RANGE ADAPTIVE PROTOCOLS FOR WIRELESS MULTI-HOP NETWORKS

By

Nattavut Smavatkul

Dissertation submitted to the Faculty of the Virginia Polytechnic
Institute and State University in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY
in
Electrical Engineering

Dr. Scott F. Midkiff, Chair
Dr. Peter M. Athanas
Dr. Nathaniel J. Davis, IV
Dr. C. Patrick Koelling
Dr. Jeffrey H. Reed

8:00 am to 10:00 am
1st November 2000
Torgerson Hall 2010A
Blacksburg, Virginia

Keywords: Wireless ad hoc network, Spread-spectrum multiple access, Adaptive MAC protocol, ADIM-NB, CAMEN

RANGE ADAPTIVE PROTOCOLS FOR WIRELESS MULTI-HOP NETWORKS

Nattavut Smavatkul

Dr. Scott F. Midkiff, Chair

Electrical Engineering

(ABSTRACT)

Recent accomplishments in link-level and radio technologies have significantly improved the performance of wireless links. Wireless mobile *ad hoc* networks, however, typically only take limited advantage of these enhancements. In this research, the medium access control protocol and *ad hoc* routing protocol are extended to take advantage of radios offering multi-user interference cancellation and direct-sequence spread-spectrum functionality, by encouraging multiple simultaneous connections and adaptively changing communication parameters on a per-packet basis. Through its environment characterization techniques, the adaptive direct sequence spread spectrum MAC protocol for non-broadcast multiple access networks (ADIM-NB) improves several aspects of the wireless mobile *ad hoc* network performance, including throughput, delay, stability, and power consumption, through its use of spread-spectrum multiple access and four different adaptive algorithms. The four adaptive algorithms change processing gain, forward error correction coding rate, transmit power, and number of simultaneous connections.

In addition, the *ad hoc* routing protocol is extended with the clustering algorithm for mobile *ad hoc* network (CAMEN). With ADIM-NB in mind, CAMEN discourages the use of broadcast messages, supplements ADIM-NB's functionality at the network level, and improves the network scalability by aggregating nodes into clusters. Both protocols are intended to lead to more powerful and flexible communication capabilities for wireless nodes.

Simulation models have been developed and simulated to verify the performance improvements of both protocols at the network-level as well as provide a means to perform trade-off analysis. Results indicate that the network capacity is increased between 50% in a moderately loaded network to 100% in a heavily loaded network over a non-adaptive MAC protocol. The delay also improve significantly in most scenarios of interest.

Acknowledgments

I am eternally grateful for the assistance of many people involved in this research development. This research effort will not be materialized without their contributions. Chief among these is my advisor, Dr. Midkiff. I thank him for accepting me as one of his Ph.D. students. He was always there to provide invaluable guidance and support; and for that I am greatly indebted. Dr. Midkiff also goes well above and beyond his call of duty by always sharing with me every possible opportunity to improve myself in research and academic environments.

I thank my committee members, Dr. Peter M. Athanas, Dr. Nathaniel J. Davis, IV, Dr. C. Patrick Koelling, and Dr. Jeffrey H. Reed for their time and valuable guidance, each shaped the research. Their inputs and suggestions for this research are invaluable. I also thank Lori Hughes for her dedication in reviewing this rather lengthy dissertation.

I thank my parents for making me understand the importance of a good education and providing support throughout my studies. They gave me every opportunity to pursue a higher education to the best of my ability. My deepest gratitude goes to my little sister, who took care of our parents during the course of my study. They have been a great inspiration to my life. Special thanks are also offered to family members and friends, both at Virginia Tech and elsewhere, who have always been understanding and supportive.

This work has been supported in part by the DARPA Global Mobile Information Systems (GloMo) program through funding for the project “Configurable and Robust Wireless Communications Nodes” (FBI Contract N65236-97-K-7806).

Any shortcomings or weaknesses which remain in this dissertation must necessarily be attributed to the author.

Table of Contents

List of Figures

List of Tables

Chapter 1. Introduction

1.1 Background

Spread-spectrum technology has been around for half a century [RoM50]. Surprisingly few contention-based medium access control (MAC) algorithms fully exploit the principal characteristic of spread-spectrum, which is that multiple traffic sessions or channels share the same frequency spectrum. Most contention-based MAC protocols that use spread-spectrum technology are based on one spreading code per subnet such as the IEEE 802.11 wireless local area network (LAN) [IEE99] and wireless asynchronous transfer mode (ATM) network [DHM98]. For such networks, there can only be a single traffic session active at any given time in a particular subnet. Arguably, using a single code leads to a simpler MAC algorithm and a smaller spreading factor, resulting in a more efficient use of the frequency spectrum. Multiple-code techniques do have their own advantages, such as increased security, reduced need for coordination, improved support for multimedia and real-time traffic, greater freedom to change transmission parameters, and the ability to take advantage of new technologies, such as multi-user interference rejection [ZMR96], adaptive antenna arrays [PeR95], and powerful error correction codes [VaW97].

However, multiple-code techniques do have weaknesses. To realize their full potential, complex hardware and computationally intensive algorithms are necessary. As computing power becomes more widely accessible, an increasing number of complex signal processing techniques will be available for wireless devices. Many of these complex algorithms can only be fully exploited by MAC protocols that use multiple-code techniques. For example, the multi-user interference rejection algorithm demodulates the desired user’s signal from other received signals, each having a different spreading code. With increased hardware capabilities and advanced signal processing techniques, the MAC protocol specified in this research, the adaptive direct-sequence spread-spectrum MAC protocol for non-broadcast multiple access network (ADIM-NB), is intended to lead to more powerful and flexible communication capabilities for wireless nodes.

Multiple-code issues also exist in other communications domains. This research can also be applied to future local area networks using Wavelength Division Multiplexing (WDM), a relatively new technology that has recently become practical. Because there are multiple carriers in each optical transmission medium, the ability to hop from one carrier to another among communicating peers is critical. This problem is somewhat similar to a multiple-code MAC protocol for wireless networks, with the exception of lower latency and a much more predictable interference level in the system. The multiple-code techniques are also known as code division multiple access (CDMA) protocols because each wireless node uses a different code to share the same communications medium in multiple access networks.

Because of the many new functions and behaviors introduced in ADIM-NB, traditional *ad hoc* routing protocols are not sufficient to exploit all of the advantages. Consequently, an existing *ad hoc* wireless routing protocol [GaM97] is extended with the clustering algorithm for mobile *ad hoc* network (CAMEN) and integrated with ADIM-NB. This integration allows both MAC and networking protocols to be jointly optimized.

An *ad hoc* network is a network that does not require any infrastructure in order to operate. Users can co-locate several terminals and expect them to communicate with each other with minimal configuration. Adding an access point allows this network to communicate with a traditional wired network. The *ad hoc* networks of interest use wireless links and support rapidly changing network topology. Each node moves independently with a speed of up to 65 Kph. A more descriptive name for such a network is a wireless multi-hop network. This differentiates an *ad hoc* network from a traditional wireless LAN in which stations communicate over a single wireless hop and nodes in multiple wireless LANs are connected via a wired infrastructure. In a wireless multi-hop network, each packet is routed across one or more wireless hops to the destination without using any wired infrastructure. An *ad hoc* network plays a limited role in today’s wireless networks. Frequently cited applications include communications for military, rescue, and disaster recovery. The primary reasons for this limited deployment are the high complexity of the required algorithms and the inefficient use of the frequency spectrum.

Spread-spectrum techniques also possess a unique ability to trade a lower data rate for a higher signal to noise ratio (SNR). By manipulating the coding rate, the varying noise level in the system can be accommodated, or signal strength fading due to the varying distance between transmitter and receiver can be tolerated [MDM97]. This research relies on this unique characteristic of spread-spectrum to improve system performance.

1.2 Research Goals

This research explores the capabilities, strengths, and weaknesses of using multiple spreading codes in a single subnet and the associated performance trade-off between data rate and transmission range. After this fundamental knowledge is obtained, an integrated MAC and *ad hoc* routing protocol based on a spread-spectrum multiple access (SSMA) scheme and various adaptive algorithms are developed and evaluated.

The primary goal is to conceive, design, and evaluate a MAC protocol that uses the two fundamental techniques, the SSMA scheme and a data rate/error rate trade-off algorithm. Other adaptive algorithms such as adapting the forward error correction (FEC) coding rate and adapting the transmitted power are also investigated. The secondary goal is to optimize throughput and delay performance of the *ad hoc* routing protocol in a dynamic environment to take advantage of ADIM-NB and several new link-level techniques that have been developed in the past few years, as well as those that seem feasible in the near future. These techniques include, but are not limited to, adaptive antenna arrays [PeR95], multi-user interference rejection [ZMR96], turbo coding [VaW97], reconfigurable computing [KaA97], and multicarrier CDMA

[HaP97]. Together, these techniques represent a new generation of radios commonly called "smart radios."

An adaptive antenna array is a software steerable antenna that can adjust itself to the direction of the signal's source. Because the algorithm has to obtain the direction of the incoming transmission in order to steer the antenna to the right direction, a longer frame length and training sequence are desirable for an adaptive antenna array to realize its full potential. Prior knowledge of the communicating peer is also desirable to reduce the need for a long training sequence. Since the multiple-code technique allows a longer session time, it may play a crucial role in improving the performance of the adaptive antenna array.

A multi-user receiver with interference rejection [Swa98] can receive data from more than one transmitter at the same time. However, that receiver cannot transmit anything during the period. Thus, coordination among participants is critical. This technique, along with multicarrier CDMA, can significantly improve the performance of a CDMA system.

Turbo codes are a new error correction coding technique that provides performance that is near the theoretical limit for long blocks of data with practical computing requirements. The key to utilizing turbo codes is to enable a long frame. With CDMA technology, longer frames no longer inhibit other nodes from communicating. Alternative approaches include a hybrid turbo/convolutional code system in which turbo codes are used in a large frame size or a hostile environment and convolutional codes are used in normal situations.

A reconfigurable computing platform allows radio hardware to be reprogrammed dynamically, which enables a single instance of radio hardware to perform multiple tasks that require multiple modules in a traditional radio. Both MAC and routing protocols may need to be aware of the reprogramming techniques and the associated delay.

The multicarrier modulation scheme, often called orthogonal frequency-division multiplexing (OFDM), divides information into several carriers and transmits them simultaneously to combat a frequency-selective slow Rayleigh fading channel. This technique relies on having a multiple-code MAC protocol.

To achieve these goals of designing the highly adaptive MAC protocol and the corresponding *ad hoc* routing protocol, OPNET Technologies' OPNET modeler®[1] is used as a simulation tool to evaluate the performance of both ADIM-NB and CAMEN. This simulation provides valuable insight into system performance under various operating conditions. The relative performance improvements and under what circumstances these occur can then be studied during this research. The results can be used to optimize the protocols under various conditions. In conjunction with the available simulator, a custom simulator is studied to provide additional link-level fidelity, such as an FPGA-based hardware simulator to model realistic physical-layer characteristics.

---

[1] OPNET Modeler® is a registered trademark of OPNET Technologies.

Both ADIM-NB and CAMEN are compatible, to a reasonable extent, with the IEEE 802.11 wireless LAN standard [IEE99] and TCP/IP protocol suite. Minimal effort is required to integrate this network with existing networks. In addition, ADIM-NB also conforms to the WINGs' Radio Application Program Interface (API) specification [BeL97] from the Defense Advanced Research Projects Agency's (DARPA) Global Mobile Information Systems (GloMo) program [SRI98].

Several contributions are resulted from this dissertation. It investigates the impact of the adaptive medium access control protocol in a non-broadcast multiple access environment, or ADIM-NB. This research focuses on a wireless mobile *ad hoc* network environment using a radio equipped with multi-user receiver interference cancellation functionality. The ADIM-NB protocol establishes a unique communication channel for each packet and dynamically changes its communication parameters by observing the current environment condition, as well as encourages the utilization of multi-user receiver algorithm. At the network layer, CAMEN, a hierarchical extension to a wireless mobile *ad hoc* routing protocol, improves the network scalability, exploits new features introduced by the ADIM-NB protocol, and reduces the dependency of the routing protocol on radio broadcast functionality.

1.3 Document Overview

This chapter provided an introduction to the problem under investigation. The ADIM-NB MAC protocol and the CAMEN routing algorithm were briefly discussed. Specifically, the importance of and potential improvement from using a SSMA technique as an access mechanism was discussed. Justification for this research was briefly presented along with several essential concepts, and the goals of the research were briefly summarized.

Chapter 2 presents a review of several important wireless MAC protocols, existing *ad hoc* routing protocols, and the link-level techniques mentioned in this chapter. It concludes with a discussion of related research efforts.

Chapter 3 presents the objectives and methodologies employed in this research. This chapter delineates the scope of the research. The simulation tools and performance metrics are introduced, and several comparable algorithms and protocols are presented as candidates for comparison. The detailed procedures in this research effort and steps toward the final results are also described.

Chapter 4 contains a more comprehensive and formal description ADIM-NB and its problem statements. In its simplest form, ADIM-NB uses three classes of spreading codes: low data rate/long range, medium data rate/medium range, and high data rate/short range. Based on the available information, each communicating peer intelligently selects a code from the appropriate class. To set up a connection, nodes exchange request to send (RTS) and clear to send (CTS) messages to negotiate unique spreading code and communication parameters. This process leads to the use of different spreading codes in each subnet at the same time. It also helps to combat the hidden terminal problem. ADIM-NB also changes FEC coding rate and transmitted power based on the information prior to and during the negotiation process. Chapter 4 also describes other important features of and motivation for using ADIM-NB. Examples of the protocol in action are also provided throughout the chapter.

Chapter 5 contains a more comprehensive and formal description of the CAMEN hierarchical *ad hoc* routing protocol and its problem statements. CAMEN is based on a clustering concept. Many nodes within each other's transmission range form a cluster. One cluster head is selected in each cluster to do minimal administrative tasks and routing among clusters. The primary purpose of using this clustering approach is to make the routing protocol scalable to a large number of nodes. An additional advantage of this clustering scheme is that it provides a natural mechanism to integrate this *ad hoc* network with an existing infrastructure-based network by forcing a wired station to be the cluster head.

The design of both ADIM-NB and CAMEN take into consideration several novel MAC and routing protocols that were recently introduced, including the IEEE 802.11 wireless LAN [IEE99], floor acquisition multiple access (FAMA) [FuG95a], the hybrid proactive-reactive routing algorithm [Haa97], multimedia wireless networks [LiG97b], and cluster management [LiG97b]. Additional information about these protocols is provided in the literature review in Chapter 2. Chapter 5 also describes other important features of and motivation for using CAMEN.

Chapter 6 describes the simulation environment, along with model verifications and validations. Starting with a description of the simulation tool, this chapter completely describes the simulation. Simulation parameters and factors are defined and assigned values. Aggregate throughput is defined as the response variable of primary interest for the simulations.

Chapter 7 presents the simulation results of this research effort, including comparisons with other MAC protocols, effects from node mobility, and impacts from each individual component. It is shown that ADIM-NB performs better than standard carrier sensing multiple access (CSMA) MAC protocol or the non-adaptive ADIM-NB/B MAC protocol in most environment situations. Aggregate throughput for a wireless network is improved when using the adaptive ADIM-NB MAC protocol.

Chapter 8 concludes the dissertation with a summary of the research findings, including important concepts, techniques behind this research effort, and the significance of research results. Finally, the chapter presents recommendations for future direction of the research area.

Appendix A provides a high-level overview of the current simulation model, which includes process models, a node model, and a network model developed for the MAC and routing protocols. It describes all of the concepts and functionality of the model. The concepts, however, are justified in Chapters 4 and 5.

Appendix B provides a summary of abbreviations that are used in this document.

Chapter 2. Background and Prior Work

This chapter describes several important protocols and techniques that are the fundamental building blocks of this study. The first two sections describe medium access control protocols and routing protocols. Many of these are applicable to both wireless and wired environments. These protocols have been implemented and proven successful, stable, and reliable. The third section describes many of the current research efforts that are related to this proposal, most of which are active research topics that have not been fully implemented and will likely improve over time. Many of them are only proposed at this time. The fourth section explains the multiple link-level techniques that this study proposes to use to their full potential within a network in a wireless mobile *ad hoc* environment. Common problems in a wireless mobile *ad hoc* network are described in the last section.

The information in this chapter does not represent a complete description of every protocol but only the features relevant to this research. This chapter is intended as an introduction to each essential related technique.

2.1 Medium Access Control Protocols

Most local area networks (LANs) and metropolitan area networks (MANs) consist of collections of devices that must share the network's transmission capacity [Sta94]. Some means of controlling access to the transmission medium is needed to provide for an orderly and efficient use of that capacity. This is the function of a medium access control (MAC) protocol.

2.1.1 Carrier Sensing Multiple Access

Many MAC protocols use carrier sensing multiple access (CSMA) as their underlying scheme. With CSMA, as the name implies, a station senses the transmission medium before transmission. It transmits if and only if the channel is idle. There are several variations of the CSMA technique that behave differently if the channel is busy. These include non-persistent CSMA, 1-persistent CSMA, and p-persistent CSMA [BeG92]. As an example, p-persistent CSMA sends the queued data with probability p after the channel becomes available, while non-persistent CSMA waits for a random amount of time before trying to re-sense the channel. The 1-persistent CSMA is a special case of the p-persistent CSMA algorithm with p equal to 1 and non-persistent is a special case with p equal to 0. There are several extensions to CSMA that behave differently in the case of collision, such as collision detection (CSMA/CD) and collision avoidance (CSMA/CA). CSMA/CD is used in the IEEE 802.3 Ethernet LAN standard [Sta94] and CSMA/CA is used in the IEEE 802.11 wireless LAN standard [IEE99].

2.1.2 IEEE 802.11 Wireless LAN Standard

IEEE 802.11 [IEE99] is a widely accepted wireless LAN standard. It is intended for an operating environment consisting of a small to moderate number of fixed wireless stations. There are two operating modes, Point Coordination Function (PCF) and Distributed Coordination Function (DCF). The PCF mode provides a contention-free access scheme that relies on a coordinator, or access point, to grant access to each individual station. This coordinator is usually a fixed host attached to a wired network. The DCF mode is a contention-based accessing scheme that uses non-persistent CSMA/CA. CSMA/CA uses several timeout variables and backoff mechanisms to help stations avoid collisions.

The IEEE 802.11 standard allows transmission at different speeds by using different waveform modulation methods. For the direct-sequence spread-spectrum (DSSS) implementation, each packet consists of a standard speed (1 Mbps differential binary phase shift keying, or DBPSK) header immediately followed by a higher speed (2 Mbps differential quaternary phase shift keying, or DQPSK) data payload, if appropriate. However, the actual algorithm of when and how to change the transmission speed is still an active research topic. The faster IEEE 802.11b standard uses yet another type of modulation called complimentary code keying (CCK), which allows 8 bits of data to be encoded using 8 chips, resulting in 5.5 folds increase in transmission speed over the QPSK modulation.

The ADIM-NB MAC protocol is based on a non-persistent CSMA technique. It also implements several functions of the IEEE 802.11 standard, including the CSMA/CA technique and IEEE 802.11 framing format. The research outcomes can be easily applied to the IEEE 802.11 standard to improve its transmission speed selection algorithm.

2.1.3 Floor Acquisition Multiple Access with Pauses and Jamming

The floor acquisition multiple access with pauses and jamming (FAMA-PJ) protocol [FuG95b] is a MAC protocol designed to combat the hidden terminal problem (explained in Section 2.5.1) experienced in a wireless multi-hop network by requiring stations to gain control of the channel, or floor, before sending data.

Whenever the mobile host wants to send a packet, it listens to the channel. If the channel is busy, the mobile host backs off and tries to retransmit later. If the channel is clear, it sends an RTS message to the destination and waits for one round-trip time plus the time needed for the destination to reply with a CTS message. If the CTS packet is corrupted or is not received within the time limit, the sender goes into a backoff state and tries the retransmission later. Once the initiator receives the CTS reply from the destination, it begins transmission.

Other stations that hear either RTS or CTS packets are requested to back off for a period of time until they encounter an acknowledgment message at the end of the data transmission. This prevents an attempt to transmit from any stations that could corrupt the established communication path between the sender and the receiver. All communications occur on a single channel shared among all stations in the network.

ADIM-NB extends this concept to the CDMA environment in which multiple concurrent transmissions and multiple transmission speeds are feasible.

2.2 Routing Protocols

The primary function of a packet-switched network is to accept packets from a source station and deliver them to a destination station. To accomplish this, a path or route through the network must be selected; generally, more than one route is possible, thus the necessity of a routing function [Sta94].

2.2.1 Path Finding Algorithm

The path finding algorithm [GaM97] is an extension of the distributed Bellman-Ford (DBF) routing algorithm. It uses the length and the second-to-last hop (predecessor) information of the shortest path to each destination to eliminate the counting-to-infinity problem of the DBF algorithm [Gar89]. Using this predecessor information, it can determine the complete path to any destination from a distance vector routing table. This allows the mobile host to find out if a route to the destination contains loops before updating its routing table.

As an example, assume that the path from a source A to a destination D is ABCD and that A wants to find the complete path from A to D using its routing table. Since A knows that the predecessor of D is C (from the routing table), A looks in its routing table for C. A will learn that B is the predecessor of C (again, from the predecessor information in the routing table) and the predecessor of B is A (itself). A then has the complete path from A to D by using the predecessor information in the routing table. This process has the advantage of the link state algorithm with a minimal modification to the distance vector routing algorithm. Specifically, each route entry in the routing table has one additional field, the predecessor information.

2.2.2 Wireless Routing Protocol

The wireless routing protocol (WRP) [MuG96] is an extension to the path finding algorithm. It adds sequence numbers and a reliable update of the routing information to the original path finding algorithm. This extension effectively reduces the number of temporary loops that occur in the original path finding algorithm. The WRP also limits route update messages to only include entries that are affected by the changes in network topology.

The clustering algorithm for mobile *ad hoc* network (CAMEN) is an extension to the WRP. It adds hierarchical structure to the flat *ad hoc* network. This improves the scalability, increases the battery life, and further reduces the number of routing updates in both the static and dynamic environments. It is also more suitable in a CDMA environment because it reduces the WRP’s dependency on broadcast messages, which are expensive in a CDMA environment.

2.2.3 Clustering Concept

Baker and Ephremides presented a concept of a clustering *ad hoc* network and the associated linked cluster algorithm in 1981 [BaE81]. The algorithm consists of two control phases in which each mobile host periodically broadcasts its topology information to neighbors. This information exchange is performed on a synchronized basis such that each station transmits in its own time slot during the control phases. After these two phases of exchanging the topology information, all cluster heads and gateways are identified.

Although the algorithm rapidly converges, it has a few deficiencies because the clustering structure can dramatically change every time the topology is updated. This change in the cluster structure makes it difficult for the networking algorithm to re-use the previously computed routing table, which makes the linked cluster algorithm appeal only to a fixed or low mobility wireless *ad hoc* network. This problem can be addressed by introducing a cluster maintenance algorithm in addition to the cluster setup algorithm. The linked cluster algorithm also requires two reserved time slots in the control channel for each node in the network, which may cause a scalability problem with large networks.

CAMEN, however, establishes a cluster structure quickly and does not require a dedicated time slot for each host. It also maintains the existing cluster structure so that old routing information can be used quickly.

2.3 Related Research Efforts

Ad hoc networking algorithms and adaptive link control techniques are active research topics. In the past several years, there have been many studies on these topics. This section presents a brief description of related research efforts, along with their relationships to this research.

Most of the *ad hoc* routing protocols described in this section rely on the ability of a radio to broadcast, which may be expensive or unavailable in a CDMA environment. This research proposes to extend the concept of these *ad hoc* routing algorithms to a CDMA environment by minimizing their dependency on broadcast messages.

2.3.1 An Adaptive Wireless Local Area Network Protocol

Mullins, Davis, and Midkiff present a concept of adaptive link control in [MDM97]. Their adaptive link control technique (CATER) is an extension to the IEEE 802.11 wireless LAN standard in the PCF mode. It performs relatively well in many situations where the standard IEEE 802.11 protocol experiences a total failure.

CATER consists of two operating modes, a default 11-chips/bit mode and a high interference 63-chips/bit mode. In the high interference environment, the communicating peers, on an individual basis, reconfigure to the 63-chips/bit mode, which reduces the information rate to combat extra noise in the system. The reconfiguration mechanism relies on a short reconfigure request frame to switch operating modes. Due to its relatively short length and high transmitted power, this frame is assumed to be more likely to arrive successfully at the destination. After finishing the data transfer, both peers switch back to the default 11-chips/bit mode.

The ADIM-NB MAC protocol proposed in this research takes into consideration several additional input parameters, include operating history, failure rate, SNR, and the number of error bits in each packet. Its CDMA operating environment also gives ADIM-NB more freedom to change its communication parameters, including power, coding rate, and forward error correction (FEC) coding rate.

2.3.2 Proactive versus Reactive Routing Mechanisms

There are two primary techniques to obtain dynamic routing information: proactive and reactive schemes [Haa97]. The proactive technique is a widely used mechanism in wired networks. Nodes periodically exchange routing information to pre-compute routes to all destinations in the network before they are actually needed. This technique enables all nodes to learn the paths to all destinations in one common operation, which reduces latency and the amount of traffic being transferred. However, a problem occurs when the network topology is so dynamic that these pre-computed routes become obsolete before they are needed. Constant changes in the dynamic network also generate too much routing information for a typical routing protocol to handle.

The reactive technique only computes routing information when it is needed. Although this behavior introduces additional delay and overhead into the routing process, it performs better in a highly dynamic environment because the routing information is less likely to be obsolete before it is needed. There are several methods to obtain the routing information on demand but the most common practice is to flood, either systematically or non-systematically, the whole network or subnet in search of the desired destination. This flooding mechanism is popular due to its low latency and low complexity. Other nodes can also learn portions of the network topology in the process.

CAMEN is based on a proactive technique that does not propagate the routing information if nodes are just moving within the cluster. Because of the large cluster size (with a diameter of up to twenty-five kilometers), the likelihood of nodes moving between clusters is small, which will reduce the amount of routing update information that is exchanged. In addition, the proactive mechanism is a natural extension to the clustering approach because the cluster head has to broadcast its cluster information periodically to its members. Adding routing information to this broadcast is more efficient than creating a whole new route discovery packet.

2.3.3 Cluster-Based Routing Protocol

The cluster-based routing protocol (CBRP) tries to employ a clustering mechanism to reduce the amount of flooding during the dynamic route discovery process [DSY99]. This is accomplished by restricting flooding messages to go through only the cluster head and gateway, not the regular nodes. CBRP also supports unidirectional links, which are not uncommon in an *ad hoc* network environment.

CBRP is different from CAMEN in three aspects. First, it is based on a reactive scheme. Secondly, every node has to periodically broadcast hello messages, which somewhat contradicts the traditional reactive scheme. Finally, each node must maintain a complete routing table.

2.3.4 Area-Based Link-Vector Algorithm

The area-based link-vector algorithm (ALVA) is designed primarily to improve distributed maintenance of the routing information in a very large internetwork [BeG98]. In ALVA, nodes are aggregated into areas with multiple levels of hierarchy. Each router maintains a database that contains a subset of the network topology at each level in the hierarchy. This subset corresponds to links used in the preferred paths to reach the destinations. The router keeps direct paths for destinations inside the same intermediate area, including all nodes in the lower hierarchy that belong to this area. If the destination is in remote areas, its packet will be forwarded to a higher level hierarchy instead. ALVA is based on a link-state algorithm, intended primarily for a wired network, that does not require complete topology information at each level in the hierarchy.

The main idea behind the aggregation scheme is that a router keeps one entry per node for close nodes in its database and a single entry for a *set* of nodes further away (by collapsing a group of nodes into one virtual node). A number of virtual nodes can also be aggregated to form a higher level virtual node. This scheme, however, requires carefully assigned addresses and a pre-engineered network topology to encapsulate hierarchical information.

2.3.5 Dynamic Source Routing

Johnson introduces a reactive routing protocol, called dynamic source routing (DSR) in [Joh94]. It is a classic reactive routing protocol in which network flooding is required each time a node needs to discover a new route. Additional mechanisms for efficient network flooding and a technique to remove obsolete routing information are also part of the DSR specification.

To discover a new route, the mobile host broadcasts a route discovery message to all of its neighbors. If the neighbor has the requested routing information in its cache, it replies to the source with the routing information. Otherwise, it stamps its address on the packet, records a reverse route to the request originator in its routing table, and forwards the request to its neighbors. If the request propagates to the destination, the destination records the route back to the source using address stamps of intermediate nodes, stores this path information in a route reply packet, and replies to the source via source routing. All subsequent data transfers use source routing to reduce the dependency on the intermediate nodes. There is no need for periodic route advertisement or neighbor discovery.

The choice of reactive or proactive routing protocols depends on the network dynamics. Because of a long radio transmission range and the cluster encapsulation scheme employed, the target operating environment in this research is expected to have relatively low network dynamics, i.e., the topology is expected to change relatively slowly, and thus a proactive scheme is more suitable.

2.3.6 Zone Routing Protocol

The zone routing protocol (ZRP) blends the reactive and proactive routing techniques [Haa97]. It uses a proactive mechanism to obtain information regarding neighboring hosts located a few hops away and thus reduces latency and the need to unnecessarily flood the network. This is achieved by a neighbor discovery algorithm and periodic route advertisement. For remote hosts, ZRP uses the reactive technique to obtain the routing information as in DSR. This is a simple yet effective way to bring together the strengths of both mechanisms. The optimum balance between proactive and reactive schemes depends on the characteristics of each network.

Although both ZRP and CAMEN try to conceal the local routing information, ZRP still has to flood the whole network to obtain remote routing information. Its view of the network hierarchy from each node is also different, prohibiting the network from sharing or grouping the routing information effectively.

2.3.7 Asynchronous Multimedia Multi-Hop Wireless Networks

Lin and Gerla employ the clustering concept in an *ad hoc* network for the purpose of resource management [LiG97b]. In their work, a cluster head is responsible for allocating resources to its members for communication between clusters. This resource allocation mechanism uses a locally synchronous, slotted, time division multiplexing (TDM) frame that is shared between data and real-time traffic. In addition, a multiplexing scheme using direct-sequence spread-spectrum in each time slot is also envisioned.

This scheme, however, does not use clustering to reduce the amount of routing information or to add hierarchy into the system. Furthermore, all participating stations must be routers that are required to periodically broadcast their topologies.

2.4 Link-Level Communications Technologies

This section describes multiple schemes for link-level communications that this research effort attempts to use to their full potential at the network level in a wireless mobile *ad hoc* network (MANET) environment. A MANET, unlike a cellular wireless network, does not have a full-duplex communication capability [StT89]. To operate in a full-duplex mode, stations must be categorized into two groups, uplink and downlink, for example. Each group uses a different reception channel to which all stations in that group listen. The full-duplex capability is then available for inter-group communication via the different reception channels. To enable any-station-to-any-station communications, as is the case with an *ad hoc* network, all stations must listen to the same channel, thus eliminating any possibility for full-duplex transmission. This lack of full-duplex capability makes the coordination among participants critical to the overall network performance. Most studies regarding link-level performance assume the availability of a full-duplex connection, which makes their impact in an *ad hoc* network environment uncertain.

2.4.1 Adaptive Antenna Arrays

An antenna array consists of a set of antenna elements that are spatially distributed at known locations with reference to a common fixed point [Ron96]. Antenna arrays are known for providing interference reduction benefits in an analog network.

The adaptive antenna array changes its beam patterns automatically in response to the signal received. It can modify its beam pattern or other parameters by means of internal feedback-based control while the system is operating. By changing the phase and amplitude of the exciting currents in each of the antenna elements, it is possible to electronically adjust the main beam and/or place nulls in any direction. Most adaptive antenna algorithms are derived by first creating a performance criterion and then generating a set of iterative equations to adjust the weights so that the performance criterion is met.

By adding the capability to adjust configuration dynamically and independently for each traffic source, an adaptive antenna array provides significant improvement over a traditional antenna array for the multi-access environment.

A longer frame length and training sequence are desirable for the adaptive antenna array to realize its full potential. Prior knowledge of the communicating peer is also desirable to reduce the need for a long training sequence. Since ADIM-NB is based on a CDMA environment, it allows a longer session time, which is well suited for adaptive antenna arrays.

2.4.2 Multi-User Receiver

The major source of interference in a CDMA system is multiple access interference (MAI), which results from other users sharing the same frequency spectrum. The multi-user interference rejection technique reduces this interference by estimating the waveforms generated by other users and subtracting them from the original signal [Swa98]. This joint demodulation of multiple users in the system reduces interference, decreases overall probability of error, and increases system capacity. An additional benefit of this technique is that it allows simultaneous multiple receptions from multiple sources. However, in an *ad hoc* network environment, the mobile host cannot transmit while receiving, making it difficult to set up another connection after the existing session begins.

There are two typical types of interference cancellation techniques: successive interference cancellation and parallel interference cancellation. The successive interference cancellation algorithm estimates, regenerates, and subtracts the strongest user signal from the received signals. The procedure is then repeated to demodulate the weaker signals of other users in the system. This is a preferred technique when the power levels of the received signal are widely disparate.

The parallel interference cancellation algorithm demodulates the received signals and subtracts the estimate of each user's signal in parallel. The procedure is then repeated to improve the quality of the estimated signal for each user. This approach performs well when the received signal powers are roughly equal, like those within a typical cellular system.

In this proposal, the multi-user receiver enabled MAC protocol (ADIM-NB/MU) allows each node to simultaneously receive data, acknowledgement messages, administrative messages, channel setup requests and reply messages (RTS/CTS), and other broadcast messages from its peers.

2.4.3 Turbo Codes

Turbo codes are error correction codes that provide performance near the theoretical limit for long blocks of data with practical computing requirements. Through simulation, Berrou, Glavieux, and Thitimasjshima show that turbo codes are capable of achieving an arbitrarily small bit error rate of 10^{-5} at an E_b / N_o ratio of just 0.7 dB with block size of 64 Kbits [BGT93].

Turbo codes are parallel concatenations of two or more systematic codes. Each block of input data is replicated into several streams. Each stream is individually scrambled and encoded by a different encoder. All streams are then parallel concatenated, along with the original data, to form the final code word. The final code word is often punctured, systematically selecting a subset of the coded stream, to increase the code rate to some desired value. This puncturing technique can also be used to dynamically adjust the error correction capability needed in the system. For example, when the channel is relatively clean or the information is relatively unimportant, puncturing can be used to achieve one-half rate, i.e., a coded data block is twice as large as the original data block. If the channel becomes noisy or the information requires extra protection, this rate can be decreased to one-quarter rate.

Turbo codes favor a large frame size. With CDMA, longer frames no longer inhibit other nodes from communicating. Alternative approaches include a hybrid turbo/convolutional code system where turbo codes are used with large frames or in a hostile environment and convolutional codes are used in a normal situation.

2.4.4 Configurable Computing

Configurable computing is an emerging technology that, unlike general purpose microprocessing, enables hardware to rapidly reconfigure the functionality and interconnectivity of the computing resources to match the computational requirements of the specific applications [SHA98]. Rapid reconfiguration provides the illusion of having a much larger (virtual) hardware platform. With this approach, specific application properties such as parallelism, locality, and data resolution can be exploited by creating custom operators, pipelines, and interconnection pathways.

The runtime reconfigurable computing platform operates on a slightly different concept than a dedicated controller. In a configurable computing machine (CCM), data usually stays at the same location while the hardware keeps changing to perform different operations on that data. There are several limitations on the reconfiguration techniques, such as the latency involved in a reprogramming procedure and the technique used to retain state information during the reprogramming process. Both MAC and routing protocols have to be aware of the reprogramming techniques and associated delays.

2.5 Well-Known Problems in a Wireless Environment

This section addresses well-known problems in a MANET environment, which are challenging as well as difficult to address. Although this research effort attempts to minimize the effect of these problems, they still have some impact on the overall network performance.

2.5.1 Hidden Terminal Problem

A CSMA MAC protocol attempts to avoid collisions by testing for the presence of a transmitted signal at the station wanting to start a new transmission. However, collisions occur at the receiver, not the transmitter. Thus, carrier sensing does not provide all the necessary information to avoid a collision. This is one of the major causes of inefficiency in multi-hop CSMA networks and is referred to as the *hidden terminal* problem [LiG97b].

The simplest scenario includes three stations, each located on a single straight line as shown in Figure 2-1. The leftmost station is currently sending data to the middle station. The rightmost station is too far away from the leftmost station to notice that a transmission is taking place and, therefore, attempts to send its data to the middle station. This causes a collision at the middle station.

Left → Middle ← Right

Figure 2-1. The hidden terminal problem.

To prevent this problem from occurring, the middle station must inhibit the rightmost station from initiating the transmission, usually via a CTS message. Fortunately, the CDMA environment gracefully reduces the severity of the hidden terminal problem. The transmission between the leftmost station and the middle station occurs on a separate channel. Any transmission attempts from the rightmost station will briefly increase the interference, but the connection will still be maintained.

2.5.2 Near-Far Problem

The near-far problem is a significant disadvantage of using a DSSS/CDMA system in a MANET environment. If one user is closer to the receiver than the others, that user’s signal will be received at a higher power level than the other user’s signal, assuming an equal transmit power. This degrades the signal quality from other distant transmitters. A good power control algorithm is necessary to ensure that all signals are received at approximately the same power level. This power control algorithm is difficult to implement in a packet switching network, especially when all transmitters and receivers have to share the same frequency spectrum, as in a typical *ad hoc* network, using the half-duplex transmission medium.

ADIM-NB addresses this problem by dynamically adjusting the transmitted power and processing gain using information from the RTS packet.

2.6 Summary

This chapter reviewed several MAC protocols, routing protocols, and link-level communications technologies that are fundamental to this study. The proposed ADIM-NB MAC protocol and CAMEN routing protocol are derived from the techniques mentioned here. Related research efforts are described, along with their relationships with this research. Common problems in a wireless mobile *ad hoc* network are briefly summarized in the last section.

As strengths and weaknesses from each protocol are well understood, the next chapters try to combine their strengths and address their weaknesses in a wide range of operating environments. The next chapters explain a MAC protocol that dynamically adapts its operating parameters and a cluster-based *ad hoc* routing protocol.

Chapter 3. Objective and Methodology

The methodology used in this research and its assumptions are explained in this chapter, including evaluation strategy and analysis procedures. Jain [Jai91] has provided a guide to systematically approach the performance evaluation effort by breaking down the procedure into ten different steps. These steps are listed below and the first three steps are extensively discussed in this chapter. Steps 4 and 5 are briefly mentioned here but will be discussed more thoroughly in Chapter 7. Steps 6, 7, and 8 will be discussed in Chapter 6, along with other simulation-specific topics. The simulation findings in steps 9 and 10 will be discussed in Chapter 7.

1. State the goals of the study and define the system boundaries.
2. List system services and possible outcomes.
3. Select performance metrics.
4. List system and workload parameters.
5. Select factors and their values.
6. Select evaluation techniques.
7. Select the workload.
8. Design the experiments.
9. Analyze and interpret the data.
10. Present the results.

3.1 Goals and System Boundaries

Chapter 2 discussed the various MAC protocols, including their advantages, disadvantages, intended operating environment, and assumptions. They all have one thing in common: most of them try to be radio-independent. By not assuming the radio-specific functionality, they can operate across a wide variety of radios. However, this also means that they cannot fully utilize the radio-specific capability or take advantage of that functionality at the transport or application layer.

One example is the link characteristic. A spread-spectrum technology allows certain dynamics in its operating parameters, such as the processing gain. In addition, it is possible to adjust transmitted power and forward error correction rate in most radios. By manipulating these parameters dynamically during normal system operation, the network capacity can be improved when the conditions permit, or the network connectivity can be maintained when the conditions deteriorate.

3.1.1 Research Objective and Methodology

The overall objective of this research is try to determine how and when the MAC protocol can be customized to take advantage of a unique radio capability to improve the network performance. Specifically, an approach to take advantage of the spread-spectrum nature of the radio by dynamically changing operating parameters and encouraging multiple simultaneous CDMA sessions in the network is addressed. To achieve this objective, this investigation addresses the following specific areas.

A new, unique, and adaptive DSSS MAC protocol, ADIM-NB, is developed for a non-broadcast multiple access medium. ADIM-NB is based on a request-reply sequence that collects information during the negotiation process. It changes its processing gain, FEC coding rate, and transmitted power on a per-packet basis. It also encourages multiple CDMA sessions in the network by setting up connections with multiple neighbors at the same time and aggressively sending out the connection request messages. These multiple CDMA sessions generate little interference with each other, thanks to the use of a multi-user parallel interference cancellation (PIC) algorithm. Since spread-spectrum transceivers must agree on certain transmission parameters, such as the spreading code, coordination between stations is necessary to change such parameters.

Although the negotiation sequence at the beginning of each packet adds additional delay, this overhead significantly improves the protocol's stability and capacity. In a congested network, the request-reply negotiation sequence leads to the loss of a small control packet rather than a larger data packet.

Although past studies have investigated the impact of recent physical layer improvements, such as DSSS [Mul97] and multi-user interference rejection [TsH98], most of them concentrated on cell-based wireless networks or point-to-point links. Effects at the network-level in an *ad hoc* environment are less well understood because there are significant differences between a cellular network and a MANET. One of the major differences is that a cellular network usually operates in a star configuration. All nodes within a single cell communicate only with a central node. In an *ad hoc* network environment, however, nodes communicate freely. Another difference is that an *ad hoc* network typically has a half-duplex link [StT89], which introduces a different set of problems and issues.

Successful completion of this research requires solving new problems in wireless networks. Changing operating parameters on the fly requires accurate environment information, coordination between communicating peers, and precautions to ensure the system stability. Changing multiple communication parameters at once can also cause some interactions among different parameters, leading to an unpredictable outcome.

This research extends the existing body of knowledge within the wireless MAC protocol domain. A novel approach to take advantage of a highly configurable software radio is proposed along with supporting studies. Specifically, this investigation extends existing knowledge by defining when and how changing different communication parameters can take advantage of dynamic channel management. Anticipated benefits include increased overall throughput of the system from adjusting the processing gain, reduced delay by setting up multiple connections, better reaction to interference by modifying FEC coding rate, and reduced power consumption by controlling the transmitted power.

The secondary goal is to develop a hierarchical *ad hoc* routing protocol to take advantage of the CDMA-based radio and new functionality in ADIM-NB. Traditional *ad hoc* routing protocols cannot fully exploit the many new behaviors and extra features, e.g., broadcast, unreliable unicast, and reliable unicast, that are introduced by ADIM-NB. Additionally, many *ad hoc* routing algorithms heavily utilize or depend upon the broadcast capability of the radio, which is either unavailable or inefficient for an unicast-based CDMA MAC protocol. For these reasons, the WRP [MuG96] is augmented with CAMEN and integrated with ADIM-NB. CAMEN is designed to improve network scalability as well as to reduce reliance on radio broadcast capability. In particular, CAMEN reduces reliance on periodic HELLO messages and does not flood the network with route discovery messages.

Generally, there are two approaches to studying the performance of routing and MAC protocols: analytical and simulation. Although it is possible to employ an analytical technique to study the performance of a MAC protocol with highly simplified parameters, a routing protocol is usually too complicated to be modeled by a set of equations. Because the interaction between MAC and routing protocols is critical to the performance of ADIM-NB, using a simulation approach will provide more capability and flexibility. A wide range of operating parameters and performance metrics are used to determine the characteristics of the MAC protocol in various environment conditions.

Computer networking covers a wide range of issues including configurations, protocols, and standards. The following sections define the scope of this research and explain the issues discussed above.

3.1.1.1 Network Under Investigation

This research focuses on performance issues related to wireless mobile *ad hoc* networks. Because of its relatively long transmission range of thirteen kilometers, the propagation delay plays an important role in the design of such a MAN. Wireless LANs, on the other hand, have a higher channel capacity and much lower propagation delay, which introduces a different set of problems and issues that are not considered here. Wireless wide area networks (WANs) usually rely on different enabling technologies, such as satellites, which are not considered here either.

The network (that requires no infrastructure) under consideration is based primarily on CDMA technology. Each session can share the frequency spectrum using a different spreading code. Whereas traditional MAC protocols only allow one active session in the network, multiple sessions in the network are encouraged.

3.1.1.2 MAC and Physical Layer Implementations

Maintaining conformance with standards is an underlying objective in order to hasten acceptance by the wireless network community. The protocols developed in this research are based on an IP network. Packet format and network behavior are also the same as those in the IP network. The addressing space is a 32-bit address used by the Internet Protocol version 4 [Pos81a].

Most information and radio functionality required by an ADIM-NB are readily available from most radios. The speed at which the radio can obtain its operating environment conditions and the radio's ability to change operating parameters on a per-packet basis, however, are currently available only in a limited but increasing number of radios. Some of the recent protocols, such as the IEEE 802.11, also assume the same type of flexibility in the radio.

3.2 System Services and Possible Outcomes

The system services supported include data communications via packet-switched transmission of data frames. Each station is allowed to send data frames to another station within the wireless network. Figure 3-1 shows a time diagram for successful data transfer.

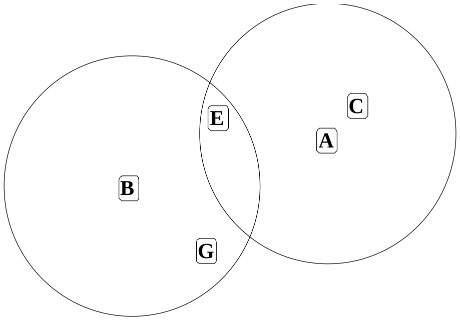
The possible outcomes are expected to depend on traffic characteristics, mobility patterns, and the network topology. ADIM-NB will be compared against a non-persistent CSMA MAC protocol and an RTS/CTS-based CDMA MAC protocol (ADIM-NB/Baseline). Network throughput, delay, utilization, error rate, and other metrics are collected and compared for all three protocols for different operating environments and parameters. Both transient and steady-state behaviors are studied. Interactions between ADIM-NB and other external factors are studied with the intent to identify areas that require further study, as well as the direction of the overall research.

Two possible outcomes are expected.

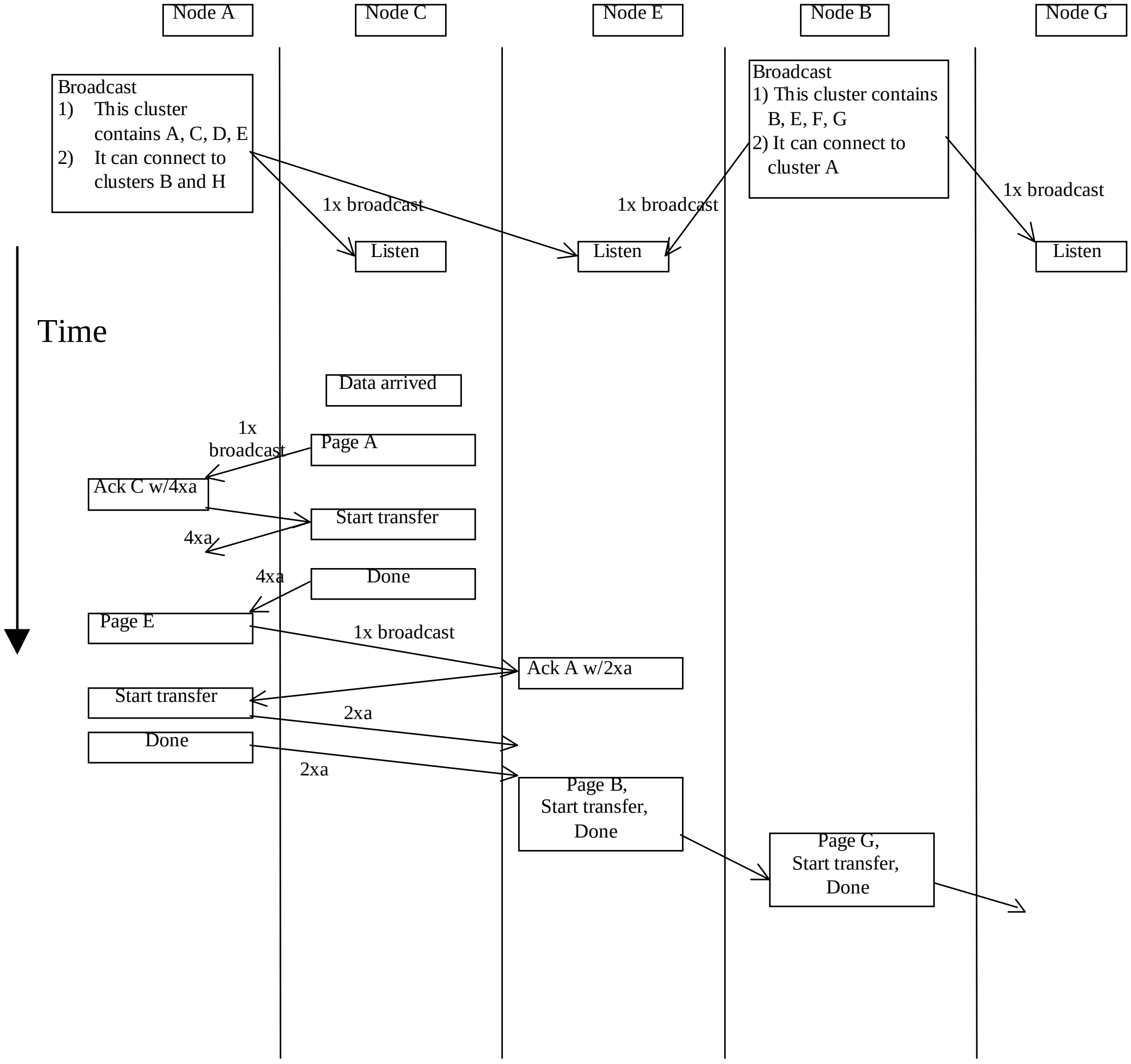
- An improvement will occur in all scenarios, or network and load configurations, under study.
- An improvement will occur in most scenarios under study.

3.3 Performance Metrics

Three main metrics are used in the performance evaluation: connectivity, throughput, and delay. Connectivity refers to the probability that a source has a valid route to the destination in its routing table, given that such a route is physically possible. Throughput is the transfer rate from all sources to all destinations in the network, i.e., the combined end-to-end throughput of all nodes in the network, in bits per second. Throughput is typically defined as the number of bytes of user data transferred per second from one station to another. It represents the ability of the network to move data in spite of various overheads. Delay is the average end-to-end delay of data transfers, in seconds. Other metrics are also considered and are defined in Chapter 6. Their suggested interpretations are presented along with the results in Chapter 7.



a) Position of reference nodes



b) Time line of events

Figure 3-1. Illustration of successful communication for node C to G.

Table 3.1 summarizes objectives and constrains of the research. There is no goal for transmission delay except that there should be no excessive buffer lengths that lead to the network instability.

Table 3.1. Development Objectives.

Network connectivity	Over 90%, given that it is physically possible
Adaptive algorithms	Speed, power, code rate, transmitted power
Primary goals	Maximize throughput and minimize delay
Secondary goals	Maximize channel utilization and minimize collision

3.4 System and Workload Parameters

This section describes the system parameters that are considered normal operating assumptions of the system model. It describes the approach used in developing the simulation models, system parameters, operating parameters, and workload parameters. Some parameters are described in greater detail in Chapters 4, 5, and 7.

