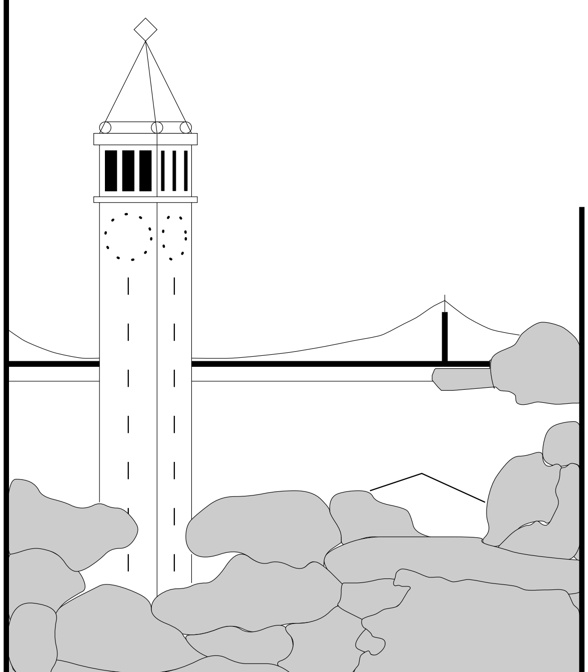
Reducing the Energy Consumption of Group Driven Ad-hoc Wireless Communication

Sharad Agarwal, Randy H. Katz, Anthony D. Joseph



Report No. UCB/CSD-1-1127

January 2001

Computer Science Division (EECS)
University of California
Berkeley, California 94720

This work was supported by DARPA Contract No. N00014-99-C-0322.

Abstract

Mobile ad-hoc networking involves peer-to-peer communication in a network with a dynamically changing topology. Energy efficient communication in such a network is more challenging than in cellular networks since there is no centralized arbiter such as a base station that can administer power management. In this report, we examine energy efficient networking protocols for ad-hoc networks. We propose MAC (media access control) and routing protocols that significantly impact the energy conservation and throughput of ad-hoc networks. We introduce a comprehensive simulation infrastructure consisting of group mobility, group communication and terrain blockage models. We employ these models to evaluate a power control loop, similar to those commonly found in cellular CDMA networks, for ad-hoc wireless networks. We show that this power control loop reduces energy consumption per transmitted byte by 10 - 20%. Furthermore, we show that it increases overall throughput by 15%. It performs significantly better in the new proposed simulation models than in simulations with random node mobility and communication models. We also discuss sleep cycle and low energy routing mechanisms.

1 Introduction

Ad-hoc wireless networking is receiving renewed attention. It enables many interesting usage scenarios but poses several challenges. Traditionally, wireless networking has been applied to cellular telephony and Internet connectivity via radio modems. These systems provide single hop connectivity to a fixed, wired base station. Ad-hoc wireless network systems attempt to form multi-hop networks without pre-configured network topologies. There is *peer-to-peer* interaction among nodes, unlike in cellular networks where nodes communicate with a centralized base station. Ad-hoc networks are characterized by dynamically changing topologies, a direct result of the mobility of the nodes. Such systems can offer many advantages. They do not rely on extensive and expensive installations of fixed base stations throughout the usage area. With the availability of multiple routes to the same node or base station, they can perform route selection, based on various metrics such as robustness and energy cost. They can use more direct routes to communicate between nodes without using more distant base stations, and thus can conserve energy and improve bandwidth. These systems enable various applications, ranging from the monitoring of herds of animals to supporting communication in military battlefields [JTRS ????] and civilian disaster recovery scenarios.

Many of these applications require that nodes be mobile and be deployed with little network planning. The mobility of nodes limits their size, which in turn limits the energy reserves available to them. Thus energy conservation is a key requirement in the design of ad-hoc networks. In wireless networks, bandwidth is precious and scarce. Simultaneous transmissions in domains which use the same bandwidth interfere with each other. Thus bandwidth re-use is also important.

Power management techniques such as power control help combat long term fading effects and interference. When power control is administered, a transmitter will use the minimum transmit power level that is required to communicate with the desired receiver. This ensures that enough transmit power is used to establish link closure but not higher than that necessary. This minimizes interference caused by this transmission to others in the vicinity. This improves both bandwidth and energy consumption. However, unlike in cellular networks where base stations make centralized decisions about power control settings, in ad-hoc networks power control needs to be managed in a *distributed* fashion.

In Section 2 we present a power control loop for ad-hoc wireless networks, which reduces energy consumption by 10 - 20% and improves throughput by 15%. We also describe the common sleep cycle mechanisms available for wireless networks. They further reduce energy consumption by turning off the transmitter and receiver circuitry when no messages are to be transmitted and no receptions are anticipated. We provide an estimate of the performance gain we can achieve by using sleep cycling in conjunction with power control.

A single best algorithm for all of the issues in ad-hoc wireless networking, such as media access control and routing, over all usage scenarios, is unlikely to be found. Unfortunately, the performance metrics by which we should compare these algorithms are still debated in the research literature. We need to examine many aspects to make fair comparisons between alternative algorithms - overall energy consumption, distribution of energy starved nodes, throughput, latency, and robustness in face of mobility.

To compare different ad-hoc wireless networking algorithms, we need to measure these metrics in the context of usage scenarios for which the solutions are being developed. Most of the ad-hoc wireless networking literature has focused on networks of small sensor-driven devices. The typical example is a large number of mobile or immobile sensors that have been scattered in a field or building that are collecting data about their environment. With ad-hoc wireless networking, data from these sensors can be disseminated among the sensors and to the end users, and control information can be passed down from the end users to the sensor devices [Heinzelman et al. 2000, Estrin et al. 1999]. However, such networks of sensors are still being developed. Thus, no mobility and traffic patterns from such systems exist. Most of the published works in this area employs random mobility and traffic patterns, making it methodologically difficult to compare different works and to ascertain their true impact in real scenarios.

In Section 3 we describe the simulation infrastructure we have built to simulate realistic usage in ad-hoc networks. We have made an effort to model the node mobility, communication traffic and environment likely to be experienced in the scenarios with which we are concerned.

We illustrate the power of our infrastructure and models by using them to evaluate our power control loop and approximate sleep cycle mechanism in Section 4. We emphasize the value of our simulation models by showing that the power control loop performs better than when only random node mobility and communication models are considered. Our power control loop improves energy consumption by 10% and throughput by 5% in the random models, but by 10-20% and 15% respectively in the group mobility models. Without these more realistic models, the better algorithms might have been missed.

As ongoing work, in Section 5 we present a low energy routing protocol that functions on top of a media access control (MAC) layer with power control and sleep cycling. This routing protocol performs route selection based on the cost of communication, rather than the latency of communication. This report concludes in Section 6.

2 Low Energy MAC Communication

In this section, we describe energy conservation techniques at the MAC layer. The goal here is to minimize the energy cost of communication between any given pair of neighboring nodes if such communication is possible. Ad-hoc networks can contain nodes of various types, of which many can have limited power capabilities and may not be able to scavenge energy from sources such as solar energy. Furthermore, many of the data gathering applications for which these networks are deployed are latency tolerant. Thus, energy efficiency rather than latency should be the principle design goal in MAC communication.

Two main mechanisms for energy conservation at the MAC layer are power control and sleep cycling. Power control loops for various cellular telephony systems have been studied extensively in the past and are used in commercially deployed systems [Ojanperä and Prasad 1998, Lee 1989]. They are especially important in ad-hoc networks due to the higher levels of interference. We have applied power control to the IEEE 802.11 MAC [1] specification [IEEE 1999], thereby achieving lower energy consumption and higher throughput.

The motivation for incorporating sleep cycling comes from the fact that a radio transceiver consumes energy if it is powered, even if no transmissions are being sent or received by that transceiver. By careful scheduling and / or through the use of additional signaling, the radio transceiver can be placed in a "hibernate" state that consumes little energy when it is idle. We provide an overview of common sleep cycle mechanisms. We also present an optimistic estimate of the performance gain that we achieve by using sleep cycling in conjunction with power control.

In this section, we begin by describing the general concept behind power control and refer to related work. In a following subsection, we describe the IEEE 802.11 MAC protocol, which is the base MAC protocol we use for evaluating power control and sleep cycling. We then describe our power control loop. We follow this with a subsection on sleep cycle mechanisms for wireless communication and describe our approximate sleep cycle mechanism. In the following sections, we describe our simulation infrastructure and results.

2.1 Power Control

In cellular systems, a base station tells mobile units to adjust their transmit powers by measuring the power received from them. Cellular systems are used for applications such as telephony where the pre-installation of a fixed base station infrastructure is feasible. Cellular systems have star topologies and every mobile unit communicates *exclusively* with

[1] Neither the IEEE 802.11 specification nor its commercial implementations such as WaveLAN [Lucent ????] use power control.

an associated base station.

An ad-hoc network on the other hand does not have a centralized arbiter which can tell each node the transmit power to use to communicate with a particular receiver. Furthermore, well defined cells or domains do not exist. Thus power control in an ad-hoc network is not trivial and needs to be administered in a *distributed* manner. However, the benefits of power management remain. Instead of every node using the same transmit power, if a node uses only the power level that is required to communicate with a desired receiver, it might extend it's battery life. Furthermore, it will reduce interference seen by other simultaneous transmissions in the network.

2.2 Related Work: Power Control Loops in Cellular Networks

Power control loops for various cellular telephony systems have been studied extensively in the past and are used in commercially deployed systems [Ojanperä and Prasad 1998, Lee 1989]. The related literature is vast, and we will not attempt a complete survey. Instead, we describe the basic concept behind power control loops in CDMA systems.

One of the main goals of power control is to avoid the near-far effect. Since transmitted signals experience propagation loss, signals received by a base station from a closer mobile station will be stronger than those received from one that is further away. Thus distant mobile stations will not experience a fair share of the available throughput to the base station. We must avoid this near-far effect so that the signal strength of all mobile stations received at the base station must be the same. Similarly, another goal of power control is to reduce the interference that a mobile station experiences from different base stations near the edge of a cell. In spread spectrum networks, especially in CDMA networks, power control is necessary to reduce the average noise level. If the average noise level is too high, it is not possible to recover the spread signal from the background noise.

Both open loop and closed loop power control mechanisms have been explored in CDMA systems. Open loop control attempts to measure at the mobile station the path loss between itself and the base station. Using the received signal strength of messages and various control parameters transmitted by the base station, the mobile station can set its transmit power level. This mechanism does not always achieve the best transmit power level because the path loss experienced on the uplink and downlink will differ (especially if different frequencies, transceivers and power supplies are used for the uplink and downlink).

Closed loop power control treats uplink and downlink power control separately. The base station measures the received signal-to-interference ratio (SIR) over a short time period and decides whether the mobile station should raise or drop its transmission power level by comparing the received SIR to the optimal SIR value. This decision is transmitted to the mobile station on the downlink. The mobile station then adjusts it's transmit power levels accordingly. The base station determines the optimal SIR value by an outer control loop that considers the error rate of the uplink. CDMA systems use a similar closed loop power control to adjust the downlink transmit power levels. The base station periodically reduces it's transmit power levels. The mobile station measures the error rate of the downlink and requests additional power from the base station if the error rate is unacceptable. The downlink control loop iterates at a frequency at least an order of magnitude lower than the uplink control loop.

Reference [Narendran et al. 1997] in particular describes an adaptive closed loop power control algorithm for cellular CDMA networks that is similar to the one we propose in this report for ad-hoc networks. They describe a scheme where the receiver observes the quality of the received signal (bit error rate and signal strength) and sends control information to the transmitter if it is inadequate. This results in the transmitter boosting it's transmit signal strength and altering its CDMA code. Reference [Narendran et al. 1997] applies this technique to cellular CDMA networks, and the simulations presented consist of hexagonal cell layouts with each cell consisting of randomly moving nodes that communicate with only a base station. We apply adaptive closed loop power control to the signaling protocol in an ad-hoc wireless MAC and we present results from simulations of realistic usage scenarios for such a network.

2.3 Related Work: Power Control Loops in Ad-Hoc Networks

Ad-hoc wireless networks provide a different set of challenges than standard cellular telephony and packet radio networks. We cannot arrange the network in a pre-surveyed cellular fashion. Each node communicates directly with many nodes rather than just one base station. Thus interference may be a more significant issue.

Reference [Kwon and Gerla 1999] attempts to impose a cellular structure to an ad-hoc network topology. Each cluster head acts like a cellular base station. They use open loop and closed loop power control in a similar fashion as described above in cellular networks, but specifically to control the size of a cluster. Their main goal is to reduce the number of network topology changes as the speeds of nodes vary.

Reference [Ramanathan and Rosales-Hain 2000] is similar to [Kwon and Gerla 1999] but formulates the problem differently. They approach it as an optimization problem, minimizing the maximum transmit energy consumed while maintaining connectivity constraints. In the LILT and LINT algorithms that they propose, they restrict the number of neighbors of a node to a certain value by altering the transmit power level of the node in question for all of it's communication. They do not exchange any extra control information. They use heuristics that compare the length of the list of neighbors that most ad-hoc routing protocols maintain to the optimal length.

The power control loop mechanism we present is different from prior work in several ways. Our focus is on ad-hoc networks, not cellular systems as in references [Ojanperä and Prasad 1998, Lee 1989, Narendran et al. 1997, Kwon and Gerla 1999]. We allow each node to choose different transmit power levels for different neighboring nodes. It is not a goal of the work we present here to reduce the connectivity of the nodes (as in references [Kwon and Gerla 1999, Ramanathan and Rosales-Hain 2000]). The goal is to allow all nodes to communicate with all of their neighbors, but by having each node choose different transmit power levels for each of it's neighbors, interference will be reduced. A system that ties each node to a single transmit level for all communication, as in reference [Ramanathan and Rosales-Hain 2000], will experience higher interference because excessive transmit power will be spent on nearby nodes. We investigate whether a power control loop in an ad-hoc wireless MAC can reduce energy consumption and increase overall throughput. We investigate this issue in the context of realistic simulation models and the relevance of these models in evaluating ad-hoc wireless networks.

