system.

Figure 2 (left) shows the breakdown of the usage of time slots in each band into 6 categories: first success (orange), follow-up (2+) success (dark red), idle (yellow), quiet (cyan), collision (blue), and receiver absent (dark blue). A first success slot is one in which the medium contention ended with a single contender transmitting and winning. One slot worth of data is successfully transferred. A follow up success (2+) slot is one that follows a first success or another follow-up success. It may carry a fragment of a longer packet that has been started in the preceding slots or it may a different packet between the same sender/receiver pair who owned the previous slot. An idle slot is one in which none of the nodes in that band has any data packet destined for any other node in the same band. A quiet slot is one in which a packet could have been transferred

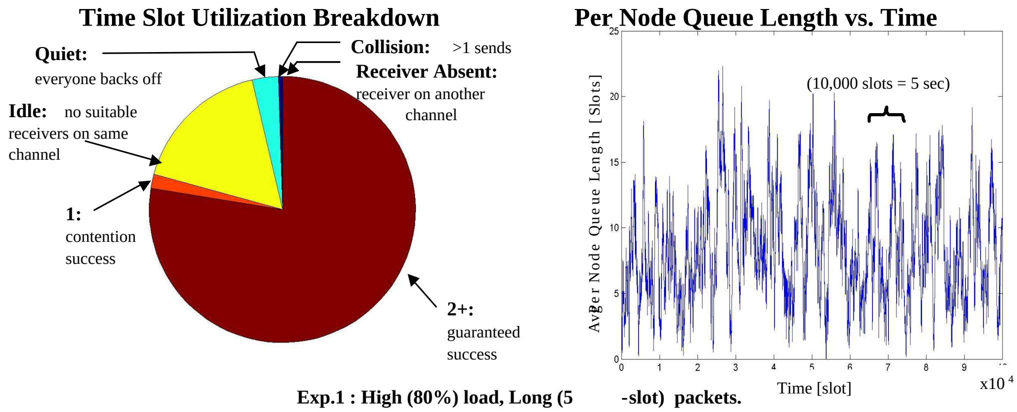

Figure 2: Time Slot Utilization and Average Queue Lengths under Heavy Load and Long Packets.

between a pair of nodes but none of the senders ended up transmitting due to random backoffs. A collision slot is one in which two or more nodes try to transmit during the same slot, resulting in a wasted slot. A slot is denoted receiver-absent when a sender wins the medium contention in a band but it turns out that the receiver is currently on a different band.

As we can see from Figure 2(left), most of the slots are follow-up successes, so every band is used very efficiently. Even though most senders are backlogged and the queues are long, as soon as a node gets hold of the medium, a larger number of the successive slots are used efficiently without contention (i.e., packet trains). As a result, the ratio of follow-up slots to first success slots is very high. This simulation scenario shows that when the packets are long, the MAC protocol is able to use the bands very efficiently.

Behavior Under Short Packets

In this experiment, we simulate the situation in which a large number of nodes send a single packet to a random destination. The data arrival rate to the network is the same as in the previous experiment but the data packets and the flow durations are shorter. This is usually the worst case for contention based MAC protocol because many nodes are contending at anytime. Furthermore, the overhead of contention cannot be amortized over several packets. From Figure 3(right), we see that the average queue length is much longer but relatively stable over time. The level of contention is high as we can see that a large fraction of slots are lost due to collision as shown in the breakdown of slots usage. Consequently, the average medium access delay reaches 1700 slots which is about 850ms. Overall, however, given the worst case traffic pattern and a very high traffic load of 80%, the system remains stable under all simulation runs we have tried. The results are encouraging as the proposed MAC scheme based on the idea of parallel contention in all bands appear to be robust to various traffic patterns and achieve reasonable delays.

5 Research Plan

As we discussed earlier, the research focuses on five tasks: 1) Design of Channels; 2) Design of Band-Hopping Sequences; 3) Synchronization; 4) Band Selection; 5) Channel Selection.