3.4.1 Simulation Models

The simulation model uses a relatively high fidelity approach to calculate packet reception results. The number of error bits in each packet is calculated using a method derived from both analytical and simulation results [CBW99]. Upon packet arrival, the received signal strength of the packet of interest and each interfering packet, along with spreading code information, white Gaussian noise, and thermal noise are used to calculate the number of error bits in each packet segment. This error allocation algorithm does not assume any orthogonal spreading codes or perfect phase synchronization of the received signals. It, however, does not take into consideration any particular fading models. Therefore, the primary cause of packet corruption is collision. The number of error bits in the packet is then compared to the error correction capability of the radio to determine the final reception result.

Everything above the physical layer, including MAC and routing protocols, is modeled using a simulation model that completely mimics the behavior of actual operation.

3.4.2 System Model Parameters

System parameters include both hardware and software parameters, which generally do not vary among various installations of the system.

3.4.2.1 Capture

Capture occurs when two or more frames collide and at least one receiver successfully receives the frame addressed to it. The spread-spectrum receivers will experience limited capture of frames. Specifically, the radio receiver will lock on to the first arriving packet regardless of its relative signal strength. Capture can also occur if one transmitter is transmitting at a much higher power level than the others. The spread-spectrum receivers considered in this research effort do not have this property.

Since each radio is equipped with four receivers, it is possible for the radio to receive more than one packet at a time if the both transmitter and receiver have the same configuration. This behavior, however, is not considered capturing.

3.4.2.2 Power Considerations

Mobile computing heavily relies on battery consumption, which is usually one of the biggest concerns. Some radio transceivers consume almost as much power as the rest of the computer while continually sensing the medium, and some transceivers incorporate a sleep mode to conserve power.

This research effort assumes unlimited battery power, meaning that mobile computers will never be inoperable just because their batteries are empty. However, transmitter power consumption is one of the performance metrics that is used to justify the advantage of using the ADIM-NB MAC protocol. Although the power used in the receiving operation also plays an important role in power consumption, it is not considered here due to the difficulties in characterizing such behavior. Because of the nature of DSSS communications, the radio receiver almost always synchronizes to a carrier signal [Com85], and therefore, almost always active. Different received power from different sources also results in different power consumption in the automatic gain control circuitry, thus difficulty in characterizing such behavior.

If the remaining battery power is low, stations can demote themselves to non-routing node to conserve battery power. However, there is no sleep mode for routing or non-routing nodes. In addition, the received power is also used in calculating the appropriate transmit power to the same peer, assuming the communication link is bi-directional. Chapter 4 provides more information regarding the adaptive power control algorithm.

3.4.2.3 Addressing Modes

Three types of addressing are available to a transmitting station: unicast to a single station, multicast to a set of stations, and broadcast to all local stations. This research effort supports all transmission modes but focuses only on the unicast session. For simplicity, all MAC addresses use the same 32-bit IP address as the network layer.

3.4.2.4 Propagation Delay

Propagation delay plays an important role in the network being studied. Since the radio transceiver has a maximum transmission range of 13 kilometers, the propagation delay is significant and is being considered in every aspect of the research. The actual propagation delay between stations is varied according to the distance between them.

3.4.2.5 Frame Length

Frame length is fixed at 18,496 bits. This is the same maximum transmit unit (MTU) as the IEEE 802.11 standard [IEE99]. Although the data transmission rates of ADIM-NB and the IEEE 802.11 standard are different, they both seem to have the best performance with the same MTU. The actual average packet size, however, is varied from the type of traffic that a station is transmitting.

3.4.2.6 System Queues

Two queues are used to transmit a data packet, one for unicast and the other for broadcast traffic. The broadcast traffic has higher priority because it tends to carry routing information. Within each queue, packets are served on a first-come-first-served basis. Each station is equipped with a finite capacity queue within the MAC layer. The actual queue size is varied in proportion to the channel speed, or chip rate, being used.

3.4.2.7 PN Code Sequence

No specific PN code is assumed. The only assumption regarding PN code is that they do not have to be orthogonal with each other [CBW99]. Several PN codes with different lengths are assumed to be available to the MAC protocol to choose from.

3.4.2.8 Modulation Format and Channel Data Rates

This research only address binary phase shift keying (BPSK) modulation format, although in theory, an ADIM-NB does not preclude or assume any type of modulation formats. Two data speeds are studied in this research, 1 Mchips/second using the Virginia Tech reconfigurable radio and 3.6864 Mchips/second using CDMA2000 radio.

3.4.2.9 Multi-user receiver

All radios have a four-user multi-user receiver employing a parallel interference cancellation (PIC) algorithm [Swa98]. Such a radio is capable of canceling spread-spectrum interference if there are no more than four simultaneous transmissions in the local subnet. The data from all four sessions is available to the MAC protocol.

3.4.3 Operating Parameters

Operating parameters are pieces of information collected during system operation to be used to make an informed decision. They can be considered as input information during the operation. These parameters can be considered environment information, which is external to the radio. Over the operating duration, a radio collects operating parameters and processes them in order to make an informed decision on how the radio will function. ADIM-NB and CAMEN react differently to various environment parameters.

3.4.3.1 Signal-to-noise Ratio

The most important operating parameter is the SNR of the most recent packet, which is used to determine the appropriate data transmission speed. Because the SNR value can vary within the same packet, only the SNR value at the time of complete packet reception is used.

3.4.4 Workload Parameters

Workload parameters are characteristics of users' requests, which vary from one installation to the next. They are the simulated application traffic to exercise the radio, MAC protocol, and the routing protocol.

3.4.4.1 Network Topology

Given the dynamic nature of the mobile computers being considered, a lot of network configurations are possible. This effort restricted the mobile computer to be within a given, predetermined boundary. The node placement within this boundary follows a uniform random distribution. Over the course of the operating period, nodes are free to move around the network using one of two mobility patterns, constant speed and direction, and "city block" [MLA98]. In "city block" mobility pattern, nodes move in a grid-like fashion, modeled as the Manhattan road grid. Node speed is varied from static to 65 Kph. The default value of the network coverage area is 55 by 85 Km.

3.4.4.2 Number of Stations

The number of stations contending for the channel affects network performance. Although the number of stations is constant when studying the ADIM-NB MAC protocol, it is changed to characterize the scalability of the hierarchical *ad hoc* routing protocol, CAMEN.

3.4.4.3 Packet Generation Rates

Since computer traffic typically involves bursty traffic, a packet switching mechanism is assumed. Each station generates two types of traffic: simple mail transfer protocol (SMTP) [Pax94] and HTTP [CBC95]. The average arrival rate is varied from experiment to experiment to study the network performance in different loading conditions. To determine the network performance in a stressful condition, the majority of traffic characteristics employed in this research are either the maximum allowable workloads before the protocol degrades or the network capacity.

3.4.4.4 Forward Error Correction

Each station can vary the FEC rate being used. In addition to the FEC, the cyclic redundancy check (CRC) in each frame provides additional capability to detect uncorrectable errors in the frame. It is also assumed that the radio can provide the actual number of error bits, up to some limits, in each frame to the MAC protocol to be used in calculating the proper FEC coding rate for future sessions.

In addition to the number of bits in error, the number of collisions is also used in adapting the FEC coding rate. Such information, up to some limit, is provided by the multi-user receiver enabled radio.

3.4.4.5 Selection of Destination Stations

The potential destination stations are divided into two groups: local and remote. Within each group, each station is equally likely to be selected as the destination station. The traffic generator selects the destination group based on the traffic characteristic being used. Although this assumption facilitates a balanced load across all stations, the stations in the middle of the network tend to receive more traffic because they have a higher connectivity and behave like an aggregation point for the entire network.

3.4.5 Factors

Factors are parameters that can be changed over time such that they significantly impact the system performance [Jai91], representing an opportunity to adapt the radio to a specific environment. Factors are a subset of workload parameters specified in Section 3.4.4. Each factor can assume different values or levels. The major factor that an ADIM-NB uses is the data transmission speed, which has three levels: base transmission speed, twice base speed, and four times base speed. Other speeds are possible with a more sophisticated spread-spectrum encoder/decoder, but they are not investigated in this research. Additional factors include transmit power, error correction coding algorithm, and various backoff parameters; these are described in greater detail in Chapters 4, 5, and 7.

The MAC protocol, which can be non-persistent CSMA, ADIM-NB/B, or ADIM-NB, can also be considered as a factor in this study. The non-persistent CSMA protocol and ADIM-NB/B do not change any parameters during their operation expect the backoff duration, which follows the exponential backoff algorithm.

3.5 Summary

This chapter presented the objectives of this research effort as well as the methods used to attain these objectives. As an introduction, the chapter presented a proven systematic approach to computer systems performance evaluation as described by Jain [Jai91]. An introduction to the problem and the objective of the research was presented in Section 3.1. Section 3.2 described the system behavior in general and the expected results. The last two sections described system metrics and parameters.

Chapter 4. ADIM-NB MAC Protocol Algorithm Definitions

This chapter describes the ADIM-NB MAC protocol developed in this research. The opportunistic ADIM-NB is designed to adapt its functionality and operating parameters to the current environment. It also takes advantage of a radio with multi-user receiver capability. Examples and flow charts are provided throughout the chapter to help the reader visualize the algorithms. Section 4.1 describes baseline functionality of the protocol (ADIM-NB/B) and the expected operating environment. The next four sections explain in detail each adaptive algorithm that makes up ADIM-NB. All adaptive algorithms are relatively independent from each other and can be enabled/disabled separately. Section 4.2 provides the description of the adaptive processing gain algorithm (ADIM-NB/PG), which decreases the DSSS coding rate when condition permits. Section 4.3 describes the power control algorithm (ADIM-NB/PC). Section 4.4 explains an algorithm that changes the FEC coding rate in response to the noise and interference in the system (ADIM-NB/FE). Section 4.5 explains the technique used to increase the utilization of the multi-user receiver capability in a half-duplex environment by synchronizing the channel setup operation (ADIM-NB/MU). The last section gives a preview of improvements from each adaptive algorithm, individually and jointly, in the presence of other adaptive algorithms.

4.1 Basic Operation in CDMA Mode

ADIM-NB/B uses a CDMA technique to separate data sessions from one another. Every station in the network shares the same frequency spectrum using a different spreading code in a DSSS CDMA system. Currently, this technique has been used primarily in circuit-switching networks, where each user is given a semi-permanent spreading code to communicate with a specific peer. This dedicated communication channel usually links between a handheld unit and a base station, such as those in the IS-95 standard [TIA95].

To allow two communicating peers to select the same spreading code, which must be different from the codes used by other users, a negotiation mechanism using a common medium is required. This process is achieved by means of a control channel. The negotiation process is illustrated in Figure 4-1. All idle stations always tune their receivers to this control channel, which can also be used for broadcast messages. Upon receiving a packet from the higher layer protocol, the source station chooses a random spreading code and encodes it into a RTS message. The selected spreading code is unique and will only be used for the subsequent data packet. The RTS packet also contains additional information regarding the size of the data packet. This packet size information will be used at the receiver to calculate an appropriate timeout value. The complete description of the RTS packet format is available in Appendix A. The source then transmits this RTS packet to the destination using the control channel.

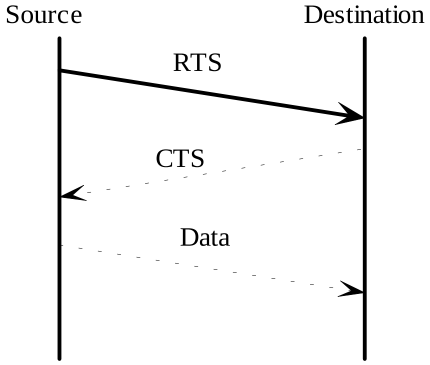


Figure 4-1. ADIM-NB/B negotiation time diagram.

Upon successfully receiving the RTS packet, the destination sets both its transmitter and receiver to the specified spreading code, calculates the duration that it will wait for the data packet using the packet size information, and replies back to the source with a CTS message. The CTS message contains various communication parameters, including processing gain, transmission power, and error correction coding rate. To avoid collision with other broadcast messages and to reduce the dependency on the radio's broadcast capability, the CTS message is transmitted back to the source using the negotiated spreading code. After receiving the CTS message, the source reconfigures its transmitter with the communication parameters recommended by the receiver and transmits the data packet using the negotiated spreading code. After finishing the transmission, both stations reconfigure their radio transceiver to the default parameters and listen to the control channel for the next session. Flowcharts of ADIM-NB/B are illustrated in Figures 4-2 through 4-7.

4.1.1 Aggressiveness

Unlike other RTS/CTS-based MAC protocols such as FAMA or BTMA, stations overhearing either RTS or CTS messages do not inhibit their transmission attempts. The reasoning behind this approach is specific to the intended radio platform, which operates in a CDMA environment. In a CDMA environment, where multiple radios can share the same communication medium using different spreading codes, simultaneous transmission should be encouraged, not discouraged. In the ADIM-NB/B MAC protocol, each station aggressively transmits RTS messages regardless of the state of their neighbors. By not requiring the station to keep its neighbors' state information, the algorithm is less complex and easier to integrate with a more sophisticated security mechanism.

If stations cannot attempt transmission when they hear either an RTS or a CTS message, there will be only one active transmission in a given area. This limitation effectively eliminates many advantages of using a CDMA radio, namely, multiplexing multiple transmissions over the same frequency spectrum with a different spreading code.

Although the impact from the hidden terminal problem that often degrades a mobile *ad hoc* network is less severe in a spread-spectrum environment, the problem still persists. ADIM-NB/B tries to reduce the severity of the hidden terminal problem through some other techniques, which will be described throughout this chapter. These techniques attempt to keep the connection active and reduce the impact from the hidden terminal problem at the same time, rather than attempting to avoid collision altogether.

4.1.2 Acknowledgement Scheme

If the destination is unable to reply with a CTS message, perhaps because it is initiating another session or simply is busy, the source will drop the data packet from its transmission queue. A packet will also be dropped if the negotiation sequence is incomplete, including a corruption of either an RTS or a CTS message. The dropped packet will be retransmitted from a retransmission queue.

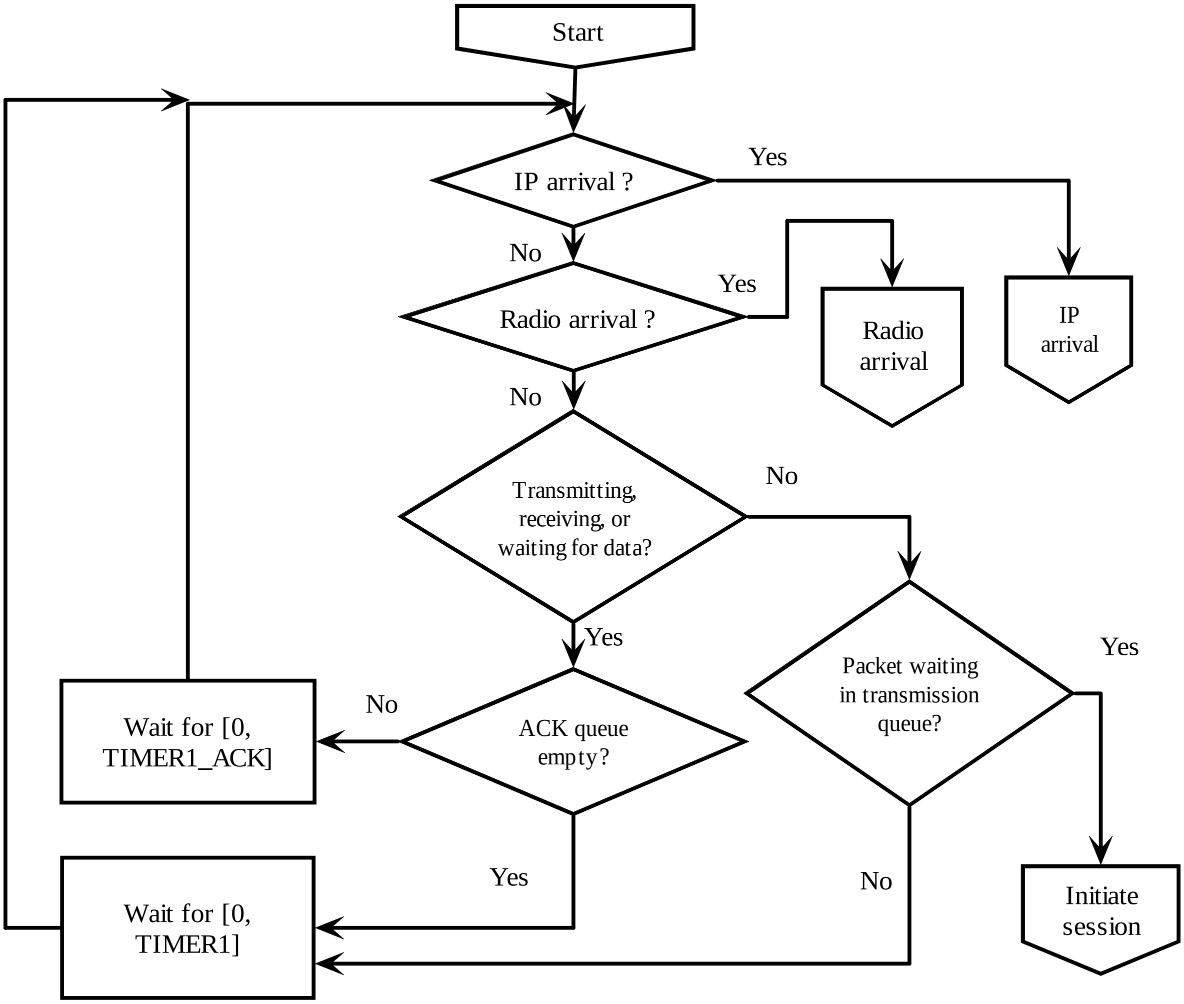


Figure 4-2. ADIM-NB/B operation flowchart (1/6).

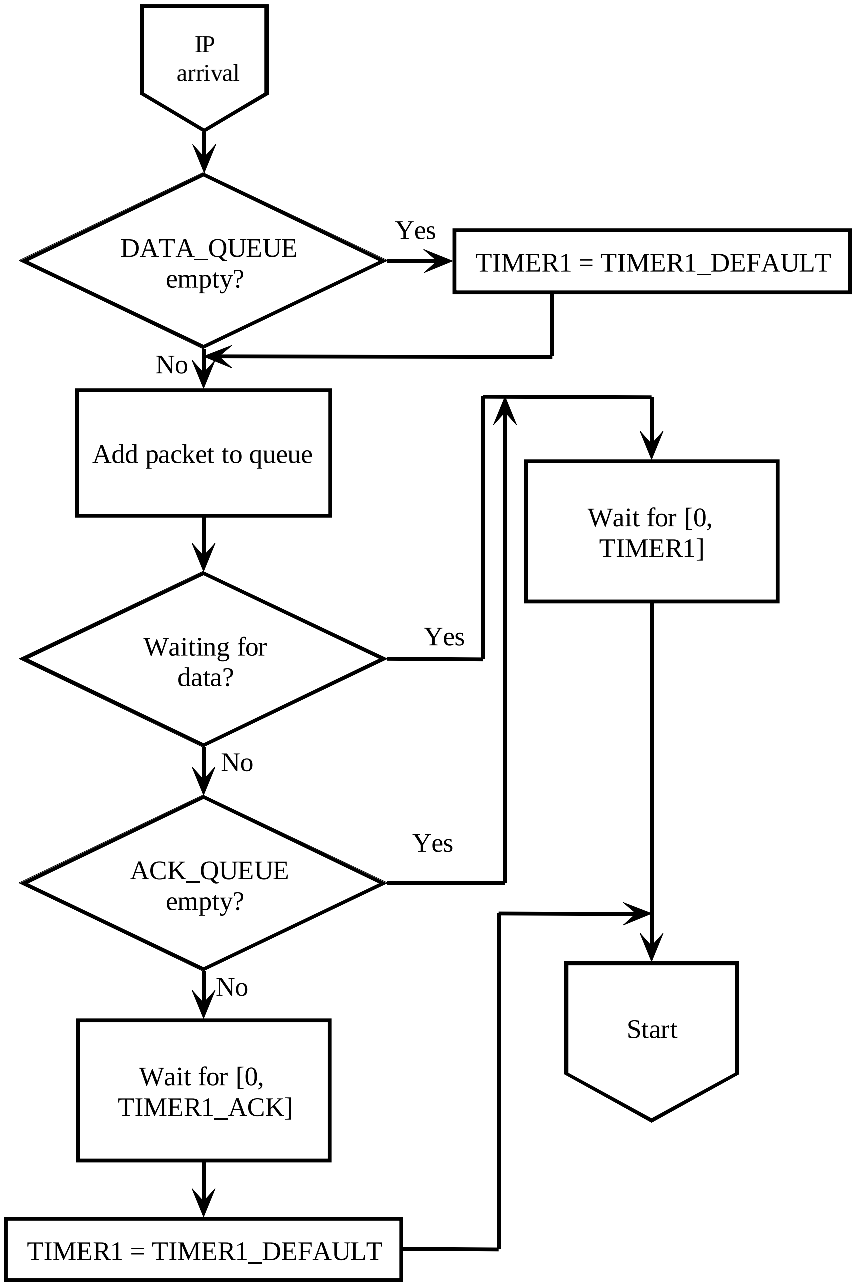


Figure 4-3. ADIM-NB/B operation flowchart (2/6).

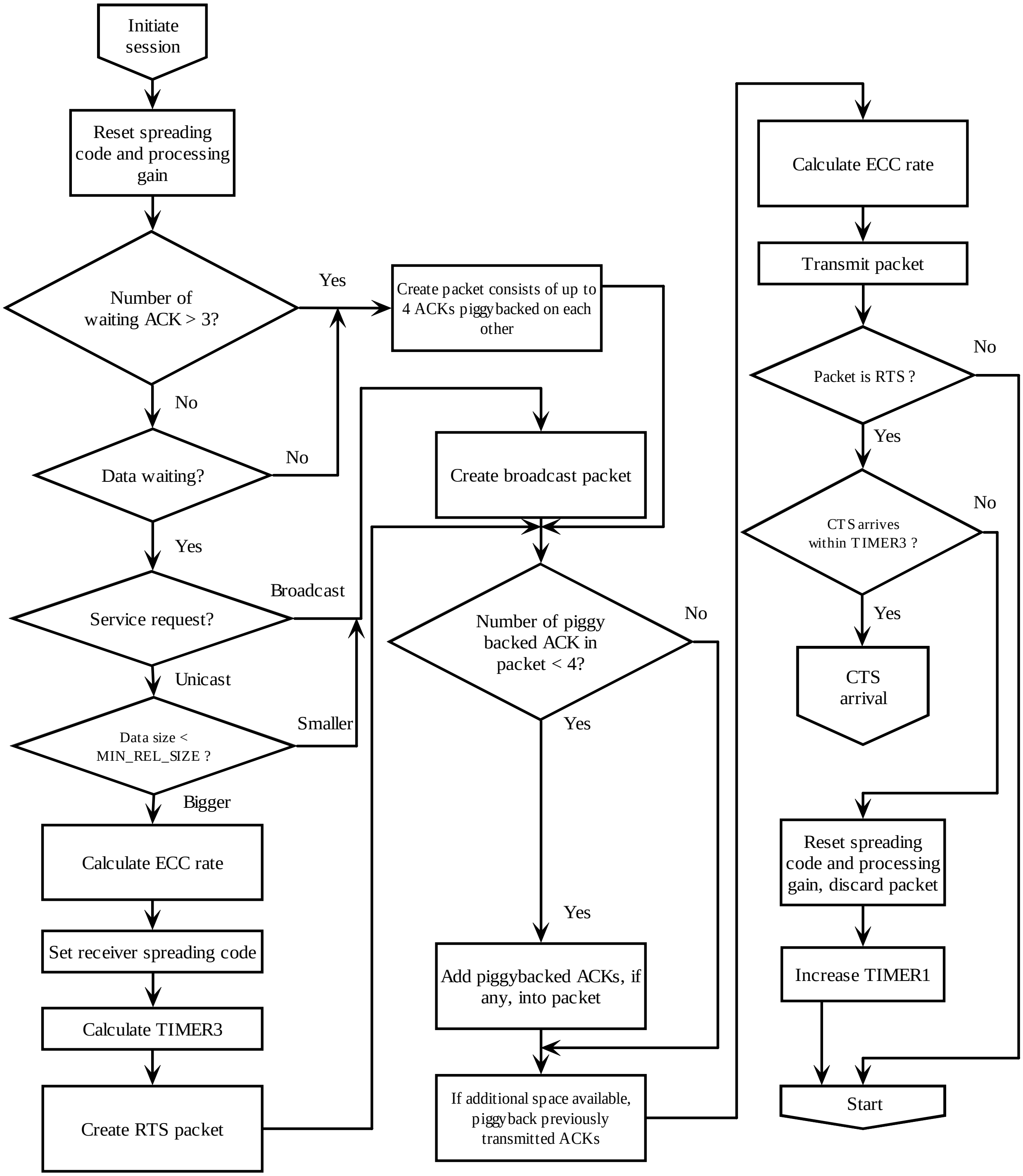


Figure 4-4. ADIM-NB/B operation flowchart (3/6).

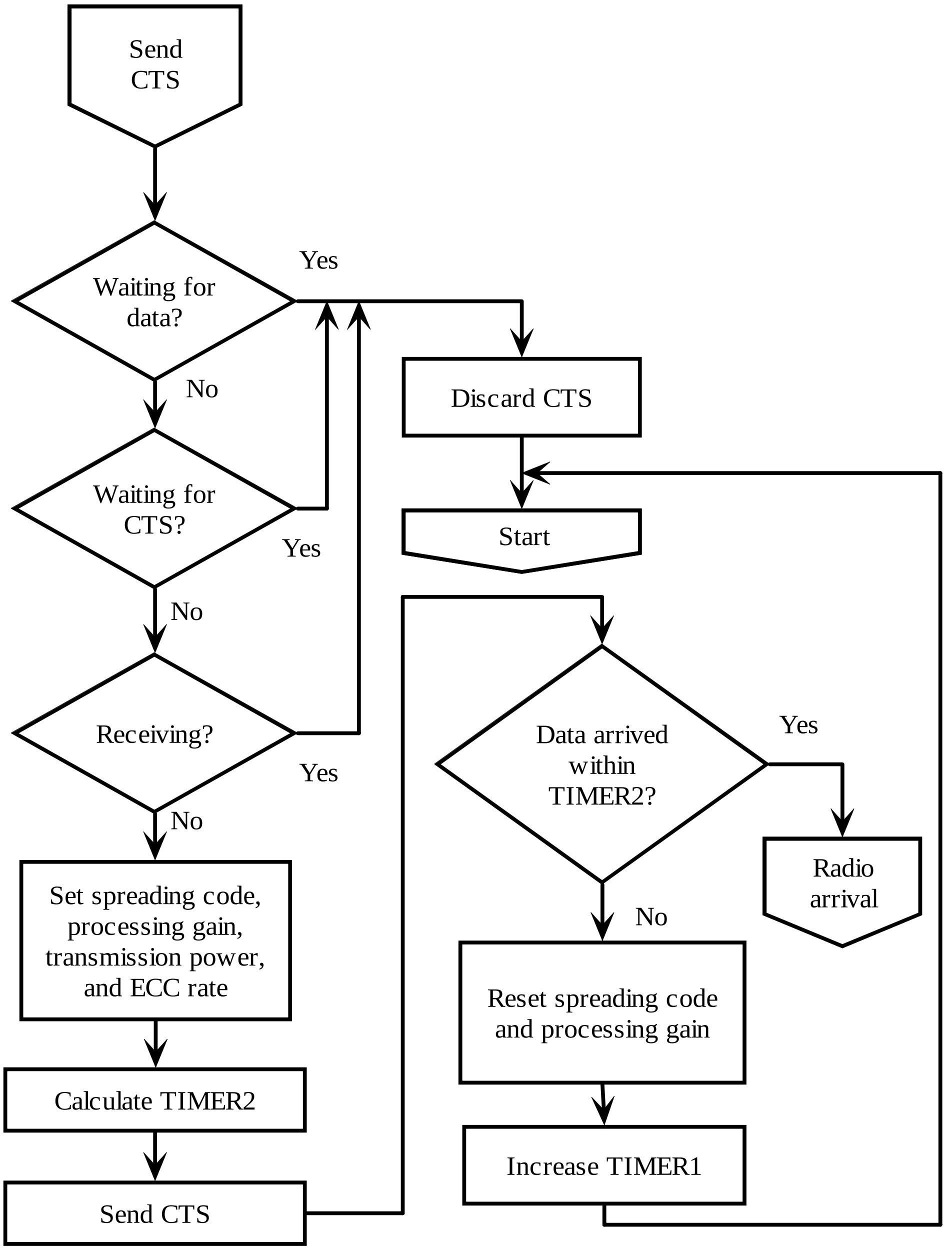


Figure 4-5. ADIM-NB/B operation flowchart (4/6).

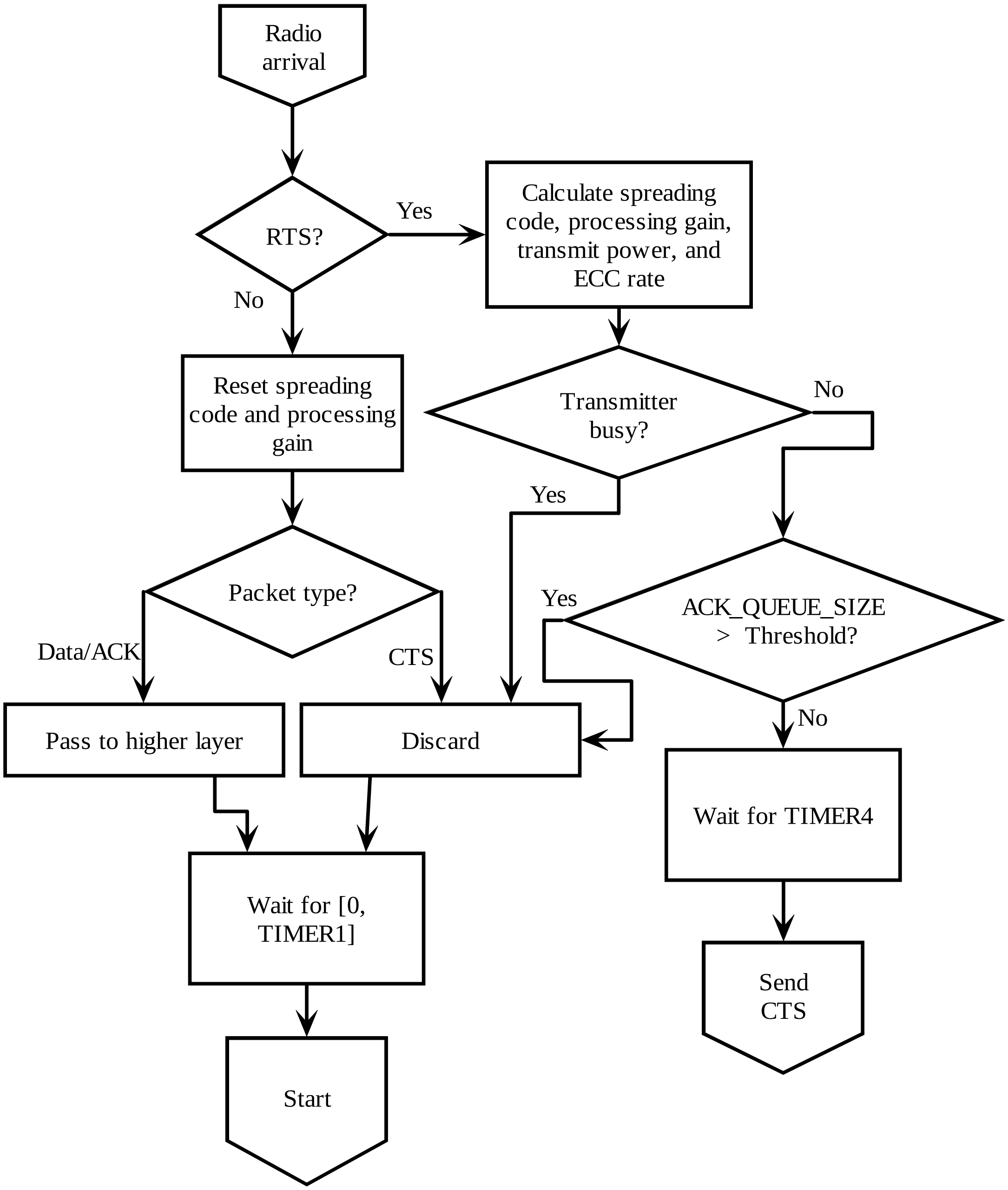


Figure 4-6. ADIM-NB/B operation flowchart (5/6).

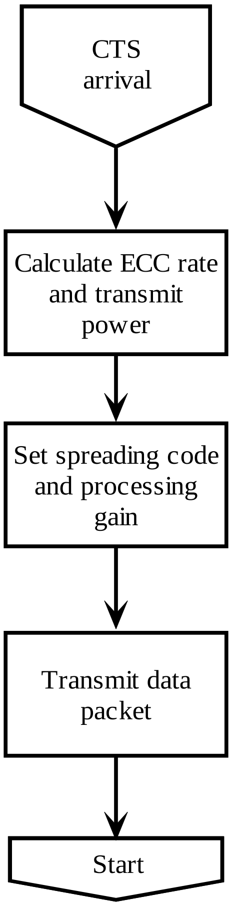


Figure 4-7. ADIM-NB/B operation flowchart (6/6).

Reliable data transfer is accomplished by a hop-by-hop acknowledgement scheme, automatic repeat request (ARQ). A selective acknowledgement scheme is chosen for its simplicity and suitability in a wireless environment. Each packet is assigned a unique sequence number and timeout value (*TIMER5*) using an algorithm similar to a TCP-style timeout mechanism [Pos81b]. This sequence number is also used to detect and discard any duplicated packets. The timeout variable is calculated using the following equations:

$$TIMER5\_Est = (\text{TIMER5\_Alpha} * TIMER5\_Est)$$

$$+ (1 - \text{TIMER5\_Alpha})*(Actual\_packet\_delay),$$

$$TIMER5 = \text{TIMER5\_Beta} * TIMER5\_Est,$$

$$\text{TIMER5\_LowerBound} < TIMER5 < \text{TIMER5\_UpperBound}.$$

A packet is retransmitted if its acknowledgement message (ACK) is not received within *TIMER5*. This retransmission process is repeated until the packet lifetime is expired. Upon packet expiration, ADIM-NB/B protocol signals the routing protocol of the delivery failure. The routing protocol can then perform the necessary steps to remove the unreachable destination from its routing table.

To reduce a per-packet overhead consisting of packet preamble and header, up to four ACK messages can be piggybacked on to an outgoing packet that uses the control channel. These four ACK messages carry their own mini-header and are not necessarily heading for the same destination. The piggybacked process is illustrated in Figure 4-4.

Because an ACK message is not transmitted immediately after the packet is received, and the station expecting the ACK message is not required to wait for it, ACK packets have a reasonably high loss rate. This non-immediate ACK transmission approach is desirable for a piggybacked ACK algorithm and necessary for the ADIM-NB/MU algorithm. If an ACK message is transmitted immediately upon packet reception, it cannot be allowed to combine with another packet because that packet will have an unfair transmission advantage over packets from other waiting stations. By not combining ACK messages with other control messages, the per-packet overhead is increased. For the ADIM-NB/MU algorithm, a receiver can receive more than one transmission from different sources. These transmissions do not always finish at the same time and the receiver cannot transmit ACK messages until all transmissions are completed. Forcing all sources to wait for the delayed ACK messages would significantly reduce the protocol effectiveness. ADIM-NB/MU will be explained in more detail in Section 4.5.

If an ACK message is lost, a packet will be unnecessarily retransmitted. To avoid such a waste of resources, one duplicate ACK is transmitted with each outgoing control packet. The added overhead from transmitting duplicate information is relatively small, compared to the packet size. All previously transmitted ACK messages are stored in a list. If there is an available space for a piggybacked ACK in the outgoing packet, one ACK is randomly picked from this list and inserted into the packet. There is an artificial limitation of at most one duplicated ACK for each outgoing packet. This retransmission list is of limited size and an old entry can be replaced with a new entry as needed. Each entry can be retransmitted more than once or not at all, depending on the ratio of incoming and outgoing traffic at a particular station.

4.1.3 Packet Priority

The MAC protocol maintains two queues, a data queue and an acknowledgement queue. The default ACK queue size is forty packets and the default data queue size is five packets. All packets that have arrived from the higher layer have to wait in the transmission queue. In Figure 4-2, ADIM-NB/B periodically senses the medium for a chance to transmit. If the medium is busy, it sleeps for a random amount of time before it re-senses the channel, i.e., a non-persistent CSMA approach. If the ACK queue is not empty, this sleep period is significantly shorter (*TIMER1_ACK*) than if the ACK queue is empty (*TIMER1*). Using this approach, stations that have a waiting ACK message wait a shorter amount of time and can re-gain the channel faster, resulting in a faster piggybacked ACK being transmitted.

In Figure 4-6, if the ACK queue size grows increasingly large, additional connection requests will be denied until the ACK queue size has been reduced to a reasonable size. In other words, after receiving a RTS message, the station will not reply with a CTS message if there are excessive ACK messages waiting to be transmitted. This behavior prevents buffer build-up at the station.

Within the data queue, a broadcast packet has a higher priority than a unicast packet and will be transmitted first. This behavior is specifically tailored to the application traffic being considered. Application data tends to use unicast service while routing and cluster maintenance messages tend to use broadcast service. Giving a higher priority to routing-related messages can help maintain network connectivity in a congested environment.

4.1.4 Reusing a Previously Allocated Transceiver

The radio transceiver is a valuable resource, which has to be reused as quickly as possible. If the receiving station does not respond within a reasonable duration, the transmitting station drops the negotiation sequence and resets its transceiver parameters to a default value so that it can communicate with other stations. Figure 4-4 shows the reclaiming algorithm if the CTS message does not arrive in time. Figure 4-5 shows the same process when waiting for a data packet. In addition, when the session is completed, both peers switch to a default communication parameter in order to communicate with other stations. The process is slightly different with the ADIM-NB/MU algorithm because of the extra transceivers available, as will be explained in Section 4.5.

4.1.5 Expected Operating Environment

The MAC protocol is specifically designed for a CDMA environment where multiple sessions share the same frequency spectrum using different spreading codes. There are many reasons for using CDMA instead of the traditional TDMA or FDMA. For example, it might be desirable to use the spread-spectrum technique for security or regulatory purposes. It may also be desirable to utilize new CDMA-based technologies such as the multi-user receiver and interference rejection in a DSSS radio. The CDMA approach also allows greater flexibility for changing the transmission parameters on a peer-to-peer basis without having to synchronize with the rest of the network.

There is no provisioning for station priority and each station has an equal chance of transmission. The design objective is to make every station equally aggressive in setting up a connection to avoid unnecessary idling. Eliminating an assumption that a central controller exists does introduce some inefficiency and additional contention into the system. This makes it unsuitable for a cellular system when compared to other MAC protocols that make use of a central controller.

4.1.6 Parameters

This section describes major parameters and variables referenced in Section 4.1. The same variable may have a different value or may be calculated from a different set of equations when a different ADIM-NB algorithm is being used, as will be explained in Sections 4.2 through 4.5.

4.1.6.1 TIMER1

TIMER1 is the maximum duration in which the ADIM-NB/B algorithm should wait before the next RTS or broadcast transmission attempt. Its default minimum value (TIMER1_DEFAULT) is 0.01 sec. This value is slightly longer than the RTS packet duration to provide opportunity for other stations to initiate their transmission requests. The default maximum value is calculated from the equation

MAX_BACKOFF = 1.0 – TIMER3 - DEFAULT_WAIT_TIME_DATA.

TIMER1 is increased every unsuccessful transmission attempt according to the equation

TIMER1 = TIMER1 * BACKOFF_RATE

After emptying the transmission queue or if the ACK queue contains one or more entries, *TIMER1* will be reverted to its default minimum value, as shown in Figure 4-3.

4.1.6.2 TIMER2

TIMER2 is the duration in which the ADIM-NB/B algorithm should wait for a data packet after transmitting the CTS message. If the data packet does not arrive within the allotted time, the transceiver is made available for another packet transmission. *TIMER2* is calculated using the equation

TIMER2 = RTT_RTS_CTS + Transmission delay + Propagation delay.

4.1.6.3 Other Constants

Table 4.1 summarizes important constants in the ADIM-NB/B MAC protocol. The default value for the DEFAULT_WAIT_TIME_DATA constant estimates the time required for the requesting node to receive a response, plus 50% margin. The default value for the *TIMER1_ACK* variable accounts for the packet acquisition time using a 64 chips preamble. The 10% default backoff rate represents an aggressive transmission scheduler as explained in Section 4.1.1. The minimum unicast packet size estimates the average packet size of a transaction-based communications such as SMTP commands.

4.2 Adaptive Direct-sequence Spread-Spectrum

Spread-spectrum techniques possess the ability to trade a lower data rate for a higher SNR. By manipulating the coding rate, the varying noise level in the channel can be accommodated or the signal strength fading due to the varying distance between the transmitter and the receiver can be tolerated [MDM97]. Using this characteristic, ADIM-NB/PG can dynamically increase or decrease the transmission rate as conditions dictate by changing the spread-spectrum processing gain on a per-packet basis. By reducing the processing gain, data can be transmitted at a higher rate.

Table 4.1. ADIM-NB/B Constants.

Symbol	Default Value	Description
RTT_RTS_CTS	0.013424 sec	Round-trip time for RTS/CTS message, equal to 2 * (Transmission delay + Propagation delay)
DEFAULT_WAIT_TIME_ DATA		Default timeout while waiting for data, equal to (1.5*RTT_RTS_CTS) + Transmission delay + Propagation delay
TIMER1_ACK	0.001 sec	Maximum duration in which ADIM-NB/B should wait before the next ACK transmission attempt
TIMER3	RTT_RTS_CTS	Default timeout after transmitting RTS and waiting for CTS, has a different value when used with the ADIM-NB/MU algorithm
TIMER4	0 sec	Delay between receiving an RTS request and transmitting a CTS reply, non-zero in the ADIM-NB/MU algorithm.
BACKOFF_RATE	10%	Backoff rate for *TIMER1*
MIN_REL_SIZE	1320 bits	Unicast packets (including all PDU headers) smaller than this value are transmitted using the control channel without negotiation

The processing gain can be changed by modifying the number of chips per bit. If the number of chips per bit decreases by one-half, the processing gain decreases by 3 dB [Skl88]. Since the transmitted chip rate remains constant, this allows data to be transmitted twice as fast. Using the ADIM-NB/PG algorithm, stations can transmit at a higher speed when they are closer to each other because the processing gain requirement is not as high as when they are far apart. The ADIM-NB/PG algorithm uses three classes of spreading codes: low data rate/long range, medium data rate/medium range, and high data rate/short range. Each communicating peer intelligently selects a code from the appropriate class based on available information.

This technique, however, has to rely on additional information from channel sensing mechanisms and operating history to select the optimal trade-off. The primary indicator employed in the ADIM-NB/PG algorithm is the SNR in the received RTS packet. The processing gain can be reduced as long as its estimated SNR is greater than a predetermined threshold, allowing data to be transmitted at a higher speed. Signal strength from a nearby source is usually strong enough such that the final signal fidelity does not degrade even when the lowest processing gain is used. This allows nodes that communicate over a short distance to use the highest possible data rate. The ADIM-NB/PG algorithm chooses the appropriate processing gain based on two variables: the instantaneous SNR of the RTS packet (*SNR_rts*) and the estimated SNR collected over multiple packets (*SNR_est*).

In Figure 4-5, upon receiving an RTS packet, the station measures the SNR of the received packet. Over the RTS packet duration, this SNR value can change depending on the source of the interference in the system. To provide the most up-to-date information, the *SNR_rts* is a value given as soon as the packet completes reception.

Every time a station receives an unicast packet, it also uses that packet's SNR to calculate *SNR_est* using the equations

SNR_est = (SNR_ALPHA**SNR_est*) + ((1-SNR_ALPHA)**SNR_rts*),

SNR_meas = Min (SNR_*est*+SNR_BETA, *SNR_rts*).

If *SNR_meas* is more than 4 dB (3dB plus a margin of 1 dB) higher than the minimum SNR (MIN_SNR), then the processing gain can be reduced by half and the transmission bit rate can be doubled. The reduction keeps going until the *SNR_meas* is less than 4 dB higher than the MIN_SNR. If the received SNR of the RTS message is already lower than the MIN_SNR, then this RTS packet will be silently discarded. After the appropriate processing gain and data rate has been determined, the destination relays this information back to the source via a CTS reply so it can begin the data transmission using the best possible transmission speed.

Figure 4-8 illustrates the basic operation of a successful negotiation. Upon receiving a packet from a higher layer protocol, the source station, Source, chooses a random spreading code, User1, and encodes it into an RTS packet. Source also sets its receiver's spreading code to User1, which is unique and will only be used for the following data packet. The RTS packet also contains additional information regarding the size of the data packet. This packet size information will be used at the receiver, Destination, to calculate the appropriate timeout value. Source then transmits this RTS packet to Destination using the control channel, which uses Default spreading code.

Upon successfully receiving the RTS packet, Destination sets both its transmitter and receiver to the specified spreading code, calculates the duration that it will wait for the data packet using the packet size information, and replies back to the source with a CTS packet. It also measures the signal strength of the RTS packet and calculates appropriate transmission speed for subsequent data transmission. For this particular example, the SNR of the RTS packet is high enough to allow the highest transmission speed, 256 Kbps/4 chips per bit, to be used. Destination then sets its receiver data speed to 256 Kbps, as well as encoding the same data transmission speed in the CTS packet. The CTS packet also contains various communication parameters, including transmission power and error correction coding rate. To avoid collision with other broadcast messages and to reduce the dependency on the radio's broadcast capability, the CTS packet is transmitted back to Source1 using the negotiated spreading code, User1. After receiving the CTS message, Source1 reconfigures its transmitter with the data transmission speed recommended by the receiver, 256 Kbps, and transmits the data to Destination. After finishing the transmission, both stations reconfigure their radio transceiver to the default parameters and listen to the control channel for the next session.

Table 4.2 contains other parameter and constant settings. For the BPSK modulation, the minimum SNR will result in an approximately 0.7% BER, which is still correctable. The SNR_ALPHA and SNR_BETA values are based on a conservative approach that emphasizes long-term effect rather than instantaneous effect.

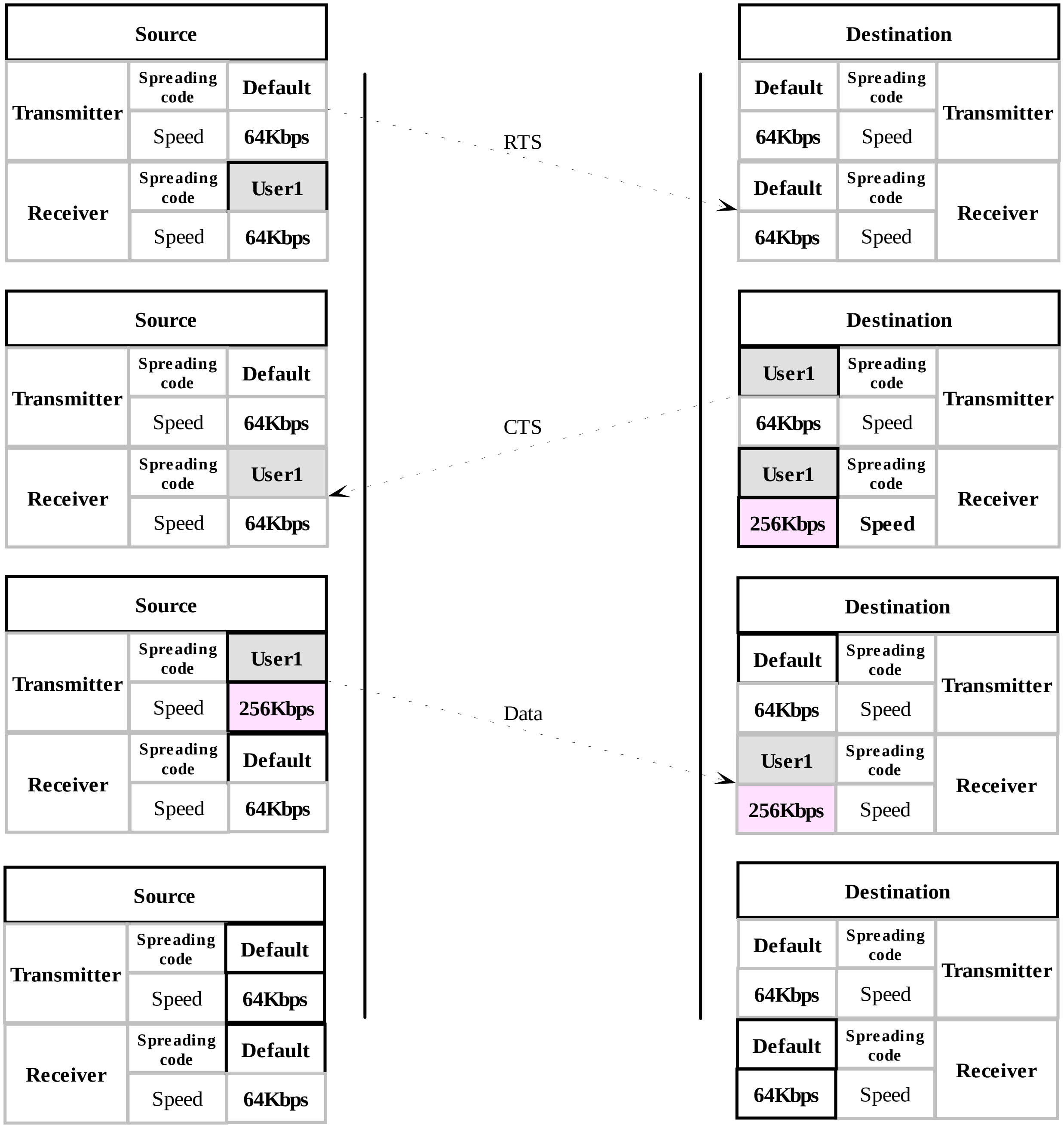


Figure 4-8. Time diagram demonstrating the ADIM-NB/PG algorithm.

4.3 Power Control Algorithm

The ADIM-NB/PC algorithm adjusts the transmission power by using the information from the RTS packet, as well as previously received unicast packets. It operates using the same principle as a regular cellular telephone power control algorithm, which attempts to equalize the received power from various transmission sources. The ADIM-NB/PC algorithm shares the same information with the ADIM-NB/PG algorithm (*SNR_meas*). After considering any reduction in the processing gain by the ADIM-NB/PG algorithm, it is possible that the estimated SNR (*SNR_meas*) is still higher than the MIN_SNR. This difference between *SNR_meas* and MIN_SNR is an excess transmission power (*excess_SNR*), which is not necessary for a correct packet reception. Thus, the transmission power can be reduced by this difference. After verifying the lower bound (-9.0 dB) and the upper bound (0.0 dB) for *excess_SNR*, the station relays this information to the requestor via a CTS packet.

Table 4.2. Constants Used in the ADIM-NB/PG Algorithm.