We have applied a power control loop to the IEEE 802.11 MAC [IEEE 1999]. In the following subsection, the relevant part of the IEEE 802.11 MAC specification is briefly described. We follow it by a description of the modifications we made to support power control.

2.4 IEEE 802.11 MAC Signaling System

We consider adding power control to ad-hoc wireless medium access to reduce energy consumption by reducing both the transmission energy and average RF (radio frequency) interference. We incorporate a power control algorithm into the IEEE 802.11 MAC protocol [IEEE 1999], which has been popular for ad-hoc networks [Broch et al. 1998b]. The power control modifications that we propose in the next subsection involve piggy-backing extra control information in the IEEE 802.11 MAC's signaling. These modifications are applicable to any ad-hoc MAC protocol that employs a signaling scheme similar to the one specified in the IEEE 802.11 standard. In the remainder of this subsection, we provide an overview of the relevant parts of this standard.

There are two basic message types that the IEEE 802.11 MAC layer generates - `(a)` broadcast messages and `(b)` messages destined for a specific host within the node's radio range (hereby referred to as addressed messages). When the MAC layer of a node generates a broadcast message, which is destined for all hosts within the node's range, the node simply transmits the message without any extra signaling.

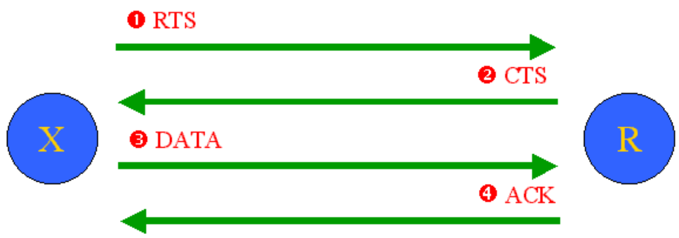


Figure 1: Signaling Between a Transmitter (X) and a Receiver (R) for Addressed Messages

When a node has to transmit an addressed message, it uses a signaling protocol (`RTS-CTS-DATA-ACK`) (see Figure 1) that is more complicated than the one used for broadcasting. This protocol includes the generation of messages that inform the destination and other neighboring nodes about a forthcoming data transmission, and thus reduces the effects of *hidden terminals* [Bharghavan et al. 1994]. These nodes will defer other transmissions during this period and thereby reduce interference. First, the transmitter sends an `RTS` (request-to-send) message. If the receiver receives this `RTS` message successfully and it is not already engaged in a different `RTS-CTS-DATA-ACK` exchange, it will reply with a `CTS` (clear-to-send) message. If no `CTS` message is sent, or if it is lost, the transmitter will time out and re-attempt the transmission of the `RTS` message. During the `RTS-CTS` exchange, other hosts that are in the radio range of the receiver and / or the transmitter will see the `RTS` and / or `CTS` message(s) in flight and can thus assume that the transmission medium is busy with that transaction for a certain period of time. Upon successful reception of the `CTS` message, the transmitter will transmit the actual addressed message as a `DATA` message. If the receiver receives this message successfully, it will reply with an `ACK` (acknowledgment) message.

2.5 Modifications to Incorporate Power Control

We now describe our modifications to the IEEE 802.11 MAC specification for addressed messages to support power control. In the original IEEE 802.11 MAC, all transmissions occur at the same transmit power level. For our power control loop, we allow this transmit level to be any one of ten levels. These levels vary linearly between the default transmit power level (the maximum) and one-tenth of this value[2]. We scale down the energy to

[2]In comparison, AMPS (Advanced Mobile Phone System) uses eight power levels [Goodman 1997]. GSM (Global System for Mobile communications) uses eight to fifteen levels, depending on the unit's maximum transmitter power.

transmit a message by the transmit power level chosen by our power control loop. Thus, if a node transmits a message at half the default power level, we will reduce it's energy budget by half the default transmit energy penalty. The energy level that a node uses in the transmission of messages is controlled by the power amplifier that drives the antenna on an RF transceiver. It is possible that the power amplifier energy consumption levels will not match the transmit power levels due to inefficiencies in it's design. We will explore the result of having such inefficiencies in the simulation results that we present in Section 4.

We also alter the message header formats for `CTS` and `DATA` messages to include a value which is the ratio of the received signal strength of the last received message to the minimum acceptable signal strength at the node currently transmitting the message [3]. When a receiver receives an `RTS` message, it will encode the ratio of the received signal strength of the `RTS` message to the minimum signal strength that is acceptable by this receiver in the header of the `CTS` reply message. As a simplification, we assume that the minimum acceptable signal strength is a known constant value for each node. Similarly, when transmitting the `DATA` message, the transmitter will encode into it the ratio of the received signal strength of the received `CTS` message to the minimum acceptable signal strength at the transmitter. Thus, during one `RTS`-`CTS`-`DATA`-`ACK` exchange, both the transmitter and the receiver inform each other about the quality of their transmitted signals. Both nodes have the opportunity to alter their transmit power levels for further communication between each other.

The MAC layer for each node maintains a small table that stores power control settings for other nodes with which this node has recently communicated [4]. Each entry contains five fields. The first field identifies which other node that table entry is currently housing settings for. The second gives the current transmit power level setting that transmissions to this other node will use (which are in one tenth increments of the maximum transmit power level). The `cf_pwr` field maintains an EWA (exponential weighted average) history of the ratio (received signal strength to minimum acceptable signal strength) received from this other node in either a `CTS` header or a `DATA` header. The `dr_pwr` field maintains an EWA history of the `cf_pwr` field at points where packet losses occurred. The last field is

[3] We could design a power control loop where the signal strength ratio is not sent in the header, but instead a simple raise or lower transmit power level signal is sent. The disadvantage with this scheme is that heterogeneous nodes with different transceivers and battery power levels would not be able to accurately gauge the optimal transmit power level for each other.

[4] In our simulations, we consider tables with ten entries. We use a random replacement policy for when a node comes across more than ten nodes in it's lifetime. The size or replacement policy of the table is not a major issue as the size of each entry is small and space for more table entries should be easily allocatable if the need arises. However, it is unlikely that a node will communicate directly with more than a few neighbors at any point in time.

a count-down timer field that dampens rapid fluctuations in transmit power levels.

When a message is sent to a node, we look it up in the table. If we do not find it, we clear an entry selected at random and allocate it for that node. We initialize the `cf_pwr` and `dr_pwr` fields to one and we set the transmit power level entry to the maximum (see Section 4 for simulation results from setting this initial value to a lower setting). When a `CTS` or `DATA` message is received from a node, we update it's entry in the table (if none exists, one will be allocated). We re-evaluate the `cf_pwr` entry by adding the ratio received in the `CTS` or `DATA` message header to the EWA history. If the `cf_pwr` field is higher than the `dr_pwr` field, we decrement the transmit power level field by one, unless the count-down timer field is not null.

When the MAC times out while waiting for a `CTS` or `DATA` or `ACK` message from a node, we update the entry in the table for that node. We increment the transmit power level field by one. We update the `dr_pwr` field by adding the current `cf_pwr` field value to the EWA history. We set the count-down field to ten. This ensures that for the next ten message transmissions to this node, the transmit power level field will not be decremented. We chose this value of ten to dampen rapid fluctuations while not hampering the overall effectiveness of the power control loop.

Our modifications apply to addressed messages between any pair of nodes. We have not extended the power control algorithm to broadcast messages where only a `DATA` message is transmitted. These messages are typically used for routing purposes (for messages that request a route to a particular node that is not a neighbor, or presence information from all neighbors). They are not destined for a specific receiver and do not involve a sequence of message exchanges. It is not possible for the MAC layer to decide what transmit power level they should use. Thus, these messages continue to be transmitted at the default (maximum) transmit power level.

The essential goal of the above algorithm is to *learn* the minimum transmit power level required for a node to successfully transmit to a remote node. Starting with an initial value for the transmit power level, the exchange and loss of messages causes the MAC layer to *ratchet* up (or down) the transmit power level. The MAC layer of a node thus learns the unique minimum transmit power level required for that node to successfully transmit to any other remote node. A lower level will result in lost packets. This level is unique for every node with this node communicates. This level can change, and so the algorithm continuously tracks the returned signal strength ratio and determines if the transmit power level can be dropped further.

We use the simulation infrastructure that we describe in the next section to evaluate our power control algorithm in Section 4. In the next subsection, we describe sleep cycling, which is another mechanism for achieving energy savings at the MAC layer. We also evaluate sleep cycling in conjunction with power control in Section 4.

2.6 Sleep Cycling

A significant amount of energy is consumed in a node when it is idle since it overhears other transmissions that it is not involved with. The goal of sleep cycling is to keep a node powered off when no transmissions need to be sent by and none are expected for it. This is typically achieved in one of two ways. Time scheduling can be used, wherein nodes that are idle (i.e., are not receiving or transmitting packets) sleep and wake up periodically. A node that wishes to transmit to a sleeping node will have to transmit a message repeatedly until it is able to reach the destination node in one of its awake periods. In an alternate scheme, a node will detect the initiation of a lengthy transmission by appropriate signaling and go to sleep for the duration of the transmission. In this subsection, we briefly describe the concept behind one such scheme, `PAMAS` (Power-Aware Multiple Access protocol with Signaling) [Singh et al. 1998]. We approximate the benefits of such a sleep cycle mechanism and evaluate it in conjunction with power control in Section 4.

In any transmission from a given node A to a neighboring node B, there might be spectator nodes in the vicinity (say node C). These nodes do not take part in the transmission or reception, but invariably overhear the transmission because they are close to either the transmitter or the receiver or both. Thus, node C will expend energy in receiving this transmission, processing it and forwarding it up to the MAC or routing layer before discarding it. Sleep cycle mechanisms such as `PAMAS` reduce the energy that node C spends on these operations.

In `PAMAS`, the duration of the lengthy transmission that determines the sleep period is indicated by an extra field embedded in `RTS` and `CTS` messages. The `RTS`-`CTS` exchange occurs in a separate channel in reference [Singh et al. 1998]. The protocol has been augmented to allow other nodes to query communicating nodes about the length of their transmissions. This additional feature enables nodes that have woken up from their sleep cycles in the middle of unrelated transmissions to fall asleep again. Reference [Singh et al. 1998] reports that `PAMAS` helps achieve about 40-70% savings in battery power. Those results do not consider power control.

While sleep cycling and power control are separate mechanisms that address different issues, there is some overlap. Since power control reduces the energy of transmissions, the range over which the transmission is overheard is smaller. Thus fewer nodes overhear transmissions and do not spend energy receiving and processing them, thereby reducing the added benefit of sleep cycling. However, in a well connected network, the number of spectator nodes cannot be reduced to zero by power control all the time. The benefit from keeping the transceiver powered down during other transmissions will remain unchanged. On the other hand, just having sleep cycling does not eliminate all the savings that power control can provide. Thus, power control and sleep cycling complement each other and can together enhance the energy savings in a network.

We have not incorporated a detailed sleep cycle mechanism into our simulation infrastructure. However, we have approximately modeled the effects of sleep cycling in our simulations. We replenish the energy consumed in a node in overhearing `CTS`, `DATA` and `ACK` messages of third party communications. However, note that since the node is unaware of the actual duration of such communication, it might have to wake up periodically to check the status of the channel. This periodic "waking up" results in some energy consumption that we do not take into account. We use this approximate model of sleep cycle behavior to evaluate the added benefit of sleep cycling when power control is present at the MAC layer. We present results of simulations with sleep cycling and power control in Section 4. In the following section, we describe the simulation models that we use to produce those results.

3 Group Mobility Simulation Models

To evaluate new algorithms for use in ad-hoc wireless networks, realistic usage patterns need to be employed. It is a common practice in published works to use random node placement, mobility and traffic patterns. These patterns do not accurately model real usage of ad-hoc wireless networks. In deployed ad-hoc networks, there are many physical obstacles to radio waves. In the scenarios we consider, nodes do not move randomly, but move with certain patterns in coordinated groups. Traffic is not random, but represents a well defined flow of data and control between the nodes.

The mobility, traffic and blockage models that we present here can be used to model many real usage scenarios. Various animals (such as wolves, birds and fish) and wilderness explorers (such as hikers and skiers) tend to travel in groups. Environmentalists wishing to track the movements of these animals may attach radio transceivers to them. They can form an ad-hoc network, allowing various location and sensor readings to propagate to distant base stations. Law enforcement officers, military troops, fire fighters and medical personnel also move and work in groups. Modeling such environments using random patterns is inadequate for the evaluation of new networking algorithms. We must apply group mobility and traffic patterns and we must model the blockages that would be experienced in real usage.

3.1 Related Work: Realistic Scenario Modeling

Much of the research literature in ad-hoc wireless networking resorts to inaccurate and unrealistic random models. We summarize in this subsection various other models that have been proposed in the past. They fall short in modeling realistic usage scenarios. What are needed are node mobility patterns, communication patterns and a propagation environment. In the remaining subsections, we present the group mobility patterns, hierarchical communication patterns and terrain blockage models that we employ in our simulation infrastructure.

One main usage scenario is the military battlefield of the future. Scattered troops and vehicles will need to communicate via a network formed in an ad-hoc fashion. Reference [Graff et al. 1998] provides an example of a hierarchical tactical military network control structure to motivate their work on the application of mobile IP and CIDR to such networks. However, they do not present simulations of such node placement and communication patterns. Reference [Stine and Veciana 1998] uses a static arrangement of nodes in a tactical network, where nodes are part of different network groupings (cohorts). One member of each of the groupings is part of a larger grouping. They use this model in simulations, but without any mobility patterns or terrain models. However, they do use bursty voice traffic models which conform to the network grouping model. Reference [Antkiewicz et al. 1998] presents a very detailed simulation of a tactical network. They support node mobility and the loss of nodes due to enemy fire and jamming attacks. However, the simulator does not generate mobility patterns; they must be specified by the user. Similarly, the user has to specify traffic probability distributions, both of connectivity and duration.