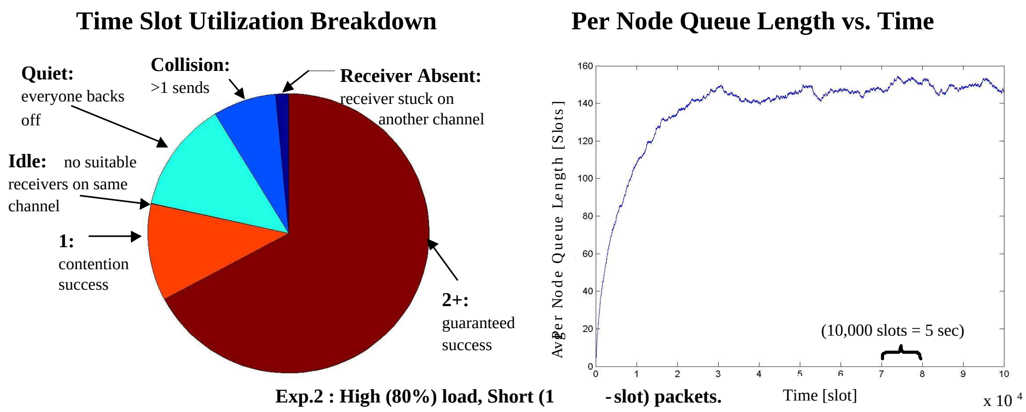

Figure 3: Time Slot Utilization and Average Queue Lengths under Heavy Load and Short Packets.

5.1 Design of Channels

The channels within each band should be design to have good orthogonality properties and to facilitate the identification of the channel to be used. In the preliminary model, we assume that a transmitter uses all the sub-carriers of its band. In that case, the different nodes in the band use a time-sharing protocol. This is the scheme that 802.11 uses. To be more flexible, and to accommodate radios with different power limitations, we will design channels that consist of sequences of subsets of carriers. One basic research question is the trade-off between orthogonality and the number of channels. Another question concerns the possibility that, for complexity and cost limitations, some radios may be able to modulate and demodulate only a subset of the sub-carriers of the band. We will study the selection of channels with these objectives and limitations.

5.2 Design of Band-Hopping Sequences

The system that we proposed uses pseudo-random band-hopping sequences. This is only one of many possible designs. One immediate question is the design of the hopping rate. This rate has an impact on the time to find a receiver and the synchronization requirements.

5.3 Synchronization

Synchronization is needed at various time scales. Within a packet, the receiver must synchronize with the sender on each OFDM symbol boundary. Pairwise synchronization can be done quite easily by adding a preamble to the beginning of each packet to allow the receiver to recover the OFDM symbol boundaries as in [14]. For an ad hoc network, however, one also needs to synchronize the different senders in the same band so that their OFDM sequences remain mostly orthogonal. When the clock of a sender slowly drifts, it will occupy the same carrier as another sender. If the overlap is only occasional, a suitable channel code can mask the resulting bit errors. However, if the senders are completely asynchronous, the system behaves like an CDMA system in which the receiver has to deal with the significant interference from other nearby senders. For best performance, synchronization is therefore needed among neighbors using a band in the same area.

The 802.11 family of protocols defines a distributed Time Synchronization Function for synchronizing the nodes in an ad hoc network. This algorithm can achieve a synchronization of $4\mu s$. which is sufficient for predicting the band hopping sequences, but insufficient for keeping OFDM sequences orthogonal within a single band. Therefore, we will explore distributed protocol mechanisms to allow nodes within one hop in the same band to synchronize to the symbol boundaries. However, one do not need to synchronize neighbor nodes on a constant basis to such accuracy when they are not communicating.

5.4 Band Selection

We will examine other rendez-vous schemes and compare them with the pseudo-random scheme. The question is whether the scheme should be based on the type of mobility of the nodes.

5.5 Channel Selection

In the simple MAC that we explored, there is no channel selection problem since the nodes use all the sub-carriers of the band. If they use orthogonal sequences of sub-carriers, then the MAC protocol could operate as follows. When a transmitter A jumps to the home band of an intended receiver B, A starts by listening to the band to discover a free channel. The algorithm uses the pattern of busy sub-carriers that A hears in a few slots. Once it identifies a free channel, A transmits its "hello B" message on that channel. Since it listens in parallel on all the sub-carriers, if idle, B should be able to hear this message as it is transmitted on an otherwise free channel. One can envision that different channels be used by nodes with different QoS requirements. One pair of nodes A and B may be sending a constant bit rate traffic using one of the channels while the other node pairs contend for the remaining channels to send bursty data traffic. Of course, these mechanism require further study to understand its performance.