Symbol	Default Value	Description
MIN_SNR	4.0 dB	Minimum desirable SNR that will result in a correct packet reception, based on a BPSK modulation
SNR_ALPHA	0.9	Indicating the importance of previous estimated SNR value in comparison with the latest measured value in calculating *SNR_est*
SNR_BETA	3.0 dB	Margin for calculating *SNR_est*

The transmission power levels for CTS and data packets are different. Because of its small size and importance, a CTS packet is transmitted with a 3 dB margin, as shown in the equations below.

CTS_power = Default power – *excess_SNR* + POWER_ADJUSTMENT_CTS.

POWER_ADJUSTMENT_CTS = 3 dB.

The 3 dB value is an optimum value obtained by studying the bit error rate (BER) statistic of each CTS packet in comparison with the BER of other types of packets. The transmission power level remains constant during the packet duration. After receiving the CTS packet, the source station adds a 0.4 dB margin to the calculated power value, according to the equations shown below.

Data_power = Default power – *excess_SNR* + POWER_ADJUSTMENT.

POWER_ADJUSTMENT = 0.4 dB.

The 0.4 dB value is also obtained from studying a BER pattern of each data packet. The ADIM-NB/PC algorithm will result in an approximately equal receiving power at the receiving end. The primary advantage of using the ADIM-NB/PC algorithm is a reduction in power consumption per bit received, though there is no increase in the network throughput. The fact that throughput does not improve indicates that the near-far problem still exists, primarily because stations cannot control the transmission power of every station in the listening area.

The RTS packet and other broadcast packets cannot be power-controlled because they are destined for multiple destinations. Because there is only one class of service, all packets share the same minimum received power, after factoring in their processing gain. Different minimum received power is desirable if additional classes of service must be supported.

4.4 Adaptive Forward Error Correction Algorithm

The objective of the adaptive FEC algorithm (ADIM-NB/FE) is not to increase the network throughput, as it is generally more effective to keep the FEC rate fixed for each type of environment. The ADIM-NB/FE algorithm is designed to enable an acceptable operation in the presence of a varying number of jamming nodes and interference levels and to eliminate the need to manually adjust the parameters. The ADIM-NB/FE algorithm keeps track of an estimated BER (*ber_est*) at each node and uses that value to adjust the FEC rate for each outgoing packet. For each incoming packet, its BER is recorded (*pkt_ber*) and used to update the *ber_est* variable according to the following rules.

- Unicast and broadcast packets are transmitted differently, and thus should be considered separately.

- Because the BER pattern is different for each packet type, information collected from different types of packets has a different weight in calculating *ber_est*.

- If there is no error bit in the incoming packet, *ber_est* will be slowly decreased.

- To improve the ADIM-NB/FE algorithm's reaction time, if the *pkt_ber* is relatively high, *ber_est* will be based solely on the most recent *pkt_ber*.

- If the packet BER is higher than the radio's maximum FEC capability, there is a high probability that the packet experiences collision, not corruption. The information collected from such a packet is discarded.

- The BER history of previous packet arrivals is equally important and is used in the *ber_est* calculation.

- When the FEC coding rate is too low, it is generally more efficient, in terms of throughput, to rely on the ARQ algorithm than to keep decreasing the coding rate. A high BER environment can be an indication of too many active users in the system. In such a case, it is generally better to retransmit the packet later than to make the transmission more robust by increasing the FEC overhead. Thus, the FEC coding rate is limited to a reasonable minimum value.

The exact algorithm to calculate the *ber_est* variable is shown in Figure 4-9, with constants defined in Table 4.3. The recommend FEC rate is calculated from the following equations.

rcmd_ber = EST_BER_BETA**ber_est.*

EST_BER_LW_BOUND < *rcmd_ber* < EST_BER_UP_BOUND.

Using the recommend FEC rate, the ADIM-NB/FE algorithm adjusts its local FEC rate to match this value. To prevent the FEC rate from changing too quickly, the local FEC rate is allowed to increase or decrease one step at a time. The default step size is 1/16.

```
If (packet is a broadcast packet) and (0 < pkt_ber < Maximum correctable BER)
    do
    ber_est = EST_BER_BCS_ALPHA *ber_est + (1- EST_BER_BCS_ALPHA)*pkt_ber
    end
If (packet is a CTS packet) and (0 < pkt_ber < Maximum correctable BER)
    do
    ber_est = EST_BER_CTS_ALPHA*ber_est + (1-EST_BER_CTS_ALPHA)*pkt_ber
    if (pkt_ber > ber_est*EST_BER_BETA)
        do
        ber_est = pkt_ber/EST_BER_BETA
        end
    end
If (packet is a CTS packet) and (there is no error)
    do
    ber_est = ber_est * EST_BER_DEC_RATE
    end
If (packet is a data packet) and (0 < pkt_ber < Maximum correctable BER)
    do
    ber_est = EST_BER_DAT_ALPHA*ber_est + (1-EST_BER_DAT_ALPHA)*pkt_ber
    if (pkt_ber > ber_est/EST_BER_DAT_ALPHA)
        do
        ber_est = pkt_ber*EST_BER_DAT_ALPHA
        end
    end
If (packet is a data packet) and (there is no error)
    do
    ber_est *= EST_BER_DEC_RATE;
    end
```

Figure 4-9. Pseudo-code for the adaptive FEC algorithm.

Each station communicates the local FEC rate to its neighbors, who store this information in their routing table. Because the amount of interference at different source/destination pairs can be different, the source station will use the FEC rate of the destination station when sending an RTS request, rather than using its local FEC rate. If an RTS packet is encoded with the local FEC rate, it can be lost when the interference level at the destination station is significantly higher than the interference level at the source. Furthermore, any subsequent RTS packets to that destination will always be lost. By using remote FEC information, which can change as the interference changes, the RTS packet has a higher chance of success while still keeping a low overhead in a low noise environment. The ADIM-NB/FE time diagram is illustrated in Figure 4-10. In comparison with a fixed FEC rate, the ADIM-NB/FE algorithm shows no significant reduction in performance in a low noise environment though it performs better in a noisy environment as will be demonstrated in Chapter 7.

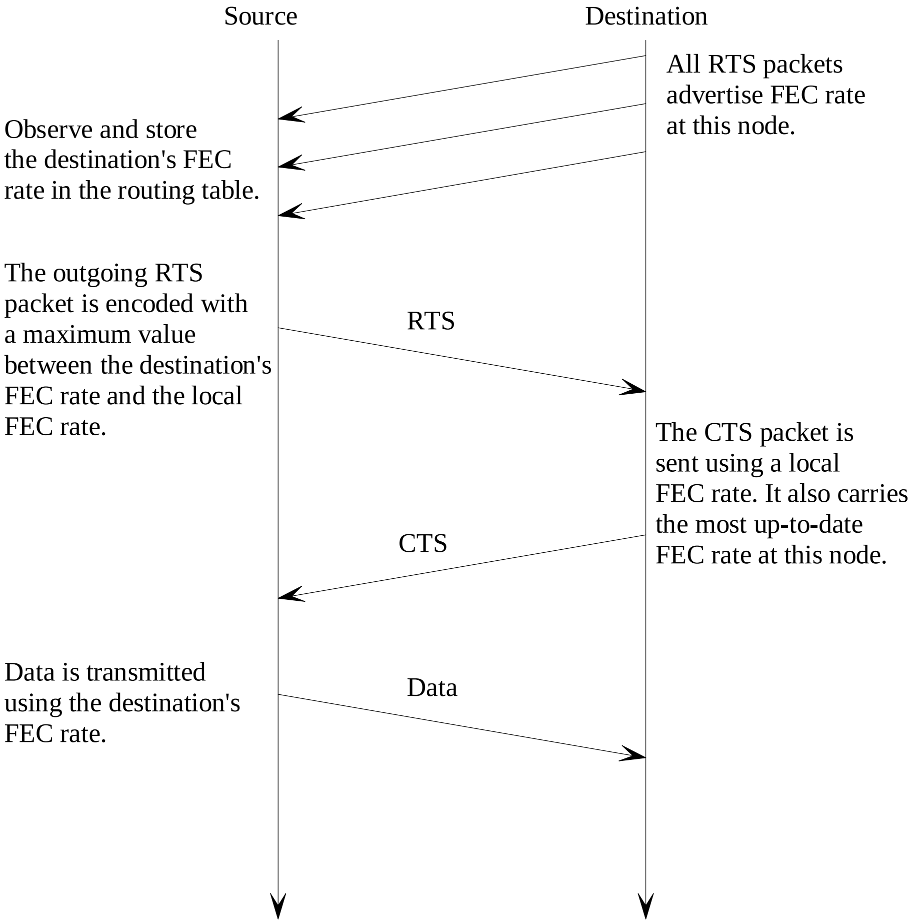


Figure 4-10. Time diagram for the ADIM-NB/FE algorithm.

An alternate algorithm is also studied in the event that determining the exact BER information for each packet is impractical. The alternate ADIM-NB/FE algorithm relies on two watermarks or thresholds to adjust its current FEC rate. If the BER in the incoming data packet is higher than the high watermark, it changes to a slower FEC rate, which offers more robust communications. If the BER in each of the last *n* consecutive data packets is lower than the low watermark, it changes to a higher FEC rate. The upper and lower limits of the FEC rate remain the same as in the previous algorithm.

Table 4.3 contains other parameter and constant settings. The maximum correctable BER correspond to a 3/4 FEC coding rate. Section 7.3 explains the reason of not using a lower FEC coding rate. Other constants are the optimal values obtained from informal trace and experimental results. In principal, the BER in a broadcast message tends to fluctuate the most, and thus, should be used cautiously. Because of the larger size of a data packet, its BER reflects the channel condition over a longer period of time. Consequently, it should play a more important role in estimating the estimated channel BER condition. The ADIM-NB/FE algorithm also uses a conservative approach when the incoming packet has no error by slowly decreases the estimated BER value.

Table 4.3. Constants Used in the ADIM-NB/FE Algorithm.

Symbol	**Default Value**	**Description**
Maximum correctable BER	0.05490	Corresponding to BCH code (255, 147)
EST_BER_BCS_ALPHA	0.98	Rate at which *ber_est* changes when broadcast packet arrives
EST_BER_CTS_ALPHA	0.95	Rate at which *ber_est* changes when CTS packet arrives
EST_BER_DAT_ALPHA	0.9	Rate at which *ber_est* changes when unicast packet arrives
EST_BER_BETA	1.5	Error margin for *ber_est*
EST_BER_DEC_RATE	0.985	Rate at which *ber_est* reduces when incoming packet has no error

4.5 Multi-User Receivers

The major source of interference in CDMA systems is MAI [CoM85], which results from other users sharing the same frequency spectrum. Multi-user interference rejection reduces this interference by estimating the waveforms generated by other users and subtracting them from the original signal. This joint demodulation of multiple users in the system reduces interference, decreases overall probability of error, and increases system capacity. An additional benefit of the demodulation process is that it allows multiple simultaneous receptions from multiple sources to provide a *multi-user receiver* service [Swa98]. Unfortunately, in a mobile *ad hoc* network, the receiving host cannot transmit during this period because of the half-duplex nature of the network [StT89]. To be able to operate in a full-duplex mode, stations must be categorized into two groups, e.g., uplink and downlink. Each group uses a different reception channel to which all stations in that group listen. The full-duplex capability is then available for an inter-group communication via the different reception channels. To enable any-station-to-any-station communications, as is the case with an *ad hoc* network, all stations must listen to the same channel, thus eliminating any possibility for a full-duplex transmission. This lack of a full-duplex capability makes coordination among participants critical to overall network performance. Nevertheless, full-duplex capability in an *ad hoc* network is still possible with such schemes as receiver directed or transmitter directed spreading code, but additional steps are required to discover neighbors’ spreading codes.

In the ADIM-NB/MU algorithm, each node can simultaneously receive data, acknowledgement messages, control messages, RTS messages, and other broadcast messages from its peers. The receiving node must negotiate connections so that all communicating neighbors will transmit at approximately the same time. Once the mobile host starts receiving from one of its neighbors, it cannot transmit, preventing future channel setup with other hosts.

The basic protocol operation flowcharts in Figures 4-4 to 4-7 need to be amended when considering the multi-user receiver algorithm. These updates are shown in Figures 4-11 through 4-14. The time diagram of an example connection setup is shown in Figure 4-15.

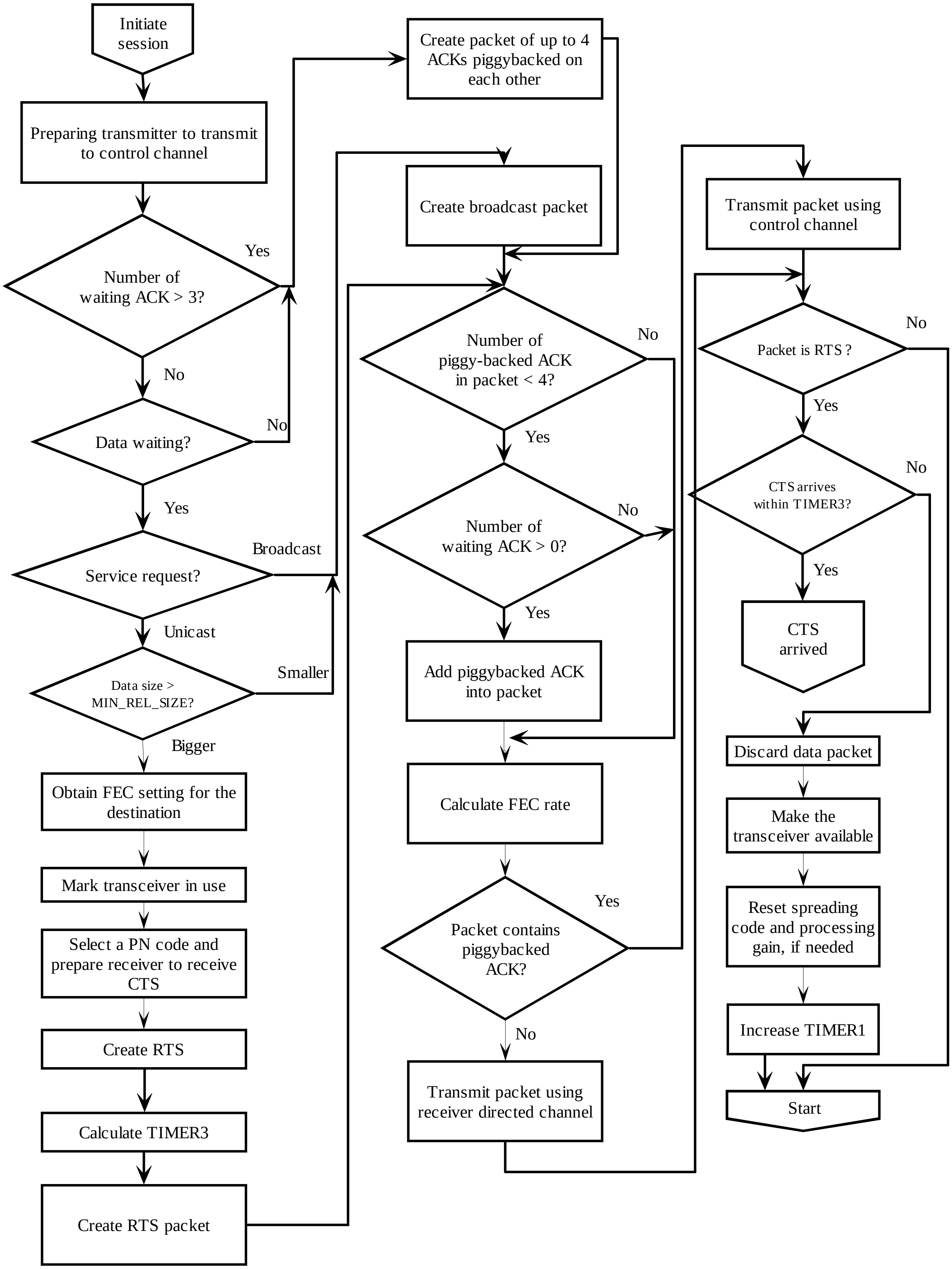


Figure 4-11. ADIM-NB/MU algorithm flowchart (1/4).

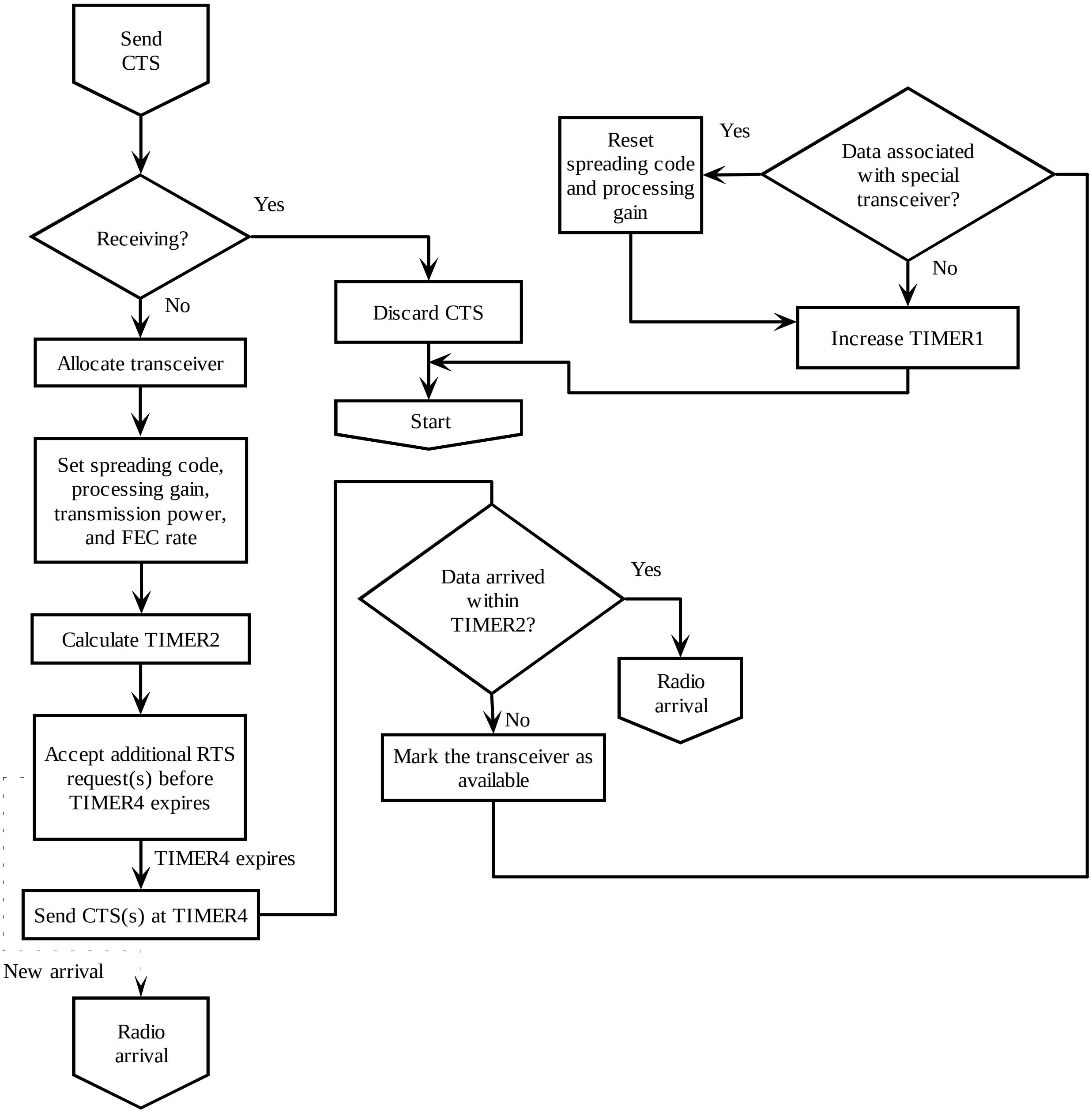


Figure 4-12. ADIM-NB/MU algorithm flowchart (2/4).

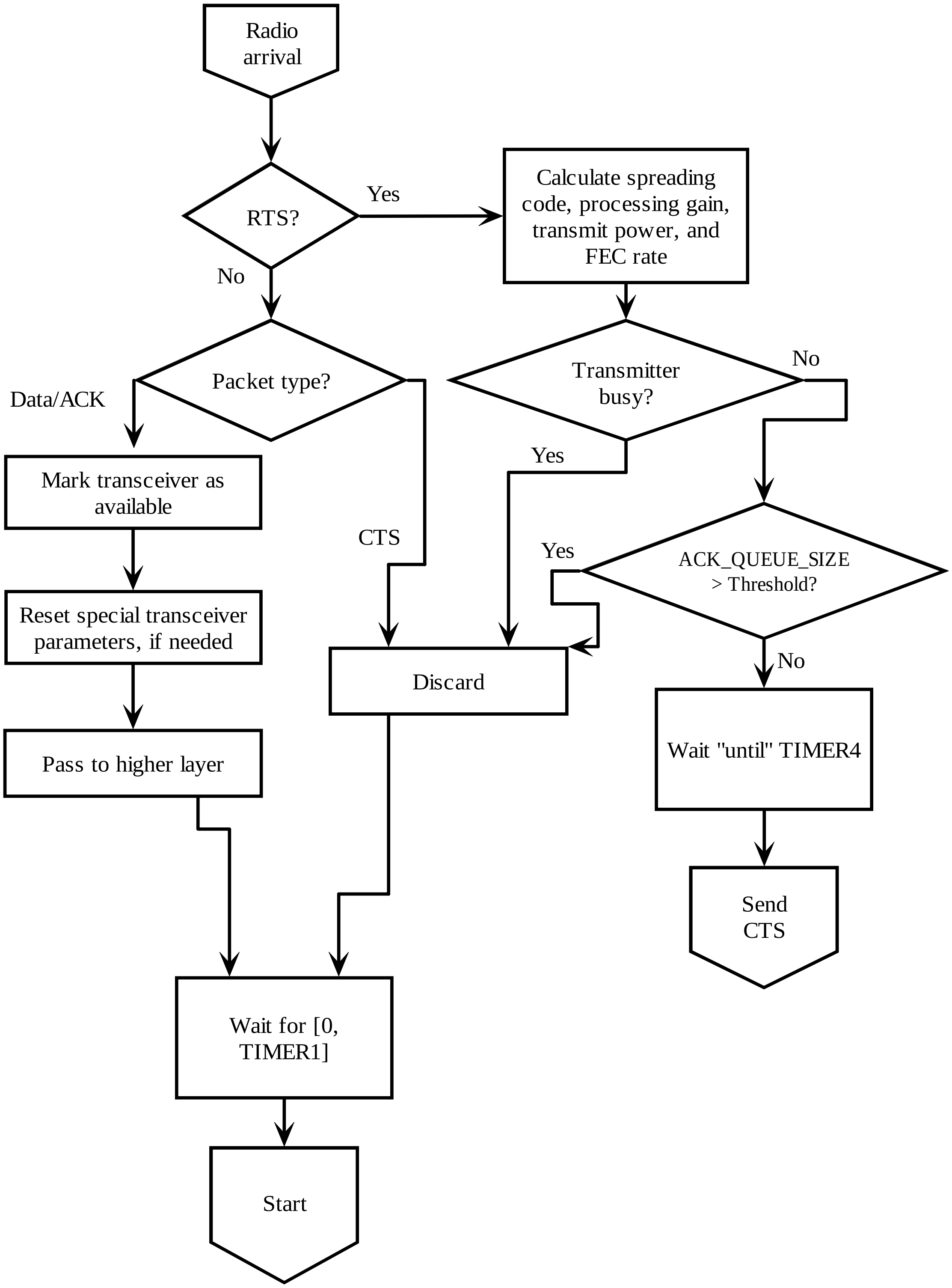


Figure 4-13. ADIM-NB/MU algorithm flowchart (3/4).

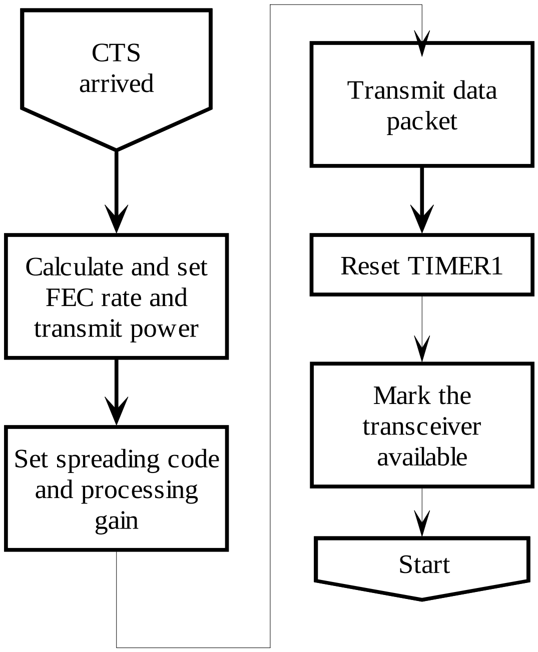


Figure 4-14. ADIM-NB/MU algorithm flowchart (4/4).

In Figure 4-15, both Source1 and Source2 attempt to transmit to Destination. Initially, the first transceivers at each node, S11, S21, and D1, are set to a common spreading code, called Default code or control channel. The last receivers at each node, S14, S24, and D4, are set to receiver-directed spreading codes. Each node is assigned a unique receiver-directed spreading code based on its address. Other transceivers, S12, S13, S22, S23, D2, and D3, are unused.

After sensing an idle control channel, Source1 transmits an RTS{S13->D4} packet to Destination. Because the RTS packet is intended for a single destination, i.e., Destination, Source1 uses a receiver directed spreading code to communicate with Destination. Knowing the address of its destination, Source1 calculates the spreading code of D4 and sets its idle transmitter, S13, to the same spreading code. Additionally, Source1 needs to allocate an idle transceiver, S12, to be used in the subsequent data session. The S12 receiver is set to a session-unique spreading code that is specified by the RTS{S13->D4} packet.

Source1 does not really need the control channel to be idle when it initiates transmission because the entire session does not involve the control channel. A busy control channel indicates broadcast traffic to in which all stations, including Source1, should listen. Therefore, Source1 should finish its reception on the control channel before attempting transmission.

Upon receiving the RTS{S13->D4}, Destination sets its second transceiver, D2, to the spreading code specified by the RTS{S13->D4} packet. D2 now has the same spreading code as S12. Destination, however, does not immediately reply with the CTS packet to Source1. It instead waits for a short duration for additional RTS requests. In this particular example, Source2 also sends RTS request to Destination within this waiting period.

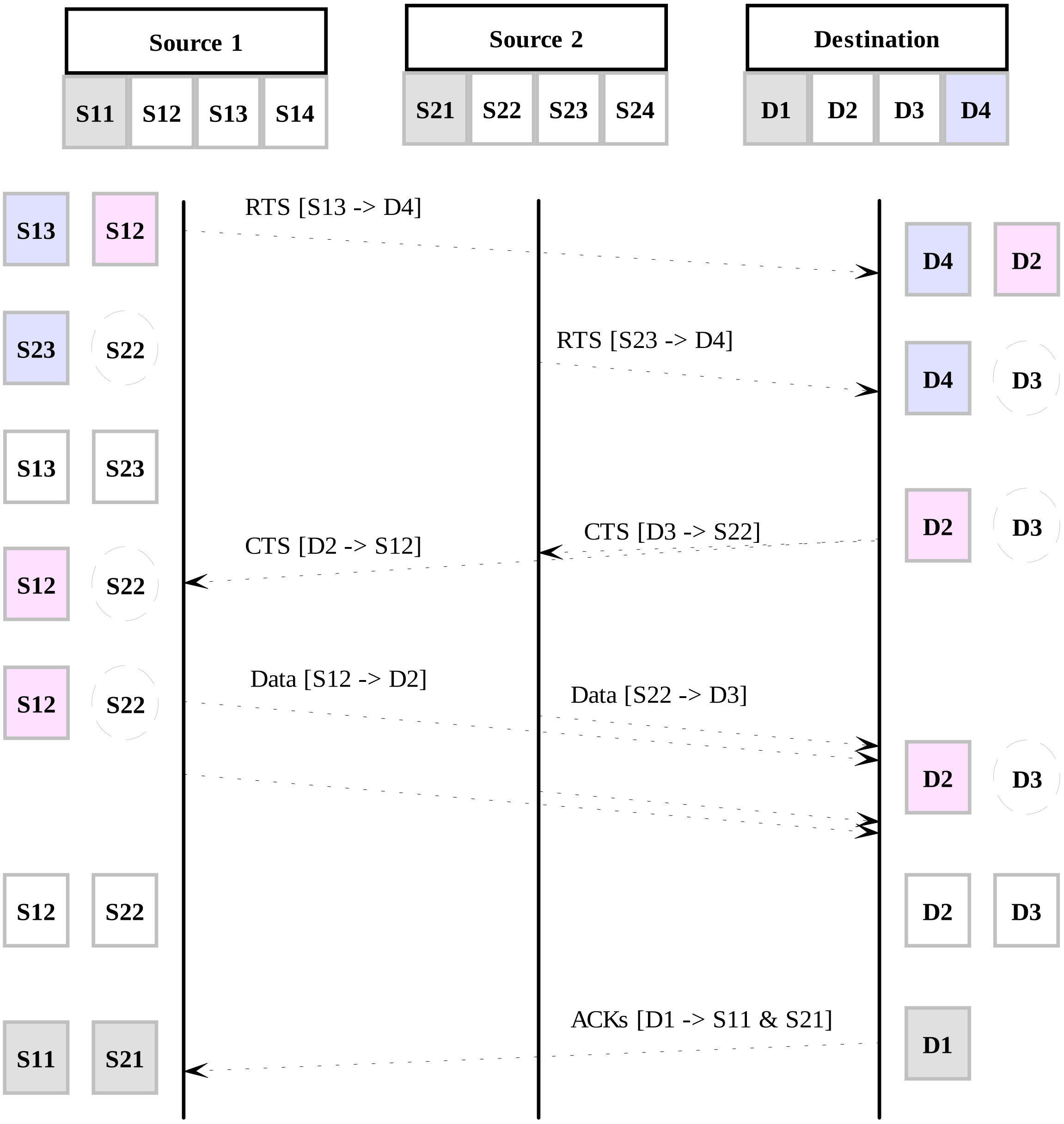


Figure 4-15. Time diagram of an example connection setup process.

The same procedure is repeated for RTS transmission from Source2 to Destination. After sensing an idle control channel, Source2 sends RTS{S23->D4} packet to Destination, using Destination's receiver directed spreading code. It also allocates an idle transceiver, S22, and sets that transceiver to the same spreading code specified in the RTS{S23->D4} packet. Upon receiving the RTS{S23->D4} packet, Destination sets its third transceiver, D3, to the information specified by the RTS{S23->D4} packet.

The probability that both RTS{S13->D4} and RTS{S23->D4} are transmitted relatively close to each other is high because both Source1 and Source2 tend to initiate the transmissions after the control channel becomes idle. A collision between the two RTS packets is also unlikely because of the carrier sensing technique employed. Therefore, the waiting period at Destination for the additional RTS can be relatively short. It is also important that this period is not too long; otherwise, Source1 will assume that the RTS{S13->D4} packet is lost.

After the waiting period is over (*TIMER4*), Destination calculates the appropriate communication parameters as described in the ADIM-NB/PG, ADIM-NB/FE, or ADIM-NB/PC algorithm, if appropriate. It then simultaneously transmits CTS{D2->S12} to Source1 using the allocated transmitter, D2, and CTS{D3->S22} to Source2 using the allocated transmitter, D3. Source1's receiver, S12, and Source2's receiver, S22, are already set to the same communication parameters in the previous step.

Upon receiving the CTS packets, both source stations make any necessary adjustments to the allocated transceivers, S12 and S22. Then, they immediately transmit their data packets, Data{S12->D2} and Data{S22->D3}, using the allocated transceivers to Destination. Consequently, all transmissions are loosely synchronized and the overall transmission duration is minimized, which also improves performance.

Destination receives these data packets using the previously allocated transceivers, D2 and D3, for their respective communication sessions. All communications from Source1 and Source2 occur independently and simultaneously. The multi-user receiver and interference rejection functionality of the radio reduces or eliminates any interference between Source1 and Source2. After the source stations finish their transmissions, Source1, Source2, and Destination reset all previously allocated transceivers, S12, S22, D2, and D3, to the default settings. During the transmission duration, Destination can still receive broadcast packets with receiver D1. It can also receive unicast packets with its receiver directed receiver, D4.

After finishing all receptions from all source stations, Destination needs to send acknowledgement messages to both Source1 and Source2. For efficiency purposes, both acknowledgement messages are combined into a single packet with multiple destination addresses. Because of the multiple destination addresses, the acknowledgement packet, ACKs{D1->S11&S21}, is transmitted using the control channel, or transmitter D1. Source1 and the first transmitters, S11 and S21, are already tuned to the control channel and can receive the acknowledgement packet without further negotiations.

4.5.1 Quality of Service Extension

Real-time support can be added to the ADIM-NB/MU algorithm to allow stations to send data at a regular interval without involving the channel setup process. This capacity reservation technique provides basic quality of service (QoS) support. However, it may provide limited improvement if the network topology is highly dynamic because the reserved path and capacity are more likely to be invalid. In addition, many applications needing QoS support require higher transmission speeds than currently supported by the system under study.

4.5.2 Multi-User Receiver Characteristics

The multi-user receiver used in this study is based on an ideal partial PIC multi-user receiver and hardware developed at Virginia Tech [Swa98]. The analytical bit-error performance of the multistage parallel-cancellation approach for user *k* at stage *s* for BPSK modulation is

$$P_k^{(s)}(E) = Q([\frac{1}{2E_b/N_o}(\frac{1-(\frac{K-1}{3N})^s}{1-\frac{K-1}{3N}}) + \frac{1}{(3N)^s}(\frac{(K-1)^s-(-1)^s}{K}.\frac{\sum_{j\neq k}P_j}{P_k}+(-1)^s)]^{-1/2})$$

where P_j is the received signal power of the k^{th} user and E_b/N_o is the received signal to noise ratio [CBW99]. The Virginia Tech radio hardware can support up to four simultaneous users ($K = 4$, $k \leq K$) and has two stages ($s = 2$) and three possible processing gains ($N = 4$, 8, or 16).

4.5.3 Interoperability with ADIM-NB/PC

In addition to multiple simultaneous receptions, it is also possible to configure ADIM-NB/MU to allow multiple simultaneous transmissions, from a single source station to multiple destination stations. In such a situation, the ADIM-NB/PC algorithm needs to be amended in order to inter-operate with the ADIM-NB/MU algorithm. In particular, if multiple simultaneous transmissions from a single source station are taking place, then, to limit self-interference, the minimum transmit power for all transmissions is limited to half of the maximum transmit power from the same station.

4.5.4 Parameters

This section describes three variables that have different meaning or value when operating with ADIM-NB/MU.

4.5.4.1 TIMER2

TIMER2 is the duration in which the ADIM-NB algorithm should wait for the data packet after transmitting a CTS packet. If the data packet does not arrive within this duration, the transceiver is reset to its default parameters and is made available for other packet transmissions. With the ADIM-NB/MU algorithm, however, there are many transceivers to spare; thus, it is not crucial to quickly reclaim the previously allocated transceivers. Therefore, when *TIMER2* expires, the allocated transceivers are marked unused but their communication settings are left untouched, allowing the receivers to keep waiting for any delayed data packets. As a result, data packets can still arrive at the transceiver after the transceiver has been marked available. *TIMER2* is calculated using the equation

TIMER2 = RTT_RTS_CTS + Transmission delay + Propagation delay.

4.5.4.2 TIMER3

TIMER3 is the maximum duration for which the protocol will wait between transmitting an RTS request and waiting for a CTS reply. If the CTS reply does not arrive within *TIMER3*, the allocated transceiver is reclaimed and the data packet is dropped. Since the ADIM-NB/MU algorithm does not immediately transmit a CTS packet when it receives a RTS packet, the RTS/CTS handshake sequence takes longer than without the ADIM-NB/MU algorithm. As a result, *TIMER3* has a larger value when used with the ADIM-NB/MU algorithm. The CTS packet transmission can be delayed by as much as

$$((1.5 * \text{RTT\_RTS\_CTS}) + \text{CTS packet duration}) \le 2 * \text{RTT\_RTS\_CTS}$$

Therefore, *TIMER3* is calculated from the equation

$$TIMER3 = 3 * \text{RTT\_RTS\_CTS}.$$

4.5.4.3 TIMER4

Upon receiving the first RTS request, the ADIM-NB/MU algorithm schedules a CTS packet transmission at *TIMER4* seconds later, rather than immediately transmitting the CTS packet, to give other RTS requests the opportunity to be received and processed. If additional RTS requests arrive, the ADIM-NB/MU algorithm schedules the corresponding CTS packet transmissions at the same time as the first scheduled CTS packet transmission. *TIMER4* is zero if the ADIM-NB/MU algorithm is not used. Because there are three allocable transceivers, *TIMER4* is calculated to allow two additional RTS packet receptions using the equation

$$TIMER4 = 1.5 * \text{RTT\_RTS\_CTS}.$$

4.6 Impacts from Each Adaptive Algorithm

Each ADIM-NB algorithm improves a different aspect of the system. The ADIM-NB/PG algorithm improves the network throughput, while the ADIM-NB/FE algorithm allows the ADIM-NB protocol to operate in a wide range of environments. The ADIM-NB/PC algorithm is primarily concerned with reducing the power consumption, and the ADIM-NB/MU algorithm slightly reduces the packet end-to-end delay. Table 4.4 provides a quick summary of the improvement from each ADIM-NB algorithm in an interference-free, static network with SMTP traffic [Pax94]. The complete results and analysis will be discussed in Chapter 7.

4.7 Summary

This chapter addressed existing problems and deficiencies in current CDMA and CSMA MAC protocols. The opportunistic ADIM-NB MAC protocol is designed to adapt its functionality and operating parameters to the current environment. It also takes advantage of a radio with multi-user receiver capability. It has four distinct features. First, an adaptive data rate selection algorithm (ADIM-NB/PG) intelligently selects appropriate data rate and processing gain based on available information. Secondly, an algorithm to utilize the multi-user receiver

(ADIM-NB/MU) by enabling multiple simultaneous connections among multiple neighbors is developed. The third feature (ADIM-NB/FE) dynamically changes the FEC code rate based on the BER of the incoming packets. Finally, a power control algorithm (ADIM-NB/PC) attempts to equate the received power from various sources. All adaptive algorithms are relatively independent of each other and can be enabled/disabled separately.

Table 4.4. Improvements from Each ADIM-NB Algorithm in a Static Network.

Percentage of Contribution	**End-to-End Network Throughput**	**Power Used per Bit**	**Delay**
ADIM-NB/ PG	96.71%	78.96%	83.58%
ADIM-NB/ PC	0.06%	13.39%	0.13%
ADIM-NB/ FE	0.92%	0.93%	1.47%
ADIM-NB/ MU	1.83%	0.94%	4.70%
2nd order interaction	0.21%	5.70%	9.09%
3rd order interaction	0.18%	0.05%	0.02%
4th order interaction	0.01%	0.00%	0.02%
Error	0.08%	0.03%	0.99%

A preview of improvements from each adaptive algorithm, individually and jointly, in the presence of other adaptive algorithms was briefly mentioned in Section 4.6.

Chapter 5. CAMEN Routing Protocol Algorithm Definitions

This chapter describes CAMEN, which is based on the WRP [GaM97]. CAMEN is designed to take advantage of the ADIM-NB MAC protocol and to improve the network scalability. It also reduces the routing protocol's dependency on radio broadcast capability, periodic hello messages, and route discovery messages. A routing protocol requires a complicated algorithm, primarily because it needs to perform a lot of functions in a distributed manner. Therefore, this chapter focuses more on the concepts and operations of the protocol in a normal situation, rather than specific environment conditions or exceptional situations. Examples and flow charts are provided throughout the chapter to help the reader visualize the algorithm. Some protocol details are difficult to describe using a time diagram or flowchart. In such cases, pseudo-code is used.

Section 5.1 describes the problem areas addressed by CAMEN and other high-level concepts. Section 5.2 describes its operation in a typical situation. Section 5.3 explains protocol descriptions and specifications in more detail.

5.1 Problem Areas Addressed by CAMEN

One of the biggest drawbacks in many *ad hoc* routing protocols is their flat structure. These non-hierarchical routing protocols assume that all participating mobile hosts are routers, which poses several significant problems as follows.

- The *ad hoc* routing protocols are primarily used by wireless handheld radios, where battery life is one of the biggest concerns. Not all radios are willing to let their batteries drain so that other nodes can route their packets or the network can stay connected, especially when other nodes are better qualified for the tasks.

- Every node must participate in the routing process in order for the routing algorithm to operate correctly. A single uncooperative node could cause a significant problem to the network as a whole.

- By running the same algorithm on all nodes, the whole network is limited by its weakest node. If more work can be allocated to a more powerful node, i.e., a node with longer battery life, larger memory, or higher processing power, the overall network performance can be improved. For example, if a router's battery is low, it simply reduces its priority so that a better suited node can take over the router functions, resulting in a longer connected network.

- If every node is a router, each node has to constantly communicate with all other nodes, which results in a large amount of routing information being exchanged indefinitely.

- For a proactive routing protocol [Haa97], one insignificant change in the network topology will cause ripples of routing update messages to be propagated across the whole network.

- For a reactive routing protocol, if the pre-computed route contains a highly dynamic node, it most likely will have a short life span and will require frequent route discovery operations. This introduces an unnecessary instability to the network. By reducing the dependency on highly dynamic nodes, network stability can be improved.

These problems can be addressed by introducing a hierarchical structure to the *ad hoc* routing protocol. In CAMEN, mobile nodes are divided into two groups: regular nodes and routing nodes. Nodes can dynamically switch between groups based on their current battery capacity and the network condition. The routers dynamically create backbone routes across the network and maintain those routes by periodically advertising routing information and sending update messages. A regular node does not participate in the routing process nor does it periodically broadcast its connectivity information. It also does not maintain a routing table but simply forwards all packets to the designated router. Routers are usually high performance nodes with low mobility. If such a node does not exist, a regular node will switch to the router mode. This is analogous to the automatic selection of base stations in a traditional cellular network.

By asking the mobile node to be a router only when necessary, the number of routers, as well as the amount of routing information, can be greatly reduced. Only a small subset of nodes must exchange information and update the topology. In addition, only the route to the destination's nearest router is adequate, which enables several nodes to share the same route entry in the routing table.

It is envisioned that a homogeneous operating environment is rare. Some of the mobile units usually have a larger power capacity than other units. In a military network, there are combinations of vehicular-mounted, satellite-based, plane-mounted, and handheld units. In a commercial network, the service provider benefits by placing several high performance units strategically across the network to improve the overall performance. In such environments, a routing protocol based on a heterogeneous assumption is more suitable.

Additional enhancements include secondary route provision when the primary route fails, detection of a redundant cluster, delayed route removal, and quick response to set up a replacement route.

5.2 CAMEN Operations

Traditional *ad hoc* routing protocols cannot fully exploit the many new behaviors and extra features, e.g., broadcast, unreliable unicast, and reliable unicast, that are introduced by ADIM-NB. Additionally, many *ad hoc* routing algorithms heavily utilize or depend upon the broadcast capability of the radio, which is either unavailable or inefficient for a unicast-based CDMA MAC protocol. For these reasons, WRP [MuG96] is augmented with CAMEN and integrated with the ADIM-NB protocol. Although WRP is based on a distance vector routing algorithm, CAMEN should be applicable to any MANET routing protocol that proactively maintains its routing table. Some features in CAMEN should also be applicable to reactive MANET routing protocols as well. CAMEN is designed to improve network scalability as well as to reduce reliance on radio broadcast capability. In particular, CAMEN reduces reliance on periodic HELLO messages and does not flood the network with route discovery messages.

The integration of WRP, CAMEN, and ADIM-NB allows both MAC and routing protocols to be jointly developed to further reduce the number of broadcast messages. ADIM-NB communicates a significant number of operating parameters to CAMEN and vice versa. This information includes SNR, BER, collision rate, error correction coding rate at other nodes, etc.

CAMEN is based on a clustering concept. Several nodes within each other’s transmission range form a cluster. One node in each cluster is elected to be the cluster head. The cluster head performs minimal administrative tasks and routes packets across the network. All nodes in the same cluster share a single route entry in the routing table, as explained below. This route-sharing technique results in a smaller routing table and improves scalability because the routing table consists only of routes to the nearest cluster head. The size of the routing table depends mainly on the number of clusters, which, in turn, depends on the network coverage area. Therefore, the maximum network size is limited by a geographic region, which can more easily be anticipated than the number of nodes in the network. An additional advantage of this clustering approach is that it provides a built-in mechanism to integrate the mobile wireless *ad hoc* portion of the network with the existing infrastructure-based network by forcing the wired stations to become cluster heads. Because a regular node does not periodically broadcast HELLO messages, exchange routing information, or maintain a routing table, its power can be used more effectively, resulting in longer battery life.

The hierarchical *ad hoc* routing protocol CAMEN divides mobile nodes into two groups, regular nodes and routers. A regular node does not participate in the routing process or advertise connectivity information but merely forwards all packets to its designated router. A router is usually a high performance node with low mobility, or simply a node with a larger battery. If such a node does not exist, a regular node can be elected to be a router. Routers include cluster heads and gateways, the intermediate nodes that link two adjacent cluster heads. The routers create backbone routes across the network, but only the cluster heads maintain these routes by periodically advertising routing update information based on WRP.

By asking a mobile host to be a router only when necessary, the number of routers as well as the amount of routing information, can be greatly reduced. Only the routers, a small subset of the overall nodes, need to update their topology information. In addition, only a route to the destination’s nearest router is necessary, enabling several nodes to share the same route entry in the routing table.

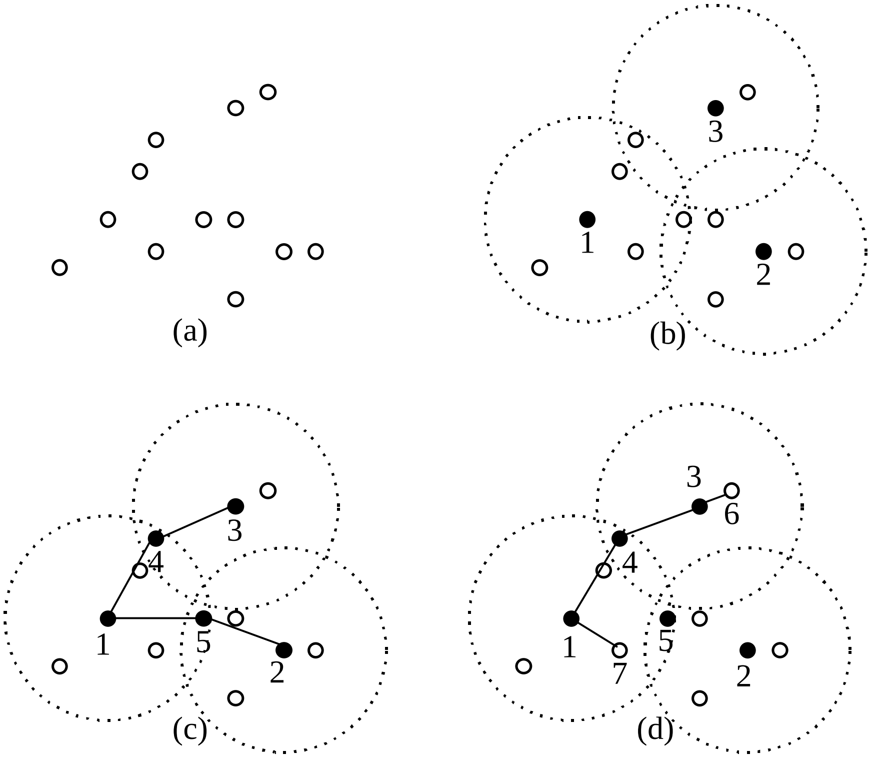
Although CAMEN introduces a hierarchical structure to the *ad hoc* network, it still uses a flat address space. Each node must be associated with a cluster head, which is responsible for advertising its members’ addresses in the route update messages. Using member information advertised by the cluster head, a complete path to any destination can be constructed without requiring a hierarchical address and without requiring the protocol to advertise a full route entry for each destination. The benefit of introducing a hierarchical structure to the network comes primarily from the reduced number of advertising nodes, rather than the reduction in the size of the route update messages. In a sense, it resembles a tree structure where branches consist of routers and leafs consist of regular nodes.

Figure 5-1 shows an example of the cluster formation and packet routing procedures. In Figure 5-1 (b), the cluster head election algorithm uses a prioritized timeout to separate a high priority node from a low priority node. At initialization, a mobile host calculates the maximum search duration to locate an existing cluster head. This value is a random variable based on the node's priority. A high priority node waits only for a short duration before declaring itself a cluster head, which makes it more likely to actually be a cluster head. After all cluster heads are identified, a backbone route is established (Figure 5-1 (c)). Node 4, using member lists of cluster 1 and cluster 3, knows that it is the highest priority node (in this case, the only node) linking the two clusters together. It then changes its status to become a gateway and starts relaying packets between cluster head 1 and cluster head 3. Node 4, however, does not generate any routing information by itself. Figure 5-1 (d) shows one example route. Node 6, knowing that node 7 is not nearby, forwards a packet to its cluster head, node 3, which uses the network backbone to route that packet to the destination cluster head, node 1, which then delivers the packet to its destination, node 7. Although limiting the packet forwarding operation to the network backbone may not yield the shortest path, it is not necessarily less effective because nodes within the network backbone will likely be high performance nodes. These nodes have more power, leading to higher speed and longer battery life, both of which are more beneficial to the overall system than providing the shortest route.

The most appropriate cluster-forming algorithm depends on several factors, including network density, network mobility, and application traffic characteristics. The cluster-forming algorithm described previously is best suited for a uniformly distributed network with moderate to high node density. Different clustering techniques may be desirable for other network environments or radio hardware, but they are not the focus of this research. Several other clustering algorithms have been proposed in the literature. The hierarchically-organized, multi-hop mobile wireless network (MMWN) keeps the number of members in each cluster relatively constant by adjusting the cluster radius [RaS98]. It also allows multiple levels of hierarchy to be used. The CBRP uses a clustering scheme similar to the one described here but employs a different election algorithm and requires some additional processing by gateways and non-routing nodes [JLT99]. The spine routing protocol describes an association between routing nodes and non-routing nodes, but does not favor any particular clustering algorithm [SDB98].

5.3 Routing Protocol Descriptions and Specifications

This section describes CAMEN operation and specification in more detail, including its operation in a normal situation and some extreme situations.



(a) Initial configuration.

(b) Nodes 1, 2, and 3 declare themselves to be cluster heads via prioritized timeout. Other nodes send registration messages to their associated cluster heads to register themselves as members of those clusters.

(c) Nodes 4 and 5 automatically identify themselves as gateways that link two clusters together. Cluster heads periodically broadcast their connectivity information while gateways passively store and forward this information to other cluster heads. The network backbone is formed.

(d) Example of route from node 6 to node 7. Only nodes 1, 3, and 4 in the route have a routing table.

Figure 5-1. Cluster formation and packet forwarding procedures.

5.3.1 Data Structure Maintained

The routing protocol (CAMEN and WRP) maintains several tables and lists to keep track of its neighbors and routes. Table 5.1 summarizes each table and its purpose. Not all tables are maintained at all nodes.

Table 5.1. Data Structure Maintained by the Routing Protocol.