Reference [Johansson et al. 1999] describes three different scenarios - conference, event coverage and disaster area. In each scenario, they place nodes in various clusters and move them randomly within the boundaries of the cluster. Each cluster has different speed settings for each node, depending on the location of the cluster. The clusters themselves do not move. They place various obstacles in each scenario that completely block all transmissions passing through them. For each scenario, they designate some nodes as data sources and others as data sinks for constant bit rate flows. They describe scenarios that involve limited group movements, numbers of nodes and deployment areas. We present more sophisticated simulation models targeted at larger scenarios involving group mobility. We move nodes in their groups as a whole, while giving each node the same trajectory as their group but with a slight variance. Reference [Johansson et al. 1999] does not explain how traffic sources and destinations are assigned. We set up traffic flows in a hierarchical manner: flows among nodes within the same group and flows between groups. Our blockage models are more sophisticated in that transmission loss is not absolute. It varies based on the loss characteristics of the obstacles and the nature of how the obstacle blocks the transmission.

Reference [Kwon and Gerla 1999] presents a survey of various mobility models and investigates the impact of group mobility on the performance of various routing protocols. They conclude that random mobility models do not accurately predict the performance of routing protocols in real usage scenarios. The group mobility model that we present is similar to theirs. However, we additionally incorporate group communication patterns and terrain models to further improve the fidelity of our simulation results.

3.2 Mobility Models

To model group movement, we pre-generate motion vectors for each node and feed them into our simulations. The persons that require such a model need to specify various high level parameters : the number of nodes to be simulated, the size of a group, the maximum speed of a node and a random number generator seed. Using this list of characteristics, our simulation infrastructure generates motion vectors for individual nodes. We initially place the nodes with their groups, and place the groups at random within the simulated field of variable size. We give each group a randomly chosen trajectory and speed. This is random because more accurate models of group movements require profiles of a specific application, which are not currently available. All the nodes within that group follow this chosen trajectory and speed but with a small random variance. This small variance is meant to model real life effects, such as animals or people moving with varying speeds and application specific responsibilities. Once this motion vector has been followed for a certain random period of time, the group will pause for a short period and will choose another vector. The group pauses at each destination to simulate the fulfillment of an application specific goal such as the investigation of an area or the collection of sensor data. This model is shown in Figure 2.

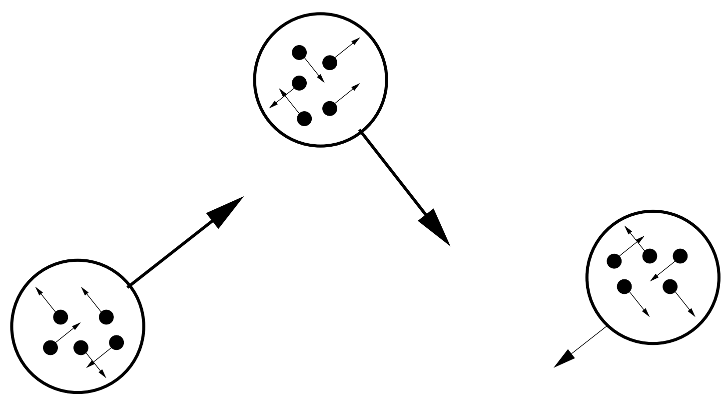


Figure 2: Group Mobility

3.3 Traffic Models

The nodes (animals, humans or machines) within a group want to share sensor and location information so that local inferences about sensor readings can be made. Thus the traffic pattern will consist of intra- and inter- group communications of data and control.

To model such traffic, we pre-generate intra-group and inter-group traffic patterns that we feed into our simulations. We assign each group an identification number (`ID`) from 0 to the total number of groups, `G`. We give each node in the group an `ID` from 0 to the total number of nodes in a group, `N`. For intra-group communications, for each group member, we pick the member node with the next higher `ID` (with a wraparound to 0). We may initiate a tcp session between these two nodes, with probability `0.5`, at a random time. We may start a second connection with the node with the second higher `ID` with probability `0.25`. For inter-group communications, for each group, we pick a random group member node and we pick a random node from the group with the next higher `ID`. We may initiate a connection between these two nodes, with probability `0.5`, at a random time. Similarly, we may start a second connection as well, with probability `0.25`, with the group with the second higher *ID*.

The resulting communication pattern will have roughly `0.75 * N` connections distributed among the nodes in each group, with each connection starting at a random time. There will be roughly `0.75 * G` connections between groups. The goal of this model is to mimic the connection pattern that will likely be used in real usage. We do not attempt to model the amount or rate of the actual data flow because we do not have traces of real usage.

3.4 Blockage Models

The areas where ad-hoc networks will be deployed are rarely open, flat terrain. It is probable that various forms of foliage, mountainous terrain, buildings, enemy RF jammers and inclement weather (rain, snow, hail) will be experienced. This interference will manifest itself as RF transmissions being impaired. Thus we make an effort to model blockages so that our simulations will more accurately reflect realistic transmission effects due to blockages.

We can accurately model every feature of every kind of blockage to infinite detail. However, it is important for us to maintain a balance between modeling accuracy and simulation run time. We need to ascertain to what extent modeling approximation impacts the accuracy of simulation results. Throughout this subsection, we identify where an approximation has resulted from a consideration of simulation run time.

For every transmission from a transmitter to every potential receiver, we must quickly determine whether the line of sight between the two nodes intersects a blockage. The more complicated the shape of the blockage[5], the more complicated the algorithm for determining the intersection. Thus we model all blockages as spheres of varying radii. It is a fairly trivial calculation to determine whether a line intersects a sphere. The shortest distance from the center of the sphere to the line of sight has to be smaller than the radius.

Nonetheless, it can be a fairly large task for us to consider all the blockage spheres for every potential receiver for every transmission. To reduce the enormity of this task, we catalog all the blockages in a simulation by their locations. We store this in a multi-level quad-cif tree by splitting the simulated field into quadrants, each quadrant being split into

[5]When considering the shape of the blockage, the issue under consideration is not the actual shape of the blockage, but rather the shape of the area of influence of the blockage. A blockage will cause interference in signals that pass through an area that is larger than the actual area of the blockage. We refer to this larger area when we use the phrase "the shape of the blockage".

more quadrants. For every potential receiver for every transmission, we pick the smallest quadrant that contains both the receiver and the transmitter. We consider only those blockages that reside either wholly or partially in that quadrant. In this way, we can keep the impact of blockage modeling on simulation run time at a minimum. Some blockages can be mobile - rain and mobile enemy jamming devices are some examples. Apart from giving blockages an initial position, we can also give them velocity vectors at any time in the simulation, much like mobile nodes. We re-arrange these blockages at the appropriate times in the quad-cif tree. At present, we allow nodes to pass through blockages. This would be accurate for simulating people walking through heavy foliage or weather conditions. As ongoing work, we will expand this model to include blockages that do not allow nodes to pass through them, such as hills and buildings.

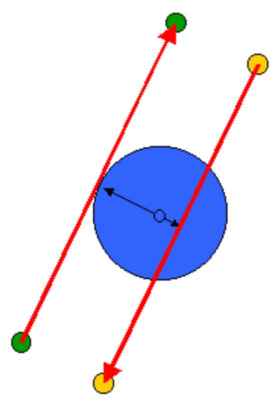


Figure 3: Blockage Model

When we consider the effect of a blockage on a transmission, we first determine whether it is blocking the transmission. If it is, we drop the energy of the transmission by a certain amount, which depends on the characteristics of the blockage. Each blockage has a certain loss factor (blockage density), L, associated with it. Taking into account the wavelength of the transmission λ, the radius of the blockage sphere R, the shortest distance between the center of the sphere and the point of intersection d and the energy of the transmission E, we calculate the drop in energy of the transmission using the following equation:

$$E * (1 - (\frac{\lambda * d}{4 * \pi * R})^2 * \frac{1}{L})$$

The above equation is a crude approximation of the true blockage loss that would occur to a transmission. The main features of the above equation are that the loss is greater the closer the intersection is to the center and the loss is greater the smaller L is. Thus in Figure 3, the communication between the lighter colored nodes will incur higher energy loss (interference) than that between the darker colored nodes. In our simulation environment, it is trivial to model more complicated blockages (e.g. those that discriminate based on frequency or use more complex blockage loss equations). This added complexity will be a direct trade off against simulation time. For each transmission, we consider blockages in no particular order. Furthermore, we apply the blockage losses after we calculate and remove the free space propagation loss from the energy of the transmission. Both of these approximations should have a minor impact on simulation accuracy.

4 Simulation Environment and Results

We have incorporated these group mobility, traffic and blockage models into our simulation environment. We use this environment to evaluate both these models and our power control algorithm. While the emphasis of this report is on realistic usage models, we also present results from completely random models to justify the need for more real usage models. In this section, we describe the simulation environment and the experimental setup using `(a)` the random models and `(b)` realistic models using the group mobility, traffic and blockage patterns. We then present our simulation results. We also evaluate our approximate sleep cycle model.

4.1 Network Simulator

We use the UCB/LBNL discrete event network simulator, NS (version 2.1b6) [NS-2 2.1b6], which is now under development as part of the VINT project. We chose NS because of its CMU Monarch project extensions that support various ad-hoc routing protocols and its extensibility. The NS simulator contains an implementation of the IEEE 802.11 MAC standard [IEEE 1999] which executes above a wireless RF (radio frequency) physical layer. The physical layer is a model of a DSSS radio interface (Lucent WaveLan direct-sequence spread-spectrum). We have modified the physical and MAC layers to support our power control loop algorithm.

4.2 Simulation Setup of Random Models

Our goal in presenting results of simulations using random models is to contrast them with those from real usage models. Here we list the simulation setup parameters that we use. We choose these values to be representative of a typical usage scenario, while not forcing a large delay on the simulation run time. It is arguable if a more realistic set of parameters could have been chosen, but these values are not unreasonable.

In the random simulation models, we do not use any blockage models. We run each of the simulations with pre-generated node placement, movement pattern and traffic patterns. Initial node placement within the field is random. We select random speeds and directions for each node at random times to generate movement patterns. For traffic patterns, we pick two nodes at random, and at a random time, we initiate a TCP session. We vary the seeds used for the random number generators to produce a set of results that we average and show the highest and lowest values in error bars. [6]

[6] We choose the seeds themselves arbitrarily. We produce twenty five sets of results for every simulation

Table 1 lists various simulation setup parameters. As we mentioned earlier, each wireless node simulates a typical WaveLAN physical layer operating at 914 Mhz with a throughput of 244 KBps. Using the given specifications for Lucent WaveLAN cards [Lucent ????], the energy budget of 1 Joule allows each node to transmit and/or receive about 800 KB at full transmit power. Most nodes die of energy starvation by the end of each simulation which lasts 10 seconds. When nodes run out of energy, they can no longer transmit or receive messages. The 10 second simulation time provides ample duration for route discovery and formation.

Table 1: Random Simulation Model Setup Parameters

Field Length	500 meters
Field Width	500 meters
Simulation Time	10 seconds
Traffic Type	TCP sessions
# of TCP Sessions	2-28
# of Nodes	30
Energy Budget / Node	1 Joule
Routing Protocol	DSR[Broch et al. 1998a]
Max Node Speeds	1, 10 & 20 m/s

4.3 Simulation Setup of Group Mobility Models

The group mobility simulation setup is similar to the random model setup with some additions. Here, we use our blockage models, group mobility and connectivity patterns. We place and move the blockages in the field using a randomly generated pattern. We simulate them with a high loss factor (i.e., L is very small). We choose these blockage settings to represent slow moving blockages, such as inclement weather or enemy jamming devices moving through a battlefield. Table 2 lists these additional parameters.

In the next two subsections, we describe the format of the graphs that we use to present our results in the following three subsections. Those results are based on the simulation parameters we have described above.

setup. The number of results is also arbitrary

Table 2: Additional Group Mobility Simulation Setup Parameters

Nodes / Group	5
# of Blockages	4
Max Blockage Radius	10 meters
Max Blockage Speed	2 m/s

4.4 Throughput Graph Description

The throughput graphs contrast the performance of the modified MAC with power control to the unmodified MAC with fixed power transmissions. Each point in the "throughput comparison" graphs (Figures 4, 5, 6, 7, 12, 13, 14, 15, 20) depicts the percentage :

$$100 * \frac{TotalThroughput(PowerControlMAC)}{TotalThroughput(FixedPowerMAC)}$$

We define throughput as the total number of TCP session data bytes successfully transferred during the 10 seconds of simulation time for each run. Data bytes that are retransmitted due to a loss in the data packets or acknowledgement messages or due to other reasons are accounted for (i.e., duplicate data bytes are not counted).

As described earlier, we conducted many simulations for each point using different random number generator seeds. Each point in the graphs shows the average value, and the vertical error bars show the *maximum* and *minimum* values.

For each of the random and cohort simulation models, we vary both the maximum speed of nodes and the total number of TCP connections that we instantiate. We first present a graph where the maximum speed is 1 m/s and we vary the number of connections on the horizontal axis. We then present a graph with 10 m/s and 20 m/s speeds. We then concatenate these three graphs and zoom in on the main data in the fourth graph, where we vary the speed and total number of TCP connections instantiated along the horizontal axis. There is no particular scale across the horizontal axes - the goal of the variation along them is to show the robustness of power control across various situations.