6 Deliverables

The outcome of this research program will be the architecture of a family of MAC and PHY protocols compatible across a wide range of powers, bit rates, and transmission ranges. The architecture will be validated through extensive analysis and simulations. Moreover, the implementation of this architecture will be explored with industrial partners. In particular, CommStack has expressed its interest in working with us to make sure that our design could be implemented using their technology.

7 Results from Prior NSF Support

7.1 Jean Walrand

Program Title: WebTP : A User-Centered Web Transfer Protocol

Sep. 1, 1998 - Aug 31, 2001. Joint proposal with Venkat Anantharam, Steve McCanne, Pravin Varaiya, and David Tse.

This project demonstrates a working implementation of a new protocol that is aware of user preferences. The protocol is receiver driven and uses application level framing. The activity of this project can be found at http://webtp.eecs.berkeley.edu. Work done includes research on the design of user-centric optimization of protocols, possible improvements of TCP, and architecture for web-optimized transmission protocols. The students supported by this grant were Linhai He, Jeonghoon Mo, and Wilson So.

[1] "The Framework of User-Centric Optimization in Web-Based Applications," Y. Xia, H-S. W. So, R. La, V. Anantharam, S. McCanne, D. Tse, J. Walrand, and P. Varaiya, *Memorandum No. UCB/ERL M 00/52*, Jan. 15, 2000.

[2] "The WebTP Architecture and Algorithms," Y. Xia, H-S. W. So, R. La, V. Anantharam, S. McCanne, D. Tse, J. Walrand, and P. Varaiya, *Memorandum No. UCB/ERL M 00/53*, Jan. 15, 2000

[3] Jeonghoon Mo, Richard J. La, V. Anantharam, and J. Walrand, "Analysis and Comparison of TCP Reno and TCP Vegas". *Proc. IEEE INFOCOM 1999*, New York, pp. 1556 -63.

[4] L. He and J. Walrand, "Stochastic Approximations and Transaction-Level Models for IP Network Design," *IEEE CDC2000*, Sydney, December 2000.

Online Statistics for ATM Networks

NCR-9628818; July 1, 1996 to June 30, 1999

The objective of this research is to clarify the possibilities and methods of control of high-speed networks. These networks include not only the asynchronous transfer mode networks (ATM) but also high-speed networks based on the TCP/IP protocols. The high-speed creates new problems related to the large rate-delay product that tends to make traditional window-based congestion control mechanisms ineffective.

The control of high-speed networks pauses new challenges. The large rate-delay product of the connections makes the usual end-to-end window-based congestion control mechanisms less effective than in lower speed networks. Intuitively, the large rate-delay product requires a large window size for a retransmission protocol to be efficient. However, the source cannot control where the large number of packets in its window are stored in the network. It may happen that most packets pile up in one router which is then congested. This intuitive argument suggests that a more effective control mechanism might be to pace the transmissions of packets with rate control mechanisms such as leaky buckets, as is re commanded for ATM networks. In addition to congestion control, we study call admission control, which is needed if quality of service is guaranteed instead of being provided on a best-effort basis.

The following students were supported in part by this award: I. Hsu, S. Papadakis, G. de Veciana, N. McKeown.

[1] Matthew Siler and Jean Walrand, "On-line Measurement of QoS for Call Admission Control," Presented at *IWQoS*, Napa, May 18, 1998.

[2] Venkat Anantharam, Nick McKeown, and Adisak Mekkittikul, and J. Walrand, "Achieving 100% throughput in an input-queued switch," *IEEE Comm,* vol. 47, no. 8, pp. 1260-67, August 1999.

[3] C.F. Su, G. De Veciana, and J. Walrand, "Explicit rate flow control for ABR Services in ATM Networks," *ACM/ IEEE Trans. Networking,* 2000.