Table	Contents	Description
Cluster member	Address, distance (~ signal strength), and time of last activity	Cluster head maintained information to keep track of its members
Link cost	Address, cost, and time of last activity	Cluster head maintained information to keep track of its neighbor clusters; necessary to send a reliable routing update message to all its neighbors
Route update message	Cluster head, members, hop counts to cluster head, and predecessor to that cluster head	Cluster head maintained information to keep track of changes in the routing table that need to be propagated to other clusters
Node unreachable	Address	Cluster head maintained information to advertise nodes that are no longer reachable from this cluster head; separating this information from route update message makes it more efficient to send the information
ACKs to be replied	Address and sequence number	Cluster head maintained information regarding acknowledgement messages to the routing update messages that are yet to be replied to nearby cluster head
ACKs not yet received	Address and sequence number	Cluster head maintained information regarding acknowledgement messages that are expected to be received from nearby clusters
Member of nearby clusters	Address	Used by cluster head to remove redundant cluster; also used by regular node to determine its gateway status
Subscribed cluster	Address and time of last activity	Gateway maintained information to unsubscribe itself from clusters that are no longer reachable; unsubscribe is crucial to keep the information at each cluster head up-to-date
Routing table	Address, hop counts, predecessor, successor, node type, back up route, creation time, time of last activity, recommended FEC rate, amount of bit forwarded, and removal status	Routing table for all destinations in the network (routers) or local destinations (non-routers)

5.3.2 Initialization

The initialization procedure can be executed when nodes are just starting up or when they did not hear from any cluster heads for a specific duration. Each node calculates a timeout based on its priority. A higher priority node waits for a shorter period while a lower priority node waits for a longer duration. If it does not hear from any cluster head for the timer duration, the node will switch itself to a cluster head mode and start advertising its availability to other nodes.

After initialization, a regular node still keeps a countdown counter to keep track of its cluster head. If it does not hear from any cluster heads for a particular time period, it will re-initiate the initialization procedure and switch itself to a cluster head mode.

The cluster head periodically advertises its cluster information, as well as other routing information, to other nodes. There is at least one advertisement in every update interval. If there are significant changes in the connectivity, the cluster head will advertise this information more frequently. Using cluster information from neighboring clusters, the cluster head can demote itself to a regular node if it discovers that all of its members are also members of another cluster. After demoting itself, the initialization procedure will be re-initiated.

5.3.3 Updating the Routing Table

Figure 5-2 shows a flowchart of a simplified route update procedure. Some of the inconsistency checking and routing table validation routines are not shown to keep the picture simple. Each node executes this algorithm once for every destination that is contained in the route update messages (as well as when it internally updates its routing table).

Using the path finding algorithm, a node can determine if, after being updated, the path to this destination will contain itself as an intermediate hop. Such action would result in a loop and, therefore, should be avoided. The new route will be recorded into the routing table only if it has a shorter distance than the old route, which means that there exists a better path to this destination. The exception to this rule is when the source of this update message is an intermediate node in the old path. Since that intermediate node is closer to the destination, it would have a better knowledge about this particular route. In that case, it means that there are some changes in the old route that need to be updated, even if it results in a longer path. Because each cluster head always retransmits the most recent route update message and the message always takes the same path, the old information will never reach its neighbors after the new route update message has been delivered. Consequently, any route update messages that originated from the intermediate hop in the path can be considered as the most accurate information available. Part of the previous route will be stored in the routing table as a back-up route in case the primarily route fails. CAMEN will attempt to use this back-up route before removing this route from its routing table and advertising the unreachable information to other nodes. This approach helps avoid momentary changes in the network topology and reduce the routing traffic.

As the final step, any changes in the routing table are propagated to other nodes only by a cluster head. CAMEN makes it unnecessary to communicate any changes that occur in gateways or regular nodes.

5.3.4 Routing Decision

Depending on the information in each packet and the state of the node, CAMEN makes a different routing decision for each packet. This decision includes where to forward the packet, if it should look up the routing table, and if the information in the routing table should be updated. Such a process is represented in flowchart format in Figure 5-3, which shows a simplified routing decision in each node. Some of the consistency checking and the node-specific validation algorithm are not shown to keep the picture simple. Each node executes this algorithm once for every arriving packet, including those generated internally.

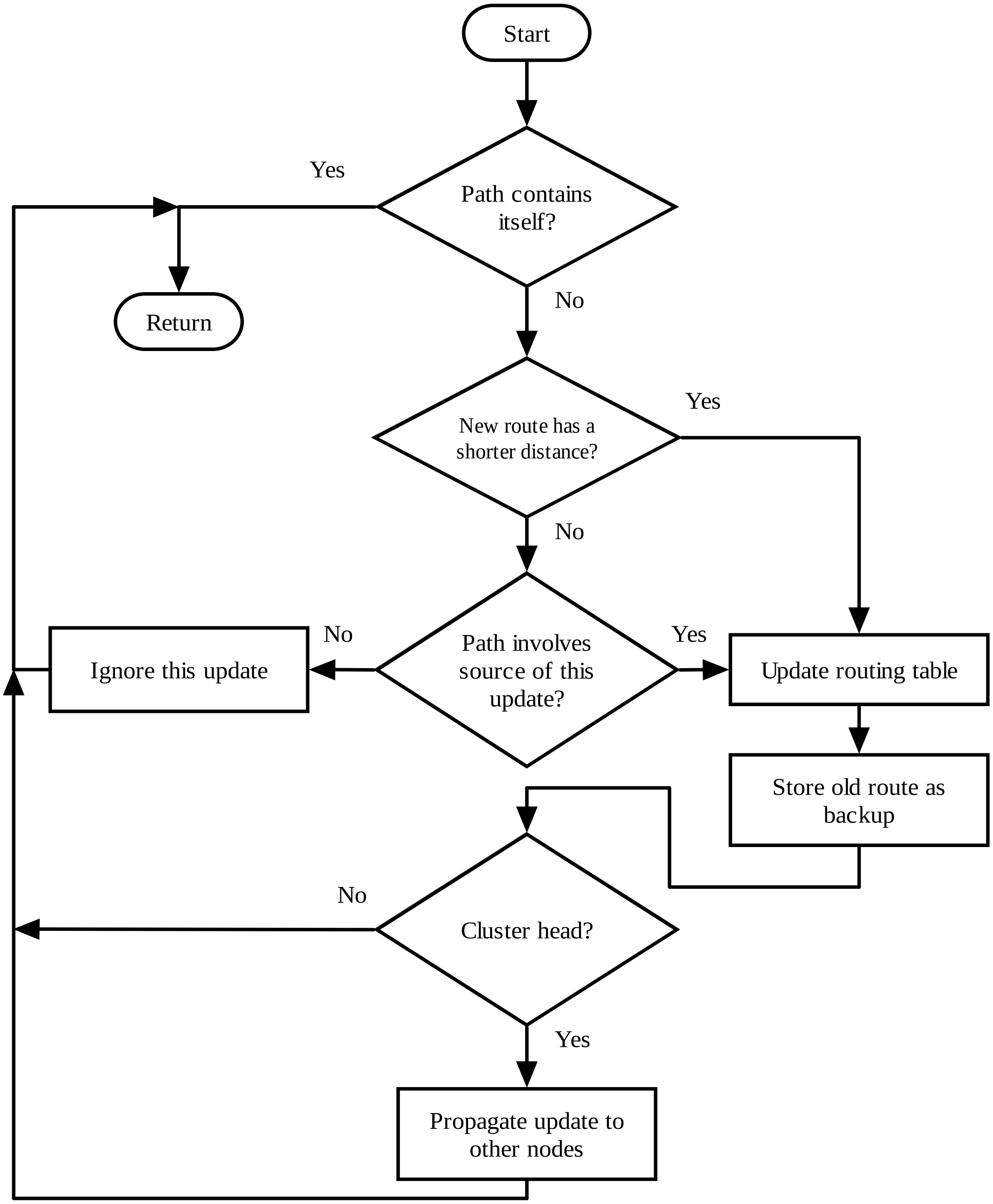


Figure 5-2. Route update algorithm.

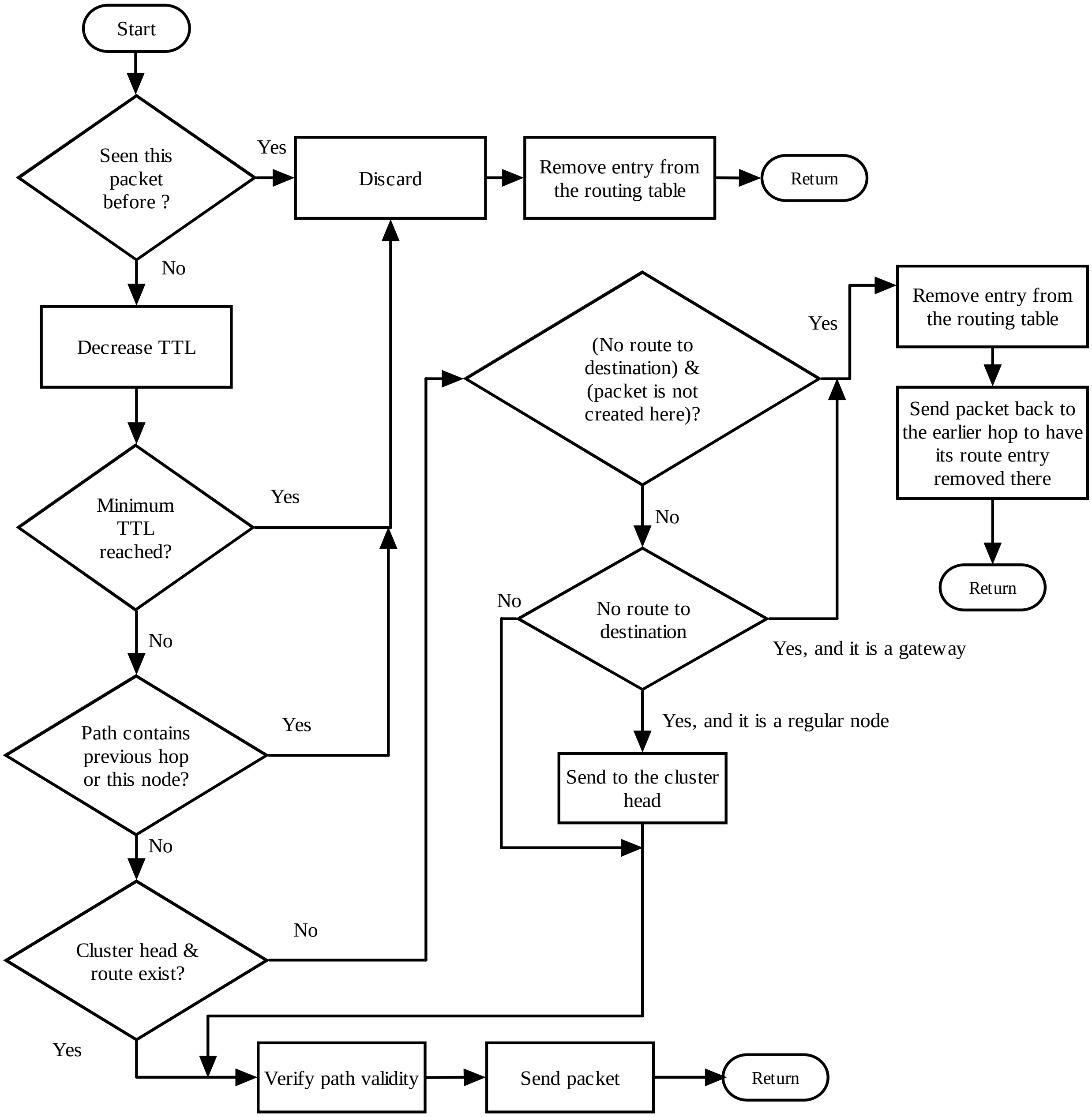


Figure 5-3. Packet forwarding algorithm.

Each packet that visits this node is recorded into a local database to detect a loop in the routing table. If CAMEN sees the same packet twice, it will simply drop that packet. Because this node previously forwards this particular packet, its routing table may contain invalid routing information and, thus, should be deleted. The deletion of route information does not have a large impact on the routing protocol because the neighboring nodes will quickly respond with their most up-to-date information regarding that deleted route. Using information from its neighbors, the node can determine the most effective route to replace the deleted route. Similarly, a packet with an expired time-to-live (TTL) may also indicate a loop in the routing table and should be discarded. The corresponding route entry in the routing table should also be deleted.

After looking up the routing table, the path finding algorithm will return a complete path along the route to the destination node. If the previous hop, which is the node that forwards this packet here, or the current node itself exists in the path to the destination, further forwarding of this packet will result in a loop. As with all earlier situations, this packet should be discarded and the corresponding route entry should be removed.

Although the non-routing node should forward all packets to its cluster head, except for those destined to known local destinations, it should not forward any packets that have not originated from itself because those packets should not have been forwarded to it. However, simply dropping this packet will not prevent this situation from happening again. To make the matter more difficult, a non-routing node is not allowed to originate or propagate any routing information to any other node. To solve this problem, the non-routing node has to reduce the packet's TTL to zero and forward it back to the previous hop. Receiving a packet with an expired TTL will cause the previous hop to drop the packet and remove the corresponding route entry from the routing table, thus preventing this situation from happening again. The same technique also applies to gateways because they are not allowed to originate any routing information. For a valid route, CAMEN reduces the packet's TTL and forwards it to the next hop in the routing table. It also records the packet size and the current time in the routing table to keep track of utilization and activity of its link to that neighbor.

5.3.5 Mobility

One thing is certain in an *ad hoc* network environment: mobility. Even in a static network, the nature of a wireless link can cause the link to be unstable. Any congestion over a single link can also make that link unresponsive, resulting in its removal from the routing table. Consequently, CAMEN must react appropriately to such changes in the network topology. Each node maintains several timers to aid it in removing idle or broken routes, which can also be placed in a hold mode before removal from the routing table if it is advantageous to do so.

5.3.5.1 Route Removal

There are many reasons to remove a route from the routing table. This section explains major events that cause route removal. In general, invalid or expired routes will be place in a hold state before being removed to avoid unnecessary changes in the routing table. Holding the route prevents it from being used, while the routing protocol tries to search for a replacement route. It also allows some information from that route to be used by the routing protocol. Nevertheless, there are also some situations in which the route should be immediately removed, without being put into a hold state.

Upon initialization, each node must register itself with the cluster head. Each registration is valid only for a certain duration. If the member node does not re-register itself before the expiration time, it will be removed from the cluster, as well as from the routing table. In addition, it is important for a gateway to de-register itself from the cluster head once it is out of range, so that the cluster head will quickly select another station to assume the functionality.

If the transmitting node does not receive a positive acknowledgement from its receiver within a certain time duration, the route to that neighbor will be put on hold. If there are no transmissions from that node during the hold period, that route will be removed from the routing table permanently.

The route entry in the routing table will be immediately removed if there are inconsistencies in the routing table or if there is a loop in the routing table. In some occasions, the secondary route will be used instead of the primary route when the primary route fails. The use of a secondary route usually applies to a destination two hops away.

5.3.5.2 Fast Response to Route Removal

The cluster head broadcasts a “node unreachable” message when a route is removed from its routing table. If such removal does not result in the same route being removed from the neighbor cluster head, the neighbor cluster head will reply back to this node with its route entry. Using such information, a new path to the destination is quickly added back to the routing table.

5.3.6 Difficulties

CAMEN introduces some difficulties because of its limitation in propagation of routing information. Because both regular nodes and gateways are not allowed to generate any routing information, they cannot propagate to their neighbors any invalid routing information that they discover. To work around that problem, those nodes have to bounce packets around, as described in Section 5.3.4, which also introduces some inefficiency.

Having a gateway automatically identify itself can also lead to no gateway at all if the cluster head does not actively remove previous cluster members from its cluster list. In addition, since two gateways cannot directly communicate with each other, they cannot create a direct link between themselves. Such a restriction causes the packet to traverse a longer path than necessary.

5.3.7 Interactions with MAC Protocol

CAMEN does not adapt itself to the operating environment as the ADIM-NB protocol does. CAMEN, however, maintains vital information for each neighboring node and passes that information to various ADIM-NB algorithms, which can adjust operating parameters on a per-neighbor basis. The ADIM-NB algorithm is also responsible for providing updated link-level information to CAMEN.

5.4 Summary

This chapter addressed existing problems and deficiencies in current flat routing protocols and introduced a hierarchical extension of the *ad hoc* routing protocol to address such problems. Several improvements to an existing wireless mobile *ad hoc* routing protocol were introduced. The proposed CAMEN algorithm reduces the system dependency on broadcast messages, allows operation in a large-scale environment, and increases overall battery life. It also exploits new features introduced by the ADIM-NB protocol and supplements the ADIM-NB protocol’s functionality at the network layer.

Chapter 6. Simulation

Generally, there are two approaches to studying the performance of protocols: analysis and simulation. Although it is possible to employ an analytical technique to study the performance of a MAC protocol with highly simplified parameters, a routing protocol is usually too complicated to be modeled by a set of equations. Because the interaction between MAC and routing protocols is critical to this research, using a simulation approach provides more capability, flexibility, and accuracy. The entire wireless MAC and routing protocols are modeled via simulation, including the ADIM-NB MAC protocol, the CAMEN routing protocol, and the underlying DSSS physical layer.

Chapter 6 describes the simulation environment, along with model verification and validation. Starting with a description of the simulation tool in Section 6.1, this chapter completely describes the simulation. Model information, including simulation parameters and factors, are defined in Section 6.2. Section 6.3 explains the simulation methodology employed, as well as transient analysis and terminating conditions. Sections 6.4 and 6.5 discuss model validation and verification, respectively.

6.1 Simulators

Selecting a proper language or tool for a simulation affects many aspects of the simulation and its results. Several simulation platforms already in use by the GloMo community [SRI98] were investigated for potential use in this study. These include the University of California at Berkeley’s Network Simulator-2 [Fal99], the University of California at Los Angeles’s MAISIE [BaW94], Rooftop/University of California at Santa Cruz’s C++ Protocol Toolkit (CPT) [Roo96], and OPNET Modeler®/Radio [MIL97]. OPNET Modeler®/Radio was chosen because of its availability, flexibility, extensibility, and extensive model libraries.

OPNET Modeler®/Radio is a commercial network simulator that has numerous built-in models and protocols available. It uses C to define a model behavior along with a graphical user interface (GUI) for higher level modeling tasks. By using C to define the behavior, it is relatively easy to make a transition from the simulation model to the actual device driver. The OPNET Modeler®/Radio has a large user base, including Virginia Tech. It also has native support for wireless links and node mobility with the capability to incorporate actual traffic and realistic channel models. In addition, it is based on an open architecture via C/C++, which makes it relatively easy to integrate the simulator with other components to provide added functionality.

OPNET Modeler®/Radio is also compatible with the SEAMLSS project [SAI99], which provides a simulation and emulation environment to test, demonstrate, and evaluate protocols, architectures, and algorithms for realistic usage scenarios. By integrating an OPNET® simulation model into the SEAMLSS project, the ADIM-NB and CAMEN protocols can be evaluated in a realistic terrestrial mobile *ad hoc* network environment with actual application traffic. In addition, the protocols can be compared against other protocols or evaluated using other types of radio hardware.

6.1.1 Simulation Platform

OPNET Modeler®/Radio runs on several computer platforms, including Intel® with Windows NT®, Sun®, and HP®[2]. This research makes extensive use of computer resources. The simulations are run on several Sun® and Windows NT® workstations. Simulation models and results are equivalent regardless of the computing platform.

6.2 Simulation Models

This section describes model-specific information. It focuses primarily on the input/output to the models without including too much about the inner working of the models. The discussion starts with model parameters, which can be considered as inputs to the simulation. The outputs from the simulation experiments, or performance metrics, are then described.

To ensure compatibility with other researchers in the GloMo community, all models are based on the GloMo radio API [BeL97]. After the models were completed and operational, they were integrated into the SEAMLSS-Lite environment [SAI99], which provided realistic application-level traffic and node movement patterns.

6.2.1 Model Parameters or Factors

The simulation models have several parameters that can be varied from simulation to simulation to study the system performance in different types of environments, as specified in this section. These model parameters are broken down into three groups: general protocol-specific parameters, environment conditions, and traffic characteristics.

6.2.1.1 Protocol Parameters

Data Speed of the Control Channel. The control channel data speed can be changed to simulate different types of radios. Two data speeds are of particular interest: 64,000 bits per second for the

---

[2] Intel® is a registered trademark of Intel Corporation. Windows NT® is a registered trademark of Microsoft Corporation. Sun® is a registered trademark of Sun Microsystems, Inc. HP® is a registered trademark of Hewlett-Packard Company.

Virginia Tech reconfigurable software radio [Swa98], and 230,400 bits per second for the CDMA2000 radio with radio configuration 6 [ITU99].

Processing Gain of the Control Channel. The control channel uses a fixed processing gain, which can be specified using this parameter. In combination with the control channel data speed, both parameters determine the chip rate of the radio.

Dynamic FEC Operation Mode. There are several operational modes for the ADIM-NB/FE algorithm as shown in Table 6.1. By setting this parameter to different values, the ADIM-NB/FE algorithm changes its behavior accordingly.

Table 6.1. ADIM-NB/FE Operation Modes.

Value	Behavior
0	No dynamic adaptation of the FEC coding rate
1	Threshold mode for radios that cannot determine the exact bit error information of the incoming packet
2	Basic estimate mode, the ADIM-NB/FE algorithm will quickly increase its FEC coding rate if there is no error in the incoming data packet
3	Exclusive estimate mode, the ADIM-NB/FE algorithm will ignore bit error information if the incoming data packet has no error, preventing oscillation in estimating the average bit error information
4	Hybrid estimate mode, this is the default mode for the ADIM-NB/FE algorithm, modes 2 and 3 are combined but use different parameters for each type of packet

FEC Coding Rate. If the ADIM-NB/FE algorithm is disabled, the MAC protocol will fix its FEC coding rate to the value specified by this parameter. If the ADIM-NB/FE algorithm is enabled, the MAC protocol will use this value as its initial FEC coding rate until the ADIM-NB/FE algorithm can gather enough information to adjust its FEC coding rate dynamically.

Multi-user Operation Mode. There are several operational modes for the ADIM-NB/MU algorithm as shown in Table 6.2. By setting this parameter to different values, the ADIM-NB/MU algorithm changes its behavior accordingly.

Power Control Operation Mode. There are several operation modes for the ADIM-NB/PC algorithm as shown in Table 6.3. By setting this parameter to different values, the ADIM-NB/PC algorithm changes its behavior accordingly.

Transmission Power. The maximum transmission power of the radio, in Watts, can be changed using this parameter. The default value is 0.01 W.

Channel Bandwidth. Bandwidth of the channel has no relationship with the channel data rate in the simulation models. It is used to calculate the amount of white Gaussian noise in the system.

MAC Protocol Queue Capacity. When the transmitter is not ready to transmit data, it stores that data in a transmission queue. The default capacity is five packets, regardless of the packet size. Setting the queue size too large will result in excessive packet delay. Since packet size is limited to a maximum value of 18,496 bits, this queue does not have unlimited capacity.

Table 6.2. ADIM-NB/MU Operation Modes.

Value	Behavior
0	ADIM-NB/MU is fully enabled, but without using receiver-directed spreading code technique; unicast packet will always go over the control channel
1	ADIM-NB/MU is completely disabled
2	ADIM-NB/MU is partially enabled and will send out only one RTS request at a time but can still receive multiple CTS messages
3	ADIM-NB/MU is completely disabled, but still uses a receiver-directed spreading code technique, all unicast packets will use a receiver-specific channel rather than the control channel
4	ADIM-NB/MU is fully enabled, receiver-directed spreading code technique is also used, all unicast packets will use a receiver-specific channel rather than the control channel, this is the default operation mode for ADIM-NB/MU

Table 6.3. ADIM-NB/PC Operation Modes.

Value	Behavior
0	ADIM-NB/PC is completely disabled
1	ADIM-NB/PC will use only SNR information from the incoming packets to calculate transmit power, this is the default operation mode for ADIM-NB/PC
2	ADIM-NB/PC will use SNR information from the incoming packets, as well as SNR of previous packets, processing gain of the outgoing packet, FEC coding rate of the outgoing packet, packet size, and number of collisions from the incoming packet to calculate the transmit power
3	ADIM-NB/PC will use SNR information from the incoming packets, as well as maximum interference power, number of collisions, bits in error, received power, and FEC coding rate of the incoming packet to calculate the transmit power

Automatic Repeat Request Queue Capacity. This queue capacity is the buffer size for storage of outgoing packets for later retransmission. The default value is fifty packets.

Acknowledgement Queue Capacity. This queue is internally used by the ADIM-NB protocol to store acknowledgement messages that will be piggybacked onto the next outgoing packet. The default capacity is forty packets.

CTS Timeout. This parameter represents the waiting period during the start of RTS packet transmission and the end of CTS packet reception. The default value is automatically calculated and equals twice the estimated round-trip time of RTS and CTS packets.

Channel Wait Time. Channel wait time is the waiting period during the start of current packet transmission and the start of RTS packet transmission for the next packet or data packet for broadcast message. This parameter introduces an opportunity for other stations to initiate their transmissions and for reducing the aggressiveness of the protocol. Default value is set to the same value as the CTS timeout.

Minimum SNR. The desired minimum signal to noise ratio is used in combination with the observed SNR in an RTS packet to determine the appropriate data transmission rate. The default value is 4 dB.

Maximum Backoff Time. Maximum backoff time is the upper limit on the backoff duration when a station experiences a busy channel. The default value is automatically calculated such that the total packet delay is less than one second per hop.

Minimum Data Size. For a small packet, the overhead of RTS/CTS exchange sequence may be too high to show any performance gain. Examples are TCP SYNC, FIN, ACK, IP PING, and telnet session packets. If the data payload size is less than this value (in bits), it will be sent using the broadcast channel without any RTS/CTS negotiation sequence. Note that reliability is not compromised because retransmission is handled by a separate ARQ module. The default value is 1,000 bits.

Backoff Rate. This parameter represents a backoff rate for the exponential backoff algorithm. If station keeps experiencing a busy channel, its backoff duration will be increased according to this parameter. The default value is 10%.

Maximum Data Speed. This variable indicates the maximum speed for data transmission, in multiples of the control channel data speed. It should be a power of two, up to the number of chips per bit for the control channel transmission. However, using chip rate as a maximum data speed provides no protection against any collisions that might happen. The default value is four times. In conjunction with the default processing gain of the control channel, it provides a 6 dB gain against interfering transmissions. Setting this parameter to one will disable the ADIM-NB/PG algorithm.

Maximum Data Size. To reduce the penalty of the negotiation overhead, the ADIM-NB protocol aggregates multiple small packets that are heading to the same destination. This parameter limits the level of aggregation such that the overall packet size must be less than this value.

6.2.1.2 Environment Conditions

Maximum Number of Nodes in the Network. This parameter is used to randomize the destination address of each generated packet. It has no relationship with the actual number of stations in the network.

Network Boundary. Nodes are not allowed to move outside the network boundary.

Node Movement Start Time. The node movement start time is time that the node begins moving, usually after the network has been initialized.

Distribution of Node Speed. Nodes are categorized into five different groups. Each group moves at 0, 5, 20, or 45 Kph, or at an arbitrary speed. These parameters assign the percentage of nodes in each group.

Node Movement Pattern. Over the course of the simulation, nodes are free to move around the network using one of the two mobility patterns, constant speed and direction, and city block movement or "Manhattan" patterns.

Network Density. This parameter reduces the network boundary to a certain size and relocates the nodes accordingly, so that each node in the network will have an average node degree of 6, 12, 18, or 24.

6.2.1.3 Traffic Characteristics

Application Traffic Type. Two types of application traffic are supported, SMTP and HTTP, which have different characteristics.

Traffic Generator Start Time. The traffic generator start time is the time that the first packet is delivered from the higher layer. This parameter allows the routing table to be created before any packet transmission attempts.

Traffic Generator Stop Time. This parameter is useful to study the time it takes to recover after the network experiences a heavily congested state.

Traffic Generator Inter-Arrival Time. This independent variable determines the amount of traffic that is offered to the network.

Average Packet Size. This parameter defines an average packet size, in bytes, for the SMTP traffic. The average packet size for the HTTP traffic is fixed and cannot be changed because of the traffic specification.

6.2.2 Performance Metrics

Performance metrics can also be called response variables. They are briefly described in this section. Chapter 7 provides more extensive information on each performance metric, along with the actual value and their interpretations.

End-to-End Network Throughput. This performance metric indicates the ability of the entire network to successfully deliver data packets to their ultimate destinations. It does not include any MAC or routing protocol overhead.

End-to-End Delay. This statistic includes the effects from several factors, including queueing, transmission, propagation, and retransmission delay and the number of hops to the destination.

Hop-by-Hop Delay. Hop-by-hop delay represents the average delay regardless of the number of hops that a packet needs to travel.

Transmitted Bit per Second. This statistic includes all transmission attempts by the higher layer, regardless of whether they are able to arrive at their destinations.

Bits Lost per Second. The bits lost per second statistic measures any transmission attempts that cannot find a route entry to the destination. It does not include expired packets or packets that are lost due to queue overflow, i.e., congestion.

Average Hop Count. This statistic represents the average number of hops it takes for a packet to reach its destination. It represents how well each node can reach any remote destinations.

Average Retransmitted Ratio per Hop. This statistic represents the average number of retransmissions per hop. Packets are retransmitted on a hop-by-hop basis until the lifetime expires.

Transmitted/Received Routing Information. These statistics represent the amount of routing information that is transmitted and received. This does not include cluster-related information. Both metrics are used to calculate the routing protocol overhead.

Transmitted Cluster Information. This statistic represents the amount of cluster-related information that is transmitted, but does not include any routing information. It is also used to calculate the routing protocol overhead.

Transmitted Power. Transmitted power measures power consumption at all nodes in the network. It depends on the period of activity of the radio transmitter, but not the radio receiver. In conjunction with the number of bits received, it measures how effective the radio uses its power source to communicate one unit of data.

Actual BER. Actual BER is the actual measurement of bit error rate in the network. It indicates the amount of interference, both from other spread-spectrum sessions and any intentional jamming nodes.

Estimated BER. This statistic indicates the value of one key variable used by the ADIM-NB/FE algorithm. It tries to estimate the current interference level in the system in order to adjust the FEC coding rate appropriately. This value should be equal to or higher than the actual BER in order to ensure adequate redundancy in the transmitted packet.

Concurrent Sessions. The concurrent sessions statistic represents the number of simultaneous transmissions in the network. The network can support multiple traffic sessions as a result of both spread-spectrum multiplexing and spatial reuse of the communication resources in different parts of the network.

6.3 Simulation Methodology

This section provides a high-level summary of the methodology used in designing and conducting the simulation experiments, as well as techniques for evaluating protocol performance.

6.3.1 Baseline Simulation Parameters

To evaluate the CAMEN and ADIM-NB protocols, a discrete-event simulator, OPNET Technologies' OPNET Modeler®/Radio [MIL97], is used to simulate their operation. The baseline radio configuration in this research has the following parameters: 1.024 MHz bandwidth, BPSK modulation, 64 Kbits/sec base transmission speed using 16 chips/bit or 1.024 Mchips/sec, (255,239) BCH error correction code [Skl88], and 13 Km maximum transmission range. Each radio has multi-user interference rejection capability, using a PIC algorithm, and can demodulate up to four simultaneous data streams [CBW99]. It also has the ability to quickly change the processing gain and the spreading code. These radio parameters are taken from a software radio design developed at Virginia Tech [Swa98].

The initial network configuration consists of 100 mobile nodes randomly positioned across a 50 by 75 Km area. The number of nodes in the network and its coverage area are chosen to provide an optimal balance between the time required to run the simulation and the spatial separation required to enable frequency reuse in different parts of the network. The IETF MANET working group [IET00] also recommends evaluating a routing protocol with 100 to 400 mobile stations. On average, each node is of degree 12, i.e., it has twelve neighbors. Nodes move at a random speed between 0 to 65 Kph. An informal simulation experiment indicates that increasing or decreasing the coverage area, while still keeping the number of nodes constant, does not significantly alter the network throughput.

The SMTP traffic used in the simulation consists of a log-normal distributed packet length with a mean of 1,024 bytes [Pax94], moving from a random source to a uniformly distributed random destination. The HTTP traffic generates four different types of traffic: local HTML pages, local images, remote html pages, and remote images [CBC95]. Because control overhead plays an important role in the ADIM-NB protocol, a realistic packet header of approximately fifty bytes is added to each packet. In conjunction with the stated network topology, each packet travels for an average of five hops before reaching its destination.

6.3.2 Simulation Fidelity

The simulation model uses a relatively high fidelity approach to calculate packet reception results. The number of error bits in each packet is calculated using a method derived from both analytical and simulation results [CBW99]. Upon packet arrival, the received signal strength of the packet of interest and each interfering packet, along with spreading code information, white Gaussian noise, and thermal noise are used to calculate the number of error bits in each packet segment. This error allocation algorithm does not assume any orthogonal spreading codes or perfect phase synchronization of the received signals. It does not, however, take into consideration any particular fading models. Therefore, the primary causes of packet corruption are either collision or interference. The number of error bits in the packet is then compared to the error correction capability of the radio to determine the final reception result.

6.3.3 Result Evaluation Method

The performance of ADIM-NB is compared with two well-known MAC protocols, the widely used non-persistent CSMA protocol and the basic CDMA-based MAC protocol

(ADIM-NB/B). Network throughput, delay, utilization, error rate, and other metrics are collected and compared for all three protocols for different operating environments and parameters. Both transient and steady-state behaviors are studied. Interactions between ADIM-NB and other external factors are studied with the intent to characterize the protocol behavior in various environments, as well as the future direction of the research.

Both steady state and transient behaviors of CAMEN are studied. In addition, its performance is compared with a non-hierarchical *ad hoc* routing protocol.

6.3.4 Experimental Design

Since the focus of this research is to adapt the protocol behavior according to operating environments, traffic patterns, and other parameters, all variables are highly correlated. A full factorial experimental design is desirable. To obtain a suitable confidence interval for each performance metric, up to six replications of each combination of factors are conducted. If the results from the different replications are greatly varied, additional investigation regarding these differences is performed.

The results from early full factorial experiments are used to design a more effective way to utilize computing resources. By excluding factors that are not correlated, the final experimental campaign using $2^{k-p}\ r$ fractional factorial designs with replications [Jai91] is derived to reduce the total simulation time.

The majority of the experiments involve comparing the protocol’s performance at different load levels against two other MAC protocols. Additional experiments include the protocol’s response to different traffic patterns, node density, and node mobility. The effect of each adaptive algorithm is isolated and identified by enabling a subset of the adaptive algorithms.

6.3.5 Transient Analysis

This section focuses on finding an appropriate simulation period, including transient removal. The simulations are run for a simulated duration of 8,000 seconds (averaging 24 hours of CPU time on a Pentium II/400 MHz PC) to study the network behavior over a long period of time. Results from one of the simulations, with random number seed equal to one, are shown in Figure 6-1.

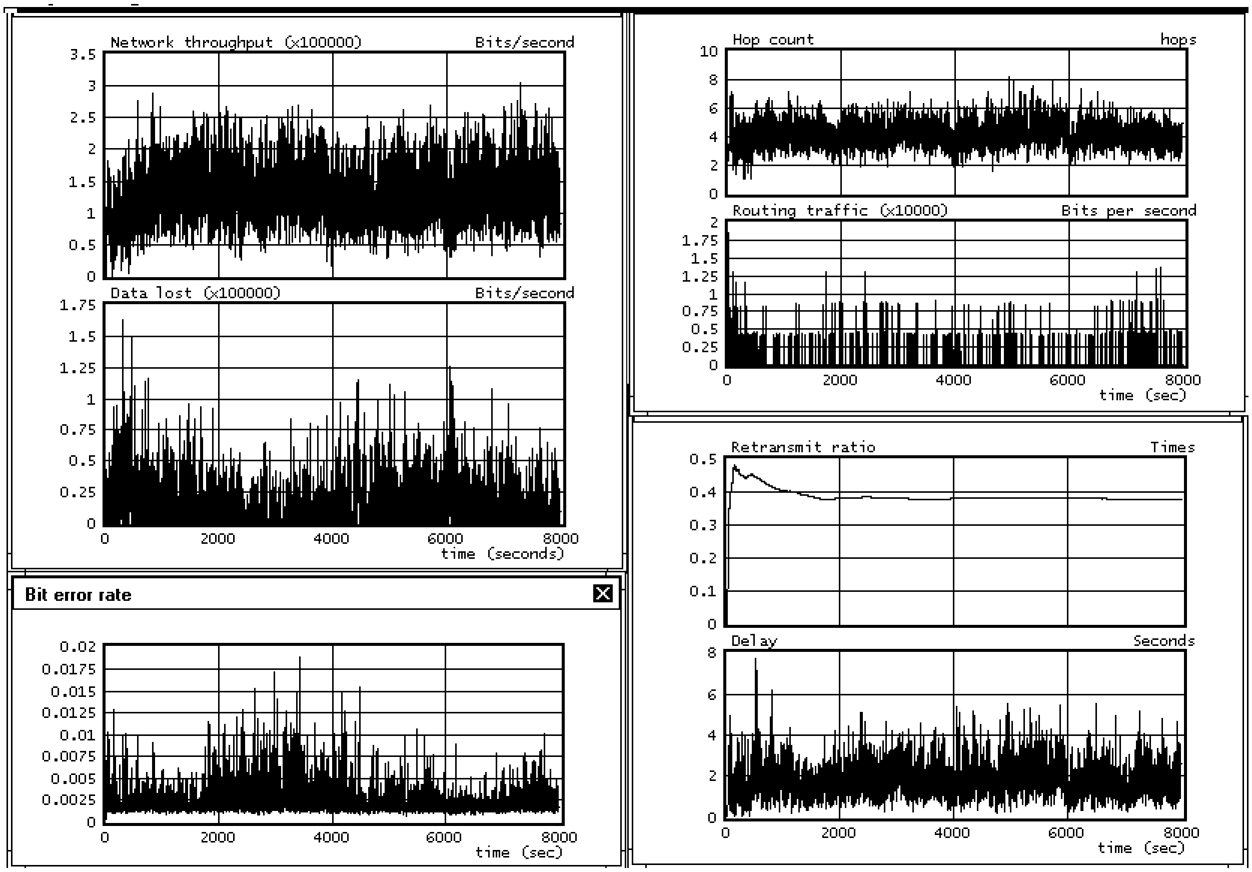


Figure 6-1. Transient analysis results.

A distance-vector routing protocol requires a number of route update cycles, equivalent to the network diameter, before it finishes calculating the initial route. Additional time is required to load the network to a desired level of utilization. The overall transient state, including both route initialization and loading, is identified by using an initial data deletion algorithm [Jai91]. In most of the simulations, the initial 1,000 seconds are ignored. Applying the truncation of moving average of independent replications algorithm [Jai91] to the end-to-end delay statistic also leads to the same conclusion.

6.3.5.1 Stopping Criterion

The appropriate simulation length is determined by variance estimation algorithms, which suggest a simulation duration of 1,000 seconds. The variance estimation algorithms base on the principal that the simulation should be run until the confidence interval for the mean response narrows to a desired width [Jai91]. The primary algorithm for the variance estimation algorithms in this research uses an auto-covariance computation of the simulation results.

6.3.5.2 Confidence Interval and Early Termination

Each simulation must be replicated several times to ensure the diversity and validity of the statistics. To avoid wasting the first 1,000 seconds in every simulation because of the transient state, a batch means technique is used. The method of batch means, also called method of sub-samples, consists of running a long simulation run, discarding the initial transient interval, and dividing the remaining observations run into several batches or sub-samples [Jai91].

After discarding the first 1,000-second transient interval, the simulation consists of several batches of size 1,000 seconds. The results from each batch are used to calculate the confidence intervals for two major performance metrics, throughput and delay. The simulation continues to another batch until either the desired confidence intervals for both performance metrics are reached or the maximum number of batches has been reached. Specifically, the simulation continues until either the 95% confidence interval width is within 5% of the mean or a maximum of seven batches has been reached. The minimum number of batches required to calculate the confidence interval is three.

6.4 Model Verification

A number of debugging and consistency checks are employed for both ADIM-NB and CAMEN protocols, including checking for long-term memory leaks, using a simplified scenario for a lightly loaded network, investigating packet losses, continuity testing for major parameters, bounds checking for all major variables, and verifying seed independence. In addition, rigorous tracing and debugging are employed to ensure the correct operation of the simulated model. Sections 6.4.1 and 6.4.2 explain a specific verification detail of each component of the study.

6.4.1 ADIM-NB Verification

Because the ADIM-NB MAC protocol is rather straight forward, the verification process includes all the standard verification techniques described in the previous paragraph, as well as using trace information to ensure that the MAC protocol performs exactly the same as its intended behavior, as described in Chapter 4.

6.4.2 CAMEN Verification

The verification procedure for CAMEN includes all the verifications steps previously described, as well as using the MAC layer statistics to verify the network layer statistics and verifying the routing protocol’s responses to various changes in the network topology, including both transient and steady state behaviors. For example, the number of packets transmitted and received by the network layer must match those values collected by the MAC layer. In addition, its transmission delay must equal the summation of all delay collected by the MAC layer at each hop along its path.

Other verification techniques employed include transient and steady state behaviors. After a change in the network topology, the amount of routing traffic surged momentarily and went back to a steady state. The model was subjected to a period of intense traffic followed by an idle period. Numerous statistics and performance metrics were collected to verify its stability during a transient state and to ensure that it reached steady state.

Three scenarios, shown in Figure 6-2, were used to verify the route update algorithm. These three scenarios force the routing algorithm to recalculate routes and switch the traffic to an equal cost route, a lower cost route, and a higher cost route, respectively. In all cases, traffic always flows from mobile_1 to mobile_2, and mobile_3 is moving in the direction indicated by the arrow. The first scenario forces the route to go through mobile_4 after mobile_3 is out of range. The second scenario creates a new lower cost route via mobile_3b's new location. The third scenario is the opposite of the second scenario where mobile_3c's movement forces the traffic to use the original higher cost route via mobile_4c and mobile_5c. Their results are shown in Figure 6-3.

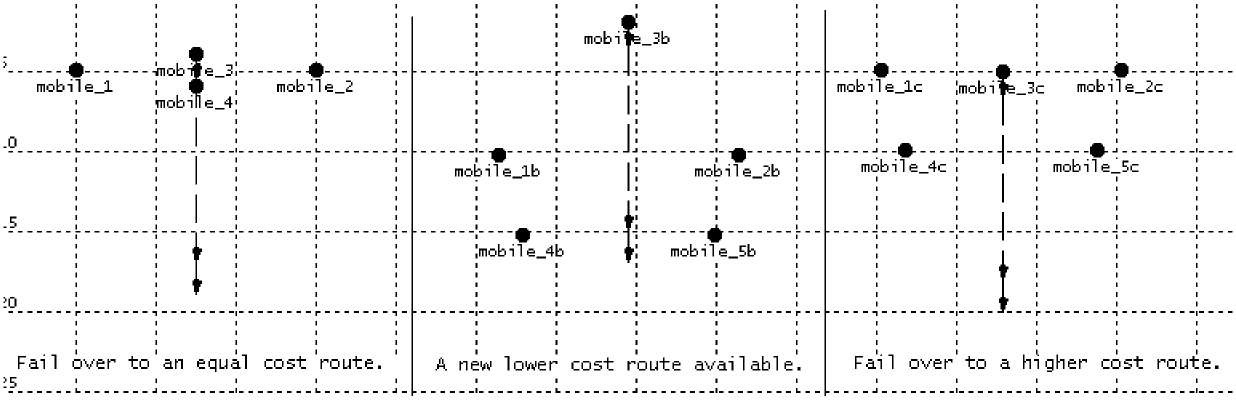


Figure 6-2. Network topology for routing protocol verification.

In all cases, data traffic continues without interruption, while routing traffic surges slightly during the rerouting period, as can be seen from Figures 6-3(a) to 6-3(c). Figure 6-3(d) shows the moments when changes occur and their corresponding path lengths over the simulation period. For the equal cost experiment, the number of hops that each packet travels always equals two. The number of hops that each packet travels drops from three to two at approximately 170 seconds in the lower cost experiment, and the hop count increases from two to three at 120 seconds in the higher cost experiment.

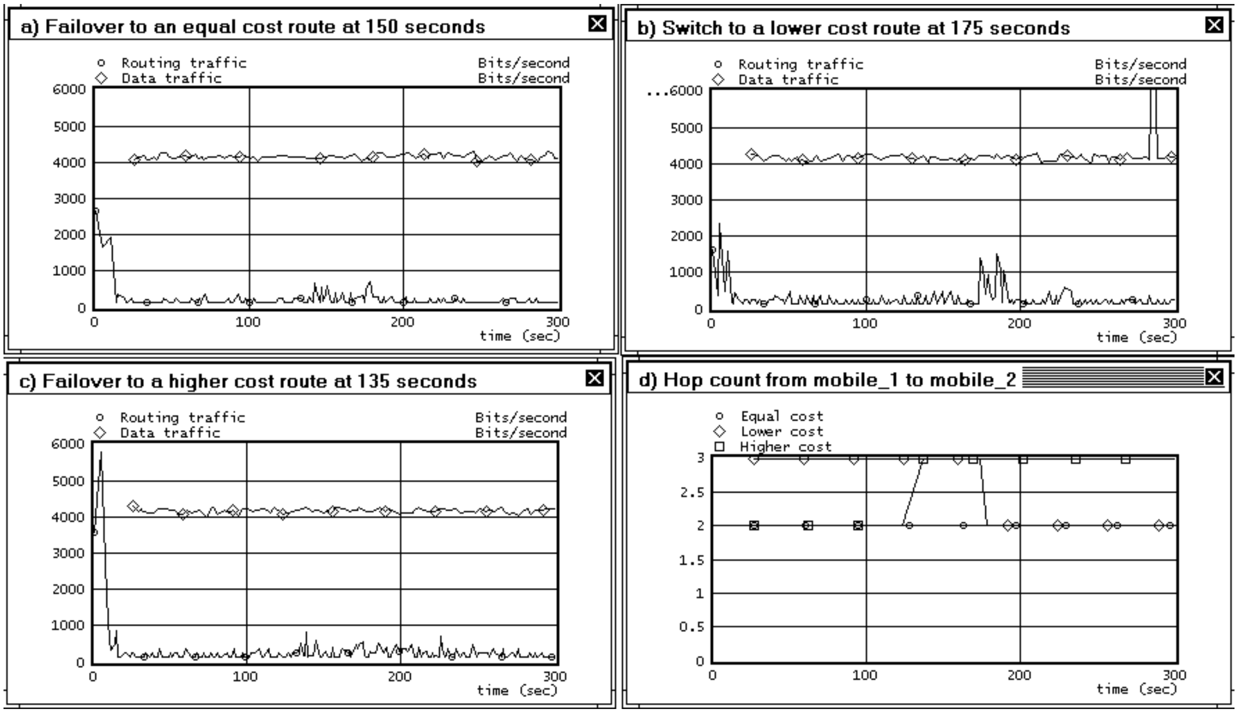


Figure 6-3. Route validation results.

Three additional scenarios shown in Figure 6-4 are used to verify the clustering algorithm. In Figure 6-4(a), gateway mobile_3 moves out of range, resulting in mobile_4 being selected as a new gateway. A new cluster is also formed by mobile_3 after the move. Figure 6-4(b) shows the situation where a regular node, mobile_5b, moves within the cluster boundary. It also presents a situation where cluster mobile_1b is destroyed because its new location is already covered by mobile_2b. Figure 6-4(c) depicts the handoff procedure where mobile_5c is unregistered from cluster mobile_2c and registered to cluster mobile_3c. Traffic from the foreign cluster, mobile_1c, is rerouted accordingly. Figure 6-5 shows the corresponding results from Figure 6-4. Figure 6-5(b) shows no significant changes in the routing traffic when movement occurs within the cluster boundary. In all experiments, extensive trace and state information are verified to ensure the proper computation of the routing table and cluster information.

6.5 Model Validation

To at least partially validate the MAC protocol model, the ADIM-NB algorithm was given a set of variables and backoff algorithms that mimic a non-persistent CSMA MAC protocol with a uniform backoff algorithm for a 200-node network. The resulting throughput was then compared to an analytical solution that assumes an infinite number of nodes [Kle76]. The two cases match within less than one percent.

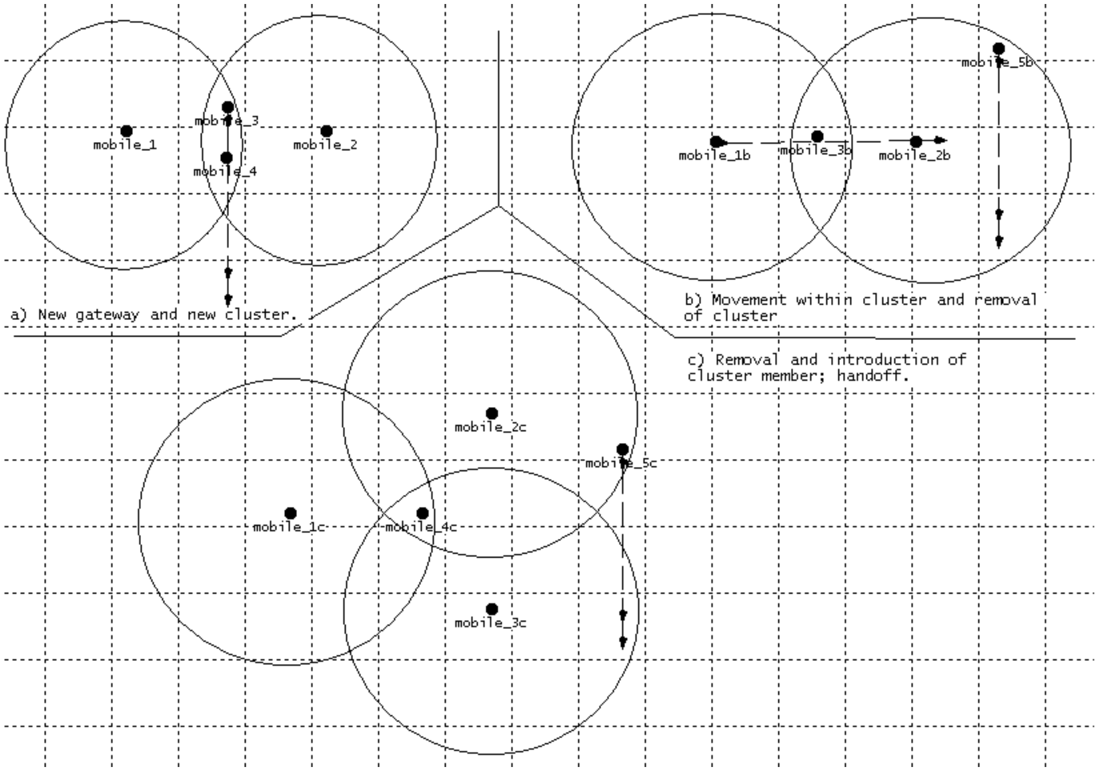


Figure 6-4. Network topology for clustering algorithm verification.