4.5 Energy per Byte Graph Description

The "energy per byte comparison" graphs are similar to the throughput graphs except that each point in the graphs depicts the following percentage:

$$100 \quad * \frac{TotalEConsumed}{TotalThroughput}(PowerControlMAC)$$

$$*\frac{TotalThroughput}{TotalEConsumed}(FixedPowerMAC)$$

NS (version 2.1b6) only accounts for the energy consumed in receiving a message and the energy consumed in transmitting a message. Thus the energy consumed value that we use indicates the energy spent only in the simulated radio transceiver. This value assumes that the transceiver does not consume any energy when idle. Each node has an energy budget and when it exhausts this energy budget, it can no longer receive or transmit messages.

4.6 Random Simulation Model Results

Figures 4, 5, 6 and 7 show the improvement in total throughput of the power control MAC versus the unmodified MAC in simulations with random placement, mobility and traffic models. A value above 100% indicates that the power control MAC modifications achieved a higher overall throughput. As we mentioned earlier, each point shows the average value of twenty five runs and the error bars show the maximum and minimum values obtained. In each of the first three graphs, the maximum speed is 1 m/s, 10 m/s and 20 m/s respectively and the fourth graph is a condensed version of these three graphs.

Figures 8, 9, 10 and 11 show the improvement in the energy consumed per transmitted byte. A value below 100% indicates that the power control MAC modifications consumed less energy overall.

Figures 7 and 11 show that the power control loop MAC achieves modest improvements in throughput and energy consumption. The variance is high, as indicated by the error bars. In simulations where communicating nodes happen to be at the maximum communication range distance between each other, the power control loop does not help. In fact, it consumes more energy and reduces throughput due to the extra control overhead transmitted in the message headers. However, in simulations where the communicating nodes happen to be close to each other, the power control loop successfully reduces the transmit power levels and reduces the total energy consumed. This can be due to both a reduction in the transmission energy consumed and due to a reduction in overall interference from other nodes. On average, it consumes about 10% less energy and improves the overall throughput of the system by about 5% compared to the unmodified MAC. However, with our group mobility, traffic and blockage models, the power control MAC achieves significant improvements on throughput and energy consumption.

4.7 Group Mobility Simulation Results

The simulations of group mobility and traffic patterns with blockage modeling produce more dramatic results. Figure 15 shows that the power control MAC offers roughly 15%

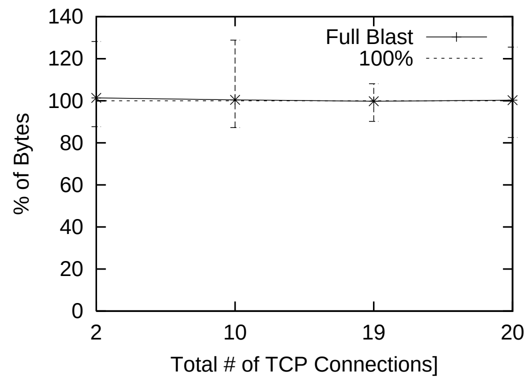


Figure 4: Throughput Comparison Between The Two MACs (Random) (Speed=1 m/s)

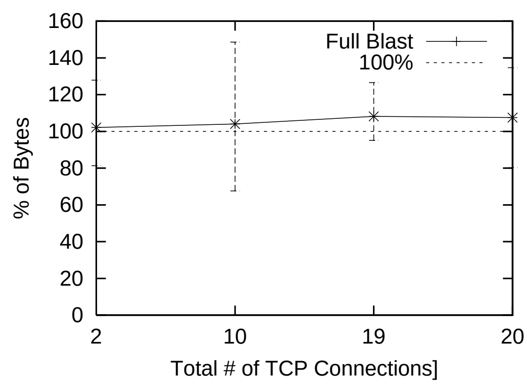


Figure 5: Throughput Comparison Between The Two MACs (Random) (Speed=10 m/s)

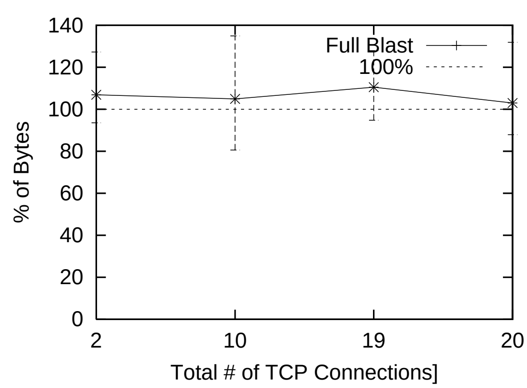


Figure 6: Throughput Comparison Between The Two MACs (Random) (Speed=20 m/s)

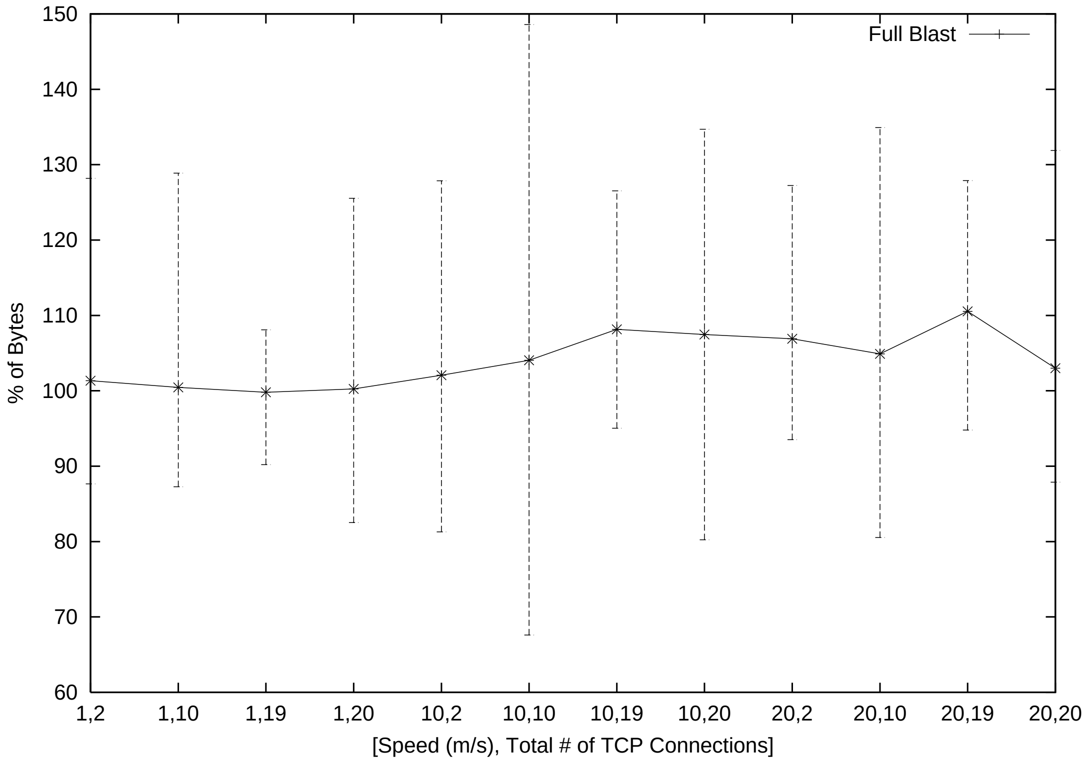


Figure 7: Throughput Comparison Between The Two MACs (Random)

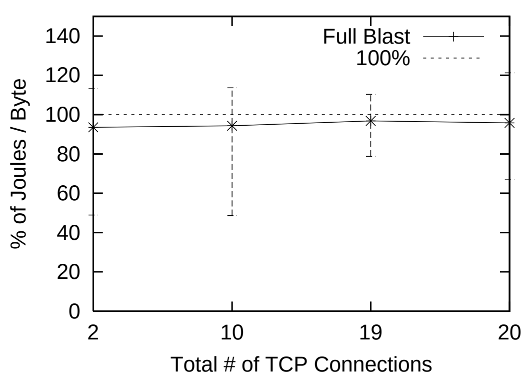


Figure 8: Energy per Byte Comparison Between The Two MACs (Random) (Speed=1 m/s)

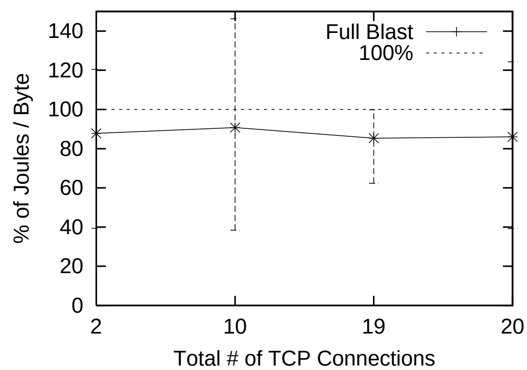


Figure 9: Energy per Byte Comparison Between The Two MACs (Random) (Speed=10 m/s)

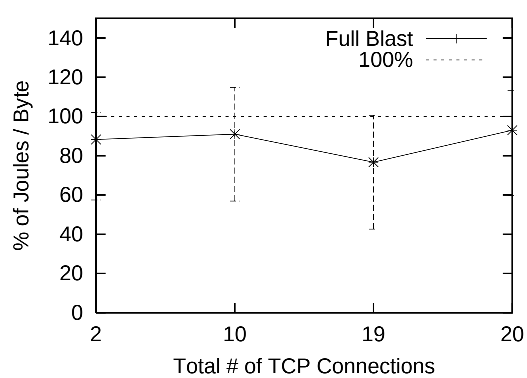


Figure 10: Energy per Byte Comparison Between The Two MACs (Random) (Speed=20 m/s)

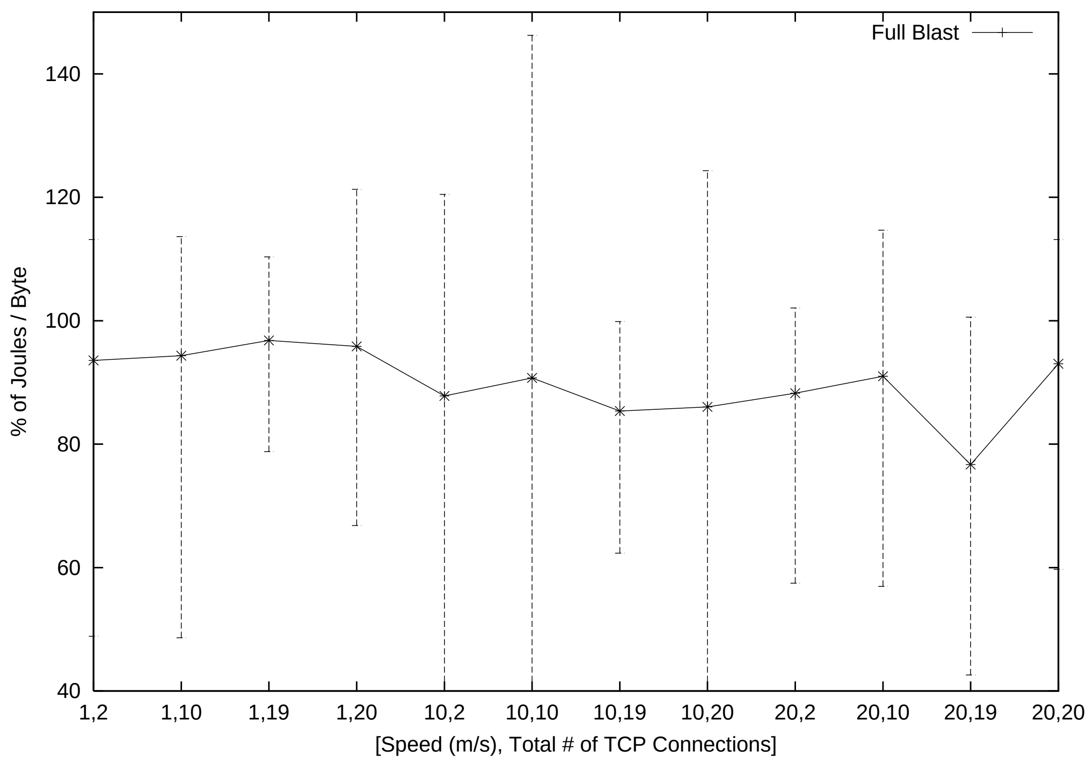


Figure 11: Energy per Byte Comparison Between The Two MACs (Random)

higher throughput than the standard MAC. Also, the minimum error bars are mostly above the 95% line and the maximum error bars are much higher in these simulation results.

Similarly, Figure 19 shows that the overall energy consumption of the power control MAC is about 10 - 20% lower than the fixed power MAC. Again, the high error bars are much lower in this graph than in the random simulation graphs and the low error bars are much lower, showing that for some cases, there is good potential for much more significant savings. These results vary significantly from the random simulations. The random simulations test many different scenarios, many of which may not be relevant to real usage scenarios. Thus it is important to test new algorithms in ad-hoc networking in models of real usage scenarios, rather than in random models to assess the real impact of these new algorithms.

As described in Section 2, the power control loop initially starts off at the highest power setting when initiating communication with a node. During the course of future communication, it ratchets the power setting down to a level just above which packet loss occurs (this shall be referred to as "full blast"). An alternative mechanism initially starts off at the lowest power setting and then ratchets up to the power level at which packets are accepted (this shall be referred to as "low blast").

The low blast mechanism has the advantage when nodes are close to each other. If only two nodes initially communicate, they will ratchet their transmission power levels to the minimum level. In full blast, if two other nodes start communicating, they start off at the highest setting, causing high interference with the two nodes already in communication. Those two nodes will then raise their power level due to this interference, and as a result, a "shouting match" ensues. With low blast, the potential for this occurring is lower.