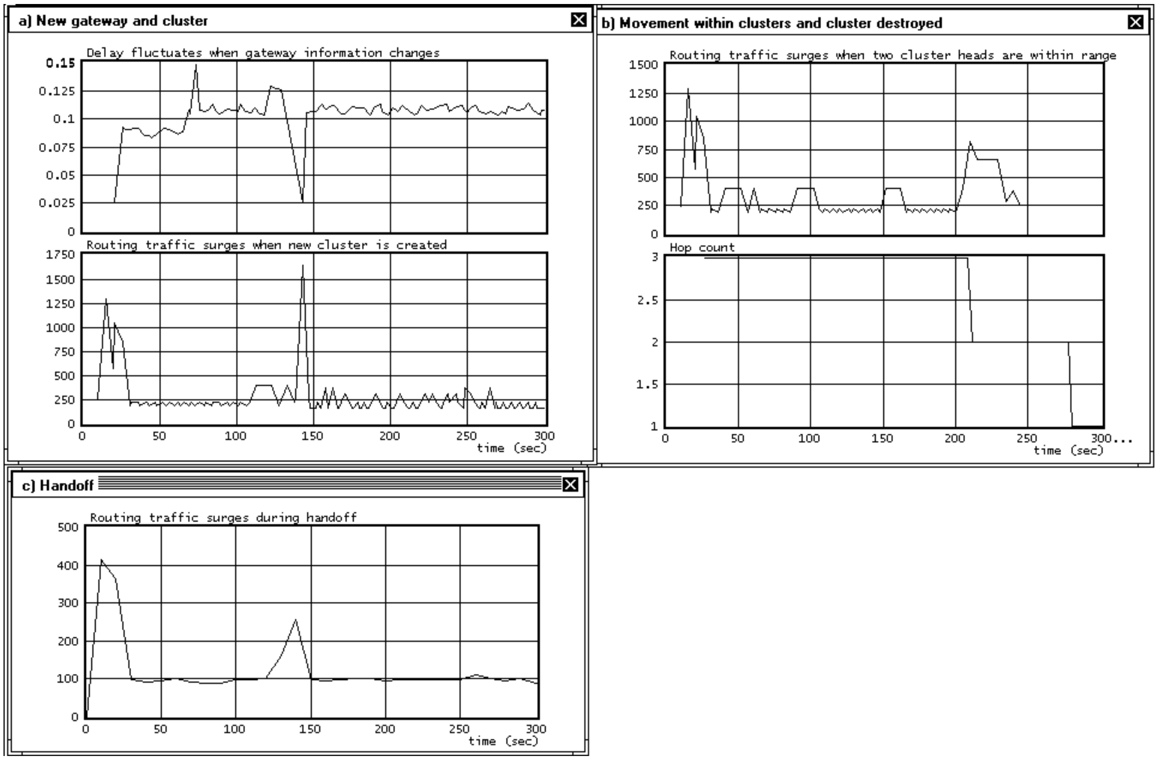


Figure 6-5. Clustering algorithm verification results.

6.5.1 ADIM-NB Validation

To validate the proposed ADIM-NB MAC protocol, it was given a set of variables and a backoff algorithm that mimics a non-persistent unslotted CSMA MAC protocol with a uniform backoff algorithm. The protocol parameters are summarized in Table 6.4. Normalized packet throughput, which is the measured packet throughput divided by total channel capacity, from the simulation results (with a 95% confidence interval) is compared to an analytical solution [Kle76] and shown in Figure 6-6. Although the comparison is between 200 simulated nodes and an infinite nodes analytical solution, both cases match with less than one percent of error.

Table 6.4. Parameters for Non-Persistent Unslotted CSMA Protocol.

Number of nodes	200 nodes
Average propagation delay	0.24 milliseconds
Packet size	4000 bits (constant)
Channel speed	64 kilobits per second

Figure 6-6. Analytical versus simulation results for non-persistent CSMA protocol.

6.5.2 CAMEN Validation

Because there is no comparable protocol to or different implementation of the CAMEN routing protocol, it is difficult to validate such an experimental clustering algorithm. Nevertheless, an extensive and thorough verification is used to offset any deficiency in its validation process.

6.6 Summary

This chapter presented the simulation-specific methodology and details about the simulation environment. The OPNET Modeler®/Radio was used as the simulation tool to construct, simulate, and analyze the protocols under study. The entire wireless MAC and routing protocols are modeled via simulation including the ADIM-NB MAC protocol, the CAMEN routing protocol, and the underlying DSSS physical layer. The actual simulation parameters and performance metrics were briefly mentioned. Model verification and validation efforts were then discussed.

Chapter 7. Simulation Results

This chapter describes the results obtained during simulation evaluations. Sections 7.1 and 7.2 discusses the effect of each individual ADIM-NB algorithm, ADIM-NB/B, ADIM-NB/PG, ADIM-NB/PC, ADIM-NB/MU, and ADIM-NB/FE, in a static network to minimize the effect from the routing protocol. Section 7.3 studies the performance of the ADIM-NB/FE algorithm in a hostile environment. Section 7.4 discusses how the ADIM-NB MAC protocol reacts to different environmental conditions and traffic characteristics. Section 7.5 uses the information from both Sections 7.2 and 7.4 to study the effect of environmental conditions on each ADIM-NB algorithm. Finally, Section 7.6 discusses the performance of CAMEN in various environments followed by a summary in Section 7.7.

Most of the simulation results presented in this chapter are based on parameters from Chapter 6, and the initial network topology in Figure 7-1. It consists of 100 mobile nodes randomly positioned across a 50 by 75 Km area. In Figure 7-1, the circle around node _97 shows an approximate transmission range in a collision-free environment with a processing gain of 12 dB. The number of nodes in the network and its coverage area are chosen to provide an optimal balance between the time required to run the simulation and the spatial separation required to enable frequency reuse in different parts of the network. The IETF MANET working group [IET00] also recommends evaluating a routing protocol with 100 to 400 mobile stations. On average, each node is of degree 12, or has twelve neighbors. Nodes move at a random speed between 0 to 65 Kph. Experiments with other network configurations and node mobility patterns are explained in Section 7.2. The descriptions of each set of statistics, operating variables, and parameters are kept to a minimum level in this chapter. Appendix A has more detailed explanations.

Most experiments focus on a moderately to heavily loaded network because it is crucial to be able to assess the network performance in such environments. For a lightly loaded network, most protocols are indifferent because a lot of channel capacity is left unused. However, general wisdom suggests that a simple protocol will have the best performance in a lightly loaded scenario because of its simplicity and minimal overhead requirement. Future experiments, however, will not include any study of a lightly loaded network because of its limited usefulness.

Section 7.1 contains the most detailed explanation of each metric as well as the suggested interpretation of the statistics. Such explanations are more limited in other sections. Their impacts in different environments, however, are experiment-dependent and are extensively explained in all sections.

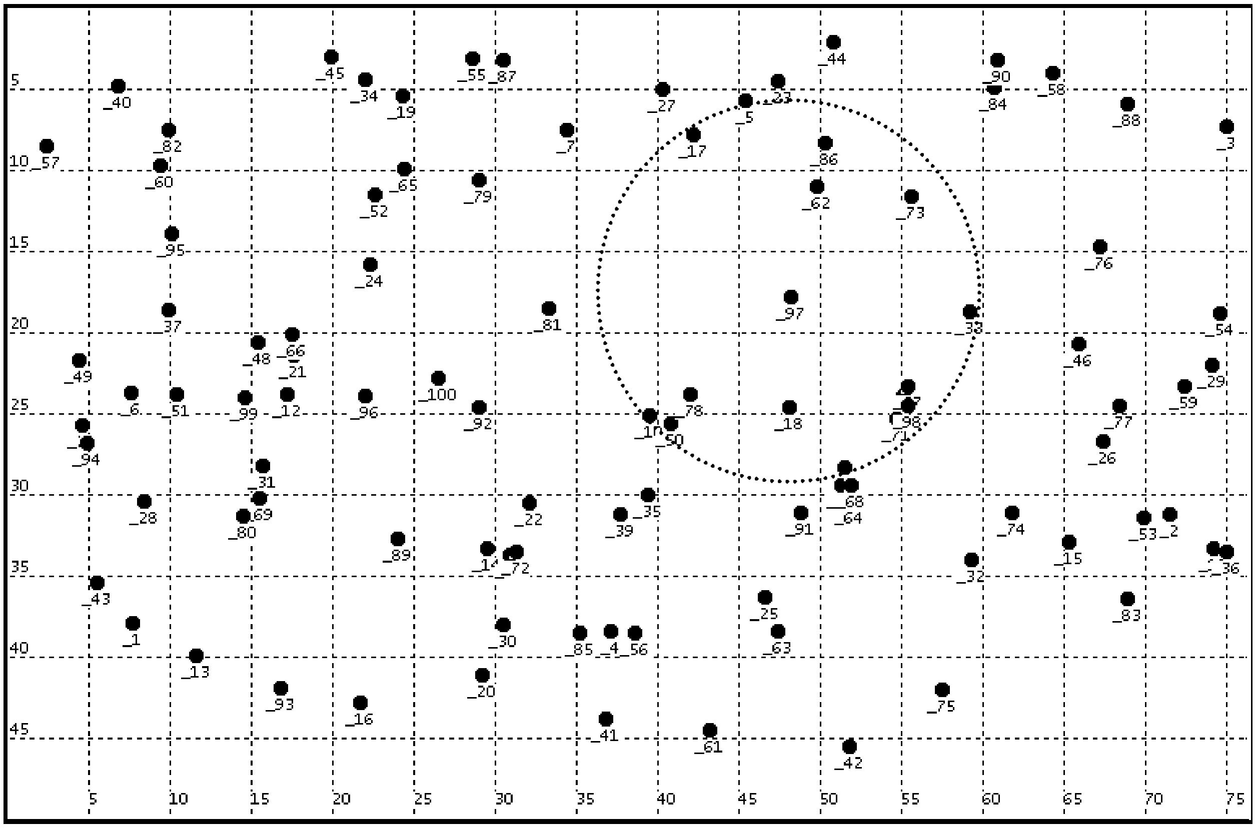


Figure 7-1. Initial network topology used for most experiments.

7.1 Comparison with Other MAC Protocols

The performance of the ADIM-NB algorithms (collectively) is compared with two other MAC protocols: a non-persistent CSMA MAC protocol [BeG92], and ADIM-NB/B, i.e., ADIM-NB without any adaptive algorithms. All MAC protocols use the same radio with parallel interference cancellation (PIC) capability. The simulation objective is not to compare the performance of a PIC-capable radio to that of a single-user receiver radio since that would be obvious. It, instead, intends to justify an improvement from a MAC protocol specially designed for the PIC-capable radio to that of a generic MAC protocol in the presence of a PIC-capable radio.

Although the non-persistent CSMA MAC protocol has been shown to be unsuitable for a narrowband MANET environment, it has reasonable performance in a spread-spectrum system with a relatively good capture characteristic [Tob87]. Consequently, it represents a worthy alternative in the environment of interest, which is a spread-spectrum environment. The use of spread-spectrum modulation is sometimes mandatory for a given environment, such as in the ISM frequency band in the United States or in a hostile environment. Additionally, comparing a CSMA MAC protocol in a narrowband environment with a CDMA MAC protocol in a spread-spectrum environment is like comparing a TDMA system with a CDMA system. Such comparisons are numerous in the literature and their results are well-known. Consequently, the non-persistent CSMA MAC protocol also uses the spread-spectrum accessing technique and uses the same processing gain of 16 chips/bit as the ADIM-NB/B protocol. The description of each MAC protocol is summarized in Table 7.1. All protocols use 1.024 Mchips/second.

Other MANET MAC protocols are not considered here because of their limited applicability. As mentioned earlier, most of them are designed for a narrowband environment which will essentially result in a single spreading code being used in a spread-spectrum environment. Any comparisons would be unfair. A comparison between CSMA and the ADIM-NB/B protocol should be used to quantify the improvement from a MAC protocol that can take advantage of a multiple spreading codes environment versus that of a generic MAC protocol. The comparison between the ADIM-NB/B protocol and the ADIM-NB protocol should indicate the improvement from a MAC protocol that can adjust its parameters to the channel environment.

Table 7.1. MAC Protocols Descriptions.

Legends	**Descriptions**
CSMA	A non-persistent CSMA MAC protocol with 16 chips/bit, or a data rate of 64 Kbits/s
ADIM-NB/B	ADIM-NB/B protocol with 16 chips/bit or a data rate of 64 Kbits/s
ADIM-NB	Collection of ADIM-NB algorithms, consisting of: ADIM-NB/PG algorithm that can automatically switch among 16, 8, or 4 chips/bit ADIM-NB/PC algorithm that can automatically lower its power by a maximum of –9 dB ADIM-NB/MU algorithm that can set up multiple simultaneous connections ADIM-NB/FE algorithm that can change its FEC coding rate among 15/16, 14/16… or 9/16

End-to-end network throughput, end-to-end packet delay, and routing overhead as well as other metrics are used to compare the MAC protocols for different operating environments and traffic levels. In this section, a node does not change its position during the experiment, which provides insight into a rapid static deployment application and the best-case performance of each protocol. The performance of the ADIM-NB protocol in a dynamic network is discussed in Sections 7.4 to 7.6. The traffic used in this section consists only of SMTP traffic [Pax94]. The destination address of each SMTP request is a uniformly distributed random variable. Each node has an equal chance of receiving or transmitting an SMTP request.

Figure 7-2 shows the end-to-end network throughput, which indicates the ability of the entire network to successfully deliver packets to their ultimate destinations. It takes into account all user data plus IP header. It does not include MAC- and physical-layer headers or routing and clustering traffic. For a fully connected network in which every node can talk directly to every other node, this is simply the channel capacity of that network. For a larger network, where each packet may need to traverse multiple hops, the same packet may have to be transmitted multiple times, each on a different hop in the network. Each unique packet contributes to this statistic only once, no matter how many hops it traverses. As a result, having a large network can potentially reduce the end-to-end network throughput. However, such a network also allows concurrent transmissions by different parts of the network, i.e., spatial reuse. The benefit of spatial reuse has been studied in [KlS87] and can increase network capacity and compensate for having to send packet across multiple hops. Consequently, this statistic represents a network capacity as a whole and has little dependency on the network size.

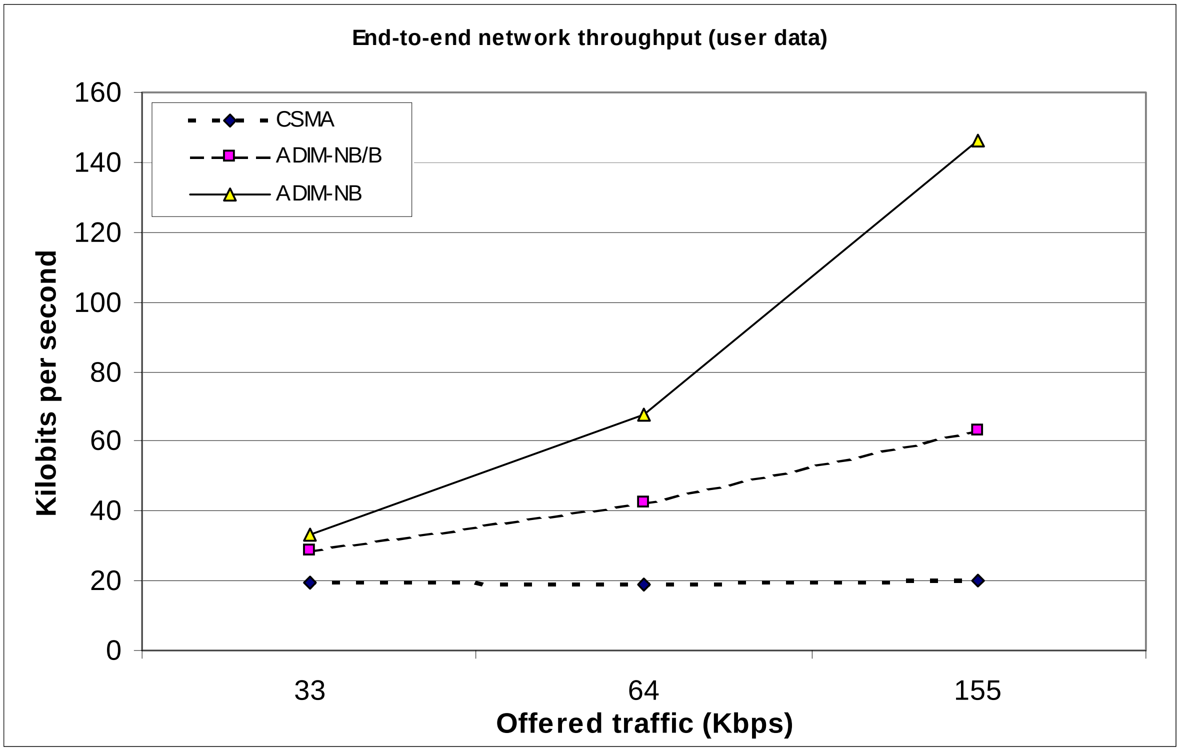


Figure 7-2. Network throughput (user data) comparison with CSMA.

The performance gain from both ADIM-NB MAC protocols can be observed from the graph. Both protocols perform better than CSMA in moderately and heavily loaded environments. In the heavily loaded scenarios (64 and 155 Kbps), the ADIM-NB/B protocol delivers more than double the amount of traffic delivered by CSMA and further improves the performance by 60% to 120%. The saturation in throughput of the CSMA curve emphasizes the instability of the "free-for-all" approach upon which the CSMA MAC protocol is based. Protocols that are based on a demand/assignment approach, such as the ADIM-NB protocol, are more stable in a heavily loaded network.

Although the offered traffic increases exponentially, the CSMA and ADIM-NB/B protocols' throughput increases linearly, indicating a limitation in network capacity. As will be apparent later, such increases in throughput of both protocols primarily result from local delivery, rather than from communicating with remote destinations. The ADIM-NB protocol, on the other hand, is able to deliver most of the offered traffic and the network throughput increases exponentially.

Although the performance improvement of the ADIM-NB protocol and the ADIM-NB/B protocol seems impressive, such improvements represent an ideal condition. In this section, all nodes remain stationary and well-connected, and there is no external factor such as a jammer. In Sections 7.2 and 7.3, the ADIM-NB protocol is broken into individual algorithms to assess the performance improvement in comparison with the ADIM-NB/B protocol. They will also be evaluated in a less ideal environment.

Other results also confirm the previous analysis. Both CDMA-based MAC protocols, especially the ADIM-NB protocol, have a lower end-to-end delay, even in a moderately loaded network, as shown in Figure 7-3. The ADIM-NB protocol and the ADIM-NB/B protocol pay a smaller penalty for each collision because a small RTS packet is lost instead of a much larger data packet in the CSMA MAC protocol. This also implies that the RTS/CTS technique used in a narrowband system [IEE99] can also prove useful in a spread-spectrum environment.

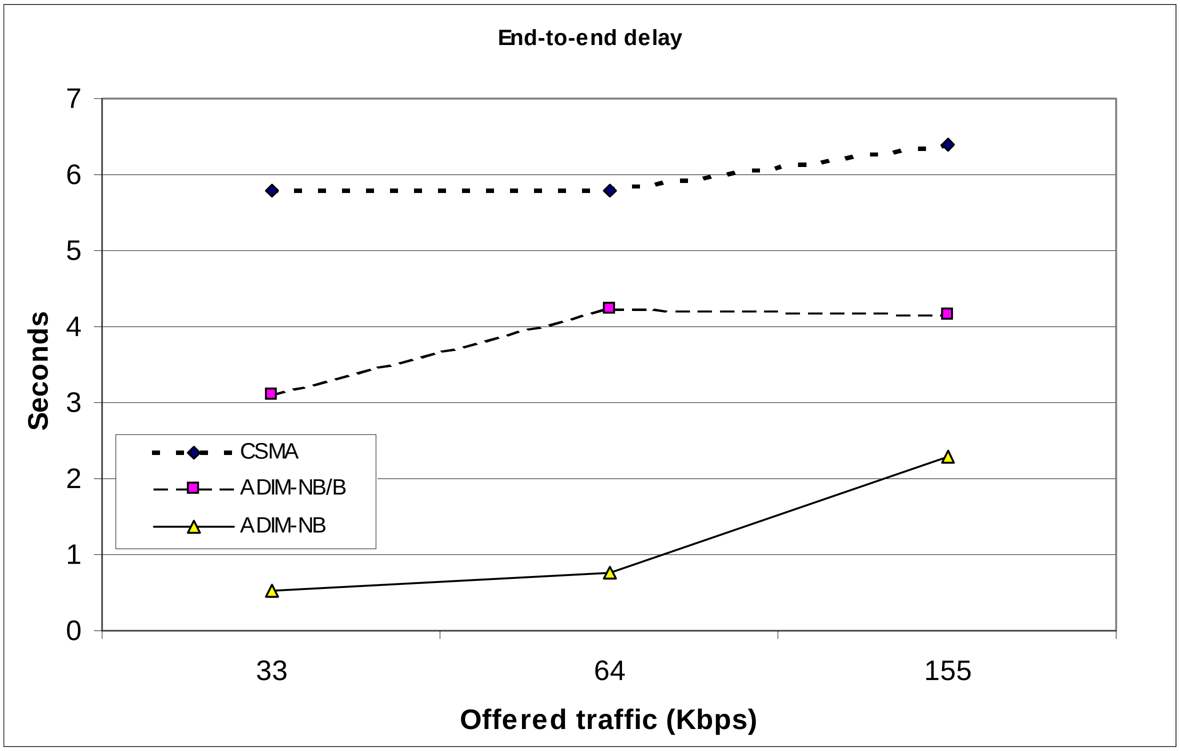


Figure 7-3. End-to-end delay comparison with CSMA.

The CSMA MAC protocol appears to exhibit constant delay, no matter how heavy the offered traffic. Combining the result of Figure 7-3 with the result from Figure 7-2, one might conclude that CSMA has the same performance regardless of the traffic level. However, the result from Figure 7-4 indicates otherwise. Although the end-to-end throughput and delay are constant, CSMA has a much smaller coverage area in a heavily loaded scenario. Because the network is heavily congested, it can only deliver a data packet to nearby destinations. Since the statistics from Figures 7-2 and 7-3 are end-to-end statistics, this indicates that throughput and delay are relatively constant even though it now delivers the packet to a closer destination. In summary, the CSMA MAC protocol does degrade when the traffic level increases.

Although the CSMA end-to-end delay appears to saturate at 6 sec, it is not a MAC protocol characteristic. In these experiments, all packets have an expiration time of 6 sec per hop. If a packet cannot be delivered within that period, it is discarded. The discarded packet will not effect the end-to-end delay statistic. Therefore, packets with excessive delay tend to expire and, therefore, do not affect the end-to-end delay statistic.

Since the ADIM-NB/B algorithm is based on a demand/assignment approach, the algorithm degrades gracefully. The end-to-end delay increases slightly even when the offered traffic is doubled or quadrupled. The high delay in a 33 Kbps scenario results from the negotiation overhead. Each packet transfer needs to establish a connection using the RTS/CTS handshake sequence.

Because of its adaptivity, the ADIM-NB protocol can further reduce the delay. It can send some data packets at a faster rate than the other two MAC protocols and can communicate with several neighbors at once. It also has other advantages, as explained in subsequent sections. As a result, using the ADIM-NB MAC protocol leads to a higher network capacity.

In most cases, the delay does not change much as the offered traffic increases, except in the ADIM-NB protocol. The networks using both CSMA and ADIM-NB/B protocols begin to saturate. The network using the ADIM-NB protocol, however, is not yet saturated and, therefore, its delay increases with the traffic level.

Figure 7-4 shows the average number of hop counts per packet, which indicates the number of hops, on average, required for the packet to reach its destination. Because all simulations in this section have the same traffic characteristic, each packet transmission attempt takes the same number of hops, on average, to reach its destination. From the moderately loaded scenario in Figure 7-4, it is safe to assume that this average number of hop counts is five hops. Since all requested packets may not be delivered to their destinations, as will be seen in the "percentage of data delivered" statistic, a selective packet delivery process occurs. In a congested network, a packet that is heading to a nearby destination is more likely to be delivered than the one destined for a more remote destination. Consequently, the average hop counts is reduced in such a network. Because the network is static, it is less likely that the average number of hop counts will be increased because of loops in the routing table. As a result, a higher hop count statistic indicates that the network is able to deliver a packet to most destinations, while a lower hop count statistic indicates that a packet destined for a remote destination cannot be delivered.

The performance gain from both ADIM-NB MAC protocols can be observed from the graph. Both the ADIM-NB/B and ADIM-NB protocols are able to deliver packets to a much farther destination than CSMA. The CSMA MAC protocol also degrades rapidly as the traffic level increases. In comparison, the ADIM-NB/B and ADIM-NB protocols degrade gracefully as the traffic level increases, although the rate of degradation of the ADIM-NB/B protocol matches that of CSMA in a heavily loaded scenario. The ADIM-NB protocol delivers the best performance and exhibits a higher network capacity and stability. The number of hop counts degrades more rapidly in a heavily loaded scenario in all MAC protocols simply because there is no congestion control mechanism between the network layer and the application layer, and, therefore, the traffic entering the network cannot be controlled. In a production network, the transport protocols or the application protocols are able to control the rate in which traffic is introduced to the network. It is important, however, to disable the congestion control mechanism for this study in order to measure the network capacity under extreme conditions.

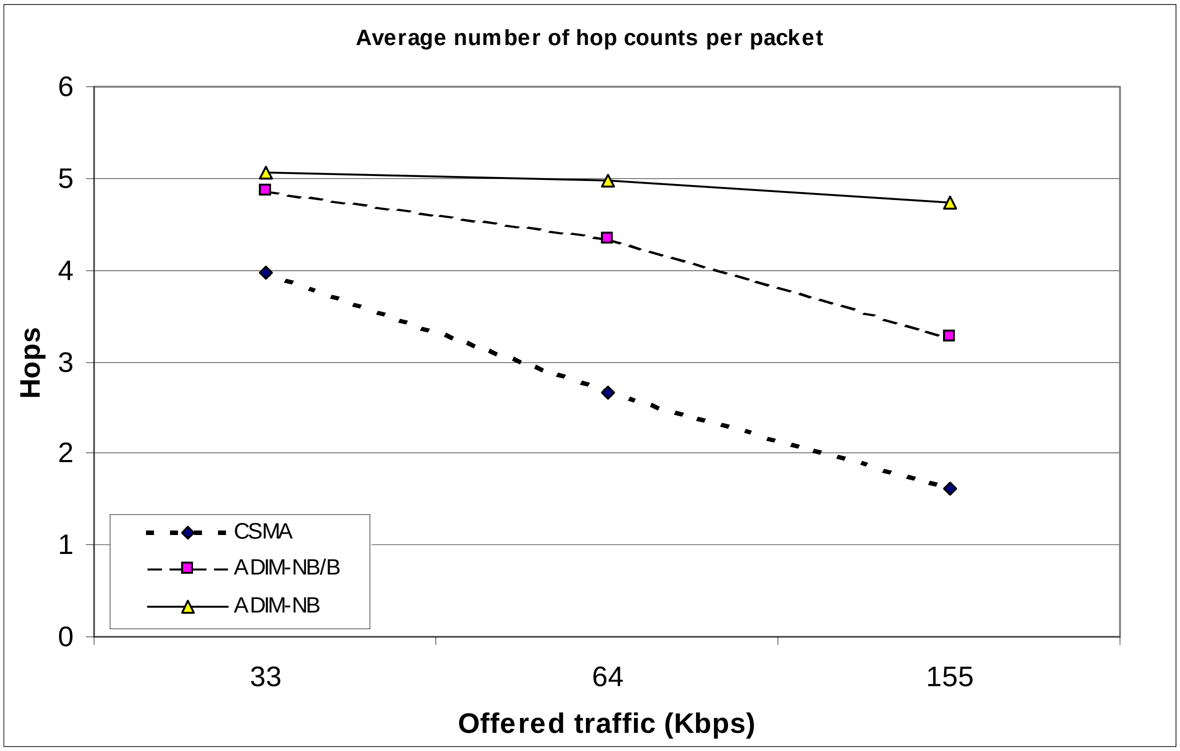


Figure 7-4. Hop counts comparison with CSMA.

Figure 7-5 shows the percentage of routing and clustering overhead over all traffic received. Because only the received traffic is of interest, any unsuccessful transmissions of routing or data packets are not included. The receiver-side statistic is chosen because it better reflects the current network condition. CSMA has the worst performance in this aspect, trailing both the ADIM-NB and ADIM-NB/B protocols by a large margin.

Figure 7-5 reveals the instability from the high delay experienced by CSMA. When a node cannot reach its neighbor within the time limit, it removes that neighbor's information from the routing table and propagates the updated routing information to other nodes. However, if that neighbor can later be reached, as is the case here, it will initiate a new neighbor discovery algorithm and, again, propagate the information to other nodes. Although CAMEN attempts to avoid such a momentary change in connectivity, the rediscovery interval of CSMA is too long for the routing protocol to track, resulting in greater routing overhead and network instability. In addition, any unsuccessful transmission attempts also contribute to a faster removal of the corresponding route from the routing table. Because CSMA has limited network capacity, as shown in Figure 7-2, it experiences more failures. This high failure rate also causes the route to be constantly removed and rediscovered, resulting in high routing traffic in the network.

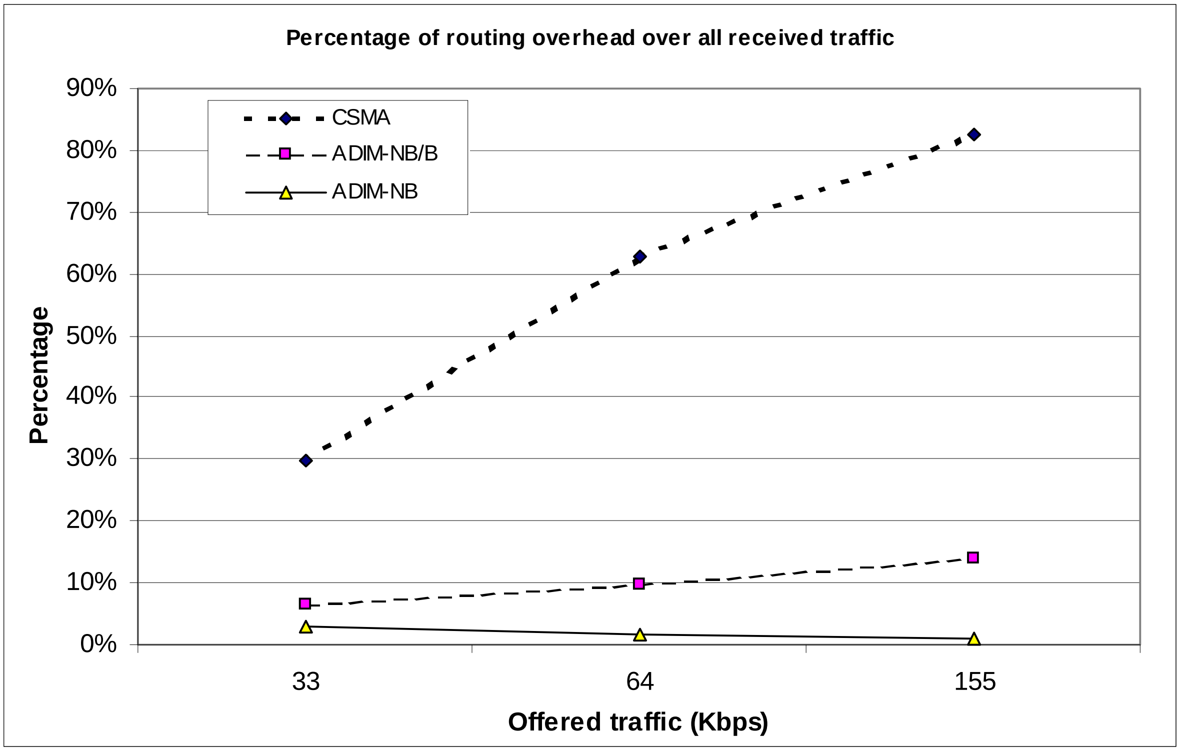


Figure 7-5. Percentage of routing overhead comparison with CSMA.

The ADIM-NB protocol maintains the amount of routing overhead relatively well. Its percentage of routing overhead does decrease when the offered load increases, because that there is a small amount of user traffic in a lightly loaded scenario. Among all MAC protocols, the ADIM-NB protocol exhibits the best stability characteristic.

The percentage of routing overhead of the ADIM-NB/B protocol slowly increases as the traffic increases, exhibiting a slight degradation in the performance, because it has a lower network capacity than the ADIM-NB protocol and it starts to experience network congestion.

Since all the experiments use the same routing algorithm, it is interesting to see why different MAC protocols result in different amounts of routing overhead. Further investigation of Figure 7-4 reveals that, although the throughput increases, the ADIM-NB protocol can deliver more data packets to a remote location than any other MAC protocols. This high level of fairness results in a better network connectivity because it does not favor local destinations. As a result, the network remains well-connected and the routing overhead remains under control.

The percentage of routing table coverage statistic shown in Figure 7-6 represents the status of the routing table. A routing table that has routes to all destinations would have 100% coverage. Because the network under consideration is not partitioned, a valid route to any destination always exists. However, congestion and repeated packet drop will result in some route removal, leading to an incomplete routing table. Consequently, this statistic shows how well a routing protocol is able to maintain a route in different environments.

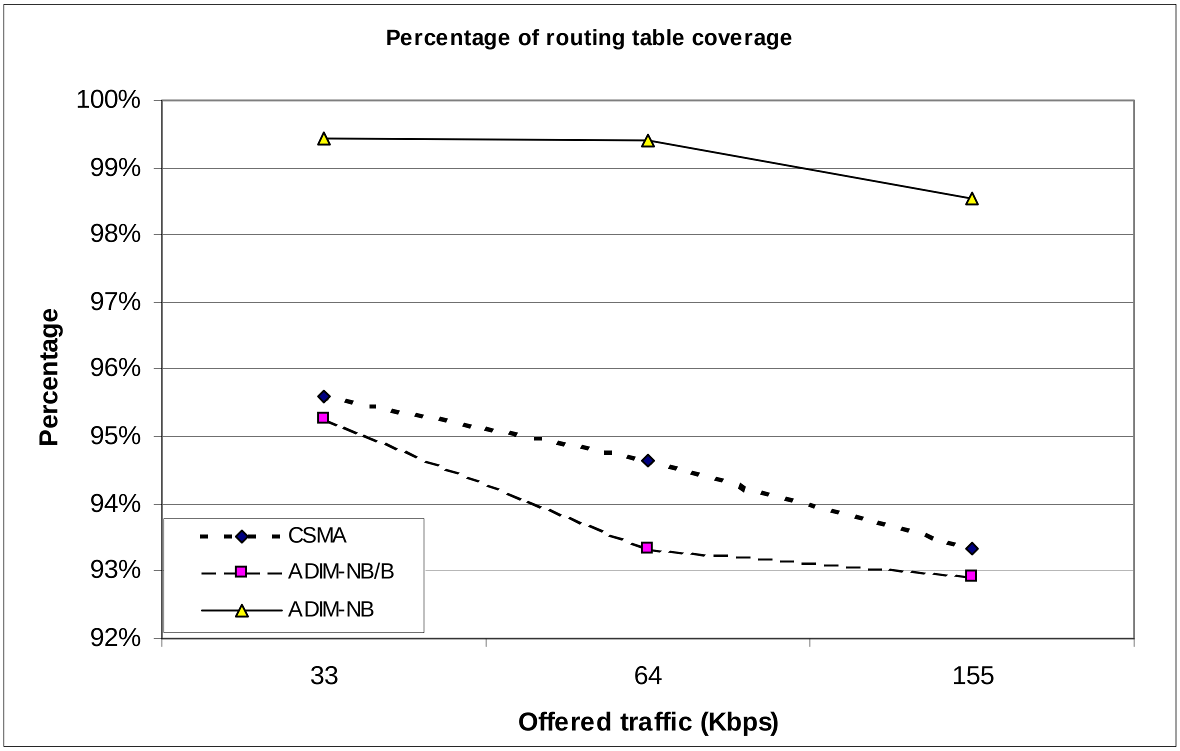


Figure 7-6. Routing table coverage comparison with CSMA.

The ADIM-NB protocol exhibits superior performance in all environments. It is able to maintain an almost complete routing table. The ADIM-NB/B and CSMA protocols, however, have slightly lower performance. Although the results show that the ADIM-NB/B protocol has lower performance than CSMA, both curves are relatively close to each other. Because all packet transmissions in CSMA are broadcast packets, they implicitly help the routing protocol maintain the network connectivity. The ADIM-NB/B protocol, on the other hand, heavily uses unicast service, to which only the intended destination can listen. Therefore, other stations cannot learn the network connectivity except from explicit routing update and control messages, hence the slightly better performance in CSMA. Since all MAC protocols are based on the same routing protocol, their performance is relatively the same. The only difference results from network congestion of each MAC protocol.

Figure 7-7 shows the percentage of data transmission requests that actually get delivered to their intended destination. Although this statistic is related to the throughput statistic in Figure 7-2, they are not identical. Because the traffic used in the simulation is SMTP traffic, the traffic has a request-response characteristic. If the request never arrives at the destination, the reply will never be generated. As a result, the offered traffic is reduced somewhat in a congested network. In addition, Figure 7-7 measures the protocol performance from another perspective, giving a sense of QoS, in addition to the traditional performance metrics, such as throughput.

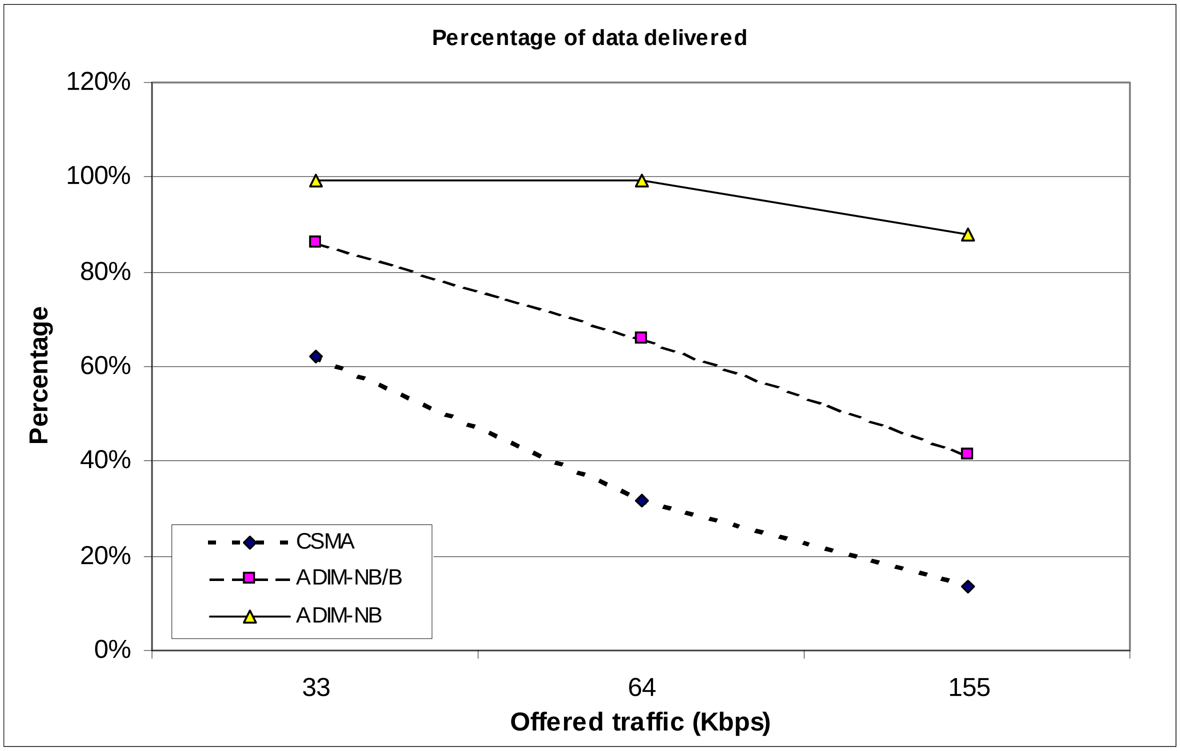


Figure 7-7. Percentage of data delivered comparison with CSMA.

The ADIM-NB protocol is consistently able to deliver most traffic to its destination. As explained earlier, there is no congestion control scheme and some traffic will eventually be lost. In heavily loaded scenarios, all MAC protocols degrade at approximately the same rate due to the lack of a flow control mechanism. The only difference among each of them is the point of degradation, which represents a desirable operating point. The ADIM-NB protocol performs better than ADIM-NB/B, which performs better than CSMA.

Figure 7-8 shows the average retransmissions per packet. All MAC protocols retransmit packets on a hop-by-hop basis. Consequently, this statistic is hop-by-hop-based. The MAC protocols treat the lack of an acknowledgement message as a transmission failure and retransmit the packet over that hop until the packet expires. There are many causes for transmission failure, including collision among data, RTS, and CTS packets, buffer overflow, and missing acknowledgements.

The ADIM-NB protocol has the lowest retransmission rate among all three protocols and shows significant improvement over CSMA and ADIM-NB/B protocols. Both CSMA and ADIM-NB MAC protocols have approximately the same performance. The ADIM-NB/B protocol's higher retransmission rate does not mean that more packets are physically retransmitted. Missing RTS or CTS messages in the ADIM-NB protocol also result in data packets being dropped before transmission. Such a drop, however, counts as one retransmission. Since the CSMA protocol does not use RTS and CTS messages, there is no such drop and all retransmissions are transmitted over the air. Hence, a retransmission in ADIM-NB/B or ADIM-NB protocols can mean that either an RTS or a data packet is retransmitted.

Consequently, a slightly better retransmission rate does not necessarily mean better performance when comparing ADIM-NB/B and CSMA protocol.

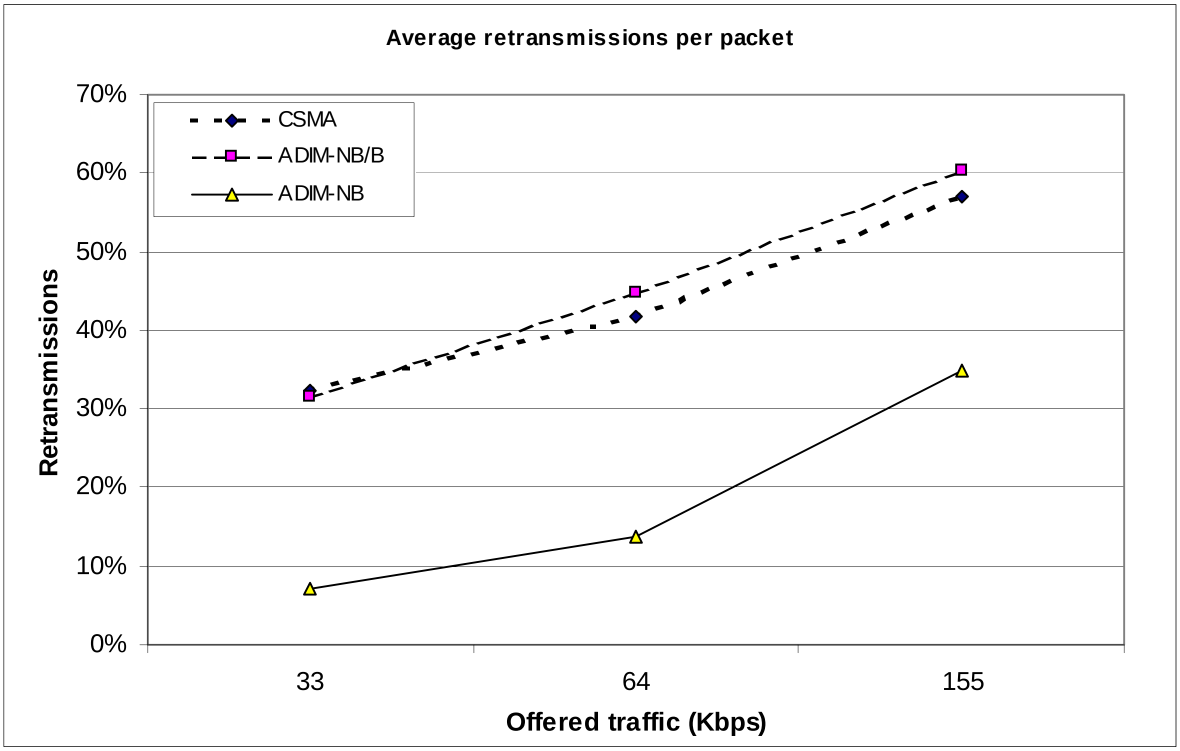


Figure 7-8. Average retransmission per packet comparison with CSMA.

As can be expected, the retransmission rate increases as the traffic increases because of the network congestion and higher collision rate. Since the retransmission rate can be relatively high in a congested scenario, resulting in an even more congested network, a congestion control mechanism to limit the network traffic is necessary in a production network.

These simulation results confirm that the ADIM-NB/B and ADIM-NB MAC protocols have a significant advantage over the CSMA MAC protocol in a moderately to heavily loaded *ad hoc* network environment. Although the larger header increases the overhead, the smaller number of collisions greatly improves network performance. Combined with the adaptive processing gain algorithm (ADIM-NB/PG), adaptive FEC coding rate algorithm (ADIM-NB/FE), adaptive power control (ADIM-NB/PC), and multiple connection setup algorithm (ADIM-NB/MU), the ADIM-NB MAC protocol significantly outperforms a non-persistent unslotted CSMA MAC protocol.

The next section focuses on the effects and contributions of each individual ADIM-NB algorithm, rather than their combined effect.

7.2 Effects from Individual ADIM-NB Algorithms

This section focuses on the effect from each individual ADIM-NB algorithm: ADIM-NB/B, ADIM-NB/PG, ADIM-NB/PC, ADIM-NB/MU, and ADIM-NB/FE. The network remains static, i.e., nodes do not change their positions during the experiment in order to minimize the effect from the routing protocol. This section also provides insight into a rapid static deployment application and the best case performance of each algorithm. A summary of the simulation results is shown in Tables 7.2 through 7.5 More information regarding each performance metric was provided in Section 7.1.

A four-factor analysis of variance (ANOVA) is used to investigate the statistical significance of the four algorithms and their interactions because they contribute significantly to four different statistics: throughput, delay, power used per bit received, and percentage of data delivered. A total of sixteen simulations corresponding to all combinations of all ADIM-NB algorithms are run in a static environment to collect the data. In each simulation, each algorithm is turned on or off independently, resulting in 2^4 simulations. An ANOVA is applied to the simulation data gathered using the method described in [Jai91]. Because it is estimated that the effect from each ADIM-NB algorithm will be multiplicative, all raw data is first transformed using a logarithmic function before the ANOVA. Tables 7.2 through 7.5 show the results of the ANOVA for the four different statistics. Different statistics are necessary because each algorithm contributes to the overall performance in a different way.

Table 7.2 shows the results of the ANOVA for the end-to-end network throughput statistic. The first column indicates the factor(s) involved in the analysis. The second column indicates the amount of influence that such factor(s) has on the throughput. The third and fourth columns represent the upper and lower limits of the 90% confidence interval. They are transformed back to the non-logarithmic domain for presentation purposes. The F-ratio in the fifth column measures how much a factor's variance or factor interaction affects the statistic in comparison with the experimental error for that factor(s). As such, a larger F-ratio indicates the factor(s) is assumed to explain a more significant fraction of the variance. The last column compares the F-ratio to the reference ninety percentiles of the $F_{[0.9;1;16]}$ value (each factor's degree of freedom = 1, experimental error's degree of freedom = 16), which is equal to 3.05. The factor or combination of factors is considered significant if and only if the F-ratio is greater than the $F_{[0.9;1;16]}$ value. An insignificant factor means that the experimental error contributes more to the variance than does the factor and the factor is not statistically significant. The estimation error in the last row indicates how reliable the ANOVA procedure is at estimating the statistical significance of each factor.

For example, the first row shows the effect solely from ADIM-NB/MU, which contributes 1.8% to the variation in throughput. The 1.8% variation is relative to the total variation when all ADIM-NB algorithms are enabled or disabled, but not relative to the baseline ADIM-NB/B MAC protocol. In this experiment, the baseline comparison when all ADIM-NB algorithms are disabled is ADIM-NB/B, which has a throughput of 68,152.67 bps. With 90% confidence, ADIM-NB/MU increases throughput by (2899.81, 3426.17), resulting in the final throughput between 71,052.48 bps to 71,578.84 bps. This confidence interval is provided by columns three and four. The F-ratio value is 361.39, which is greater than the F-variates of 3.05. Therefore, the contribution from the ADIM-NB/MU algorithm is statistically significant. In all tables, the experimental error remains small, indicating a good match between the ANOVA model assumed and the simulation results.

Table 7.2. ANOVA Algorithms Results for End-to-End Network Throughput.

Source (Throughput)	**Percentage of variation**	**90% confidence interval**		**F-ratio**	**F-ratio/ F-variates**
ADIM-NB/MU	1.833%	2899.81	3426.17	361.39	Significant
ADIM-NB/FE	0.923%	1879.70	2406.06	181.95	Significant
ADIM-NB/PC	0.059%	-778.97	-252.61	11.70	Significant
ADIM-NB/PG	96.708%	20910.44	21436.80	19071.51	Significant
MU * FE	0.024%	65.08	591.45	4.78	Significant
MU * PC	0.006%	-143.10	383.27	1.19	Not significant
MU * PG	0.145%	1266.25	1792.61	28.68	Significant
FE * PC	0.030%	10.54	536.91	5.99	Significant
FE * PG	0.005%	382.48	908.84	1.01	Not significant
PC * PG	0.000%	-348.47	177.90	0.07	Not significant
MU * FE * PC	0.121%	-963.01	-436.64	23.77	Significant
MU * FE * PG	0.033%	-542.85	-16.48	6.55	Significant
MU * PC * PG	0.001%	-308.53	217.83	0.23	Not significant
FE * PC * PG	0.020%	-494.52	31.84	4.03	Significant
MU * FE * PC * PG	0.010%	-229.70	296.66	1.91	Not significant
Error	0.081%				

Other rows can be interpreted in approximately the same way. The fifth row, for example, shows the effect of the ADIM-NB/MU algorithm in conjunction with the effect from the ADIM-NB/FE algorithm, which can also be interpreted as a correlation between the two algorithms.

From Table 7.2, most variation in throughput comes from the first order interaction, or the first four rows. The ADIM-NB/PG algorithm contributes most to the increase in throughput by increasing the end-to-end network throughput by approximately 21 Kbps, an equivalent of a 31% increase in the network capacity. Since the ADIM-NB/PG algorithm increases the transmission speed by reducing the processing gain to an absolute minimum, it is natural to provide an increase in throughput. Although distant from the ADIM-NB/PG algorithm, the ADIM-NB/MU algorithm is the second most important algorithm in terms of throughput. Since the ADIM-NB/MU algorithm depends on multiple stations sending data to a single destination at the same time, it has only a limited probability to set up multiple connections. As a result, its contribution to throughput is not that obvious. Because the ADIM-NB/FE algorithm is designed to combat jamming and dynamic interference in the network, it is surprising that it helps improve network throughput in an interference-free network as well. Its performance in a jamming environment will be investigated in the later section. Although the ADIM-NB/PC algorithm slightly degrades the end-to-end network throughput, its advantage in power reduction will later be apparent. Initially, the ADIM-NB/PC algorithm is designed to reduce the noise level in the system. From Table 7.2, such a reduction does not seem to help increase the throughput. Rather, it helps decrease the transmission power while maintaining the level of throughput when the power control algorithm is not used.

Although statistically significant, other interactions contribute little to the network throughput. They may, in fact, reduce the throughput in many combinations. As an example, MU * PG slightly increases the network throughput because the ADIM-NB/MU algorithm increases the number of concurrent connections, resulting in more opportunities for the ADIM-NB/PG algorithm to speed up those connections. Because such interactions represent a small contribution to the network throughput, they rarely occur. Therefore, the number of simulations may be too few to determine the correct cause of such an increase in throughput.

Table 7.3 shows the results of the ANOVA for power consumption in milli-Watts (mW) after 1 Mbit of user data has been received. The statistic is only concerned with the transmission power. Although the power in the receiving circuit also contributes to the overall power consumption, it is not included because there are numerous techniques in a digital communication system to reduce the received power consumption. Since none of those techniques is implemented in this research, including the receiving circuit power consumption can misrepresent the final statistic.

Table 7.3. ANOVA Algorithms Results for Power Used (mW) per Mbit Received.

Source (Power used per bit)	Percentage of variation	90% confidence interval		F-ratio	F-ratio/ F-variates
ADIM-NB/MU	0.944%	-25.467	-21.378	519.84	Significant
ADIM-NB/FE	0.932%	-25.797	-21.709	513.08	Significant
ADIM-NB/PC	13.389%	-110.590	-106.501	7372.70	Significant
ADIM-NB/PG	78.955%	-221.492	-217.404	43476.73	Significant
MU * FE	0.002%	-1.361	2.727	1.20	Not significant
MU * PC	0.008%	0.442	4.531	4.63	Significant
MU * PG	0.009%	3.926	8.014	4.71	Significant
FE * PC	0.004%	0.209	4.298	2.35	Not significant
FE * PG	0.000%	5.877	9.966	0.05	Not significant
PC * PG	5.674%	85.637	89.726	3124.36	Significant
MU * FE * PC	0.025%	1.919	6.008	13.52	Significant
MU * FE * PG	0.006%	-0.124	3.964	3.52	Significant
MU * PC * PG	0.023%	-7.780	-3.692	12.43	Significant
FE * PC * PG	0.000%	-4.820	-0.732	0.02	Not significant
MU * FE * PC * PG	0.000%	-3.450	0.638	0.02	Not significant
Error	0.029%				

The ADIM-NB/PG algorithm still contributes most to the reduction in power used, specifically 79% of the variation. Since the baseline ADIM-NB/B protocol takes 658 mW/Mbit (not shown in the table), the ADIM-NB/PG algorithm represents a 33% reduction in the power consumption. Everything else being equal, the ADIM-NB/PG algorithm uses fewer chips per bit for each transmission. This reduction in the number of chips per bit directly results in a reduction in power consumption. The ADIM-NB/PC algorithm also significantly decreases the power consumption simply because that was its design objective. The ADIM-NB/PC algorithm records the received power from the negotiation packet and determines the required minimum power to maintain the desired link quality. As a result, it rarely uses more power than necessary. Because both the ADIM-NB/MU and ADIM-NB/FE algorithms slightly increase the network throughput, such an increase also leads to the transmitted power being used more effectively, reducing the overall power consumption.

There is a significant correlation between the ADIM-NB/PC and the ADIM-NB/PG algorithms. Because the ADIM-NB/PG algorithm reduces the number of chips per bit, it also reduces the maximum possible energy per bit. Consequently, the dynamic range by which the ADIM-NB/PC algorithm can vary the transmission power is reduced considerably, resulting in a slight increase in the power consumption. Other than the ADIM-NB/PC * ADIM-NB/PG correlation, there is little correlation between other combinations of the algorithms.

Table 7.4 shows the results of the ANOVA for end-to-end delay. The statistic is only concerned with data packets, not routing packets. If a packet has to be broken into several fragments, each segment is treated as a separate packet for the purpose of delay calculation. Calculating fragment delay is advantageously independent from the traffic distribution and focuses on the delay from the MAC protocol itself.

Table 7.4. ANOVA Algorithms Results for End-to-End Delay.

Source (Delay)	**Percentage of variation**	**90% confidence interval**		**F-ratio**	**F-ratio/ F-variates**
ADIM-NB/MU	4.702%	-0.094	-0.059	75.83	Significant
ADIM-NB/FE	1.467%	-0.065	-0.030	23.65	Significant
ADIM-NB/PC	0.129%	-0.007	0.028	2.07	Not significant
ADIM-NB/PG	83.580%	-0.383	-0.347	1347.76	Significant
MU * FE	0.050%	-0.007	0.029	0.81	Not significant
MU * PC	0.182%	-0.002	0.033	2.94	Not significant
MU * PG	7.533%	-0.120	-0.085	121.48	Significant
FE * PC	0.508%	0.011	0.046	8.19	Significant
FE * PG	0.185%	-0.030	0.005	2.98	Not significant
PC * PG	0.634%	0.013	0.048	10.22	Significant
MU * FE * PC	0.004%	-0.021	0.014	0.07	Not significant
MU * FE * PG	0.009%	-0.020	0.015	0.14	Not significant
MU * PC * PG	0.006%	-0.017	0.018	0.10	Not significant
FE * PC * PG	0.001%	-0.019	0.016	0.02	Not significant
MU * FE * PC * PG	0.018%	-0.013	0.022	0.29	Not significant
Error	0.992%				

The ADIM-NB/PG algorithm has the most effect to the end-to-end delay. Reduction in an end-to-end delay value is a direct result of transmitting data packets at a higher speed. Since the ADIM-NB/B protocol has an average delay of 3.8 sec, the ADIM-NB/PG algorithm results in a 10% reduction in delay. This number is less impressive when compared to the 30% improvement in throughput and power consumption because the majority of delay comes from the RTS/CTS handshake sequence. Since the end-to-end delay is a combination of the RTS/CTS negotiation delay plus data packet transmission time, any reduction in data packet transmission time does not directly reduce the end-to-end delay at the same rate.

The ADIM-NB/MU algorithm can simultaneously set up multiple connections, so it reduces the hop-by-hop delay, resulting in reduced end-to-end delay. The ADIM-NB/FE algorithm also slightly reduces the delay, while the ADIM-NB/PC algorithm does not have any effect on the delay statistic. This further emphasizes that the ADIM-NB/PC algorithm reduces power consumption with no effect on throughput and delay in a static network.

A strong correlation between the ADIM-NB/MU and ADIM-NB/PG algorithms results from the system design. The reduction in delay indicates that the ADIM-NB/MU and ADIM-NB/PG algorithms help each other in reducing the delay. Without the ADIM-NB/MU algorithm, the ADIM-NB/PG algorithm will discard any RTS requests that have a low SNR value, which is an indication of strong interference. With the ADIM-NB/MU algorithm enabled, however, the penalty of loosing a CTS packet is not as high, and the ADIM-NB/PG algorithm will attempt to reply to all RTS requests, no matter how weak the signal is. Such behavior and tight integration across algorithms leads to an improvement in end-to-end delay.

Table 7.5 shows the results of the ANOVA for percentage of completed transmission requests, which roughly represents QoS (see Figure 7-7 for a more detailed explanation).

Table 7.5. ANOVA Algorithms Results for Percentage of Data Delivered.

Source (Percentage of data delivered)	**Percentage of variation**	**90% confidence interval**		**F-ratio**	**F-ratio/ F-variates**
ADIM-NB/MU	1.973%	1.66%	2.04%	258.48	Significant
ADIM-NB/FE	1.021%	1.08%	1.47%	133.72	Significant
ADIM-NB/PC	0.102%	-0.57%	-0.19%	13.36	Significant
ADIM-NB/PG	96.400%	11.86%	12.25%	12629.00	Significant
MU * FE	0.013%	-0.06%	0.32%	1.69	Not significant
MU * PC	0.005%	-0.13%	0.26%	0.71	Not significant
MU * PG	0.150%	0.66%	1.05%	19.59	Significant
FE * PC	0.046%	0.01%	0.40%	6.04	Significant
FE * PG	0.007%	0.18%	0.56%	0.95	Not significant
PC * PG	0.003%	-0.22%	0.17%	0.35	Not significant
MU * FE * PC	0.077%	-0.51%	-0.12%	10.11	Significant
MU * FE * PG	0.043%	-0.40%	-0.01%	5.60	Significant
MU * PC * PG	0.002%	-0.23%	0.16%	0.22	Not significant
FE * PC * PG	0.028%	-0.35%	0.04%	3.62	Significant
MU * FE * PC * PG	0.009%	-0.15%	0.23%	1.14	Not significant
Error	0.122%				

The ADIM-NB/PG algorithm provides the most improvement to the percentage of data delivered. Since the ADIM-NB/B protocol can satisfy 57% of its requests, this translates into a 21% improvement over the ADIM-NB/B MAC protocol. Although the ADIM-NB/PG algorithm provides a 31% increase in end-to-end throughput, such an improvement does not directly translate to the same increase for percentage of data delivered because the average hop count is different depending on whether the ADIM-NB/PG algorithm is enabled or disabled (see the explanation in Section 7.1).

Although not equivalent, there is a strong resemblance, in terms of variation percentage, between throughput and percentage of data delivered. Both the ADIM-NB/MU and ADIM-NB/FE algorithms slightly improve the performance while the ADIM-NB/PC algorithm has little impact. There is also little correlation between each ADIM-NB algorithm regarding data delivered.

7.3 Special Scenario – Jammer

As discovered earlier, implementation of the ADIM-NB/FE algorithm in an interference-free environment has little effect on the performance. This section focuses on the ADIM-NB/FE algorithm performance in a hostile situation. In this experiment, there is one jamming node moving counter clockwise around the network, as shown in Figure 7-9. The jammer continuously sends a jamming signal, which is between 16 to 256 times as strong as the transmission from a regular node. The network performance is compared among several FEC options summarized in Table 7.6. The experimental objective is to assess a performance improvement from the ADIM-NB/FE algorithm in the presence of dynamic interference. All other ADIM-NB algorithms are enabled and the network remains static. Because the interference at any specific node varies with time, a capability to adapt the operating parameters to accommodate the current environment is crucial to maintain the communication channel. Such adaptability is not only important for data packets, but also for control packets. Previous study [Esc95] shows that it is ineffective to use a small FEC coding rate in a mobile *ad hoc* network; hence, three large FEC rates [Skl88] are chosen as a comparison to the ADIM-NB/FE algorithm.

There are two options within the ADIM-NB/FE algorithm, depending on the radio capability. The first option relies on exact BER information in the received packet. The ADIM-NB/FE algorithm uses such information to formulate and maintain an estimated BER value, which will be used to determine the FEC coding rate for outgoing packets. The second option uses a simpler assumption and just requires the receiver to be able to categorize the packet BER into three categories: lower than the low threshold, higher than the high threshold, or between the low and high thresholds. The ADIM-NB/FE algorithm can use that limited information to effectively adapt to the current environment.

Figure 7-10 shows the end-to-end network throughput for user data. Both ADIM-NB/FE algorithm alternatives seem to have the best performance when the interference power is the lowest. The worst performance seems to occur when the algorithm fixes the FEC rate at three-fourths. Otherwise, each FEC alternative seems to have approximately the same throughput. To verify this assumption, additional analysis is performed to compare each alternative quantitatively, as shown in Table 7.7. Since each experiment is repeated in seven batches, the variations in each repetition can be used to generate a confidence interval for each alternative.

Table 7.6. FEC Algorithms Descriptions.

Legends	Description
Approx. BER	The ADIM-NB/FE algorithm does not require exact BER information to operate, but relies on low and high thresholds
Exact BER	The ADIM-NB/FE algorithm in which exact BER information is available
Fixed 3/4	Fixed FEC coding rate at 3/4
Fixed 7/8	Fixed FEC coding rate at 7/8
Fixed 15/16	Fixed FEC coding rate at 15/16

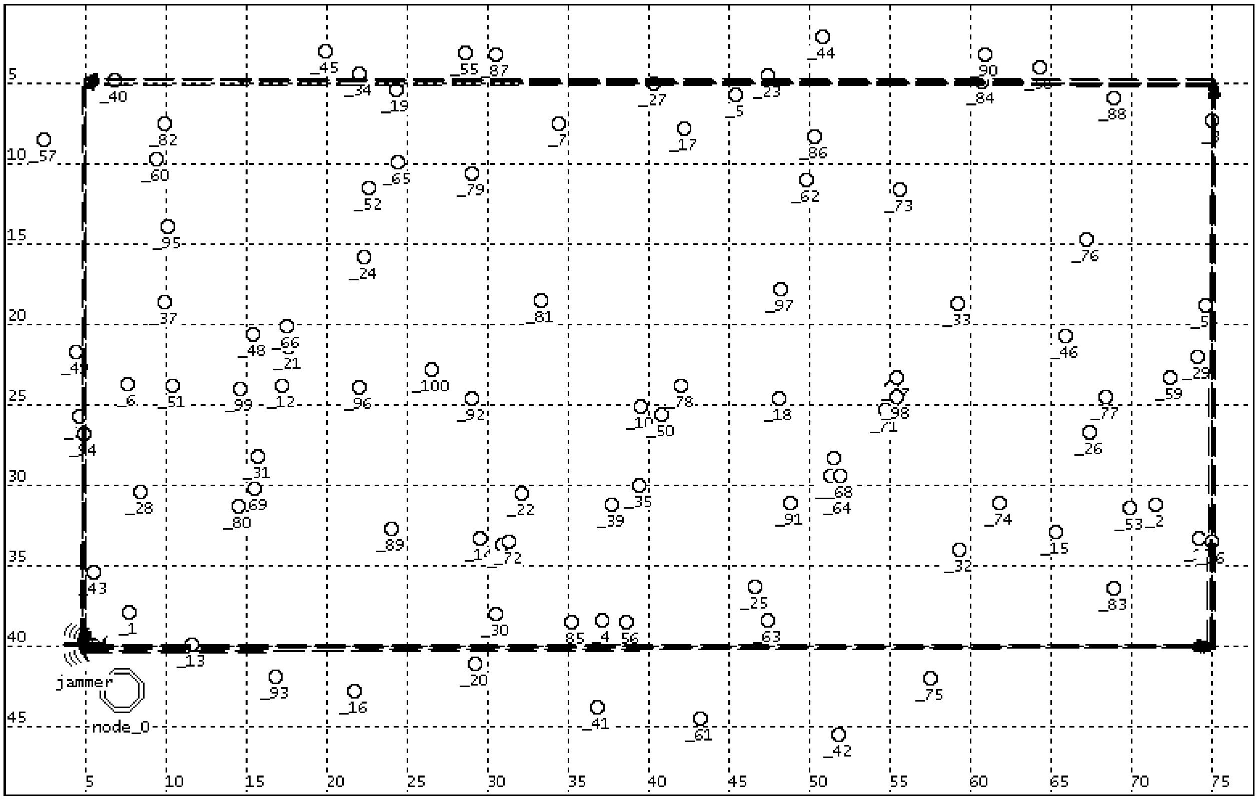


Figure 7-9. Network topology in the presence of jammer.

Table 7.7 shows a 95% confidence interval for the end-to-end throughput statistic. Two experiments are considered statistically indifferent if they have overlapping confidence intervals. Table 7.7 also indicates which experiments are statistically equivalent. For example, if the interference power is sixteen times stronger than normal transmit power, the network throughput is statistically equivalent when the ADIM-NB protocol operates with 7/8 FEC, 15/16 FEC, or the ADIM-NB/FE with exact BER algorithm.

From Table 7.7, the ADIM-NB protocol using fixed 3/4 FEC has the lowest throughput in all environments. The ADIM-NB/FE with approximate BER algorithm has the highest throughput when the interference power is equal to 16x (sixteen times the maximum transmitted power of a regular node). Otherwise, most alternatives have statistically the same throughput.

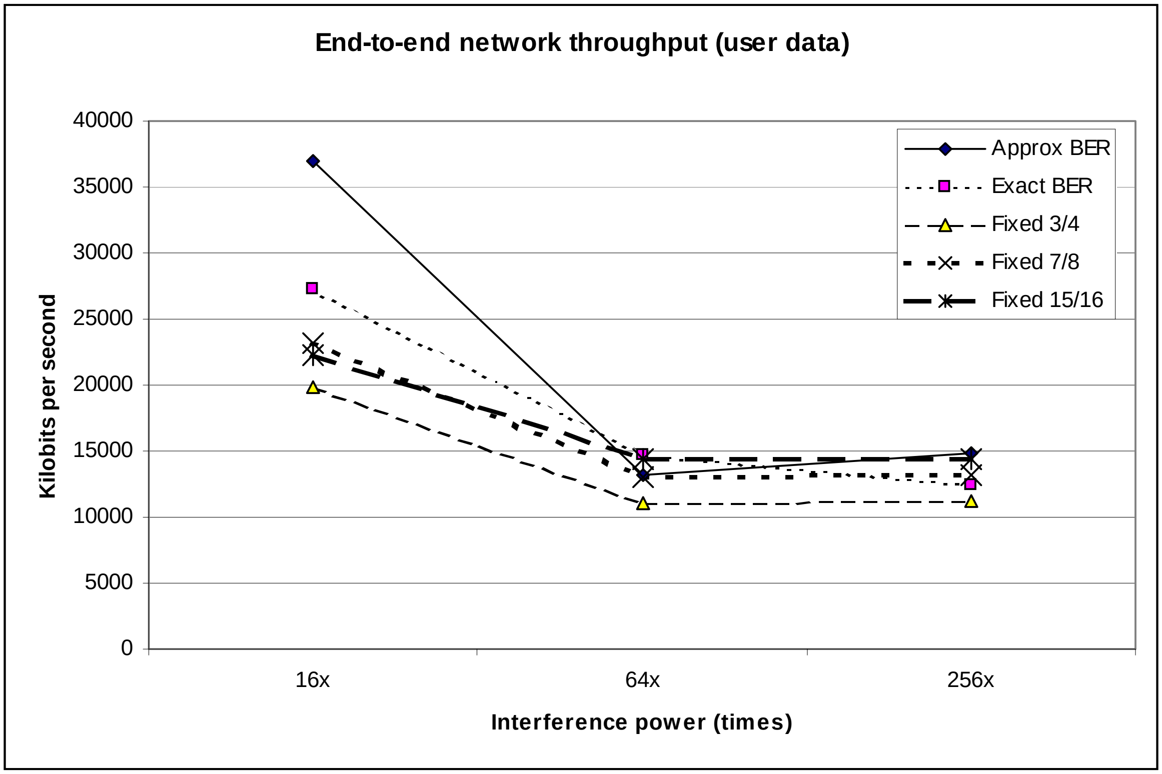


Figure 7-10. End-to-end network throughput in the presence of jammer.

Table 7.7. End-to-end Network Throughput in the Presence of a Jammer.

(Bits/second)	Approx BER	Exact BER	Fixed 3/4	Fixed 7/8	Fixed 15/16
16x		*SAME*		*SAME*	*SAME*
Confidence interval (low)	32355.606	23332.293	18323.679	21635.461	20799.493
Confidence interval (high)	41566.978	31066.014	21350.034	24703.133	23831.364
64x	*SAME*	*SAME*		*SAME*	*SAME*
Confidence interval (low)	12515.955	13346.946	10079.046	12413.148	12779.685
Confidence interval (high)	13742.418	15935.837	12062.450	13572.570	15951.716
256x	*SAME*			*SAME*	*SAME*
Confidence interval (low)	13720.697	11316.770	10297.192	12361.742	13761.477
Confidence interval (high)	15891.595	13461.021	12117.492	14012.832	14898.008

In a hostile environment, where the interference power is relatively high, any transmission attempts have a high probability of failure. The best solution, in terms of throughput, would be to defer the transmission until the channel is clear. Using a low FEC rate might be impractical and inefficient in terms of network throughput, as indicated by the relative performance of the ADIM-NB/FE algorithm and 3/4 fixed FEC alternative.

In a less hostile environment, where the interference power is relatively low, the ADIM-NB/FE algorithm with approximate BER option has the best performance because it can adjust its FEC rate dynamically, according to the environment. The ADIM-NB/FE algorithm with an exact BER option does not perform as well because it tries to make sense of information that should not be used in an exact sense. Because the interference power changes rapidly, depending on how many stations are transmitting, the relative distances of those stations, their transmit power, and any interferers in the network, the ADIM-NB/FE algorithm has to be able to distinguish among all the possible causes and adjust its parameters accordingly. Instead of using a complex algorithm to track such information, using the BER information in an approximate sense is simpler and more effective.

Figure 7-11 shows the end-to-end delay and Table 7.8 shows the corresponding 95% confidence interval. The ADIM-NB/FE algorithm with the approximate BER option has the lowest delay in two out of three scenarios. Although it is not superior in all environments, it improves, rather than degrades, the delay performance in the majority of the test scenarios. Unlike the throughput, the ADIM-NB/FE algorithm with the exact BER option performs poorly in most scenarios. Its increased delay may simply be the result of being able to deliver more traffic to the destination. The fixed 15/16 alternative has the lowest delay because it has the least amount of overhead and the penalty for retransmission is lower than the penalty for using a lower FEC coding rate.

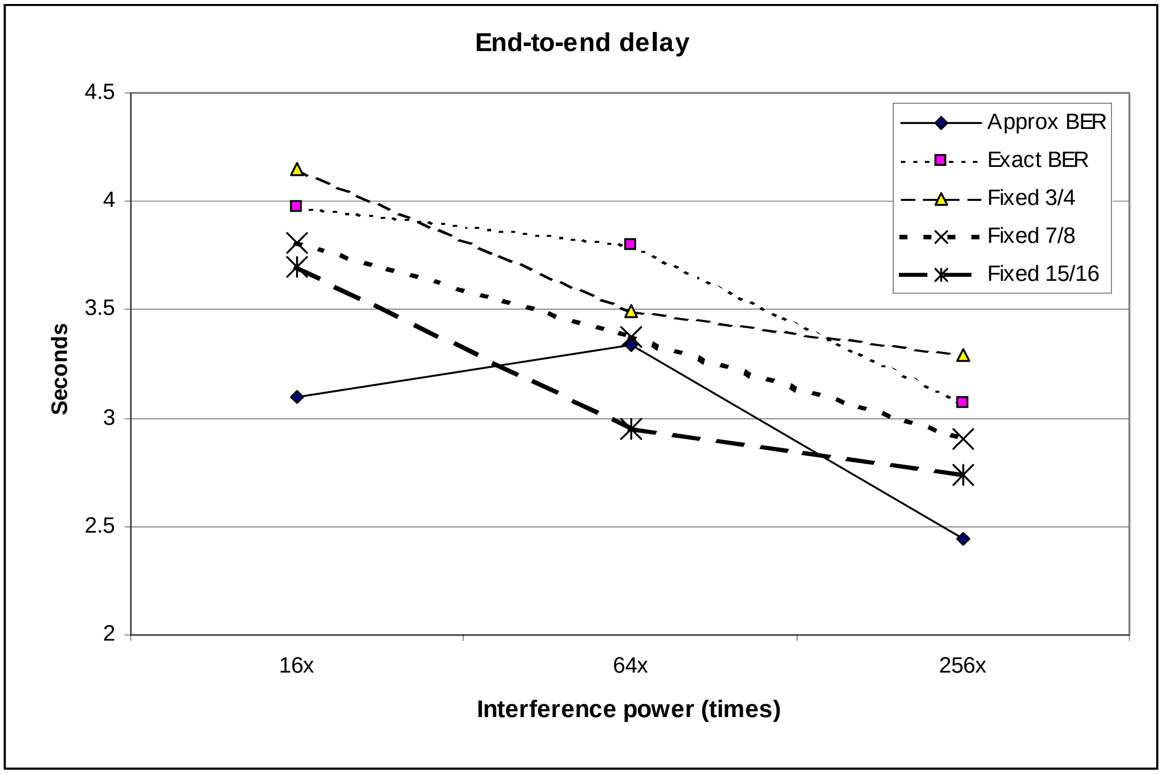


Figure 7-11. End-to-end delay in the presence of a jammer.

Table 7.8. End-to-End Delay in the Presence of a Jammer.

(Seconds)	**Approx BER**	**Exact BER**	**Fixed 3/4**	**Fixed 7/8**	**Fixed 15/16**
16x		*SAME*		*SAME*	*SAME*
Confidence interval (low)	2.674	3.803	4.012	3.634	3.557
Confidence interval (high)	3.518	4.145	4.292	3.988	3.841
64x	*SAME*		*SAME*	*SAME*	
Confidence interval (low)	3.231	3.717	3.407	3.271	2.707
Confidence interval (high)	3.442	3.872	3.575	3.485	3.193
256x				*SAME*	*SAME*
Confidence interval (low)	2.247	3.014	3.157	2.710	2.622
Confidence interval (high)	2.647	3.130	3.430	3.093	2.851

Figure 7-12 shows the percentage of data delivered and Table 7.9 shows the corresponding 95% confidence interval. Both sets of results exhibit the same trends and characteristics as the end-to-end network throughput statistic. The ADIM-NB/FE algorithm with the approximate BER option delivers the most traffic in all three scenarios, although some other alternatives have the same performance in two out of three scenarios. The fixed 3/4 FEC alternative has the worst performance in terms of the percentage of data delivered, which is also consistent with the throughput analysis.

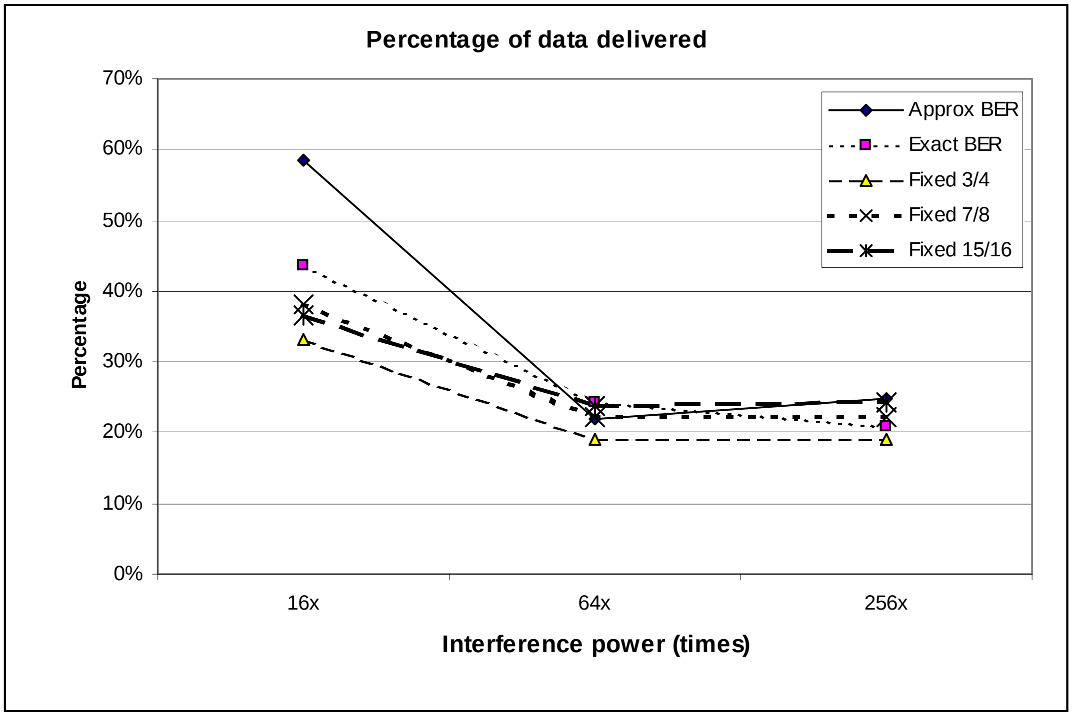


Figure 7-12. Percentage of data delivered in the presence of a jammer.

Table 7.9. Percentage of Data Delivered in the Presence of a Jammer.

	Approx BER	**Exact BER**	**Fixed 3/4**	**Fixed 7/8**	**Fixed 15/16**
16x		*SAME*		*SAME*	*SAME*
Confidence interval (low)	51.56%	37.93%	30.90%	35.77%	33.90%
Confidence interval (high)	64.48%	49.04%	35.23%	39.94%	38.80%
64x	*SAME*	*SAME*		*SAME*	*SAME*
Confidence interval (low)	21.38%	22.70%	17.62%	21.55%	21.67%
Confidence interval (high)	23.65%	26.45%	20.66%	23.07%	26.32%
256x	*SAME*			*SAME*	*SAME*
Confidence interval (low)	23.45%	19.71%	18.06%	21.26%	23.58%
Confidence interval (high)	27.08%	23.27%	20.99%	24.09%	25.61%

Figure 7-13 shows the power used per bit received and Table 7.10 shows the corresponding 95% confidence interval. All results indicate that the ADIM-NB/FE algorithm with the approximate BER option has the best performance in two out of three scenarios, implying that the algorithm can still be improved. The ADIM-NB/FE algorithm with the exact BER option has statistically equivalent performance as fixing the FEC coding rate at 15/16 or 7/8. Fixing the FEC coding rate at 3/4 is not suitable for an *ad hoc* network environment, because it has the lowest performance in most of the experiments.

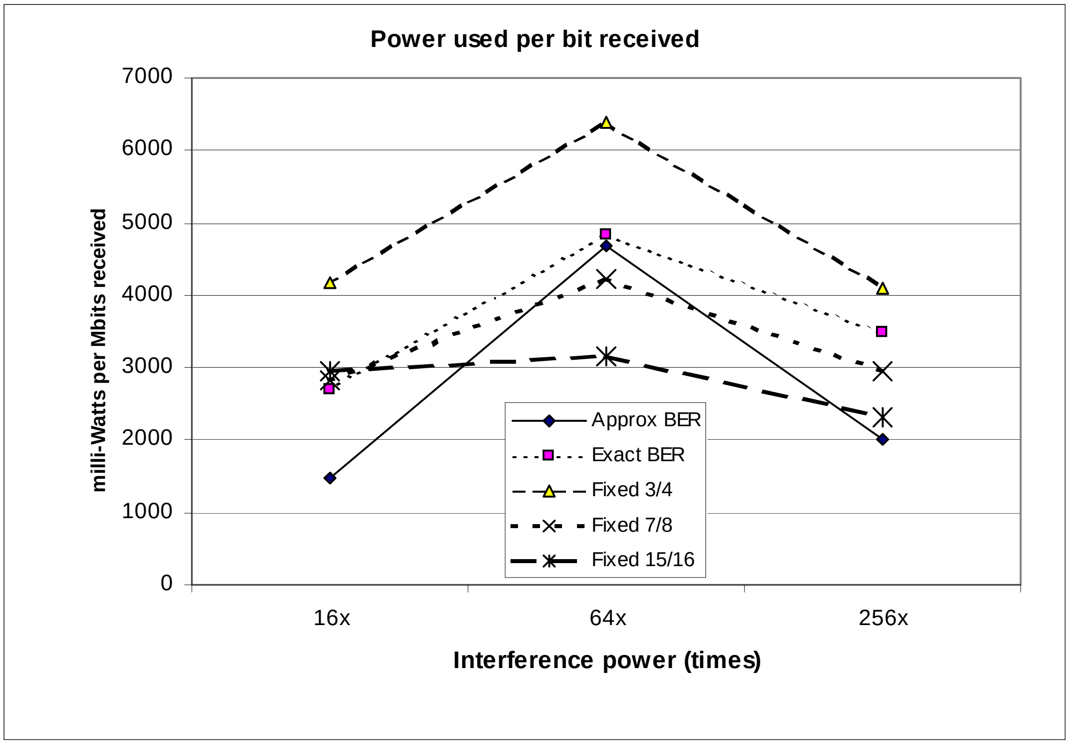


Figure 7-13. Power used per bit received in the presence of a jammer.

Table 7.10. Power Used per Bit Received in the Presence of a Jammer.

(mW/Mbit)	Approx BER	Exact BER	Fixed 3/4	Fixed 7/8	Fixed 15/16
16x		*SAME*		*SAME*	*SAME*
Confidence interval (low)	909.348	2232.944	3850.166	2563.545	2618.122
Confidence interval (high)	2055.645	3179.359	4495.875	3107.442	3271.843
64x	*SAME*	*SAME*		*SAME*	
Confidence interval (low)	4320.922	4357.144	5478.277	3844.115	2534.377
Confidence interval (high)	5057.226	5328.928	7283.937	4581.532	3774.379
256x		*SAME*		*SAME*	
Confidence interval (low)	1704.071	2938.323	3497.170	2514.432	2085.889
Confidence interval (high)	2317.381	4013.578	4700.830	3388.834	2537.601

There is a peak power consumption when the interference power is equal to sixty-four times the transmit power. However, such a peak does not exist in other statistics. This may represent an unstable point in the network. As the interference power increases, each node increases its transmission power to a certain point, where it can no longer combat the jammer. If the interference keeps increasing, such links will eventually become unusable and no more transmission attempts will take place. As a result, the power consumption goes down to a normal level.

Figure 7-14 shows the percentage of routing table coverage for all FEC options. It presents another aspect of the performance measurement. Although the fixed 3/4 FEC option has the lowest efficiency for throughput, delay, and power consumption, it has the best network connectivity. This should not be unexpected because it always uses heavy FEC coding, improving the link quality. The most efficient FEC alternative, the ADIM-NB/FE algorithm with the approximate BER algorithm, has moderate performance in terms of connectivity. Depending on the system objective, each FEC alternative has its own advantage. There is, also, an opportunity to improve the ADIM-NB/FE algorithm to combine the advantage of each FEC option in various environment conditions, resulting in a highly adaptive FEC algorithm.

7.4 Effects from Environment and Traffic Characteristics

This section studies the performance of the ADIM-NB MAC protocol in several different environments, as summarized in Table 7.11. Since the effect of each individual ADIM-NB algorithm has been studied, all individual ADIM-NB algorithms are enabled. In the next section, additional experiments to evaluate the effect of each ADIM-NB algorithm in each type of environment is investigated.

7.4.1 Effects From Environment – 2^k Factorial Experiment

To evaluate the correlation and the joint effect of each environment factor, a 2^k factorial experimentation technique [Jai91] is used. Each factor has two possible levels as described in Table 7.12. The simulation design consists of all possible combinations of factors, resulting in 2^6 or 64 experiments. Table 7.12 also includes the legends used in the subsequent table. The results are analyzed, using the same ANOVA technique as in Section 7.2, to investigate the statistical significance of the six environmental factors and their interactions as they significantly contribute to four different statistics: throughput, delay, power used per bit received, and percentage of data delivered. Because it is estimated that the effect from each environmental factor will be multiplicative, all raw data is first transformed using a logarithmic function.

Figure 7-15 shows raw data for the end-to-end network throughput (user data) statistic. Each bar represents an arithmetic mean of three to seven simulation batches. The total sixty-four possible combinations result in sixty-four bars in the figure. The overall graph can be divided into four quadrants. Each quadrant shares the same offered load level and application traffic configuration because both factors represent the majority of the variation in the statistic. The same presentation technique is subsequently applied to each quadrant to group related bars together.

From Figure 7-15, it is clear that more offered traffic load results in a higher throughput, and different traffic characteristics result in different network capacity. Other factors have more effect in a heavily loaded network than in a lightly loaded network. The detailed analysis of the impact from each factor is presented in Tables 7.13 through 7.16. They show the results of the ANOVA for four different statistics, which are necessary because each environment factor contributes to the overall performance in a different way.

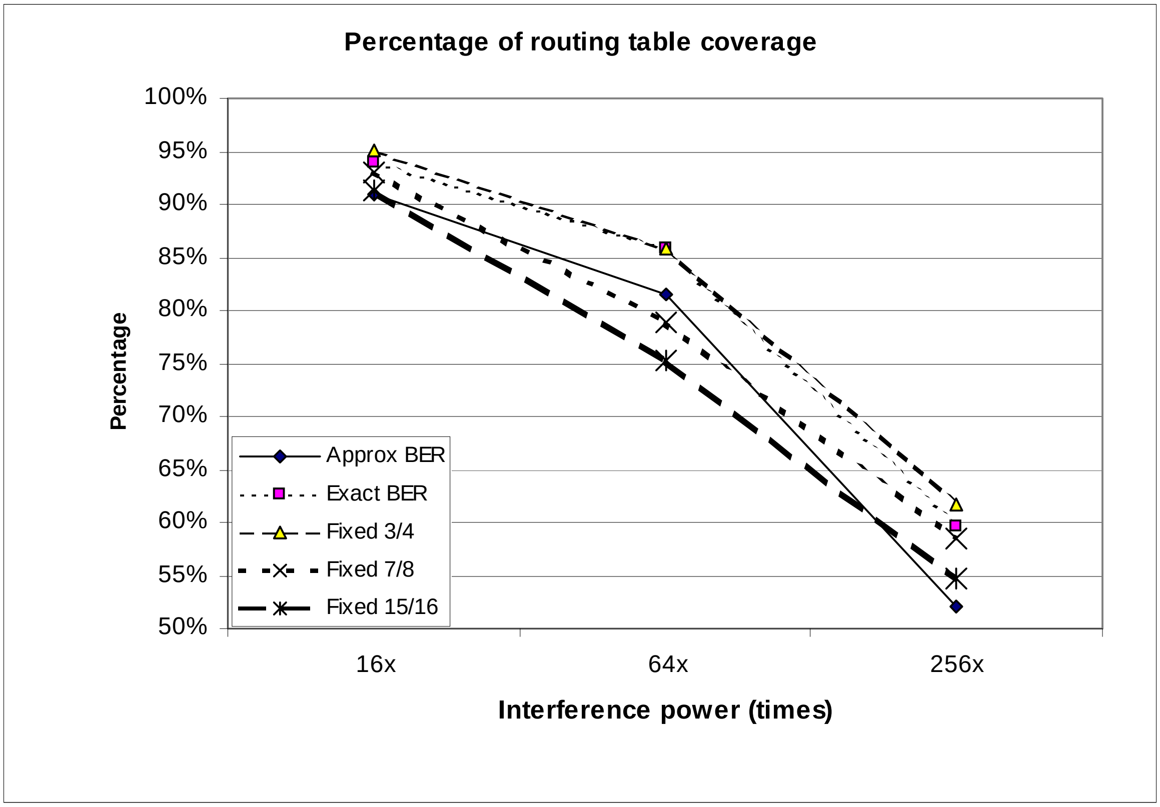


Figure 7-14. Percentage of routing table coverage in the presence of a jammer.

Table 7.11. Descriptions of Environmental Factors.

Factors	Description
Offered load	The offered load is the offered traffic to the overall network. This includes user data, all headers inside the IP packet, and the IP packet header. It does not include any MAC- or physical-layer header. Since the network capacity cannot be easily determined, it is difficult to normalize the offered load to a certain maximum value. As a result, the percentage of successful requests is used as a normalized offered load. The offered load is specified in terms of this percentage. Since such a value also changes from experiment to experiment, it is an approximation.
Traffic type	Traffic characteristics include average packet size, packet size distribution, packet inter-arrival time distribution, request/reply information, destinations selection, etc.
Node speed	This is usually specified in terms of the percentage of nodes moving at 0, 5, 20, and 45 Kph. In each speed level, the actual speed is a normal distribution with speed level as a mean value and 30% of the mean value as a variance. The minimum node speed is 0 Kph. For example, (10, 20, 30, 40) means that 20% of all nodes move at a random speed according to a uniform distribution with mean of 5 and variance of 5*0.3 = 1.5. Once initialized, a node will always move at the same speed until the end of the simulation.
Movement pattern	The movement pattern determines the direction in which each node moves. Two patterns are considered. Constant direction refers to a simple physical movement model. Each node moves in a constant direction until it hits the network boundary, in which it bounces off according to a physics law and continues in a new direction. The city block movement pattern models movement in the Manhattan grid [MLA98]. The directions are limited to North, East, West, and South. Each node constantly makes a turn decision (left, right, or straight) at the end of each block.
Degree of node	The average degree of each node, or the average number of neighboring nodes, determining how well the network is connected.
Radio type	The radio type determines the radio characteristics, including the raw bit rate, number of chips per seconds, the possible values of processing gain, etc.

Table 7.12. Levels Used in the Full Factorial Experimentation.

Factors	Levels
Offered load	95% (**L9**) and 66% (**L6**) of the requested traffic get delivered; in a static network with SMTP, 95% is approximately equivalent to 68 Kbps of offered traffic, while 66% is approximately equivalent to 163 Kbps of offered traffic.
Traffic type	SMTP (**Ts**) [Pax94] and HTTP (**Th**) [CBC95] traffic
Node speed	**S4** (0, 100, 0, 0) and **S2** (0, 33, 33, 33)
Movement pattern	Constant direction (**Mc**) and city block (**Mb**)
Degree of node	12 (**N12**) and 18 (**N18**)
Radio type	The Virginia Tech reconfigurable software radio (**Rv**) [SHA98] and CDMA2000 (**Rc**) with ITU-R/IMT-2000/IMT-MC recommendation (using radio configuration 6 with 3.6864 Mchips per second)

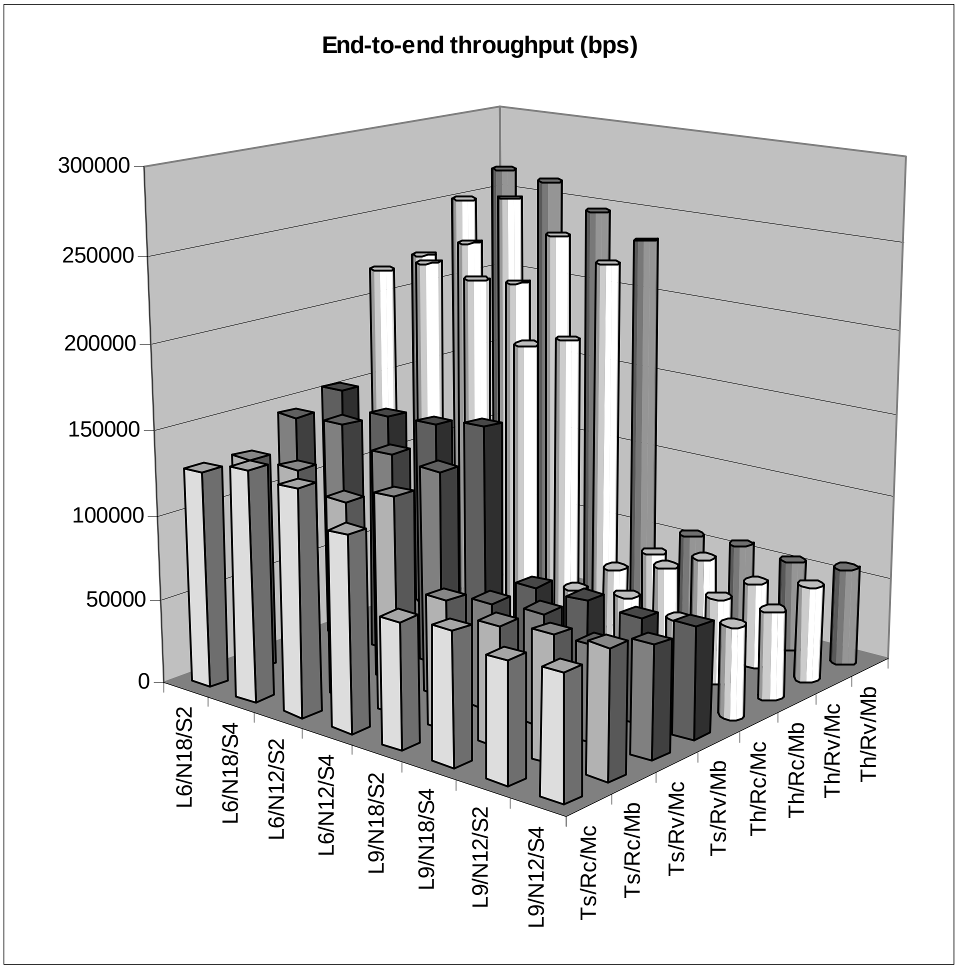


Figure 7-15. Network throughput data in various environment factors.

Table 7.13 presents the ANOVA results for the end-to-end network throughput (user data) statistic. All first order influence, second order interactions, and third order interactions are included. Only significant fourth order interactions are included. It is natural to see that higher offered traffic leads to a higher network throughput because they are directly related. From the first row, the differences in levels of offered load explains over 85% of the variation in throughput. The 90% offered traffic level (L9) results in approximately 82% lower throughput than the 66% offered traffic level (L6).

The node speed has little impact on the end-to-end throughput. A higher node speed (S2 at 23 Kph) results in an approximately 2% reduction in throughput when compared to a lower node speed (S4 at 5 Kph). Network density also has a small impact on the end-to-end network throughput. When the average degree of each node increases from twelve to eighteen, throughput increases by approximately 6%. The traffic type has some impact on the throughput because part of the traffic characteristic specifies the average hop that a packet needs to traverse. A lower average hop, or a closer destination, directly results in a less congested network and more network capacity.

Table 7.13. ANOVA Environment Results for End-to-End Network Throughput.

Source (Throughput)	Percentage of variation	90% confidence interval		F-ratio	F-ratio/ F-variates
Load (L6/L9)	85.10%	-82.63%	-82.07%	138078.5	Significant
Speed (S4/S2)	0.02%	-2.17%	-1.23%	35.7	Significant
Density (N12/N18)	0.22%	5.02%	6.00%	362.7	Significant
Traffic (Ts/Th)	1.90%	15.99%	17.02%	3083.6	Significant
Movement (Mb/Mc)	0.03%	-2.59%	-1.65%	55.8	Significant
Radio (Rv/Rc)	0.16%	-4.97%	-4.04%	254.7	Significant
L * S	0.11%	-4.32%	-3.39%	185.9	Significant
L * D	0.03%	-2.59%	-1.64%	55.5	Significant
L * T	10.70%	-34.70%	-33.91%	17363.2	Significant
L * M	0.00%	-0.78%	0.17%	1.1	Not significant
L * R	0.65%	8.97%	9.97%	1049.0	Significant
S * D	0.01%	-1.68%	-0.73%	18.0	Significant
S * T	0.00%	-0.71%	0.24%	0.7	Not significant
S * M	0.00%	-1.20%	-0.25%	6.5	Significant
S * R	0.00%	0.27%	1.23%	6.8	Significant
D * T	0.02%	1.11%	2.07%	30.9	Significant
D * M	0.01%	0.49%	1.45%	11.4	Significant
D * R	0.01%	0.42%	1.38%	9.8	Significant
T * M	0.01%	-1.33%	-0.38%	9.1	Significant
T * R	0.18%	-5.35%	-4.42%	300.0	Significant
M * R	0.01%	0.82%	1.78%	20.5	Significant
L * S * D	0.04%	1.94%	2.91%	71.2	Significant
L * S * T	0.03%	-2.40%	-1.46%	46.1	Significant
L * S * M	0.02%	-2.14%	-1.20%	34.5	Significant
L * S * R	0.00%	-0.14%	0.81%	1.4	Not significant
L * D * T	0.00%	-1.22%	-0.27%	6.8	Significant
L * D * M	0.00%	-0.11%	0.84%	1.6	Not significant
L * D * R	0.00%	-0.42%	0.53%	0.0	Not significant
L * T * M	0.01%	-1.35%	-0.40%	9.5	Significant
L * T * R	0.15%	-4.88%	-3.95%	244.7	Significant
L * M * R	0.02%	-2.24%	-1.29%	38.6	Significant
S * D * T	0.04%	-2.84%	-1.90%	69.7	Significant
S * D * M	0.01%	-1.62%	-0.67%	16.1	Significant
S * D * R	0.04%	-2.75%	-1.81%	64.6	Significant
S * T * M	0.00%	-0.50%	0.45%	0.0	Not significant
S * T * R	0.00%	-0.90%	0.06%	2.2	Not significant
S * M * R	0.01%	0.55%	1.51%	13.0	Significant
D * T * M	0.03%	-2.58%	-1.64%	55.1	Significant
D * T * R	0.00%	0.32%	1.28%	7.9	Significant
D * M * R	0.01%	-1.68%	-0.73%	17.9	Significant
T * M * R	0.01%	-1.41%	-0.46%	10.7	Significant
L * T * S * D	0.00%	-0.08%	0.88%	1.9	Not significant
L * T * S * M	0.01%	0.52%	1.48%	12.1	Significant
L * T * S * R	0.01%	-1.81%	-0.87%	22.2	Significant
L * T * D * M	0.00%	-0.79%	0.16%	1.2	Not significant
L * T * D * R	0.01%	0.35%	1.31%	8.3	Significant
L * T * M * R	0.00%	-0.41%	0.55%	0.1	Not significant
Error	0.24%				

Different packet size also has some impact on the amount of control overhead, leading to a change in network throughput. From this experiment, the HTTP traffic characteristic helps increase the end-to-end network throughput by approximately 17% over the SMTP traffic. As with the average node speed, node movement pattern has little impact on the end-to-end throughput. The constant direction movement pattern degrades the network throughput because it has a greater tendency to spread the network apart, while the city block movement pattern allows nodes to turn, giving them more opportunity to get back to their original neighbors.

Two radio types are considered in the comparison, the Virginia Tech reconfigurable software radio and a CDMA2000 radio. The former uses 1 Mchips/bit channel and the latter uses 3.6864 Mchips/bit channel. For side-by-side comparison, the end-to-end network throughput data for the CDMA2000 radio is divided by 3.6864, which makes it equivalent to the network throughput per 1 Mchip/sec channel. Surprisingly, a faster radio results in an approximately 4% degradation in network throughput. This behavior can be attributed to the fact that most timers and operating parameters of the ADIM-NB protocol are optimized for the slower radio. Nevertheless, its small percentage of variation indicates that the end-to-end network throughput statistic slightly depends on the radio data rate.

There is a large interaction between the amount of offered load and the traffic type, as shown in the L * T entry. This can be attributed to the design of the experiment. Because both traffic types significantly differ in their characteristics, the actual amount of offered load is different. For the same 66% offered traffic level (L6), the offered traffic is equal to 168 Kbps of SMTP traffic or 368 Kbps of HTTP traffic. In other words, the network that carries only HTTP traffic can service approximately 66% of its transmission requests when the offered traffic load is equal to 368 Kbps, and the network that carries only SMTP traffic can service 66% of its requests when the request is equal to 168 Kbps. Such tight integration between these two environmental factors leads to a large interaction between them.

Although other interactions exist and are statistically significant, they only contribute to a small portion of the variation in network throughput. Because the unexplained percentage of variation, or error, is only 0.24%, the model assumed in the ANOVA analysis is a good match to the simulation results. In summary, only the offered traffic load levels, types, and their interaction have a large impact on the end-to-end network throughput statistic.

Figure 7-16 visually summarizes Table 7.13. The single point on the left represents the worst case scenario when each environment factor is on the left-hand side of the legends (speed = S2, movement = city block, radio = CDMA2000, node degree = 12, traffic type = SMTP, and load offered = 95%). Each environmental factor is arranged in the reverse order of importance to aid the presentation. The network throughput is increased from 43 Kbits/sec to 126 Kbits/sec when the offered load changes from 95% to 66%, as seen in the bottom curve. In addition to using a 66% offered traffic level, if the traffic type is also changed from SMTP to HTTP, the network can deliver 89 Kbits/sec more traffic, resulting in the total network throughput of 215 Kbits/sec. Changing the node degree from twelve to eighteen additionally increases the network throughput to 227 Kbits/sec. Other environmental factors also impact the network throughput in a similar way. The top curve shows the end-to-end network throughput when all the factors are on the right-hand side of the legend.