As expected, Figures 15 and 19 show that the low blast algorithm has a lower energy consumption pattern and offers slightly higher overall throughput. However, it experiences an initial delay in communication as the nodes ratchet up their transmission power levels until communication can be established. This extra delay is only experienced when initiating communication with a node for the first time. Once connected, extra delay will not be incurred again. This behavior is similar to TCP slow start, where the number of packets that are transmitted is altered in an additive increase, multiplicative decrease manner. In full blast mode, communication is established at the first transmission itself. This increase in latency for low blast has not been quantified in this report and should be considered when implementing this power control loop in a MAC.

Figures 20 and 21 are similar to the previous two graphs except that the horizontal axes are different. All the points on the graphs simulate a maximum node speed of 10m/s and a connection count of 28 TCP sessions. These graphs attempt to show how the benefits of the power control MAC change as the density of nodes varies. We vary the simulated field

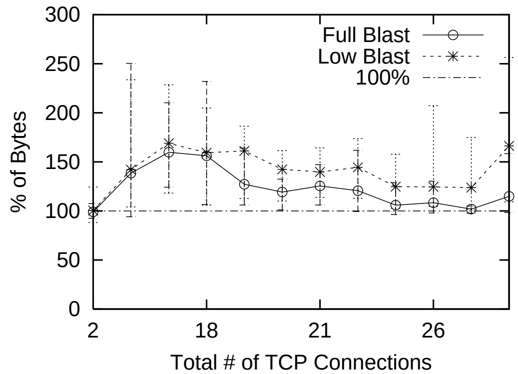


Figure 12: Throughput Comparison Between The Two MACs (Speed=1 m/s)

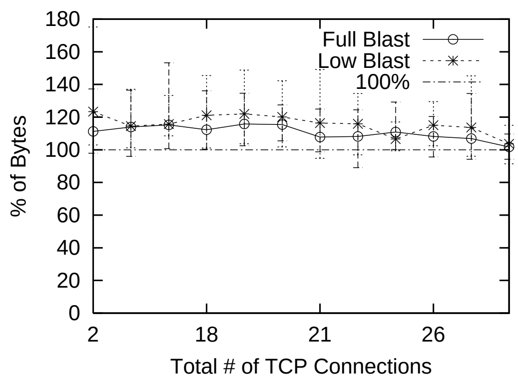


Figure 13: Throughput Comparison Between The Two MACs (Speed=10 m/s)

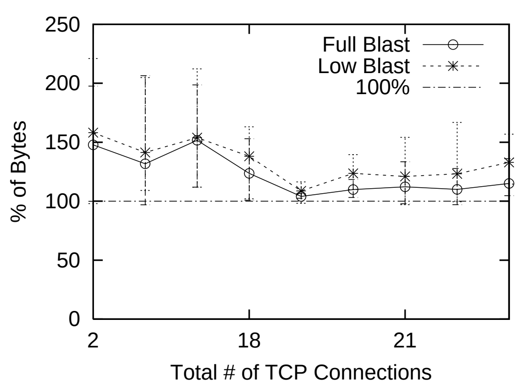


Figure 14: Throughput Comparison Between The Two MACs (Speed=20 m/s)

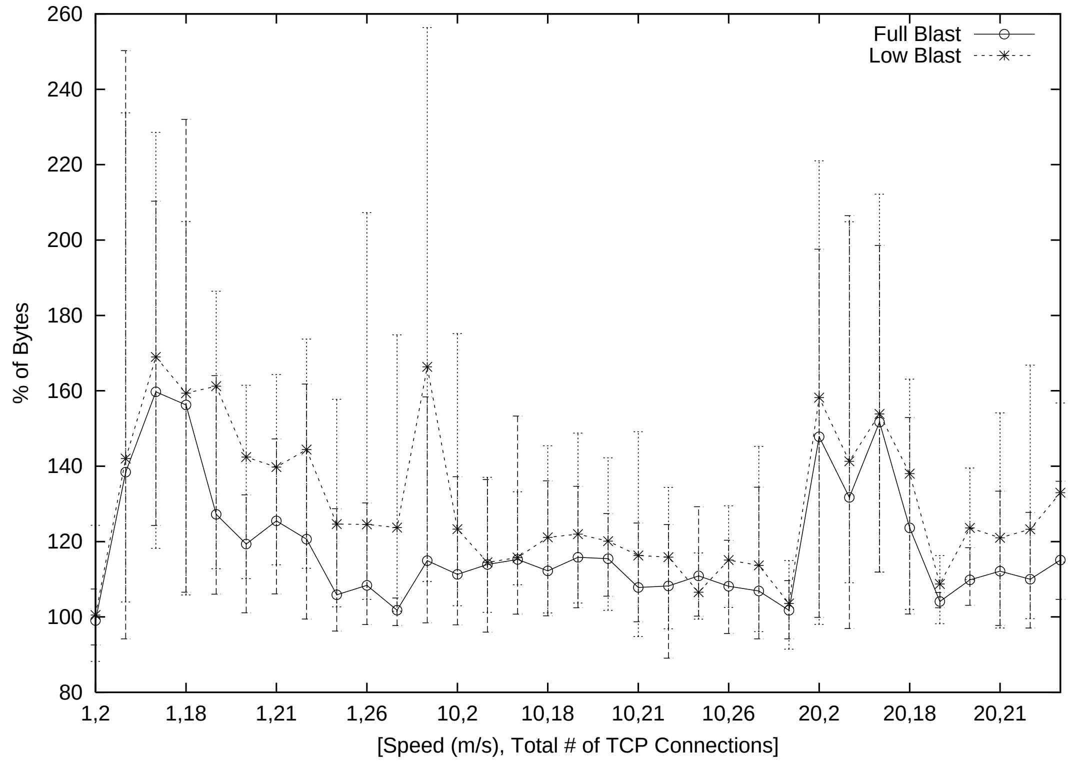


Figure 15: Throughput Comparison Between The Two MACs

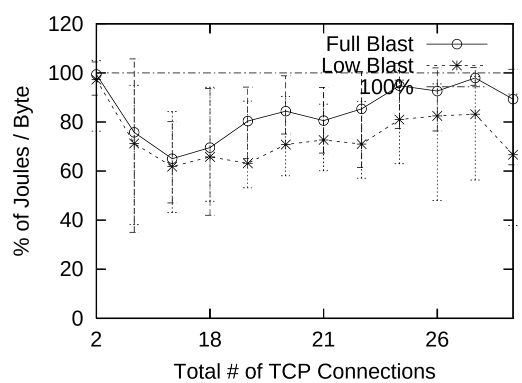


Figure 16: Energy per Byte Comparison Between The Two MACs (Speed=1 m/s)

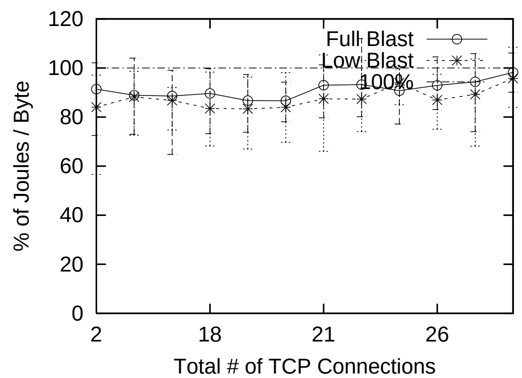


Figure 17: Energy per Byte Comparison Between The Two MACs (Speed=10 m/s)

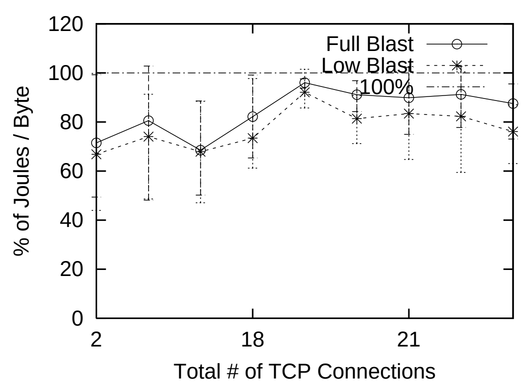


Figure 18: Energy per Byte Comparison Between The Two MACs (Speed=20 m/s)

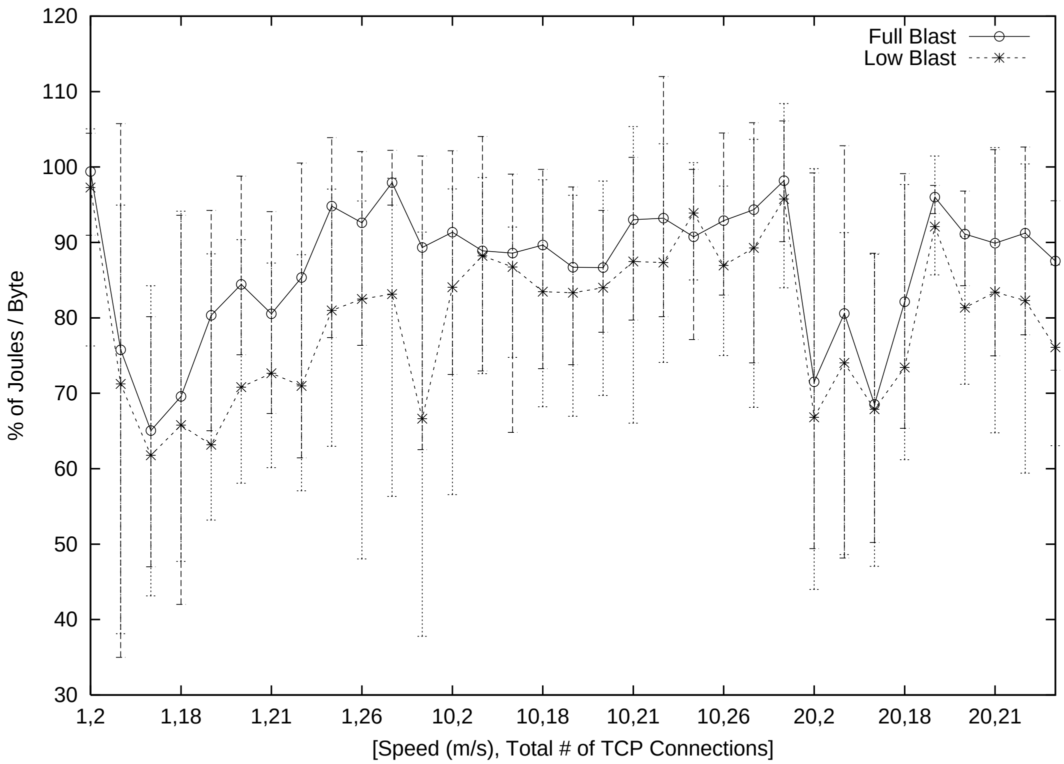


Figure 19: Energy per Byte Comparison Between The Two MACs

area from 100m by 100m to 2100m by 2100m across the horizontal axes. We keep the total number of nodes and the group size constant at 30 and 5 respectively.

As we previously hinted, the main benefits of the power control loop result from two factors. Firstly, there is the reduction in energy consumed when transmitting at a lower power level, offset by the extra energy consumed in transmitting the power control information bits in the MAC message headers. This reduction in energy consumed means that there is more energy left for transmitting more bytes, thus also increasing the throughput of the system. Secondly, by reducing the power level of transmitted signals, the power control loop reduces the average noise level. This helps to improve throughput because more data can be in flight at the same point in time. This also translates into a reduction in energy consumed because there will be fewer transmissions that result in collisions / corruptions requiring re-transmissions. Furthermore, a low average noise level is critical for CDMA and other spread spectrum networks.

One way to assess the relative benefits of these two factors is to remove the first from the simulations. By not altering the energy consumed when transmitting packets (i.e., transmitting any packet costs the same energy as in the unmodified MAC), we can study the benefit obtained only from a reduction in interference. The "Full Blast w/o Energy Budget Change" lines in Figures 20 and 21 reflect the comparison between the power control MAC and the unmodified MAC when the transmit energy budget is fixed. As is shown, up to a field size of 700m by 700m, the "Full Blast w/o Energy Budget Change" line follows the "Full Blast" line closely. This means that most of the benefits obtained by the power control are due to a reduction in interference, which is high when the nodes are distributed in a small area. As the field size increases, the "Full Blast" line rises much higher then the other line. This shows that for large field sizes, the benefits obtained are mostly due to a reduction in the transmit energy costs. In a larger field, since nodes are moving in groups, most of the intra-group TCP sessions are active and most of the inter-group sessions are not due to a lack of stable routes across the large field. The groups are spread out across the field, and so the interference between groups is relatively low. So the power control loop does not help in reducing overall interference. It does help in reducing some intra-group interference - the savings in "Full Blast w/o Energy Budget Change" at 2100m by 2100m are over 10% in both throughput and energy. Thus for scenarios where communicating groups are sparsely distributed, it is important for energy consumption and throughput to have efficient radio transceivers that can scale their energy consumption when altering transmit power levels.

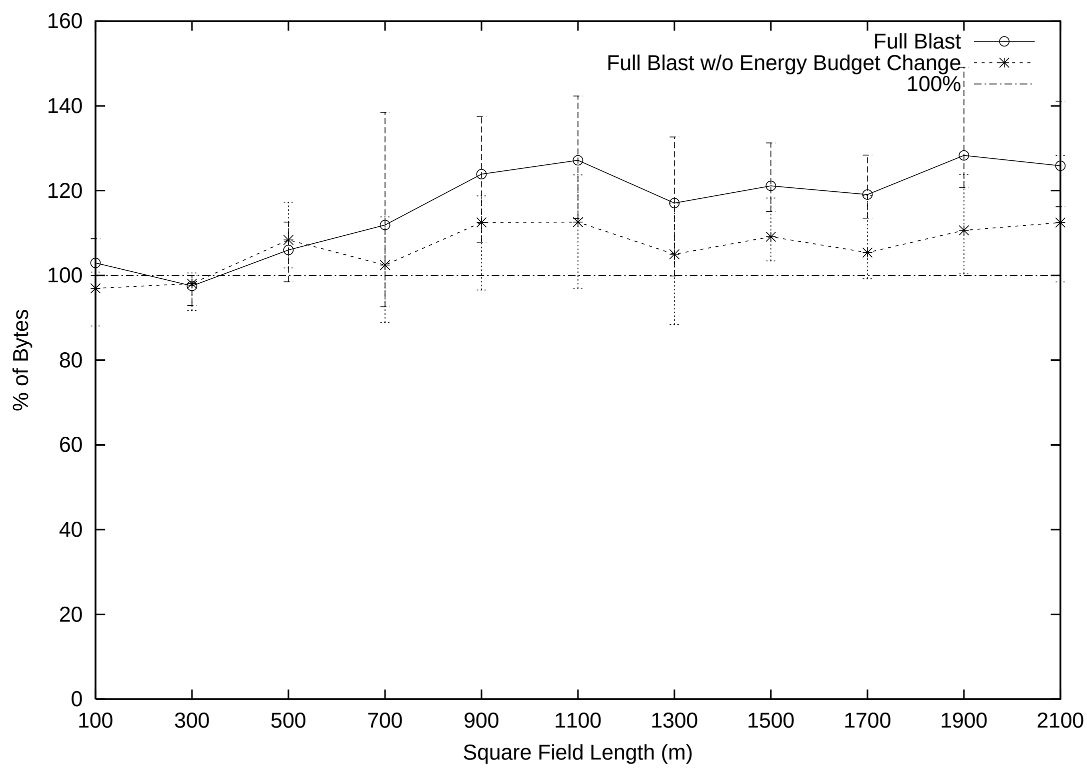


Figure 20: Throughput Comparison Between The Two MACs (Density)

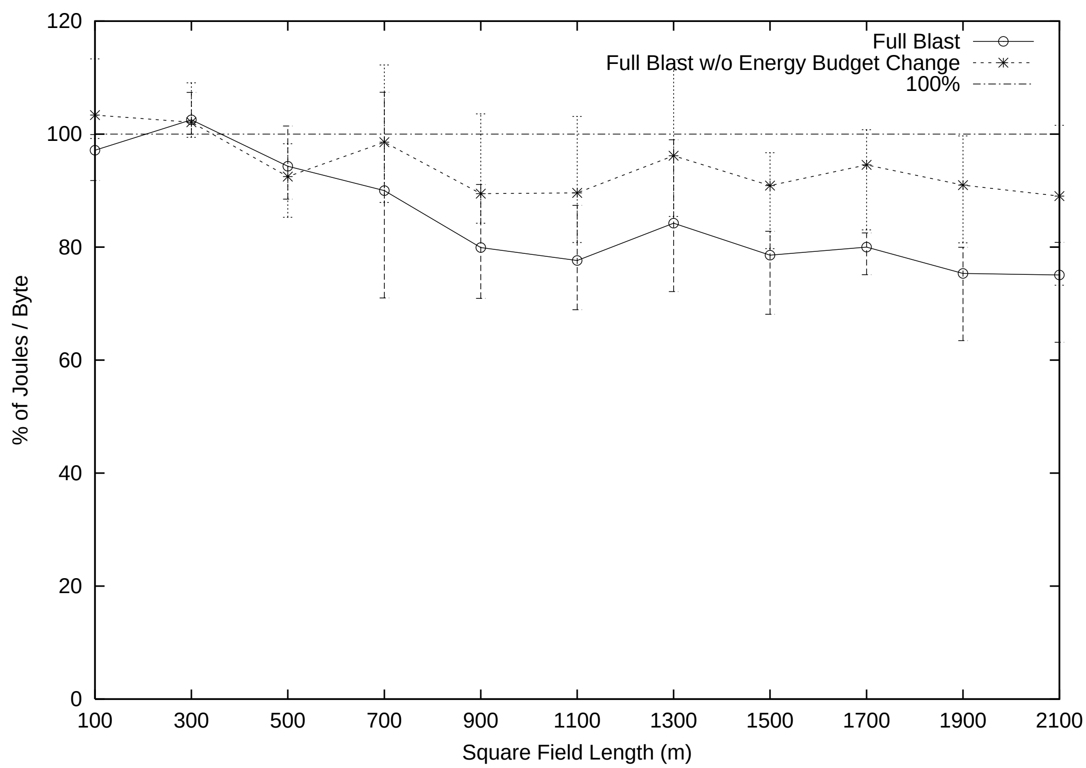


Figure 21: Energy per Byte Comparison Between The Two MACs (Density)

4.8 Added Benefit of Sleep Cycling

In the previous subsections, we have shown the benefit of using realistic simulation models to evaluate new algorithms for ad-hoc wireless networks. We have used them to evaluate our power control loop and shown that it significantly improves throughput and energy consumption. In this subsection, we now include our approximate sleep cycle model in our simulations. We now compare the standard IEEE 802.11 MAC with one that has both power control and sleep cycling.

As described in Section 2, our approximate sleep cycle mechanism replenishes the energy lost in unintentionally receiving messages. Specifically, the energy spent in receiving `CTS`, `DATA`, and `ACK` messages not addressed to the node in question will be restored. This is an optimistic model, that assumes all such receptions occur when spectator nodes are sleeping and there is no overhead in periodically waking up and sampling the medium. Thus the results presented here show the upper bound in throughput improvements and energy savings that we can achieve if we added sleep cycling to our power control algorithm.

In Figures 22, 23, 24, 25, 26, 27, 28 and 29, we show simulation results with both sleep cycling and power control. They show that we can expect to no more than double the throughput and consume no less than one-fifth the energy consumed per byte when compared with a MAC with neither power control nor sleep cycling. The throughput increases due to the energy savings. As we mentioned earlier, each node exhausts its 1 Joule in initial energy before the end of the simulation and stops transmitting and receiving messages. If each node saves energy by sleep cycling, it has more energy to transmit and receive more messages. Thus the total number of bytes transmitted over the 10 seconds of each simulation will be higher.

As mentioned, these results are approximate due to the limited sleep cycle model we have considered. Our future efforts will involve incorporating a more accurate sleep cycle mechanism into our simulation infrastructure.

In this section, we have shown that the use of simulation models based on real usage applications rather than random models are beneficial. Without them, the true benefits or pitfalls of new algorithms or optimizations may not be realized. We have quantified the advantages of using power control loops for ad-hoc wireless networks. We have also explained the cause of these improvements in throughput and energy consumption in the data that we presented. We have also presented the advantages of using MAC layers with both power control and sleep cycling.

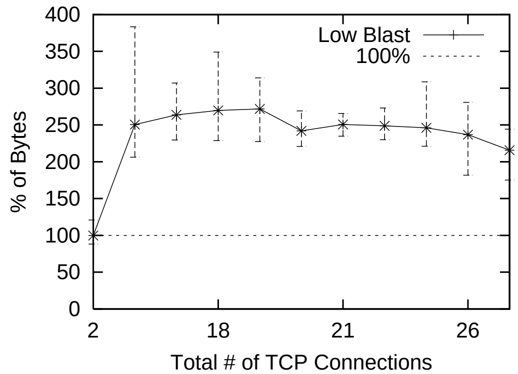


Figure 22: Throughput Comparison of Combined Sleep Cycling and Power Control (Speed=1 m/s)

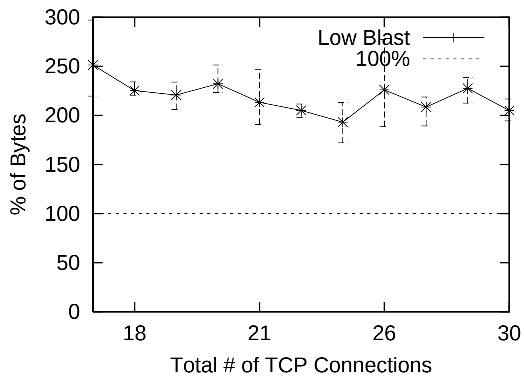


Figure 23: Throughput Comparison of Combined Sleep Cycling and Power Control (Speed=10 m/s)

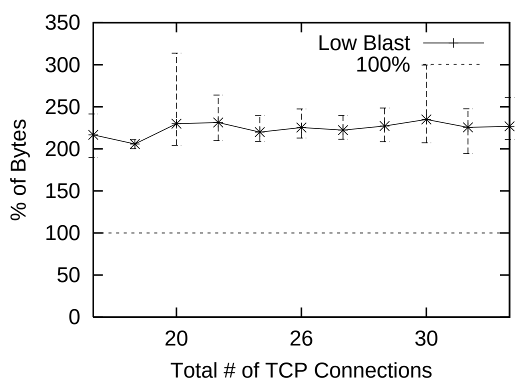


Figure 24: Throughput Comparison of Combined Sleep Cycling and Power Control (Speed=20 m/s)

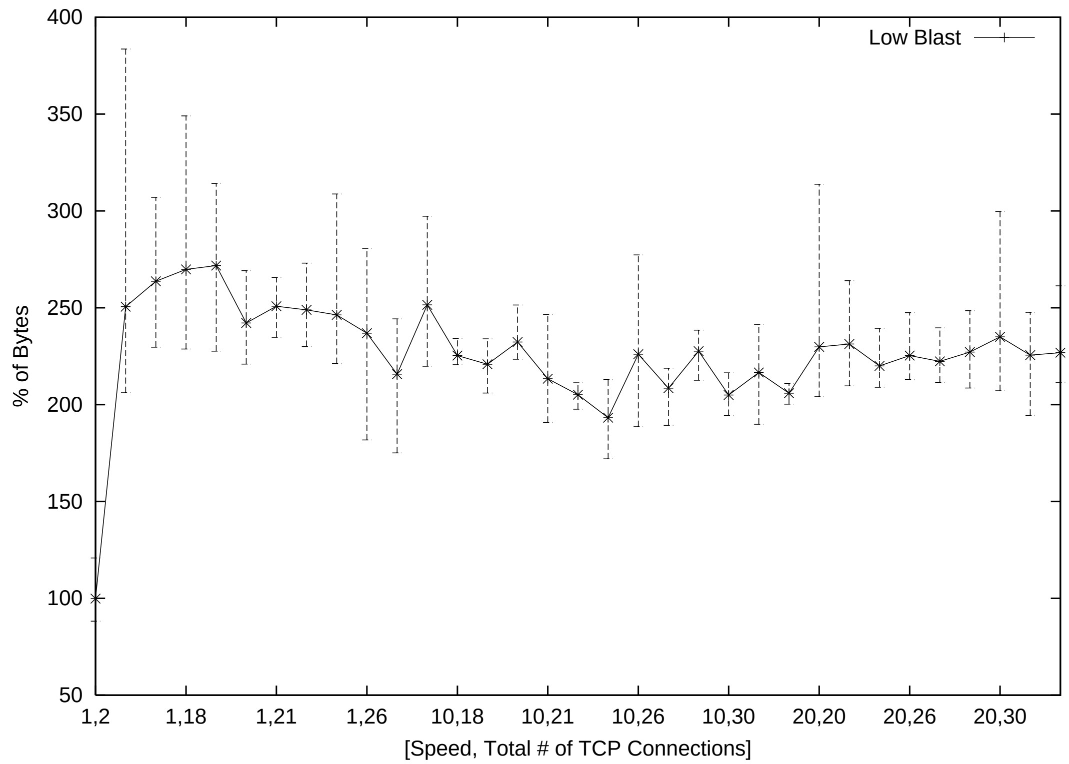


Figure 25: Throughput Comparison of Combined Sleep Cycling and Power Control

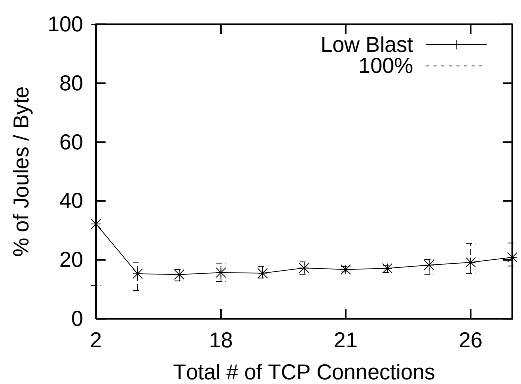


Figure 26: Energy per Byte Comparison of Combined Sleep Cycling and Power Control (Speed=1 m/s)

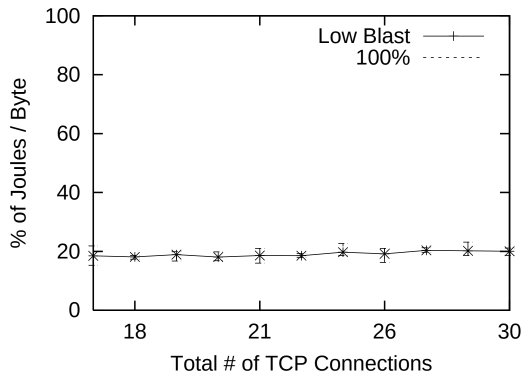


Figure 27: Energy per Byte Comparison of Combined Sleep Cycling and Power Control (Speed=10 m/s)

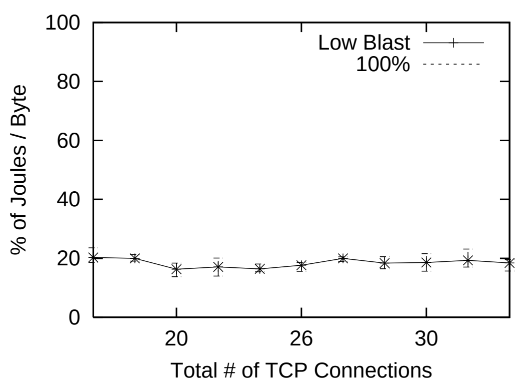


Figure 28: Energy per Byte Comparison of Combined Sleep Cycling and Power Control (Speed=20 m/s)