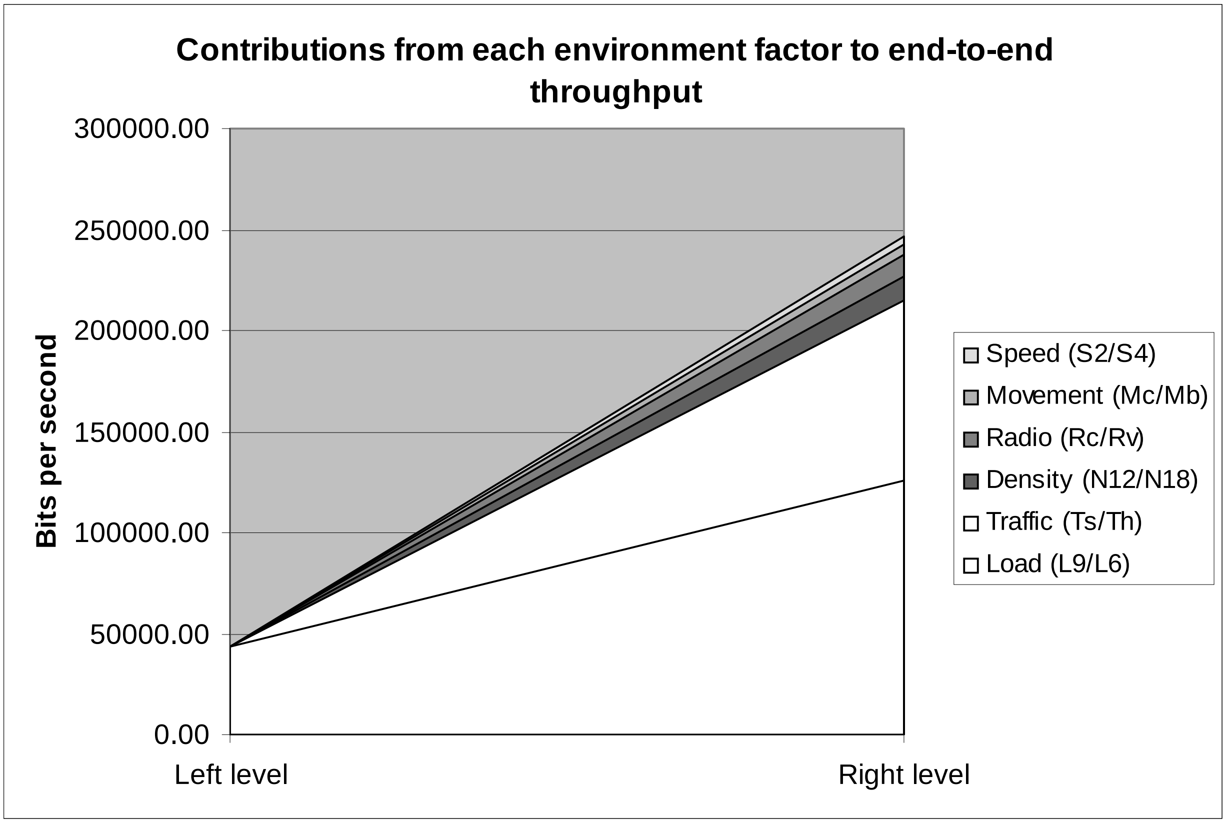


Figure 7-16. Visual representation of ANOVA results for the end-to-end network throughput.

The significance of each factor can be determined from the width of the area beneath each curve. The impact from other factors seems small when compared to the impact from the offered load and traffic type. Network mobility-related factors have the smallest impact on the throughput.

Table 7.14 presents the ANOVA results for the percentage of data delivered statistic. All interactions that result in a percentage of variation smaller than 0.1% are not shown to keep the table compact. In addition, those interactions only explain a small percentage of the variation in the statistic.

The offered load no longer causes almost all variation in the percentage of data delivered. It now explains approximately half of the variation. Other factors, such as traffic type and radio type come in to play. An increase in node speed results in a reduction in the percentage of data delivered, as can be expected. Faster node movement causes the network to be disconnected and more traffic gets dropped. Naturally, a higher network density leads to more data being delivered because the network is better connected and the data packet needs to traverse fewer hops.

A higher throughput does not result in a higher percentage of data delivered for different types of traffic. Although the HTTP traffic helps increase the network throughput, it also reduces the percentage of data delivered. This may simply mean that the network throughput statistic does not increase as fast as the offered traffic, resulting in a negative trend in the percentage of data delivered statistic. It can also be attributed to the amount of local and remote traffic. Throughput can remain relatively constant while the percentage of data delivered is reduced because more local traffic can be delivered while the remote traffic cannot. Additional degradation may come from an invalid routing table. To determine whether the destination is remote or local, each source looks into its routing table. If the routing table does not contain up-to-date information, it can wrongly assume that some remote destinations are local. Such behavior can cause data packets to travel at a greater distance.

Table 7.14. ANOVA Environment Results for Percentage of Data Delivered.

Source (percentage of data delivered)	Percentage of variation	90% confidence interval		F-ratio	F-ratio/ F-variates
Load (L6/L9)	45.80%	31.00%	31.85%	17280.7	Significant
Speed (S4/S2)	0.16%	-2.06%	-1.33%	58.8	Significant
Density (N12/N18)	1.69%	5.30%	6.06%	636.4	Significant
Traffic (Ts/Th)	23.06%	-20.01%	-19.34%	8700.0	Significant
Movement (Mb/Mc)	0.21%	-2.34%	-1.60%	79.5	Significant
Radio (Rv/Rc)	10.84%	-14.05%	-13.36%	4088.2	Significant
L * S	0.66%	-3.84%	-3.11%	248.8	Significant
L * D	0.18%	-2.20%	-1.47%	69.0	Significant
L * T	6.47%	10.88%	11.67%	2440.1	Significant
L * R	6.29%	10.72%	11.51%	2373.8	Significant
D * T	0.23%	1.69%	2.44%	85.4	Significant
T * R	0.70%	-3.93%	-3.20%	262.3	Significant
M * R	0.13%	1.16%	1.91%	47.4	Significant
L * S * D	0.31%	2.05%	2.80%	117.5	Significant
L * S * T	0.15%	-2.03%	-1.29%	56.5	Significant
L * M * R	0.12%	-1.86%	-1.12%	45.5	Significant
S * D * T	0.20%	-2.31%	-1.57%	77.1	Significant
S * D * R	0.29%	-2.69%	-1.96%	110.9	Significant
D * T * M	0.19%	-2.23%	-1.50%	71.2	Significant
Error	1.02%				

Movement pattern still explains only a small variation in the statistic. As with the result from Table 7.13, a constant movement pattern tends to cause more stress to the network and result in a lower percentage of data delivered. Surprisingly, a faster radio results in a lower percentage of data being delivered. One of the reasons is that the Virginia Tech radio may have a better parameter optimization while the CDMA2000 radio may not. Additionally, path loss is higher with increased frequency and bandwidth. Finally, because the transmission range can be as far as 12 Km, the effect of propagation delay is more severe with a higher transmission speed.

The reason for a strong correlation between the offered load and traffic type comes from the design of the experiment, as explained in analysis shown with Table 7.13. Another strong correlation between factors is between the offered loads and radio types. The faster radio, CDMA2000, performs approximately 11% better when operated with a smaller offered load.

Since the amount of routing information is approximately constant in a lightly loaded network, regardless of the radio type, the percentage of routing information to the amount of data traffic is smaller for the CDMA2000 radio. Such a small overhead leads to a better network performance, as witnessed by the percentage of data delivered statistic.

Once again, since the statistics are normalized to the chip rate of 1 Mchip/sec, this analysis does not imply that a faster radio has a lower performance, but that the performance improvement is not linear with the increased radio speed.

Table 7.15 presents the ANOVA results for the end-to-end delay statistic. If a packet has to be segmented into several fragments, this statistic represents the delay of each fragment. All interactions that result in a percentage of variation smaller than 0.1% are not shown in the table.

Table 7.15. ANOVA Environments Results for End-to-End Delay.

Source (end-to-end delay)	Percentage of variation	90% confidence interval		F-ratio	F-ratio/ F-variates
Load (L6/L9)	18.85%	-40.05%	-38.17%	3866.8	Significant
Speed (S4/S2)	1.07%	-11.22%	-9.00%	219.9	Significant
Density (N12/N18)	2.16%	-15.27%	-13.10%	442.4	Significant
Traffic (Ts/Th)	18.27%	46.33%	49.23%	3747.9	Significant
Movement (Mb/Mc)	0.03%	0.56%	2.92%	6.1	Significant
Radio (Rv/Rc)	25.78%	56.45%	59.47%	5288.3	Significant
L * T	0.43%	5.47%	7.89%	88.1	Significant
L * R	5.69%	24.09%	26.73%	1167.8	Significant
S * R	0.13%	2.49%	4.87%	27.1	Significant
D * T	0.92%	8.60%	11.06%	187.9	Significant
T * M	0.30%	-6.52%	-4.25%	60.9	Significant
T * R	13.30%	-34.39%	-32.44%	2729.3	Significant
L * D * T	0.57%	6.48%	8.91%	116.2	Significant
L * T * R	3.80%	-19.68%	-17.56%	780.1	Significant
S * D * T	0.26%	-6.17%	-3.89%	53.1	Significant
S * D * M	0.53%	6.21%	8.63%	108.3	Significant
S * T * M	0.35%	4.79%	7.20%	71.1	Significant
S * T * R	0.58%	-8.61%	-6.36%	118.8	Significant
D * T * M	0.15%	-4.93%	-2.64%	29.9	Significant
D * T * R	0.25%	3.93%	6.33%	52.3	Significant
T * M * R	0.11%	-4.42%	-2.12%	22.3	Significant
L * T * S * D	0.23%	-5.85%	-3.57%	46.4	Significant
L * T * S * R	0.23%	-5.87%	-3.59%	46.8	Significant
L * T * D * M	0.13%	-4.76%	-2.46%	27.2	Significant
L * T * D * R	0.30%	4.35%	6.76%	61.2	Significant
L * T * M * R	0.34%	-6.92%	-4.65%	70.4	Significant
Error	1.87%				

Unlike the previous two tables, there is no single factor that has a much higher percentage of variation than other factors. Each factor tends to equally contribute to the variation in the end-to-end delay statistic, except those related to network dynamics and topology. In addition, each higher degree interaction has a higher percentage of variation than the earlier two statistics. Such behavior indicates that delay can be highly dynamic and depends on many factors.

Naturally, a higher offered load results in a higher delay because of the increased network congestion. Unexpectedly, a higher node speed slightly reduces the end-to-end delay. This can be because some packets that need to travel across a long path tend to get lost in a highly dynamic network, although the percentage of variation is too small to make any valid assumptions. Higher network density reduces the number of hops that each packet has to travel, resulting in a slightly smaller delay.

Because the network can deliver more HTTP traffic than SMTP traffic, as seen in Table 7.13, the extra traffic also leads to a higher delay for each packet. Although the movement pattern has little impact on the delay, it is consistent with earlier analysis. A constant movement pattern causes more stress to the network and results in a higher delay.

The type of radio has the most significant impact on the end-to-end delay and increases the delay by as much as 59%. Nevertheless, this degradation is consistent with the earlier analysis that performance does not necessary increase with a faster radio.

The increase in the percentage of data delivered shown in Table 7.14 also results in an increase in the delay, as can be seen in the L * R entry of the Table 7.15. Network density also has a small interaction with the traffic type, primarily because a higher network density tends to blur out the distinction between local and remote traffic. Since D * T increases the percentage of data delivered, the delay also increases proportionally. The high correlation between traffic type and radio type indicates that a faster transmission channel leads to a higher percentage of local traffic, which also leads to a reduction in the end-to-end delay. There is also a high correlation among the three most important factors: offered load, traffic type, and radio type.

Table 7.16 presents the ANOVA results for the power used per bit received statistic. All interactions that result in a percentage of variation smaller than 0.1% are not shown in the table.

Unlike the earlier three statistics, the offered load plays no part in the power used per bit received statistic, excluding higher order interaction. This behavior suggests that the power consumption is constant regardless of the congestion level. A higher node speed slightly increases the power consumption because the packet may not always take an optimal route to its destination. Both higher network density and HTTP traffic significantly reduce the power consumption because they reduce the number of hops that a packet needs to travel. As with earlier analysis, a constant movement pattern slightly degrades the power consumption because it puts more stress on the network. As is consistent with previous analysis, the CDMA2000 radio degrades the performance by increasing the power consumption.

Table 7.16. ANOVA Environment Results for Power Used per Bit Received.

Source (power used per bit received)	Percentage of variation	90% confidence interval		F-ratio	F-ratio/ F-variates
Load (L6/L9)	0.00%	-0.79%	0.20%	1.0	Not significant
Speed (S4/S2)	0.09%	0.76%	1.76%	17.8	Significant
Density (N12/N18)	21.61%	-19.21%	-18.31%	4387.8	Significant
Traffic (Ts/Th)	36.54%	-24.48%	-23.60%	7418.9	Significant
Movement (Mb/Mc)	0.66%	2.96%	3.97%	133.3	Significant
Radio (Rv/Rc)	28.78%	23.52%	24.63%	5844.2	Significant
L * R	6.07%	-10.64%	-9.70%	1232.1	Significant
S * D	0.18%	-2.27%	-1.29%	36.2	Significant
S * M	0.46%	2.40%	3.41%	94.0	Significant
D * M	0.10%	0.87%	1.87%	21.2	Significant
T * M	0.13%	1.02%	2.02%	26.0	Significant
M * R	0.23%	-2.51%	-1.52%	46.4	Significant
L * S * T	0.19%	1.38%	2.38%	39.6	Significant
L * T * R	0.15%	-2.10%	-1.12%	29.6	Significant
L * M * R	0.19%	1.33%	2.34%	37.8	Significant
S * D * T	0.27%	1.72%	2.72%	55.1	Significant
S * D * R	0.14%	1.10%	2.10%	28.7	Significant
S * T * R	0.21%	-2.41%	-1.42%	41.9	Significant
S * M * R	0.47%	-3.39%	-2.41%	96.3	Significant
L * T * S * M	0.23%	-2.51%	-1.53%	46.7	Significant
Error	1.89%				

There is a significant interaction between the offered traffic and the radio type. The faster radio, the CDMA2000, consumes approximately 10% less power when operating with a smaller offered load. As previously explained, such a combination reduces the percentage of routing overhead, resulting in a smaller overall power consumption.

Figures 7-17 to 7-19 visually summarize information from Tables 7-14 to 7-16 to present the relative effects of each environment factor for each performance metric. Because graphs can only represent a limited interaction among factors, they show a cumulative effect of each factor listed in the legend on the right-hand side. Because they do not accurately represent the stand-alone effect of each factor, a direct and careful comparison of the effect between these graphs and the original tables is needed.

Figure 7-18 deserves additional explanation. At first glance, it seems like the CDMA2000 radio has a lower delay than the Virginia Tech reconfigurable software radio. This is not entirely true, however. Because of the interaction between factors, the Virginia Tech radio degrades the performance when the offered load is 66% and the traffic type is HTTP. After considering the effect of each factor and their interactions, the Virginia Tech radio does, in fact, improve the delay performance.

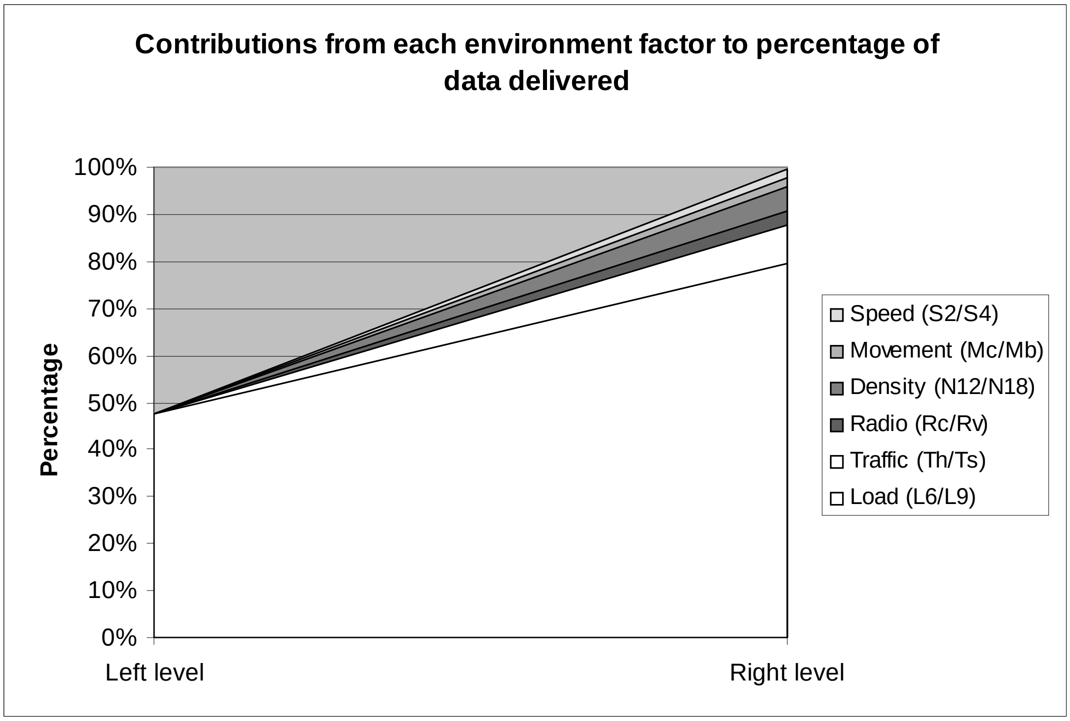


Figure 7-17. Visual representation of ANOVA results for the percentage of data delivered.

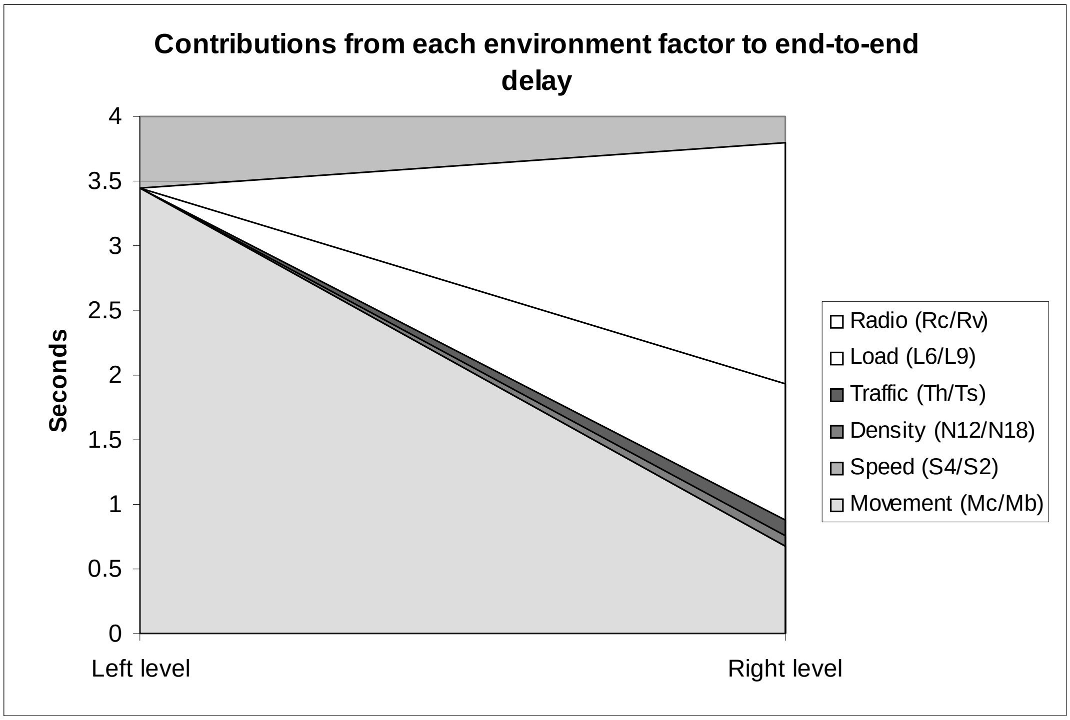


Figure 7-18. Visual representation of ANOVA results for the end-to-end delay.

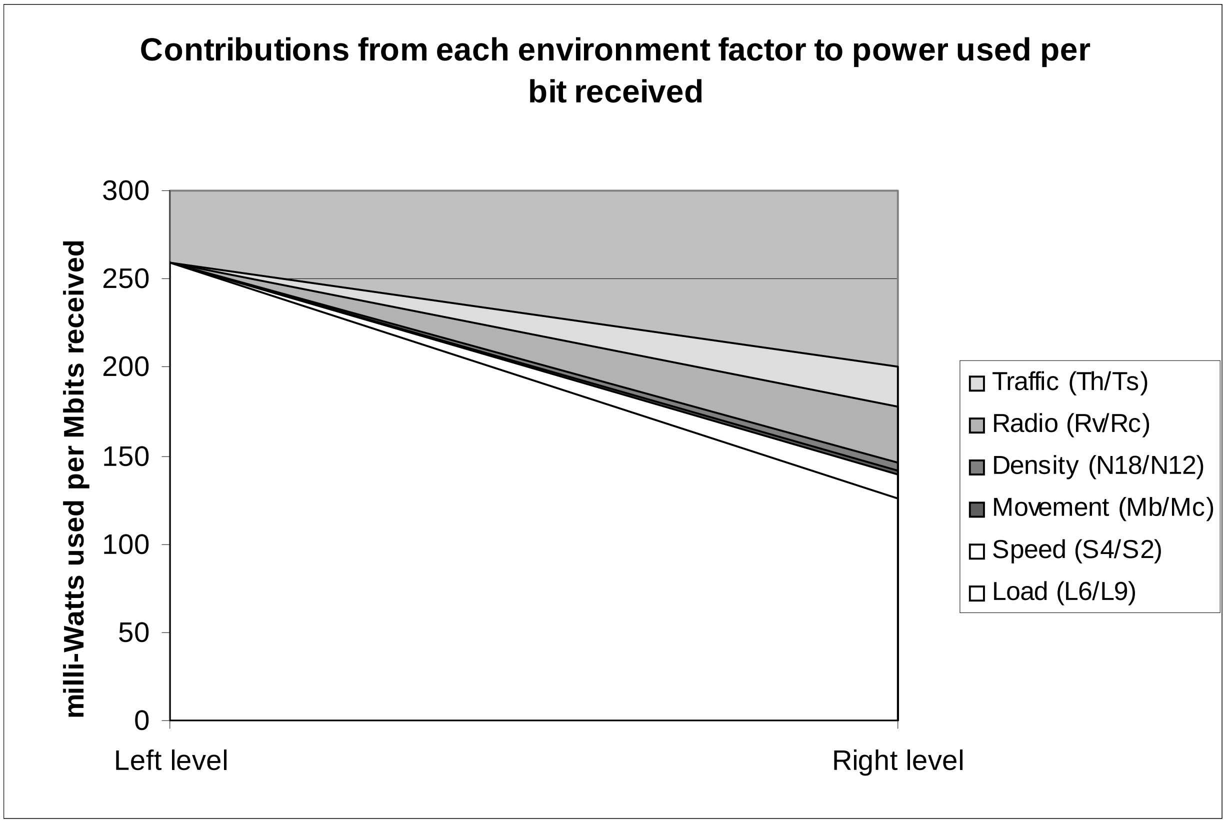


Figure 7-19. Visual representation of ANOVA results for the power used per bit received.

7.4.2 Effect from Environment – Individual Factor Experiment

Using data collected from Section 7.4.1, the importance of each factor can be measured and its interactions with other factors can be assessed. Such knowledge is used to design another set of experiments to study the impact of each factor in more detail, using multiple levels for each factor rather than the two-level approach in the previous section.

Table 7.17 outlines the levels that are used in the individual factor experiments. Three experiments are performed: the effect of varying the node speed from 0 Kph to 65 Kph using a constant direction movement pattern; the effect of network density by varying the node degree from 6 to 24; the combined effects of offered load level, traffic type, and radio type.

7.4.2.1 Effect of Node Speed

In this experiment, all factors except speed are fixed (66% offered load, HTTP traffic, constant direction, node degree of twelve, and Virginia Tech radio). Since previous results indicate that node speed has little effect or correlation with other factors, varying other factors should not result in a significantly more accurate analysis.

Table 7.17. Levels Used in Individual Factor Experimentation.

Factors	Description
Offered load	95% (**L9**), 66% (**L6/Default**), and 50% (**L5**) of the requested traffic get delivered
Traffic type	SMTP (**Ts**) [Pax94] and HTTP (**Th/Default**) [CBC95] traffic
Node speed (0, 5, 25, 45 Kph)	**S1** (0, 33, 67, 0), **S2** (0, 33, 33, 33), **S3** (100, 0, 0, 0), **S4/Default** (5 Kph), **S5** (25 Kph), **S6** (45 Kph), and **S7** (65 Kph)
Movement pattern	Constant direction (**Mc/Default**) and city block (**Mb**)
Degree of node	6 (**N6**), 12 (**N12/Default**), 18 (**N18**), and 24 (**N24**)
Radio type	The Virginia Tech reconfigurable software radio (**Rv/Default**) [SHA98] and CDMA2000 (**Rc**) with ITU-R/IMT-2000/IMT-MC recommendation (using radio configuration 6 with 3.6864 Mchips per second)

Figure 7-20 shows the network throughput when nodes are moving at different speeds. The top and bottom curves represent the low and high limit of the 90% confidence interval, respectively. Although there seems to be a slight dip in performance when the node speed equals 15.5 Kph, the network throughput for all node speeds less than or equal to 23.33 Kph are statistically equivalent. Any results are considered statistically equivalent if they have an overlapping confidence interval.

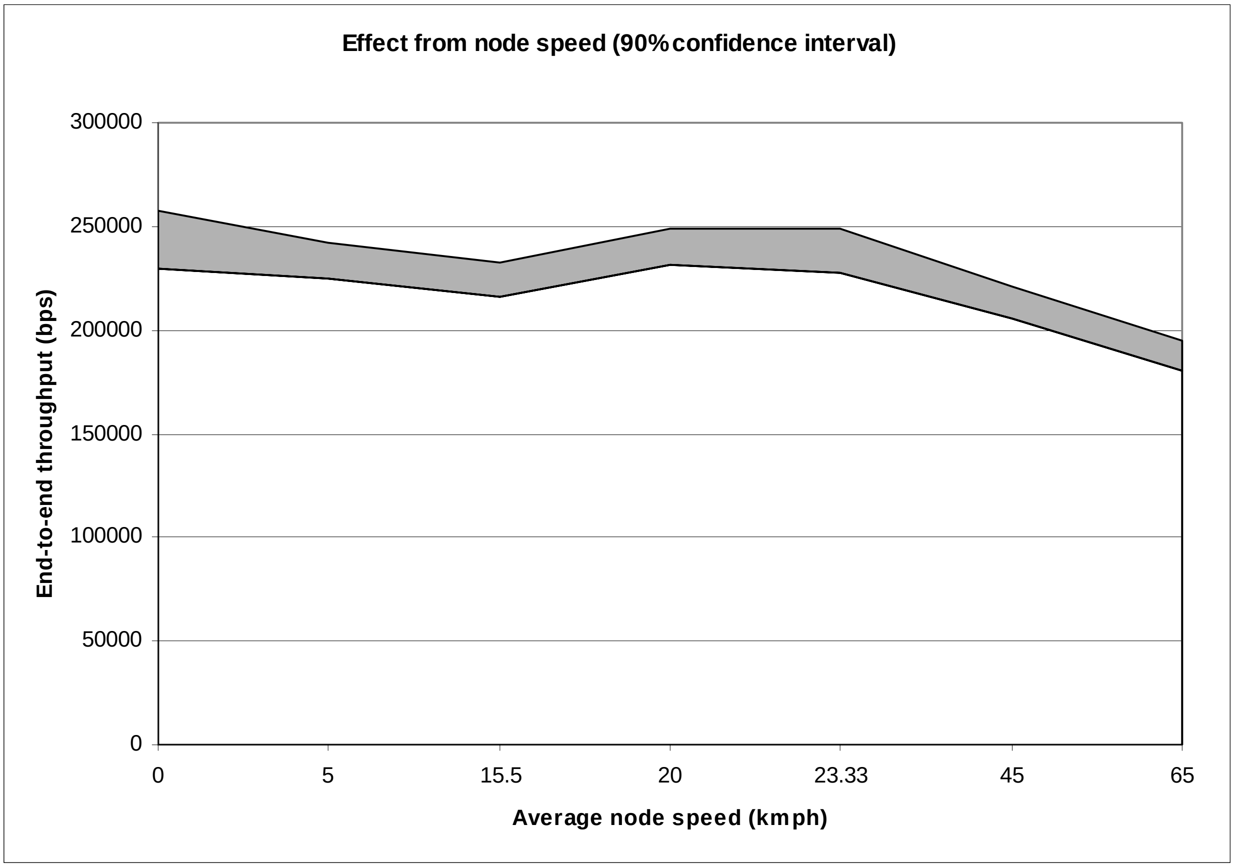


Figure 7-20. Effect of node speed on the end-to-end network throughput.

In this experiment, the performance, in terms of throughput, does not suffer as long as node speed is less than or equal to 23.33 Kph. The network throughput decreases steadily as the speed increases to 45 and 65 Kph. Their graceful degradation is an indication of stability in both the ADIM-NB and CAMEN protocols.

Table 7.18 summarizes the error from various experiments. The first column shows the variation that can be explained by the ANOVA analysis while the second column shows unexplained variation, or experimental error. As previously mentioned, node speed has little effect on the overall performance. That conclusion is further emphasized by a large percentage of experimental error, which indicates that variations in the result are not significantly due to the changes in the levels used.

Table 7.18. Variation in Each Statistic as a Result from Changes in Node Speed.

Percentage of variation	ANOVA model	Experiment error
End-to-end throughput	70.02%	29.98%
Percentage of data delivered	73.02%	26.98%
End-to-end delay	51.45%	48.55%
Power used per bit received	81.21%	18.79%

Different node speeds seem to cause more change in the power consumption and less change in the end-to-end throughput and percentage of data delivered. Because of its relatively long transmission range, low node speed has little impact on network connectivity and can cause even less change in the end-to-end delay because the negotiation process reduces the effect of the node speed.

Figure 7-21 shows the percentage of data delivered when nodes are moving at different speeds. The top and bottom curves represent the low and high limit of its 90% confidence interval, respectively. As with the earlier statistic, there seems to be a slight dip in performance when node speed is equal to 5 and 15.5 Kph. However, the network can deliver the same amount of data, statistically, when node speeds are less than or equal to 23.33 Kph and data delivery degrades gracefully as node speed increases.

Figure 7-22 shows the end-to-end delay when nodes are moving at different speeds. The end-to-end delay does not seem to be significantly affected by different node speeds. The network has statistically the same performance when node speeds are less than or equal to 23.33 Kph. It also has the same delay when node speeds are either 45 or 65 Kph. As shown in Table 7.18, node speed has the least impact on the packet delay.

Figure 7-23 shows the power used per bit received when nodes are moving at different speeds. The power consumption tends to increase as node speed increases. Because power consumption depends on several factors, including the number of hops, throughput, level of retransmission, and fairness in the network, it changes quite unpredictably. Nevertheless, the power consumption degrades gracefully as the node speed increases. Table 7.18 shows that node speed has the greatest impact on power consumption.

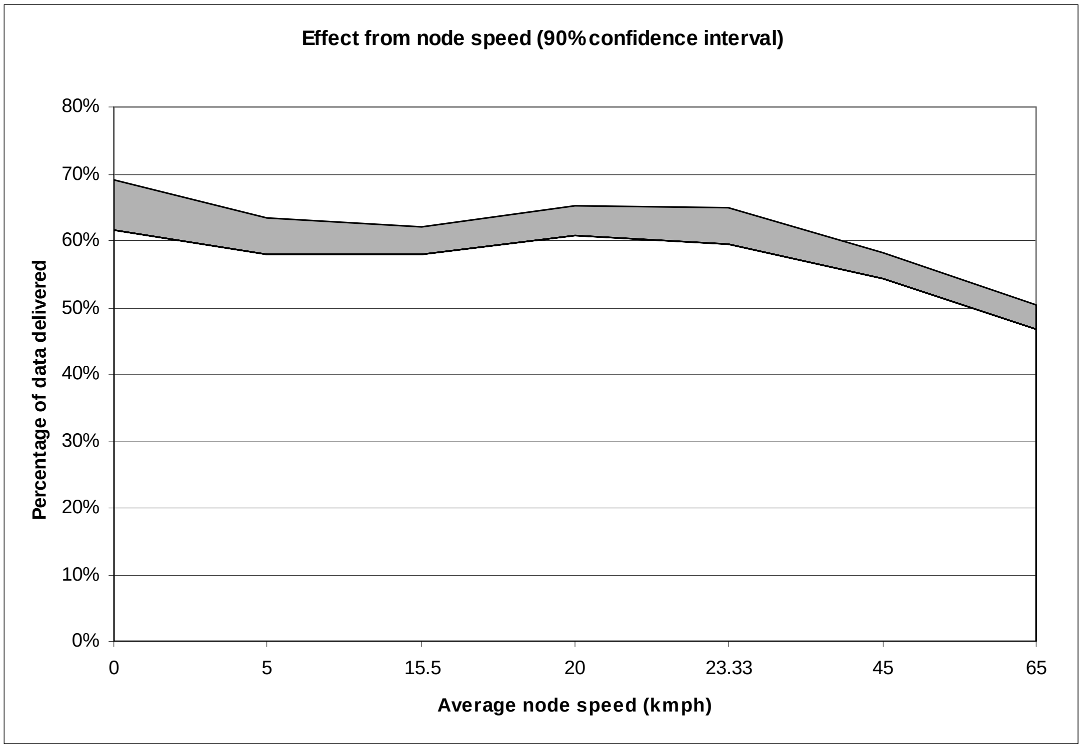


Figure 7-21. Effect of node speed on the percentage of data delivered.

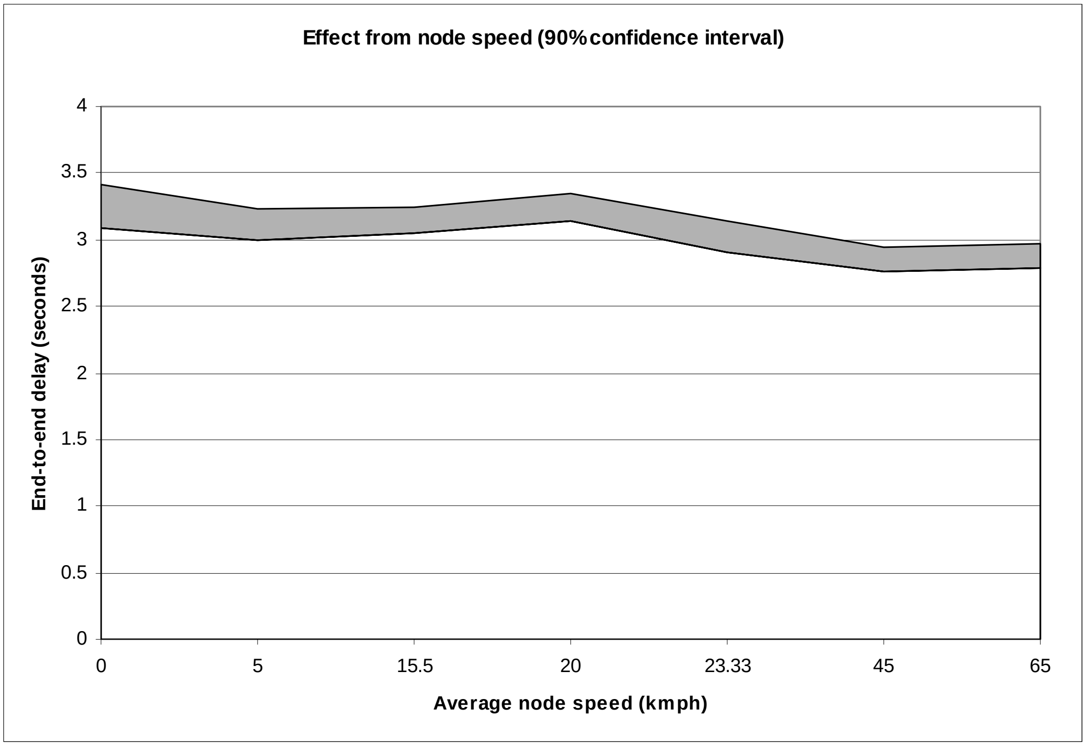


Figure 7-22. Effect of node speed on the end-to-end delay.

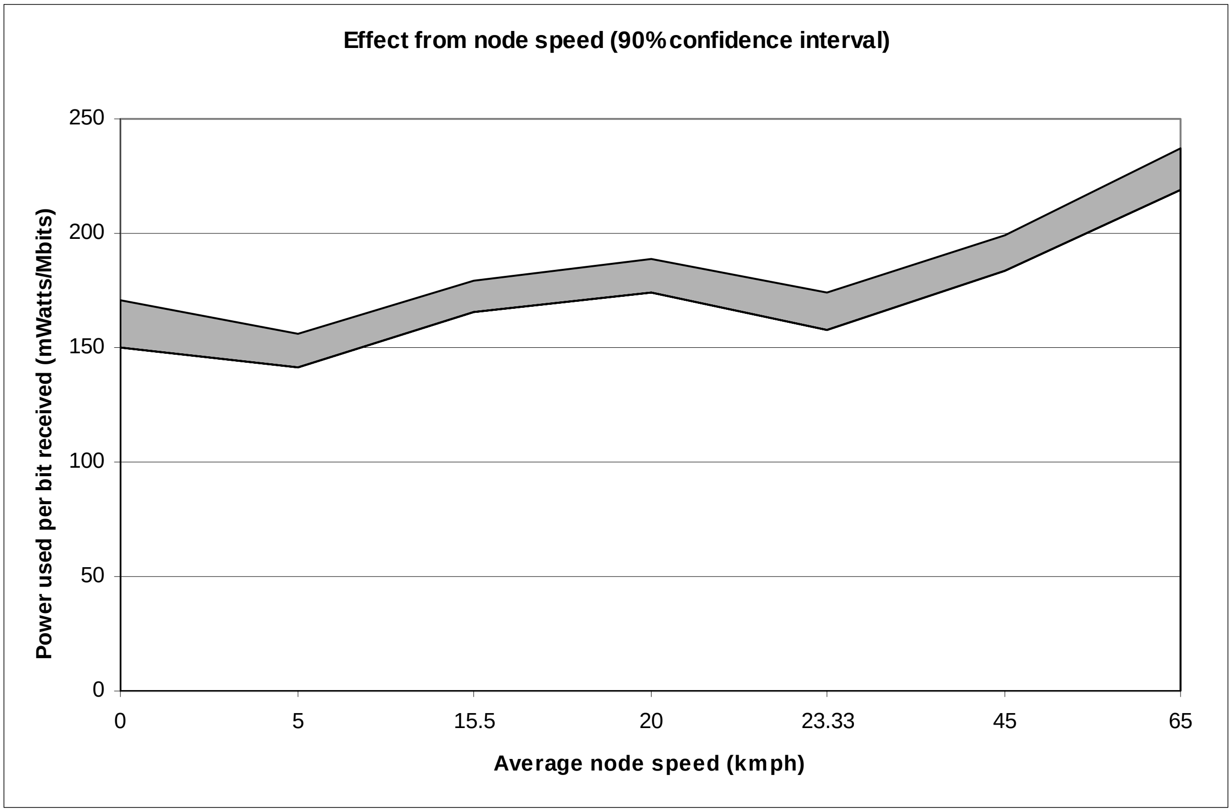


Figure 7-23. Effect of node speed on the power used per bit received.

7.4.2.2 Effect of Network Density

In this experiment, all factors except the degree of each node are fixed (66% offered load, HTTP traffic, constant direction, 5 Kph, and Virginia Tech radio). Since previous results indicate that node degree has little effect on and a small correlation with other factors, varying other factors will not result in a significantly more accurate analysis.

Figure 7-24 shows the network throughput when the degree of each node varies from a node degree of six to a node degree of twenty-four. The top and bottom curves represent the lower and higher limits of its 90% confidence interval, respectively.

The end-to-end network throughput increases steadily as the network density increases, until each node has a degree of eighteen. At this point, the network is already tightly packed and there is no performance gain from additional connectivity. The increase in throughput primarily results from the lower number of hops that each packet has to traverse, the more robust network connection, and the higher transmission speed when nodes are closer to each other.

Table 7.19 summarizes the error from various experiments. The first column shows the variation that can be explained by the ANOVA analysis and the second column shows an unexplained variation, or experimental error. As previously mentioned, network density has little impact on the overall performance. That conclusion is further emphasized by a large percentage of experimental error. A large percentage of experimental error indicates that variations in the result are not significantly due to changes in the levels used.

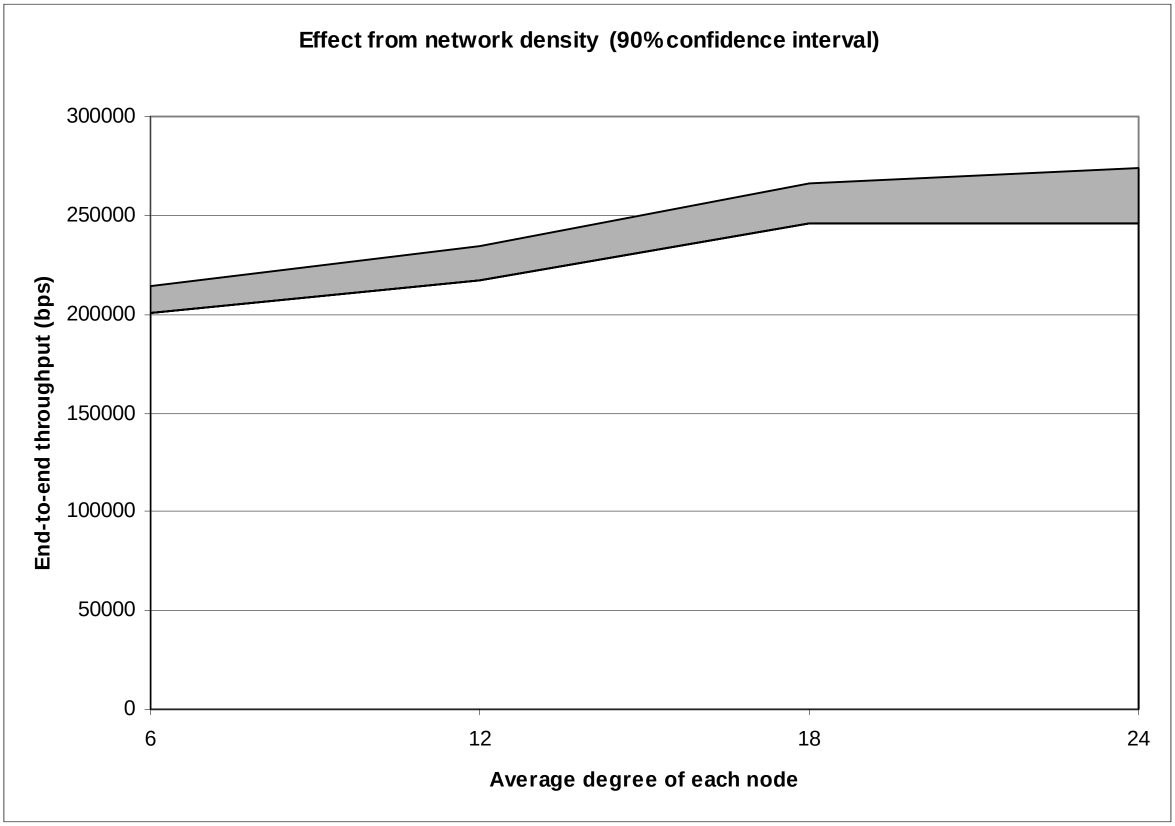


Figure 7-24. Effect of network density to the end-to-end network throughput.

Table 7.19. Variation in Each Statistic as a Result of Changes in Node Degree.

Percentage of variation	ANOVA model	Experimental error
End-to-end throughput	75.90%	24.10%
Percentage of data delivered	77.79%	22.21%
End-to-end delay	90.67%	9.33%
Power used per bit received	98.57%	1.43%

Different network density seems to cause greater change in the delay and power consumption and a smaller change in the throughput and percentage of data delivered. Both end-to-end delay and power consumption are highly sensitive to the number of hops that each packet has to traverse. A denser network significantly reduces the number of hops and improves the performance of these two statistics. In addition, being closer to each other allows the use of lower transmission power, resulting in a further reduction in power consumption.

Figure 7-25 shows the percentage of data delivered at different node degrees. As earlier statistics indicate, the performance keeps increasing until each node has a degree of eighteen.

Because the network throughput and percentage of data delivered are closely related, they exhibit the same characteristics and trends.

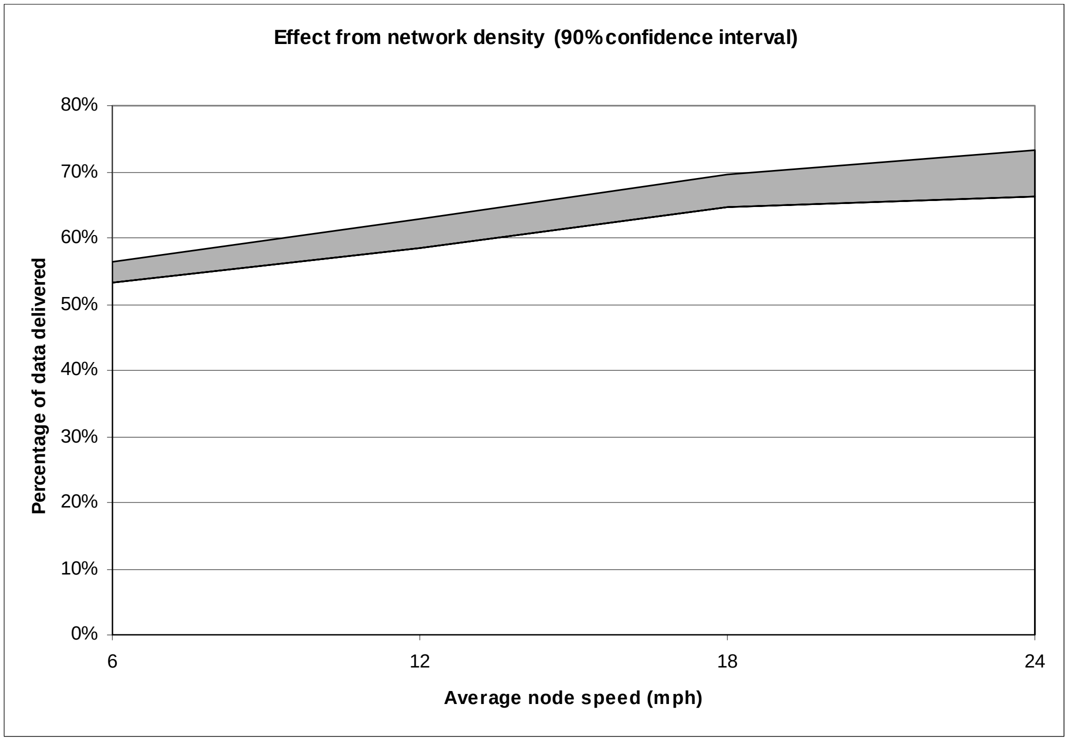


Figure 7-25. Effect of network density on the percentage of data delivered.

Figure 7-26 shows the end-to-end delay at a different node degree. Unlike the previous two statistics, the delay keeps decreasing as the network density increases, including when each node has a degree of twenty-four. As shown in Table 7.18, the node degree has a significant impact on the packet delay. A higher network density not only reduces the number of hops that each packet has to traverse, but also reduces the propagation delay. Since the radio under consideration has a long transmission range, any reduction in delay most likely results in a performance improvement. The delay also has a rather small confidence interval, indicating a high level of consistency among several replications of the simulation.

Figure 7-27 shows the power used per bit received when nodes have different node degrees. The power consumption tends to decrease as the node degree increases because of the lower number of hops that each packet has to traverse and the reduced power requirement as nodes are closer to each other. The drop in power consumption is apparent from a network with a node degree of six to that with a node degree of twelve. Because power consumption depends on several factors, including the number of hops, current throughput, level of retransmission, and fairness in the network, its trend can be difficult to predict. The small number of nodes in the network may be another reason for differences in the reduction rate of power consumption. Nevertheless, power consumption steadily decreases as the node degree increases. As shown in Table 7.19, node degree has the most impact on power consumption.

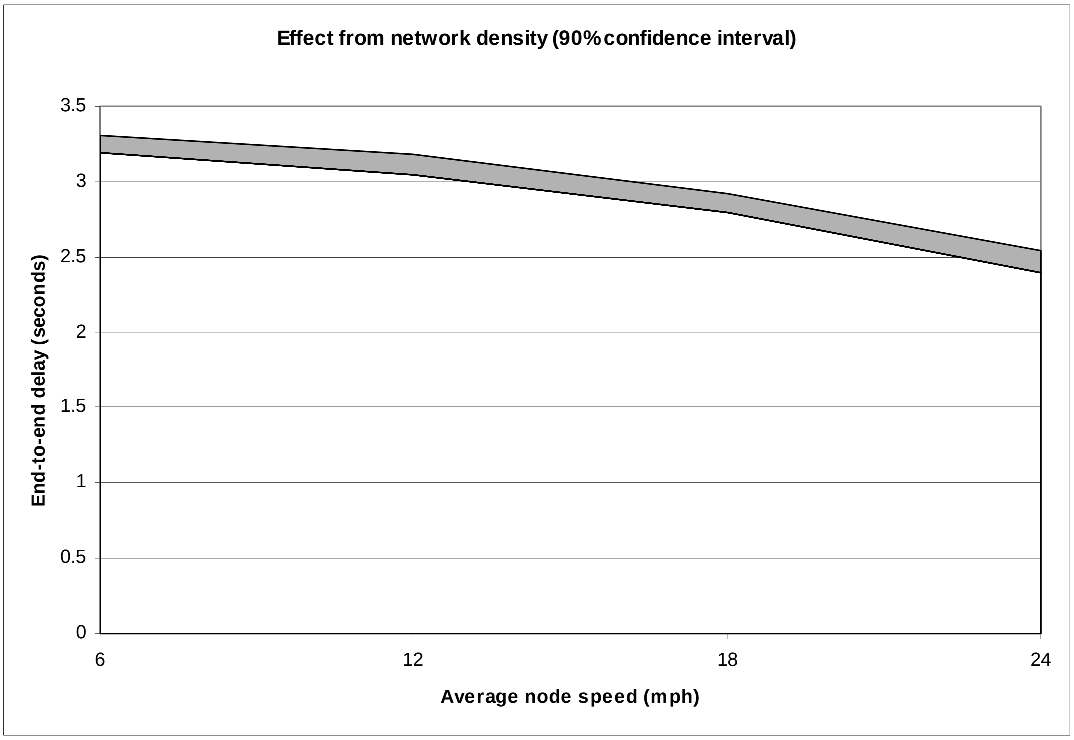


Figure 7-26. Effect of network density on the end-to-end delay.