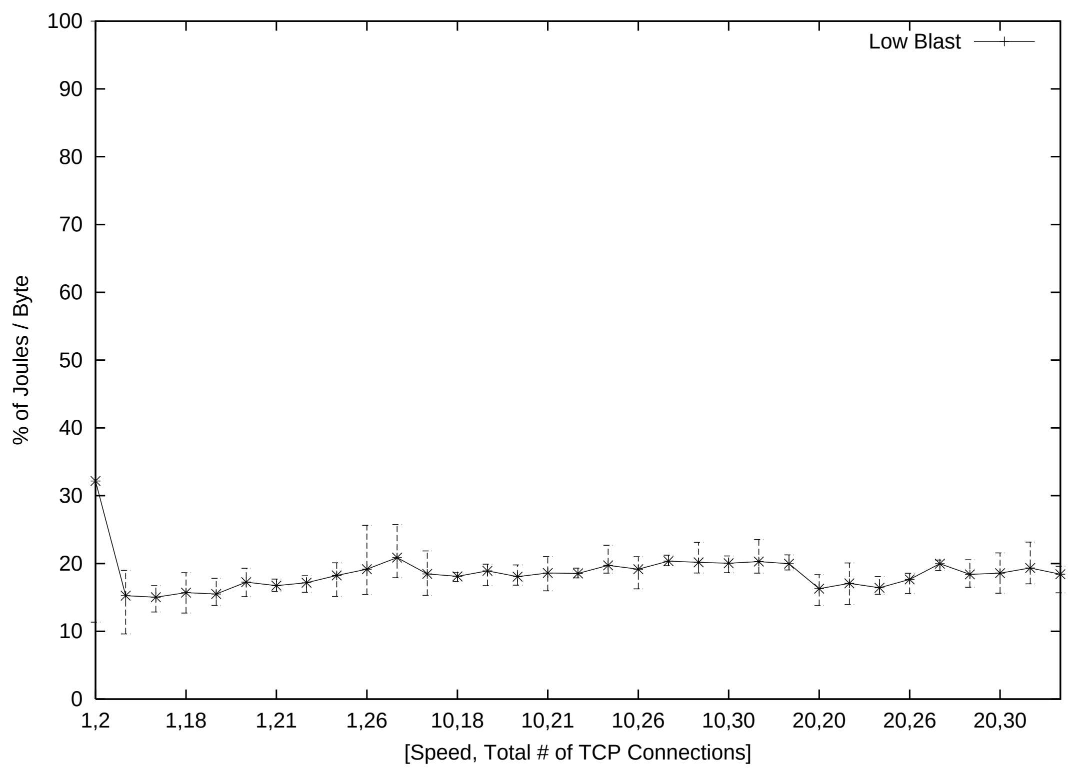


Figure 29: Energy per Byte Comparison of Combined Sleep Cycling and Power Control

5 Future Work: Energy Conservation at the Routing Layer

Power control and sleep cycling at the MAC layer reduce the cost of communication between neighboring nodes. Low energy route selection at the routing layer reduces the cost of communication between distant nodes. Most of the prior work on routing protocols for ad-hoc wireless networks has focused on minimizing latency. For the applications that we envision, energy conservation is more important than the latency incurred in communication. In this section we describe some ongoing work on energy conservation at the routing layer.

For energy conservation, the routing layer should consider the energy cost of sending a message along a route, rather than simply the latency of the route. This cost should incorporate both the transmission energy spent and the energy reserves at nodes comprising each route. We will describe a low energy routing *feature* that we are adding to the well known Dynamic Source Routing (DSR) [Broch et al. 1998a] protocol.

To allow the routing layer to pick routes based on transmission energy costs and energy reserves, we provide an interface by which we expose the power control settings at the MAC layer and the battery energy levels at the physical layer up to the routing layer. We have modified DSR to include an extra 8 bit field in the header that it adds to packets. We add this to all packets that are passed down from the higher layers (TCP, routing) to the MAC layer (i.e., all `DATA` packets). The impact of these extra 8 bits on the overall energy consumption and throughput is negligible [Agarwal et al. ????a]. This information is propagated from node to node as a packet traverses a route. It is updated at each hop along the route and indicates the health, in terms of energy reserves and transmit energy costs, of the route traversed so far. When a new packet is generated, the value in this field is zero. Beginning at the next hop and at every hop along the route, including the destination node, the following value is added to this 8 bit field:

$$\frac{TransmitEnergyCostToPreviousHop}{EnergyLevelAtThisNode}$$

Thus for a packet that is relayed from node A to node D via node E in Figure 30, the energy cost field will initially be set to zero at node A. When the packet arrives at node D, this value will be

$$\frac{TransmitCost(E \rightarrow A)}{EnergyLevel(E)} + \frac{TransmitCost(D \rightarrow E)}{EnergyLevel(D)}$$

Thus node D knows the cost of sending a packet on the *reverse* route (D to A via E). Each intermediate node will also know the cost of sending a packet from it back to the

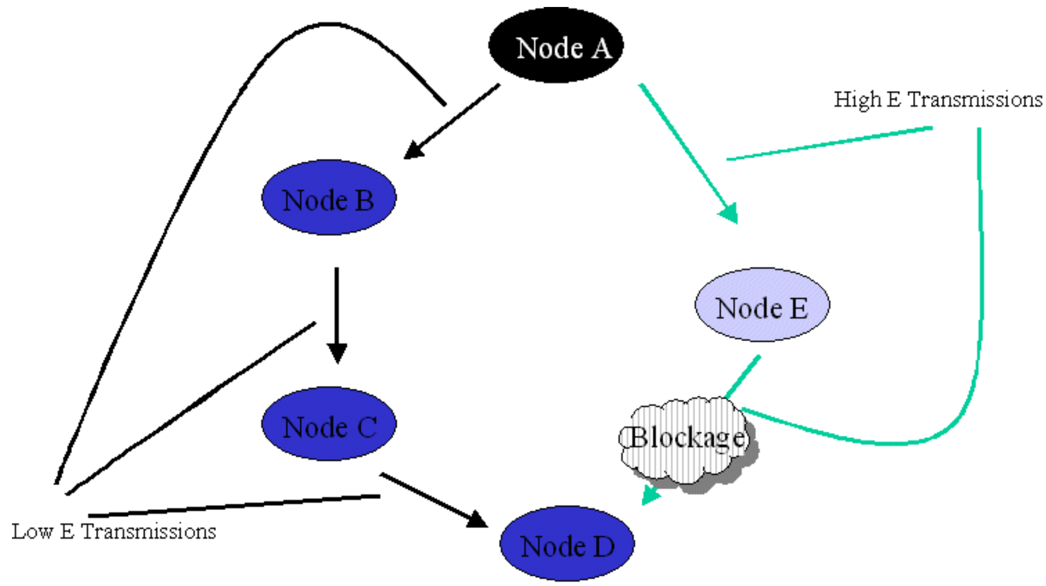


Figure 30: Low Latency Routing versus Low Energy Routing

originating node. This cost function attempts to measure the overall impact of using this route. A node along the route that has low energy reserves will drive the cost function up, to discourage the usage of this route. A node along the route that has an unusually high transmit cost will also drive the cost function up. Thus by picking the route with the lowest cost, the routing layer can minimize the overall impact of sending a message. The impact of the protocol is to increase the longevity of the network by picking routes that consist of nodes with low transmit energy costs and high energy reserves. This occurs at the possible expense of latency. In Figure 30, the low latency route from node A to node D is via node E. However, this route uses two high energy transmission links and a node which is low on energy, viz. node E as a relay. The alternate route, via nodes B and C, uses more hops and hence has a higher associated latency. However, it uses nodes with larger energy reserves and links with lower energy costs. Thus our energy efficient routing protocol picks the latter route. We refer the reader to reference [Singh et al. 1998] for a discussion on the impact of using different energy cost functions.

We now include our energy efficient MAC protocol, our approximate sleep cycle mechanism and our low energy routing algorithm in our simulations. In Figure 31, we present results from simulations of the scenario in Figure 30. We assume two CBR (constant bit rate) flows, one from node A to node D and one from node D to node A. We only show the energy consumption patterns of nodes E and B. Our low energy routing protocol produces significant changes in the energy consumption patterns of nodes B and E when compared with the pattern observed when DSR is used. The energy consumption patterns of nodes A

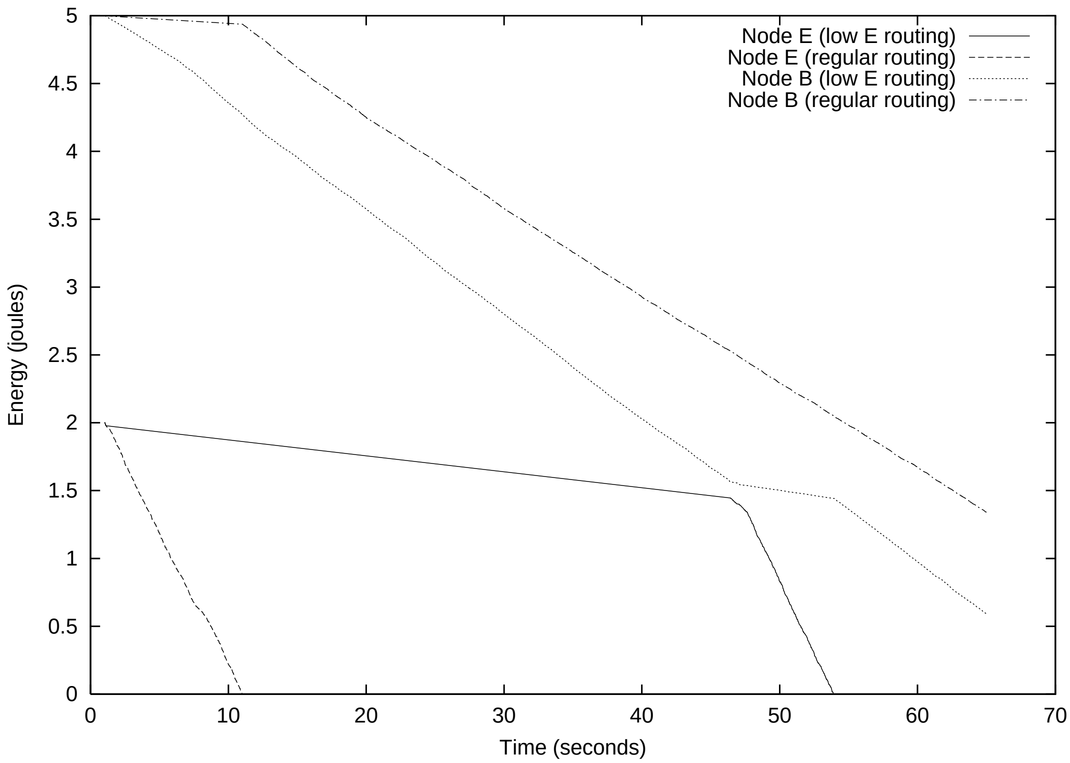


Figure 31: Node Energy Consumption with Low Energy Routing

and D remain relatively unchanged. The energy consumption pattern of node C is similar to that of node B. Low latency routing (shown in the "regular routing" graphs in Figure 31) initially picks the low latency routes between nodes A and D which involve node E. Thus node E (initially with only 2 Joules of energy) quickly runs out of energy after 10 seconds. Low latency routing then picks longer routes that involve nodes B and C. Low energy routing first picks routes via nodes B and C as these nodes have higher energy reserves (initially 5 Joules each) and have lower transmit costs. After about 50 seconds, nodes B and C are drained to a point where the routes via node E are more energy efficient, in terms of the cost defined in the above equations. Thus, low energy routing switches over to these routes at that point in time. This is the behavior that we expect from a routing framework that attempts to minimize the energy cost of communication.

However, note that at the end of the simulation, near 65 seconds, node E has depleted its energy reserves in both cases, but node B ends up with lower energy reserves in the low energy routing case. This is due to the overhead of transmitting route energy information during route selection and maintenance. We are continuing to reduce this overhead.

We are continuing to integrate a more sophisticated sleep cycling mechanism into our simulation infrastructure. We are also continuing to evaluate our low energy routing algorithm by using many different simulation scenarios [Agarwal et al. ????a].

6 Conclusions

Ad-hoc wireless networks will be employed in situations where the communicating nodes will not have access to wired power sources such as an electricity grid. These nodes vary in size, but many will be small units and thus will have very limited energy cells and energy scavenging mechanisms. A significant portion of a node's energy budget will be spent in communication. Thus it is important to explore new algorithms that minimize the energy cost of communication.

We have described three main mechanisms for achieving low power communication. Power control at the MAC layer selects the minimum amount of transmit energy needed to pass a message between any pair of neighboring nodes. Sleep cycling at the MAC layer turns off nodes that are not involved in any communication to conserve their energy. Low energy route selection at the routing layer selects low energy cost routes for sending messages between nodes that are not neighbors.

To evaluate the effectiveness of new algorithms and optimizations to the communication mechanism, we need simulation models that are based on realistic usage. Random mobility and traffic models are not realistic. We have presented a simulation infrastructure that incorporates group mobility patterns, novel group traffic patterns and novel blockage models. These models recognize that nodes (animals, humans, etc.) move in groups and that various terrain features, weather and man-made obstacles will affect radio communication. Even though these models are simple in design, we have shown that they dramatically affect simulation results.

We used this realistic simulation infrastructure to evaluate the effectiveness of a power control loop for ad-hoc wireless networks. We show that our power control loop improves energy consumption and throughput, and does so more significantly in the group mobility simulations. Our power control loop improves energy consumption by 10% and throughput by 5% in the random models, but by 10-20% and 15% respectively in the group mobility models. These benefits vary as the density of nodes in the field varies, due to a change in overall interference. We have also evaluated the added benefit of sleep cycling via an approximate sleep cycle mechanism. We have also described ongoing work in energy conservation at the routing layer. We have shown that the behavior of our low energy routing mechanism is desirable.

Acknowledgments

Much credit goes to my research advisor, Randy H. Katz, for his support and endless advice. He has provided much needed direction for this work. His help with this report and previous publication submissions have greatly improved my writing and presentation skills. Many thanks go to Srikanth V. Krishnamurthy and Son K. Dao at HRL Laboratories, LLC, for their advice, support and direction. I would like to thank the second reader, Anthony D. Joseph, for taking the time to read and provide valuable, detailed comments on my report. This work was supported by DARPA (Defense Advanced Research Projects Agency) Contract No. N00014-99-C-0322.

References

S. Agarwal, R. Katz, S. Krishnamurthy, and S. Dao. Energy efficient routing in mobile ad-hoc wireless networks. *Work in Progress*, ????a.

S. Agarwal, R. Katz, S. Krishnamurthy, and S. Dao. Ns-2 modifications to support power control, energy efficient routing, group mobility, traffic and rf blockages. *http://www.cs.berkeley.edu/ sagarwal/research/wireless/ns2-msreport.tgz*, ????b.

R. Antkiewicz, A. Manikowski, A. Najgebauer, and T. Nowicki. A computer simulator of a tactical communication system. *IEEE MILCOM*, 1:893–897, Oct 1998.

V. Bharghavan, A. Demers, S. Shenker, and L. Zhang. Macaw: A media access protocol for wireless lans. *SIGCOMM*, September 1994.

J. Broch, D. Johnsson, and D. Maltz. The dynamic source routing protocol for mobile ad hoc networks. *Internet Draft*, pages draft–ietf–manet–dsr–00.txt, March 1998a.

J. Broch, D. Maltz, D. Johnson, Y. Hu, and J. Jetcheva. A performance comparison of multi-hop wireless ad hoc network routing protocols. *MOBICOM*, October 1998b.

D. Estrin, R. Govindan, J. Heidemann, and S. Kumar. Next century challenges : scalable coordination in sensor networks. *MOBICOM*, pages 263–270, 1999.

D. J. Goodman. *Wireless Personal Communications Systems*. Wireless Communications Series. Addison Wesley, 1997.

C. Graff, M. Bereschinsky, M. Patel, and L. F. Chang. Application of mobile ip to tactical mobile internetworking. *IEEE MILCOM*, 1:409–414, Oct 1998.

W. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. *IEEE Hawaii International Conference on System Sciences*, January 2000.

IEEE. Medium access control (mac) and physical (phy) specifications. *IEEE P802.11/D10*, Jan 1999.

P. Johansson, T. Larsson, N. Hedman, B. Mielczarek, and M. Degermark. Scenario-based performance analysis of routing protocols for mobile ad-hoc networks. *MOBICOM*, pages 195–206, 1999.

JTRS. Joint tactical radio system. *http://www.jtrs.sarda.army.mil/*, ????

T. Kwon and M. Gerla. Clustering with power control. *IEEE MILCOM*, 2:1424–1428, Nov 1999.

W. C. Lee. *Mobile Cellular Telecommunications Systems*. McGraw-Hill, 1989.

Lucent. Wavelan specifications. *http://www.wavelan.com*, ????

B. Narendran, J. Sienicki, S. Yajnik, and P. Agrawal. Evaluation of an adaptive power and error control algorithm for wireless systems. *IEEE ICC*, 1:349–355, June 1997.

NS-2. Network simulator. *http://www-mash.cs.berkeley.edu/ns/*, 2.1b6.

T. Ojanperä and R. Prasad, editors. *Wideband CDMA for Third Generation Mobile Communications*. Universal Personal Communications. Artech House Publishers, 1998.

R. Ramanathan and R. Rosales-Hain. Topology control of multihop wireless networks using transmit power adjustment. *IEEE INFOCOM*, 2:404–413, March 2000.

S. Singh, M. Woo, and C. Raghavendra. Power-aware routing in mobile ad hoc networks. *MOBICOM*, pages 181–190, Aug 1998.

J. A. Stine and G. Veciana. Tactical communications using the ieee 802.11 mac protocol. *IEEE MILCOM*, 1:575–582, Oct 1998.

Appendix

Code Structure

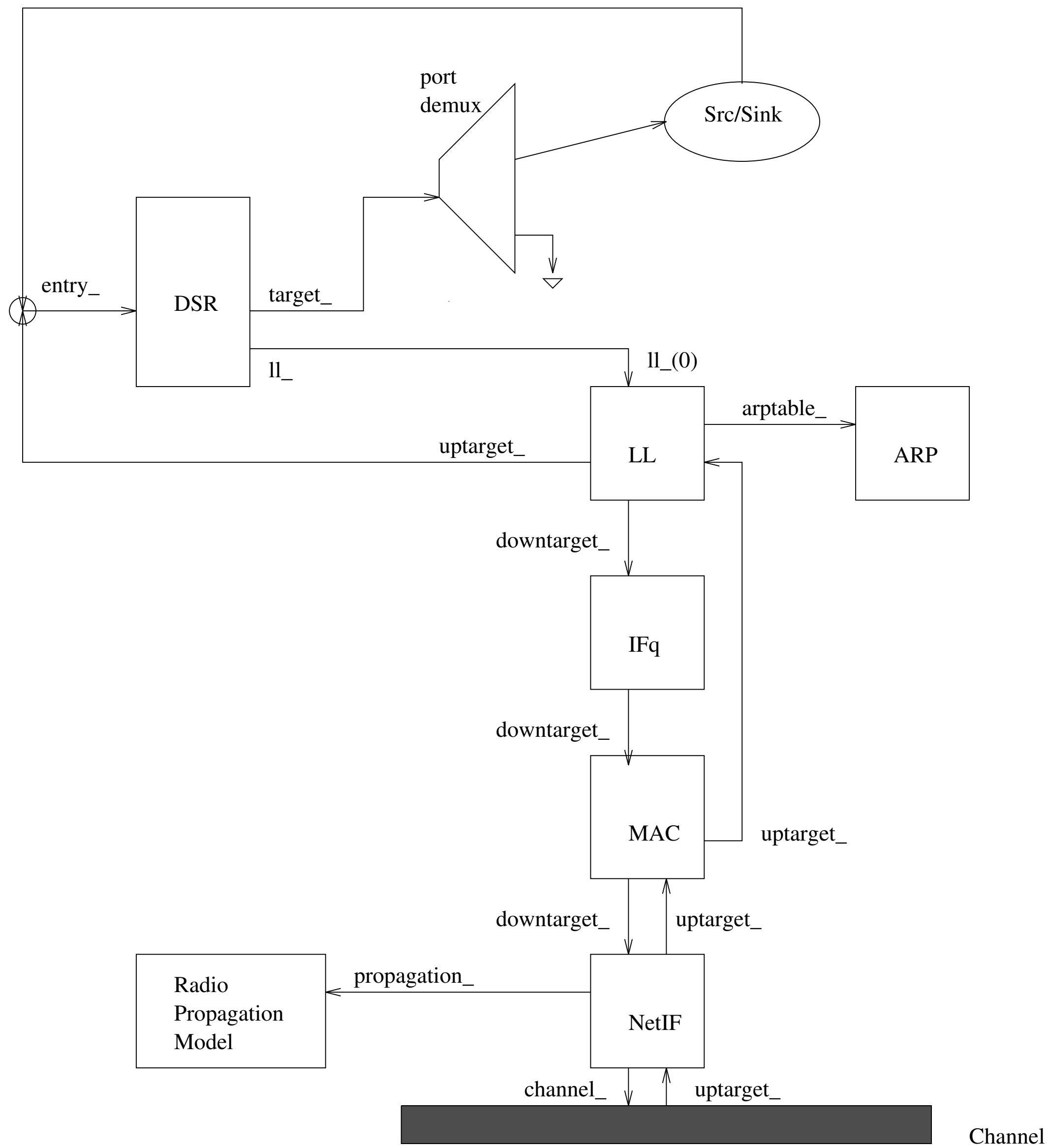


Figure 32: Structure of DSR Wireless Nodes in NS

In this section, we describe how we engineered our source code into NS's modular structure. Figure 32 shows the code structure of wireless nodes in NS that run the DSR routing protocol. The structure of those that run other ad-hoc routing protocols is only slightly different.

We modified the network interface (`ns-2/wireless-phy.{h,cc}`) to query our blockage manager (`ns-2/erebus.{h,cc}`) every time a packet is received. The network interface has to determine if a received packet can actually be delivered to the given node based on the radio propagation model. We modified it to consider both the radio propagation model and our RF blockage model. The blockage manager itself determines the loss due to transmissions through blockages.

We altered the MAC layer (`ns-2/mac-802_11.{h,cc}`) to support power control. We augmented the `CTS` and `DATA` message headers with power control values. We added our power control algorithm to the transmit and receive mechanisms in the MAC. We also modified the interface between the MAC layer and the physical layer (`ns-2/packet-stamp.h`) to allow the MAC to set the transmit power level for outgoing packets and to read the received power level for incoming packets. The network interface also scales the energy consumed by each node during transmission by the transmit power level.

Extensive modifications to DSR (`ns-2/dsr/dsragent.{h,cc}`) were required to allow it to route packets in a low energy fashion rather than with low latency. We altered the structure of route maintenance packets (`ns-2/dsr/hdr_sr.h`) to include various fields that carry link and node energy state. We similarly altered internal route storage structures (`ns-2/dsr/path.{h,cc}`) to also store these energy values. We changed various route caches (`ns-2/dsr/mobicache.cc`, `ns-2/dsr/routecache.{h,cc}`) and route tables (`ns-2/dsr/requesttable.{h,cc}`) to select low energy routes over low latency routes.

Running Simulations

In augmenting NS with our blockage, power control and low power routing algorithms, it was our goal to be able to selectively use some, all or none of these additions at run time. We added extra options to various modules (`ns-2/tcl/lib/{ns-erebus,ns-lib,ns-mobilenode}.tcl`, `ns-2/tcl/mobility/dsr.tcl`) so that simulation scripts can select which features to use at run time.

Figure 33 shows how we expect wireless simulations to be run in NS. The simulation T-CL script contains standard simulation setup, such as the routing protocol to use and turning on power control. This script refers to two other files. The node mobility pattern file is generated by programs such as our group mobility generator (`ns-2/indep-utils/cmu-scen-gen/setdest-troops/*`). The traffic patterns can be generated by our group traffic generator (`ns-2/indep-utils/cmu-scen-gen/troop-tcpgen.tcl`).

Our group mobility generator allows the simulation runner to specify various para-

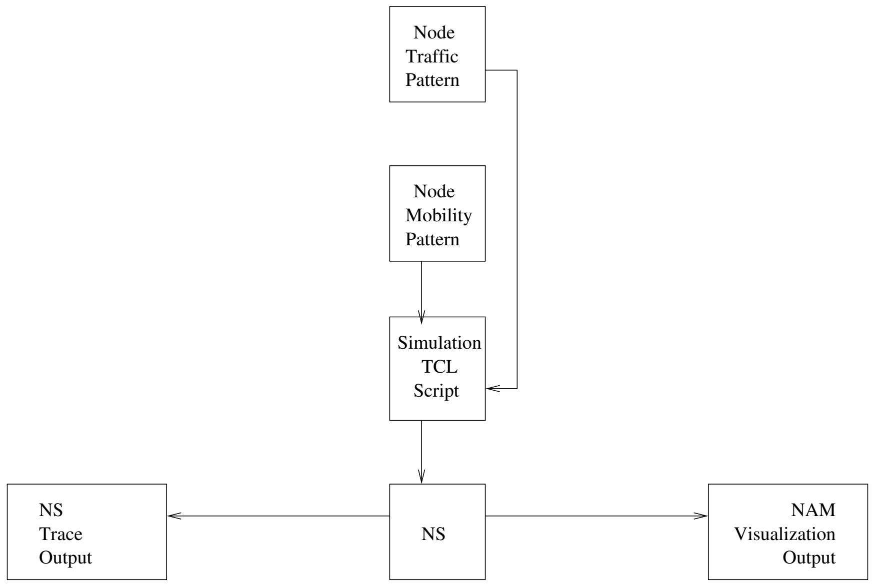


Figure 33: Running Simulations in NS

meters that will control the number of nodes, their placement and their movement. These parameters are

- NODES (total number of nodes to simulate)
- COHORTS (total number of cohorts)
- MAXX (length of simulation field)
- MAXY (breadth of simulation field)
- MAXTIME (total simulation time)
- MAXSPEED (maximum speed of a node)
- MAXE (maximum energy of a node)
- MAXXMITE (maximum transmit energy of a node)
- PAUSE (time to pause between velocity vectors)
- DIVERGENCE_FACTOR (maximum divergence of a node from cohort)
- BLOCKAGES (total number of blockages to simulate)

- BLOCK_MAXR (maximum radius of a blockage)
- BLOCK_L (blockage density factor)
- BLOCK_MAXV (maximum speed of a blockage)

Our group traffic generator allows the simulation runner to specify various parameters that will control the number, distribution and length of traffic simulated. These parameters are

- NN (total number of nodes to simulate)
- COHORTS (total number of cohorts)
- MTIME (total simulation time)
- SEED (seed for random number generator)
- TYPE (type of traffic - TCP / CBR)
- INTERVAL (CBR rate)
- MC (maximum number of connections)

Once our simulation infrastructure is mature enough, and given enough interest from the NS developers community, we plan on integrating our improvements into the root NS-2 source tree so that other researchers can benefit from our work. Until then, our code can obtained via our website [Agarwal et al. ????b].