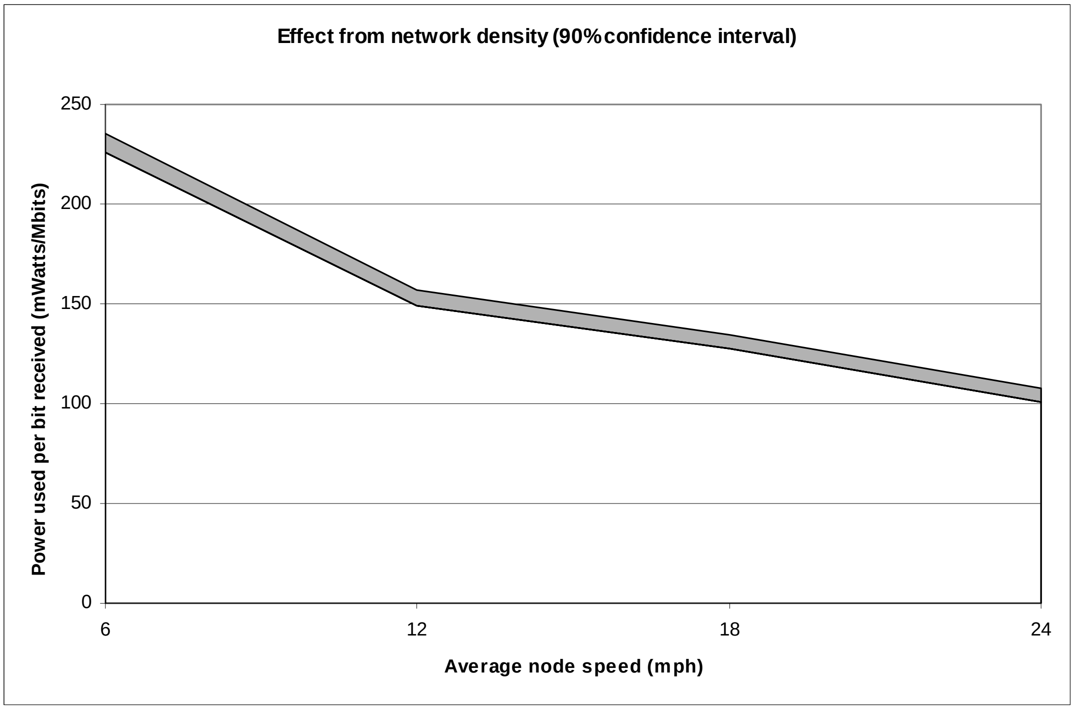


Figure 7-27. Effect of network density on the power used per bit received.

7.4.2.3 Effect of Offered Load, Traffic Type, and Radio Type

Since the offered load level, traffic type, and radio type are highly correlated with each other, the analysis of one must take the others into consideration. One new level is introduced into the experiment, 50% offered load, while other factors are still the same (constant direction, node degree of twelve, and 5 Kph speed). The simulation results are analyzed using the ANOVA technique to determine the importance of each factor as well as their correlation. Table 7.20 summarizes the factors and their corresponding levels used in this experiment. Since the effect of each factor is already described in detail in Section 7.4.2, this section will only revisit the relative significance of each factor and will not go into detail about which levels have the best performance or why.

Table 7.20. Levels Used in Joint Load-Traffic-Radio Experimentation.

Factors	Description
Offered load	95% (**L9**), 66% (**L6**), and 50% (**L5**) of the requested traffic get delivered.
Traffic type	SMTP (**Ts**) [Pax94] and HTTP (**Th/Default**) [CBC95] traffic
Node speed (0, 5, 25, 45 Kph)	**S4/Default** (5 Kph)
Movement pattern	Constant direction (**Mc/Default**)
Degree of node	12 (**N12/Default**)
Radio type	The Virginia Tech reconfigurable software radio (**Rv/Default**) [SHA98] and CDMA2000 (**Rc**) with ITU-R/IMT-2000/IMT-MC recommendation (using radio configuration 6 with 3.6864 Mchips per second)

Figure 7-28 shows the raw data for the end-to-end network throughput statistic. Each bar represents an arithmetic mean of three to seven simulation batches. The total twelve possible combinations result in twelve bars in the figure. Although more offered traffic always results in greater throughput, the increase in throughput is varied with the type of traffic and radio. Since this type of relationship is difficult to analyze visually, an ANOVA analysis is applied to the simulation data. The detailed analysis of the impact of each factor is presented in Tables 7-21 through 7-24, which show the results of the ANOVA for four different statistics.

Table 7.21 presents the ANOVA results for the end-to-end network throughput (user data) statistic. Because different radios have different data transmission speeds, the throughput is normalized to the amount of data received per megahertz of channel bandwidth using BPSK modulation. All main effects and their interactions are shown in the table.

Most of the variation in throughput can be attributed to the main effect, which accounts for approximately 88% of the total variation. The most important effect is the offered load level, which accounts for 81% of the total variation. It is natural to see that more offered traffic leads to a higher network throughput because they are directly related. The difference in radio types explains only 1% of the variation in throughput, indicating that throughput is quite independent from the channel speed. The difference in traffic types explains about 7% of the total variation because the HTTP traffic has a slightly different characteristic than the SMTP traffic, particularly regarding the destination selection and the amount of local traffic, as previously explained in Section 7.4.2.

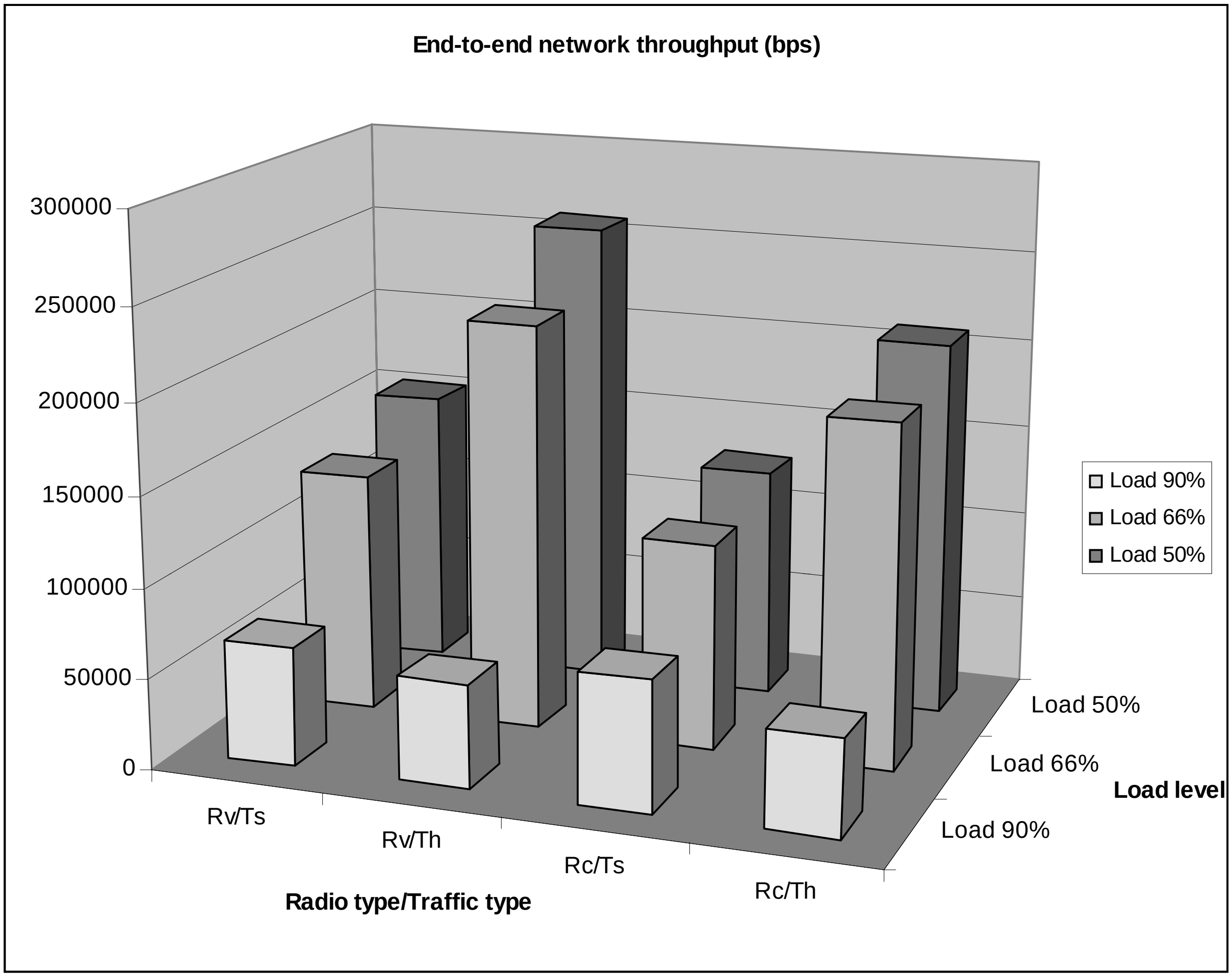


Figure 7-28. Network throughput data in various environment factors.

Another significant interaction is that between the offered load and traffic type, which is also a result of the experimental design. The small percentage of error indicates a good match between the model and the data. Such a small percentage of error also makes all effects and their interactions statistically significant. In summary, most variations in the throughput can be attributed to the levels of offered traffic and the traffic characteristics, which is an expected behavior.

Because the network throughput is highly dependent on the level of offered traffic, it is advantageous to look at another performance metric, the percentage of data delivered, which provides a good indication of the network throughput without too much dependency on the offered traffic because it is already normalized with the offered traffic level. Table 7.22 presents the ANOVA results for the percentage of data delivered statistic. All main effects and their interactions are shown in the table. Unlike throughput, the percentage of data delivered is less dependent on the levels of offered traffic and on traffic types. It still depends the most on the offered load because a higher offered load leads to a congested network, which can serve a smaller number of transmission requests.

Table 7.21. ANOVA Joint Load-Traffic-Radio Results for End-to-End Throughput.

Component	Sum of squares	Percentage of variation	Degrees of freedom	Mean square	F-computed	F-ratio	
y	2175.165		84				
$\bar{y}$	2170.642		1				
y- $\bar{y}$	4.523	100.000%					
Main effect	3.994	88.287%	4	0.998	3299.499		
Offered **L**oad	3.644	80.557%	2	1.822	6021.249	2606.60	Significant
Radio type	0.054	1.186%	1	0.054	177.321	65.19	Significant
Traffic type	0.296	6.543%	1	0.296	978.178	359.62	Significant
First order interactions	0.504	11.144%	5	0.101	333.199		
L * R	0.034	0.745%	2	0.017	55.721	24.12	Significant
L * T	0.466	10.293%	2	0.233	769.334	333.05	Significant
R * T	0.005	0.106%	1	0.005	15.885	5.84	Significant
2^{nd} order interactions							
L * R * T	0.004	0.087%	2	0.002	6.523	2.82	Significant
Error	0.022	0.482%	72	0.000			

Table 7.22. ANOVA Joint Load-Traffic-Radio Results for Percentage of Data Delivered.

Component	Sum of squares	Percentage of variation	Degrees of freedom	Mean square	F-computed	F-ratio	
y	4.341		84				
$\bar{y}$	2.854		1				
y- $\bar{y}$	1.486	100.000%					
Main effect	1.353	91.073%	4	0.338	1424.709		
Offered **L**oad	0.871	58.629%	2	0.436	1834.340	794.09	Significant
Radio type	0.173	11.648%	1	0.173	728.866	267.97	Significant
Traffic type	0.309	20.796%	1	0.309	1301.289	478.42	Significant
First order interactions	0.115	7.746%	5	0.023	96.936		
L * R	0.043	2.877%	2	0.021	90.010	38.97	Significant
L * T	0.068	4.578%	2	0.034	143.232	62.01	Significant
R * T	0.004	0.291%	1	0.004	18.198	6.69	Significant
2^{nd} order interactions							
L * R * T	0.000	0.031%	2	0.000	0.964	0.42	Not significant
Error	0.017	1.151%	72	0.000			

Traffic type also has a large impact on the percentage of data delivered because of the different characteristics of different traffic types. A different radio also results in a different percentage of data delivered. Although not shown in the table, a slower radio tends to be able to deliver more data, partly because the ADIM-NB protocol is optimized with a slower radio and partly because the same propagation delay has a greater impact with a faster radio.

The interaction between factors for the percentage of data delivered is smaller than that of the end-to-end throughput statistic because it is normalized. The highest percentage of variation is still the interaction between offered load and traffic type, although it is reduced by half. The interaction between the offered load and radio type is a result of routing overhead. As discussed earlier in Section 7.4.1, since the amount of routing information is approximately constant in a lightly loaded network, regardless of the radio type, the amount of routing information compared to the amount of data traffic is smaller for the faster CDMA2000 radio. Such smaller overhead leads to better network performance. The second order interaction is virtually zero, indicating that the normalization algorithm removes a lot of correlation from the results. The small percentage of variation for error also implies that there is only a small mismatch between the simulation results and the ANOVA model.

Another common performance metric, the end-to-end delay, is shown in Table 7.23. Because they measure the system performance from a slightly different perspective, the percentage of variation is slightly different from the throughput statistic. Unlike the previous two tables, there are significant interactions between factors, and the model error contributes to a higher percentage of variation.

Table 7.23. ANOVA Joint Load-Traffic-Radio Results for End-to-End Delay.

Component	**Sum of squares**	**Percentage of variation**	**Degrees of freedom**	**Mean square**	**F-computed**	**F-ratio**	
y	17.314		84				
$\bar{y}$	14.795		1				
$y-\bar{y}$	2.520	100.000%					
Main effect	1.010	40.065%	4	0.252	133.648		
Offered **L**oad	0.393	15.594%	2	0.196	104.034	45.04	Significant
Radio type	0.522	20.729%	1	0.522	276.586	101.69	Significant
Traffic type	0.094	3.743%	1	0.094	49.938	18.36	Significant
First order interactions	0.967	38.393%	5	0.193	102.456		
L * R	0.542	21.511%	2	0.271	143.512	62.13	Significant
L * T	0.006	0.245%	2	0.003	1.632	0.71	Not significant
R * T	0.419	16.637%	1	0.419	221.990	81.61	Significant
2nd order interactions							
L * R * T	0.407	16.146%	2	0.203	107.719	46.63	Significant
Error	0.136	5.396%	72	0.002			

Delay is a rather sensitive statistic because it depends on many factors. Every re-route decision or loss of packet has a higher impact on the delay than the network throughput, especially in a congested network. The queuing theory suggests that the network delay can increase exponentially if the network capacity is reached [BeG92]. As a result, the variation, as well as any errors in the ANOVA model prediction of the delay statistic, is higher than in the previous two tables.

Naturally, different load levels cause changes in the delay because the system experiences different levels of congestion. The type of radio also has a lot of impact on the delay statistic because any delay values regarding control packet overhead and propagation remain the same while the radio speed is different. As a result, different types of radios experience different amounts of delay because the relative effect of control overhead is different. Different traffic types have a slight effect on the end-to-end delay statistic, even though they have a large impact on the percentage of data delivered. As can be seen from three different statistics, each factor has different influence on different performance metrics. The end-to-end packet delay is dominated by both negotiation delay and retransmission delay, while the end-to-end throughput is dominated by traffic characteristics and network connectivity.

A large interaction between the offered load and radio type is a result of the experimental design. A large interaction between radio type and traffic type indicates that a faster transmission channel leads to a higher percentage of local traffic, which also leads to a reduction in the end-to-end delay, as discussed in Section 7.4.1. There is also a high interaction among all three factors.

Table 7.24 presents the ANOVA results for the power used per bit received statistic. The main effects dominate almost all variation in the statistic, accounting for 92%. There is a small first order interaction of 3%. The error from the ANOVA model account for approximately 4% of the variation in the power used per bit received.

In contrast with the earlier three statistics, the offered load makes little contribution to the power consumption. In general, the radio always uses approximately the same power to transmit. If there are excessive packet retransmissions, the power consumption will increase. The level of congestion in the network has little to do with variation in power consumption, indicating that the amount of retransmission is approximately the same in all load levels. The radio type has the most effect on the power consumption because different radios have different characteristics. The higher speed radio has less power per bit, because the bit duration is smaller. The power used per bit received value will also be different if the unit used is Jules/Mbit received rather than Watts/Mbit received. A high data transmission speed also results in an increased path loss, resulting in a higher retransmission rate and more wasted power. Traffic type also has a large impact on the power consumption because the traffic type defines the average number of hops that a packet needs to travel. The power consumption is directly dependent on the number of hop counts per packet because it is an end-to-end statistic. Different packet sizes also results in different amounts of packet headers, leading to different power consumption. Other first and second order interactions are relatively small in comparison with the main effect.

Since different factors have different effects on different statistics, it is important to understand the objective of the system and the desired characteristics before evaluating the effect of each factor and algorithm.

Table 7.24. ANOVA Joint Load-Traffic-Radio Results for Power Used per Bit Received.

Component	Sum of squares	Percentage of variation	Degrees of freedom	Mean square	F-computed	F-ratio	
y	454.995		84				
$\bar{y}$	454.334		1				
$y-\bar{y}$	0.662	100.000%					
Main effect	0.609	91.998%	4	0.152	400.516		
Offered **L**oad	0.015	2.240%	2	0.007	19.506	8.44	Significant
Radio type	0.308	46.504%	1	0.308	809.821	297.73	Significant
Traffic type	0.286	43.254%	1	0.286	753.232	276.92	Significant
First order interactions	0.023	3.456%	5	0.005	12.037		
L * R	0.017	2.569%	2	0.009	22.371	9.68	Significant
L * T	0.006	0.885%	2	0.003	7.701	3.33	Significant
R * T	0.000	0.002%	1	0.000	0.039	0.01	Not significant
2nd order interactions							
L * R * T	0.003	0.411%	2	0.001	3.582	1.55	Significant
Error	0.027	4.135%	72	0.000			

7.5 Effect of Environmental Conditions and Traffic Characteristics on Each ADIM-NB Algorithm

To evaluate the correlation and the effect of each ADIM-NB algorithm in a different type of environment, a 2^{k-p} factorial experimentation [Jai91] is used. Using the results from Sections 7.2 and 7.3, any factors that do not have a significant impact on the performance are excluded from this experiment. In addition, some interactions or correlation that have little impact on the performance can also be excluded from the experiment, resulting in fewer simulations. The large number of factors (seven) involved in this experiment produce a large amount of information in the experimentation results. This chapter will focus only on the joint effect or correlation between the ADIM-NB algorithms and the environment conditions. Self-correlation among different ADIM-NB algorithms or environment factors is already covered in earlier sections of this chapter.

Table 7.25 summarizes the factors used in this section. The first three factors are the environmental conditions and traffic characteristics. Previous analysis shows that they have significant impact on the network performance as well as significant interaction among each other. Other environmental factors related to mobility are not as important and are not included in this experiment. The last four factors are the individual algorithms of the ADIM-NB MAC protocol. As discussed earlier, each algorithm improves a different aspect of the system and deserves further analysis. Table 7.25 also contains a shorthand notation in bold typeface that is used in subsequent tables within this section.

Table 7.25. Factors Used in the Joint Algorithm-Environment Experimentation.

Factors	**Levels**
Offered **L**oad	95% (L9) and 66% (L6) of the requested traffic get delivered.
Traffic type	SMTP (Ts) [Pax94] and HTTP (Th) [CBC95] traffic
Radio type	Virginia Tech reconfigurable software radio (Rv) and CDMA2000 (Rc)
ADIM-NB/**M**U	Enable/Disable multi-connection setup algorithm
ADIM-NB/P**C**	Enable/Disable power control algorithm
ADIM-NB/P**G**	Enable/Disable adaptive processing gain algorithm
ADIM-NB/**F**E	Enable/Disable adaptive forward error correction algorithm

The full factorial experimental design of seven factors would result in 2^7 or 128 experiments. Since earlier analysis indicates that the third order interaction among all ADIM-NB algorithms, ADIM-NB/MU, ADIM-NB/PC, and ADIM-NB/PG, are relatively small and considered negligible, such knowledge is used to reduce the number of experiments from 2^7 experiments to 2^{7-1} or 64 experiments. The technique is called 2^{k-p} factorial experimental design [Jai91]. As a result, some effects will be combined, as will be mentioned later.

The results are analyzed using the same ANOVA technique to investigate the statistical significance of the four ADIM-NB algorithms in the presence of three environmental factors, as well as any interactions as they significantly contribute to four different statistics: throughput, delay, power used per bit received, and percentage of data delivered. Because it is estimated that the effect from each factor will be multiplicative, before the ANOVA analysis, all raw data is first transformed using a logarithmic function.

Figure 7-29 shows raw data for the end-to-end network throughput (user data) statistic. Each bar represents an arithmetic mean of three to seven simulation batches. Half of the total 128 possible combinations of factors are not simulated because they can be inferred from other data points, based on their correlation characteristics. The remaining sixty-four combinations are shown as sixty-four bars in the figure. The overall graph can be divided into four quadrants. Each quadrant shares the same offered load level and ADIM-NB/PG setting because both factors represent the majority of the variation in the statistic. The same presentation technique is subsequently applied to each quadrant to group related bars together.

Because of the additional information and correlation, Figure 7-29 is more difficult to understand than its cousin, Figure 7-15. It is shown here to present a high level overview of the relative performance and to indicate which combinations of factors are simulated.

As more traffic is offered, the network has higher throughput. In addition, different traffic characteristics result in different network capacities. Many factors have a greater effect in a heavily loaded network than in a lightly loaded network. The detailed analysis of the impact of each factor is presented in subsequent tables, which show the results of the ANOVA for four statistics, because each factor or combinations of factors contribute to the overall performance differently.

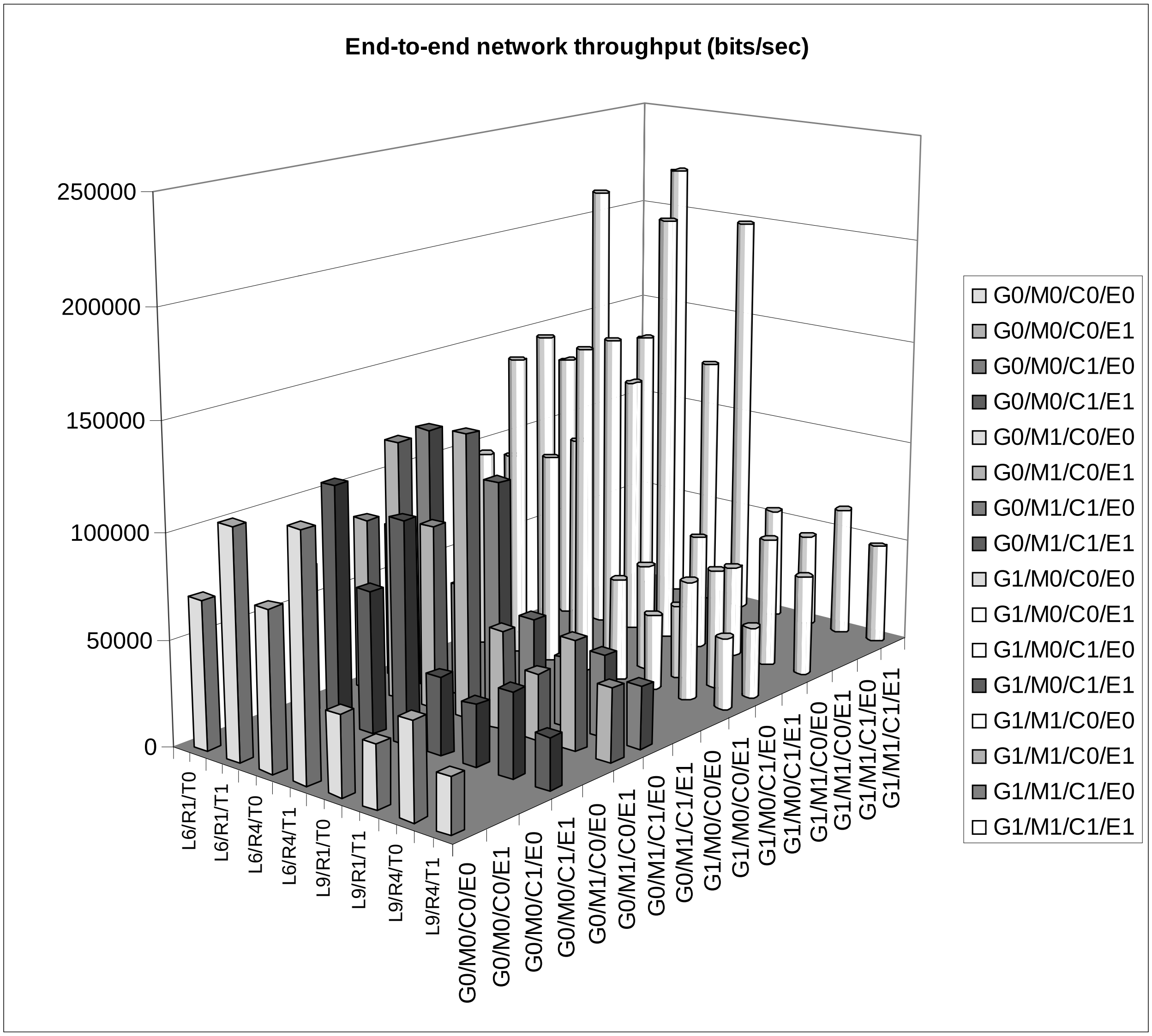


Figure 7-29. Network throughput of different ADIM-NB algorithms in various environments.

Table 7.26 presents the ANOVA results for the end-to-end network throughput (user data) statistic. All first order effects, second order interactions, and third order interactions are shown in the table. Some of the effects are combined together in a single entry because of the limitation of the 2^{k-p} factorial experimental design. In all cases, the second term in each combination is expected to be negligible, based on the previous analysis.

The interactions among different environment factors are explained earlier and will be briefly mentioned here. Any interactions among different ADIM-NB algorithms, as well as the stand-alone effects have also been previously discussed and will be briefly explained here. This chapter will focus on the correlation or interaction between environment factors and the different ADIM-NB algorithms or combinations of ADIM-NB algorithms.

With the exception of seven entries that consist solely of L, T, and R, all other entries are a combination of two interactions. The L.M + L.C.G.F entry, for example, is a combined effect of a correlation between load and the ADIM-NB/MU algorithm, and a correlation among load and the ADIM-NB/PC, ADIM-NB/PG, and ADIM-NB/FE algorithms. The second correlation, L.C.G.F, is expected to be small and negligible, and the first correlation explains most of the variation in throughput. The three most significant causes of an increase in throughput are the offered load, the use of the ADIM-NB/PG algorithm, and the interaction between offered load and traffic type.

There is a slight interaction between the offered load and the ADIM-NB/PG algorithm. The network is able to deliver more traffic in a congested environment if the ADIM-NB/PG algorithm is also enabled. This behavior implies that the ADIM-NB/PG algorithm provides greater improvement in a heavily loaded scenario, which is reasonable. In a lightly loaded network, other factors, such as routing and clustering traffic, also impact the throughput of user data but have less impact in a heavily loaded network.

The 0.12% variation from T.M indicates that the ADIM-NB/MU algorithm performs better with the more uniform distribution of HTTP traffic. SMTP traffic tends to involve the transfer of a large amount of data over a connection with one destination. HTTP traffic includes several requests to several destinations. Having more uniformly distributed traffic has a positive effect on the ADIM-NB/MU algorithm because this algorithm has more opportunities to set up multiple connections, resulting in a higher network throughput.

The 0.16% variation from R.C indicates that the ADIM-NB/PC algorithm performs better with the slower Virginia Tech radio. Because noise power increases with operating bandwidth, the CDMA2000 radio experiences a higher level of Gaussian noise. As a result, the system may experience a lower average SNR value, resulting in a reduction in the dynamic range in which the ADIM-NB/PC algorithm can operate. Such behavior lends favor to the use of the slower Virginia Tech radio.

The two most significant third order interactions will not be explained because of the complexity of the interpretation. In addition, many reasons can explain such a small variation in throughput, which can potentially result in an incorrect analysis. Nevertheless, all correlation between environmental factors and the ADIM-NB algorithms are under 0.5%, or not highly significant, suggesting that any throughput improvements from the ADIM-NB algorithms are quite independent of from the environmental conditions. Because the unexplained percentage of variation, or error, is only 0.55%, the model assumed in the ANOVA analysis is a good match to the simulation results. Such a small error also makes most effects and interactions in Table 7.26 significant.

Table 7.27 presents the ANOVA results for the percentage of data delivered statistic. All interactions that result in less than 0.1% variation are not shown to keep the table compact. In addition, those interactions only explain a small percentage of the variation in the statistic.

Table 7.26. ANOVA Joint Algorithm-Environment Results for Network Throughput.

Source (Throughput)	Percentage of variation	90% confidence interval		F-ratio	F-ratio/ F-variates
L (L6/L9)	72.98%	-80.25%	-79.34%	50502.5	Significant
T (Ts/Th)	0.29%	5.70%	7.27%	198.3	Significant
R (Rv/Rc)	0.00%	-0.01%	1.51%	2.7	Not significant
M+C.G.F	2.23%	17.78%	19.43%	1541.6	Significant
C+M.G.F	0.18%	-5.77%	-4.29%	126.3	Significant
G+M.C.F	9.40%	39.19%	41.01%	6505.1	Significant
F+M.C.G	0.03%	1.29%	2.82%	20.3	Significant
L.T	11.95%	-37.85%	-36.62%	8268.5	Significant
L.R	0.03%	1.45%	2.98%	23.6	Significant
L.M+L.C.G.F	0.00%	-1.31%	0.20%	1.5	Not significant
L.C+L.M.G.F	0.01%	0.46%	1.99%	7.2	Significant
L.G+L.M.C.F	0.42%	-8.29%	-6.83%	289.6	Significant
L.F+L.M.C.G	0.01%	-2.05%	-0.55%	8.3	Significant
T.R	0.07%	-3.79%	-2.30%	45.8	Significant
T.M+T.C.G.F	0.12%	3.47%	5.01%	85.7	Significant
T.C+T.M.G.F	0.03%	1.36%	2.89%	21.6	Significant
T.G+T.M.C.F	0.01%	0.10%	1.62%	3.6	Significant
T.F+T.M.C.G	0.01%	-1.98%	-0.47%	7.4	Significant
R.M+R.C.G.F	0.03%	-2.84%	-1.35%	21.6	Significant
R.C+R.M.G.F	0.16%	-5.41%	-3.93%	108.7	Significant
R.G+R.M.C.F	0.08%	2.67%	4.21%	56.5	Significant
R.F+R.M.C.G	0.03%	1.18%	2.71%	18.1	Significant
M.C+G.F	0.04%	-3.25%	-1.75%	30.9	Significant
M.G+C.F	0.28%	5.62%	7.19%	193.6	Significant
M.F+C.G	0.18%	4.32%	5.87%	123.2	Significant
L.T.R	0.06%	-3.67%	-2.17%	42.2	Significant
L.T.M+L.T.C.G.F	0.01%	0.44%	1.97%	7.0	Significant
L.T.C+L.T.M.G.F	0.01%	-1.75%	-0.24%	4.9	Significant
L.T.G+L.T.M.C.F	0.01%	0.21%	1.73%	4.6	Significant
L.T.F+L.T.M.C.G	0.01%	0.57%	2.09%	8.5	Significant
L.R.M+L.R.C.G.F	0.08%	2.71%	4.25%	58.0	Significant
L.R.C+L.R.M.G.F	0.00%	0.03%	1.55%	3.1	Significant
L.R.G+L.R.M.C.F	0.01%	0.37%	1.89%	6.2	Significant
L.R.F+L.R.M.C.G	0.00%	-0.28%	1.24%	1.1	Not significant
L.M.C+L.G.F	0.02%	1.11%	2.64%	17.0	Significant
L.M.G+L.C.F	0.15%	-5.36%	-3.88%	106.7	Significant
L.M.F+L.C.G	0.01%	-1.66%	-0.16%	4.1	Significant
T.R.M+T.R.C.G.F	0.09%	2.89%	4.43%	64.1	Significant
T.R.C+T.R.M.G.F	0.00%	-0.55%	0.97%	0.2	Not significant
T.R.G+T.R.M.C.F	0.00%	-1.56%	-0.05%	3.2	Significant
T.R.F+T.R.M.C.G	0.00%	-1.27%	0.24%	1.3	Not significant
T.M.C+T.G.F	0.01%	0.57%	2.10%	8.6	Significant
T.M.G+T.C.F	0.03%	1.21%	2.73%	18.7	Significant
T.M.F+T.C.G	0.03%	-2.84%	-1.34%	21.5	Significant
R.M.C+R.G.F	0.04%	-3.20%	-1.71%	29.7	Significant
R.M.G+R.C.F	0.00%	-0.57%	0.95%	0.2	Not significant
R.M.F+R.C.G	0.14%	3.72%	5.27%	96.2	Significant
Error	0.55%				

Table 7.27. ANOVA Joint Algorithm-Environment Results for Data Delivered.

Source (percentage of data delivered)	**Percentage of variation**	**90% confidence interval**		**F-ratio**	**F-ratio/ F-variates**
L (L6/L9)	26.13%	35.96%	37.54%	7130.1	Significant
T (Ts/Th)	26.32%	-31.72%	-30.59%	7183.2	Significant
R (Rv/Rc)	2.34%	-10.48%	-9.21%	638.7	Significant
M+C.G.F	6.62%	17.00%	18.45%	1806.3	Significant
C+M.G.F	0.60%	-5.69%	-4.39%	163.4	Significant
G+M.C.F	28.10%	37.45%	39.04%	7669.3	Significant
L.T	1.30%	6.97%	8.36%	354.3	Significant
L.R	0.30%	2.96%	4.32%	81.6	Significant
L.G+L.M.C.F	1.11%	-7.49%	-6.20%	303.9	Significant
T.R	0.10%	-2.73%	-1.41%	27.1	Significant
T.M+T.C.G.F	0.50%	4.02%	5.39%	135.4	Significant
T.G+T.M.C.F	0.15%	1.86%	3.21%	39.7	Significant
R.C+R.M.G.F	0.47%	-5.14%	-3.84%	129.4	Significant
R.G+R.M.C.F	0.22%	2.42%	3.77%	59.1	Significant
M.C+G.F	0.14%	-3.08%	-1.76%	37.1	Significant
M.G+C.F	0.92%	5.73%	7.11%	250.2	Significant
M.F+C.G	0.57%	4.36%	5.73%	155.7	Significant
L.R.M+L.R.C.G.F	0.28%	2.82%	4.18%	75.4	Significant
L.M.G+L.C.F	0.40%	-4.76%	-3.45%	107.9	Significant
T.R.M+T.R.C.G.F	0.31%	3.05%	4.41%	85.6	Significant
T.M.G+T.C.F	0.14%	1.78%	3.13%	37.2	Significant
R.M.C+R.G.F	0.11%	-2.88%	-1.56%	31.2	Significant
R.M.F+R.C.G	0.40%	3.53%	4.89%	108.8	Significant
Error	1.41%				

The largest interaction between traffic characteristic and the ADIM-NB algorithm is between the offered traffic and the ADIM-NB/PG algorithm. Other large interactions are between traffic type and the ADIM-NB/MU algorithm and between radio type and the ADIM-NB/PC algorithm. Because the statistic exhibits the same trend and relative performance as the throughput statistic, they can be explained by the same explanation associated with Table 7.26.

As with the throughput statistic, all correlation between environmental factors and ADIM-NB algorithms are small, or not highly significant, suggesting that any improvements from ADIM-NB algorithms are largely independent from the environment condition. Because of the small unexplained percentage of variation, or error, the model assumed in the ANOVA analysis is a good match to the simulation results. Such a small error also makes most effects and interactions significant.

Table 7.28 presents the ANOVA results for the end-to-end delay statistic. The delay statistic is normalized by the radio speed, making it a delay per unit duration of data received statistic. In particular, the delay from the CDMA2000 radio is multiplied by 3.6, which is the number of chips that fit into one micro-second. All interactions that result in a smaller than 0.1% variation are not shown to keep the table compact.

Table 7.28. ANOVA Joint Algorithm-Environment Results for Delay.

Source (percentage of data delivered)	Percentage of variation	90% confidence interval		F-ratio	F-ratio/ F-variates
L (L6/L9)	10.07%	-24.29%	-23.03%	3473.6	Significant
T (Ts/Th)	6.56%	20.61%	22.19%	2263.4	Significant
R (Rv/Rc)	0.52%	5.05%	6.52%	178.2	Significant
M+C.G.F	11.53%	-25.84%	-24.59%	3977.3	Significant
C+M.G.F	0.31%	3.74%	5.20%	107.0	Significant
G+M.C.F	33.29%	-41.49%	-40.36%	11480.7	Significant
F+M.C.G	0.05%	1.02%	2.46%	16.4	Significant
L.T	1.72%	9.95%	11.45%	594.7	Significant
L.M+L.C.G.F	1.61%	-10.51%	-9.15%	556.5	Significant
L.G+L.M.C.F	6.38%	-19.73%	-18.43%	2202.1	Significant
T.R	0.63%	-6.91%	-5.53%	218.5	Significant
T.M+T.C.G.F	1.62%	9.62%	11.12%	559.6	Significant
T.G+T.M.C.F	5.69%	19.07%	20.64%	1962.4	Significant
R.M+R.C.G.F	1.11%	7.80%	9.29%	383.4	Significant
R.C+R.M.G.F	0.16%	2.47%	3.92%	55.0	Significant
R.G+R.M.C.F	0.82%	6.57%	8.05%	281.9	Significant
M.C+G.F	0.40%	4.36%	5.82%	138.2	Significant
M.G+C.F	9.67%	-23.85%	-22.59%	3336.4	Significant
L.T.R	0.29%	-4.91%	-3.51%	99.2	Significant
L.T.G+L.T.M.C.F	0.21%	2.96%	4.42%	73.1	Significant
L.R.M+L.R.C.G.F	0.32%	3.81%	5.27%	110.1	Significant
L.M.G+L.C.F	0.64%	-6.94%	-5.56%	220.9	Significant
T.R.M+T.R.C.G.F	1.41%	-9.87%	-8.51%	484.8	Significant
T.R.G+T.R.M.C.F	0.43%	-5.84%	-4.45%	148.7	Significant
T.M.G+T.C.F	1.04%	7.51%	9.00%	358.3	Significant
R.M.G+R.C.F	1.22%	8.24%	9.73%	422.5	Significant
Error	1.11%				

The largest interaction between a traffic characteristic and a ADIM-NB algorithm is between the offered traffic and the ADIM-NB/PG algorithm, which is 6%. This is also consistent with both throughput and percentage of data delivered statistics. The next largest interaction is between traffic type and the ADIM-NB/PG algorithm. The delay will decrease if the ADIM-NB/PG algorithm is enabled and the traffic type is SMTP. This indicates that the ADIM-NB/PG algorithm favors SMTP traffic. Because SMTP traffic is usually destined to a single destination, at every transmit node, the packets will most likely arrive from the same node and depart to another node. Such a stream of packets tends to discourage new transmissions from neighboring nodes and allows the ADIM-NB/PG algorithm to operate with a higher efficiency because of the reduced interference. Because each packet does not have to wait at each intermediate node, the end-to-end delay is decreased. Throughput, on the other hand, does not improve because the network has limited capacity.

The interaction between traffic type and the ADIM-NB/MU algorithm indicates that the ADIM-NB/MU algorithm also favors SMTP traffic. Because SMTP traffic has a smaller packet size, which is almost half of the HTTP packet size, there is more opportunity to set up multiple connections between several source/destination pairs, resulting in an even lower delay. The ADIM-NB/MU algorithm can then be more effectively utilized.

The interaction between the offered load and the ADIM-NB/MU algorithm indicates that the ADIM-NB/MU algorithm slightly favors a less congested network. Since the ADIM-NB/MU algorithm attempts to set up multiple connections, a less congested network also means less interference, which can lead to a more successful negotiation rate. There are two main limitations in setting up multiple connections: the orthogonality of the spreading code and the number of simultaneous users that the PIC algorithm can track. Because the spread-spectrum processing gain is just 16 chips/bit and the maximum number of simultaneous users for the PIC algorithm is four, the network experiences some limitations in having a lot of simultaneous CDMA connections. In a production network, it is more likely that the processing gain will be higher and the PIC-capable radio will have a higher capacity.

The end-to-end delay is a sensitive variable, which makes the percentage of variation more distributed across several effects and interactions than the throughput statistic. The error in the ANOVA prediction model, however, is still relatively small, indicating a good match between the ANOVA model and the simulation result.

Table 7.29 presents the ANOVA results for the power consumption per bit received statistic. The power consumption statistic is also normalized by the radio speed, making it power consumption per bit received per unit bandwidth statistic. In particular, the power used per bit from the CDMA2000 radio is multiplied by 3.6, which is the ratio of the operating bandwidth between the CDMA2000 radio and the Virginia Tech reconfigurable radio. All interactions that result in a smaller than 0.1% variation are not shown to keep the table compact.

There are few interactions between traffic characteristic and the ADIM-NB algorithm. Most interactions are within the traffic characteristics themselves or within different ADIM-NB algorithms. The error in the ANOVA prediction model, however, is still relatively small, indicating a good match between the model and the simulation results. The largest such interaction is 0.64% between the radio type and the ADIM-NB/PC algorithm. As is consistent with the analysis from Table 7.26, such behavior indicates that the ADIM-NB/PC algorithm favors the slower Virginia Tech radio. Because noise power increases with operating bandwidth, the CDMA2000 radio experiences a higher level of Gaussian noise. As a result, the system may experience a lower average SNR value, resulting in a reduction in the dynamic range in which the ADIM-NB/PC algorithm can operate. Such behavior lends favor to the use of the slower Virginia Tech radio.

Table 7.29. ANOVA Joint Algorithm-Environment Results for Power Consumption.

Source (percentage of data delivered)	Percentage of variation	90% confidence interval		F-ratio	F-ratio/ F-variates
L (L6/L9)	0.14%	3.35%	5.06%	69.2	Significant
T (Ts/Th)	5.17%	-24.38%	-22.91%	2532.0	Significant
R (Rv/Rc)	0.20%	-5.65%	-4.02%	95.8	Significant
M+C.G.F	18.29%	-42.80%	-41.48%	8954.1	Significant
C+M.G.F	31.39%	-53.92%	-52.69%	15366.8	Significant
G+M.C.F	9.40%	-31.90%	-30.49%	4599.2	Significant
F+M.C.G	0.05%	-3.31%	-1.66%	25.1	Significant
L.T	0.14%	3.33%	5.04%	68.6	Significant
L.R	0.45%	-8.10%	-6.49%	221.1	Significant
L.M+L.C.G.F	0.23%	-6.06%	-4.44%	113.3	Significant
T.G+T.M.C.F	0.13%	3.11%	4.81%	61.4	Significant
R.M+R.C.G.F	0.62%	8.01%	9.76%	302.3	Significant
R.C+R.M.G.F	0.64%	8.20%	9.95%	315.0	Significant
M.C+G.F	0.39%	-7.56%	-5.94%	188.6	Significant
M.G+C.F	6.43%	-26.89%	-25.44%	3145.5	Significant
M.F+C.G	24.16%	61.40%	63.59%	11825.2	Significant
L.R.M+L.R.C.G.F	0.12%	-4.59%	-2.95%	58.1	Significant
L.R.C+L.R.M.G.F	0.12%	-4.63%	-2.99%	59.1	Significant
R.M.C+R.G.F	0.31%	5.36%	7.09%	150.2	Significant
R.M.F+R.C.G	0.22%	-5.91%	-4.28%	106.7	Significant
Error	0.78%				

There is another small interaction of 0.23% between the offered load level and the ADIM-NB/MU algorithm. Also shown in Table 7.28, since the network can have only a limited number of simultaneous connections, the ADIM-NB/MU algorithm performs better in a less congested network.

7.6 CAMEN

Although introducing yet another *ad hoc* routing protocol is not the main purpose of this research, it is interesting to look at the improvement from CAMEN and its scalability. To study the algorithm's scalability, a network with the same coverage area but a different number of stations is used. Three scenarios, having 50, 100, and 150 stations each, are studied. Since all components of the research are optimized with CAMEN enabled, any comparisons between enabling and disabling CAMEN represent the best case improvement and should be used cautiously. The network performance is then compared CAMEN enabled and disabled. The experiment uses parameters shown in Table 7.30.

Figure 7-30 shows the end-to-end network throughput statistic. With CAMEN enabled, the network throughput remains relatively constant across a different number of nodes in the network. Without CAMEN, however, the network throughput quickly degrades as the number of nodes increases, suggesting a scalability problem. Except for the fifty-node network, the throughput is higher when CAMEN is enabled than when it is disabled. Because the network is still relatively sparse in the fifty-node network, additional routers in the network help reduce the path length that each packet has to travel, resulting in a shorter path and increased network capacity. Such additional routers do not impose a significant penalty when the network density is low. They do, however, present a problem when the network density is high. In addition, the improvement when CAMEN is disabled is marginally better than when CAMEN is enabled, indicating a small effect.

Table 7.30. Levels Used in the Scalability Experimentation.

Factors	Levels
Offered load	95% of the requested traffic get delivered
Traffic type	SMTP [Pax94] traffic
Node speed	45 Kph
Movement pattern	City block
Radio type	The Virginia Tech reconfigurable software radio [SHA98]
ADIM-NB/PG	Enable
ADIM-NB/PC	Enable
ADIM-NB/MU	Enable
ADIM-NB/FE	Enable

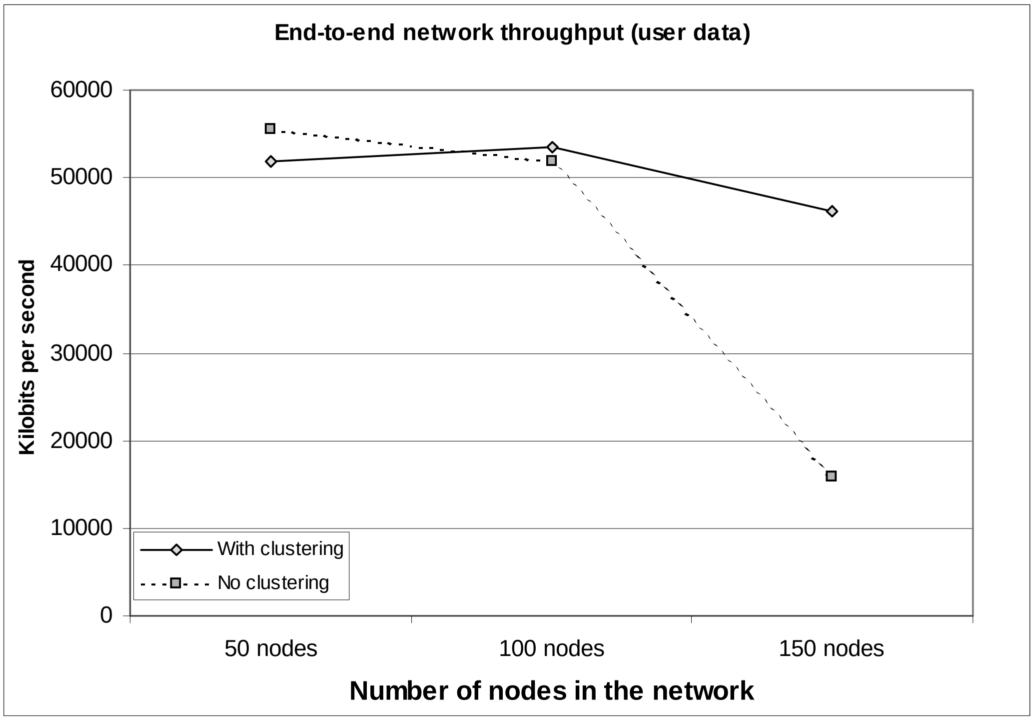


Figure 7-30. Network throughput (user data) comparison for the scalability study.

Other results also confirm the previous analysis. The network with CAMEN has a significantly lower end-to-end delay, as shown in Figure 7-31. The delay for a network with CAMEN increases linearly as the number of nodes increases, whereas the delay for a network without CAMEN increases exponentially, also suggesting that such a network is not scalable. In most cases, the performance of the network with CAMEN is as good or better than the one without CAMEN.

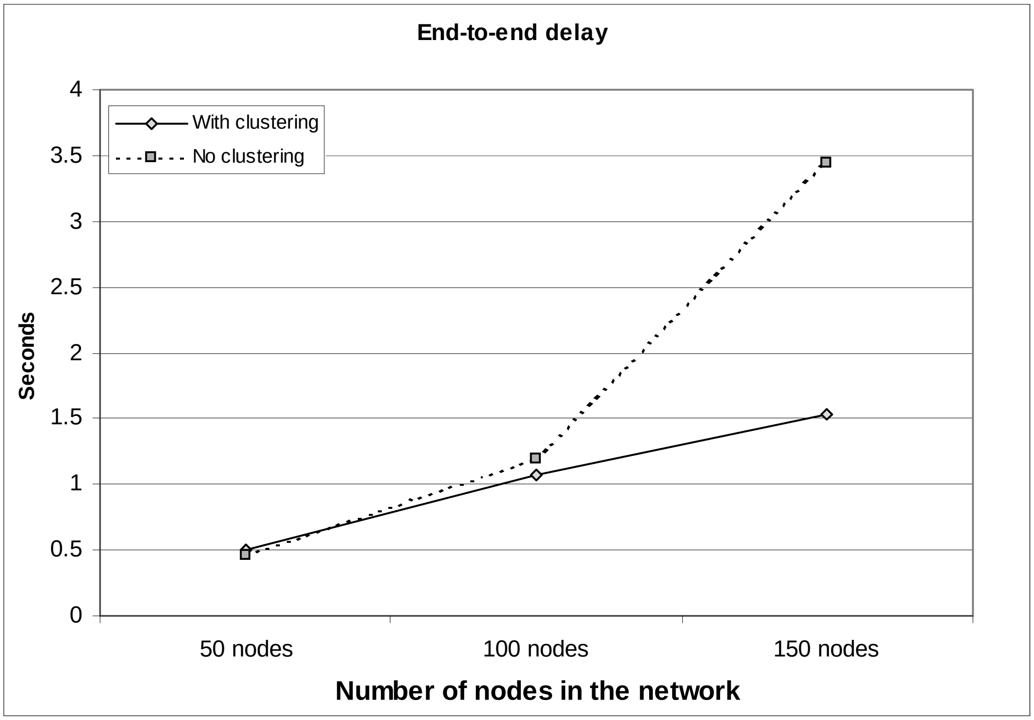


Figure 7-31. End-to-end delay comparison for the scalability study.

Figure 7-32 shows the average number of hop counts per packet. Because all simulations in this section have the same traffic characteristic, each packet transmission attempt takes the same path distance, on average, to reach its destination. The number of hop counts per packet, however, is different depending upon the path that each packet takes. The network with CAMEN uses more hops, on average, than the network without CAMEN because the former contains a smaller number of routers. The hop counts decrease slightly when the network density increases from fifty nodes to 100 nodes because an opportunity exists for a new router to appear. The number of routers becomes saturated and does not increase as the network density increases from 100 nodes to 150 nodes, indicating the scalability of the algorithm.

The network without CAMEN, however, experiences a decreased number of hops. The situation is a little bit different from the CAMEN network because all the nodes in the non-CAMEN network are routers. Since some requested packets may not be delivered to their destinations, a selective packet delivery process occurs. In a congested network, it is more likely that a packet heading to a nearby destination will be delivered than a packet destined to a distant destination. Consequently, the average hop count is reduced in such a network. Therefore, a higher hop count statistic indicates that the network is able to deliver packets to most destinations, whereas a lower hop count statistic indicates that packets destined for remote destinations cannot be delivered. The reduction in hop counts in the non-CAMEN network indicates the network's inability to deliver packets to distant destinations.

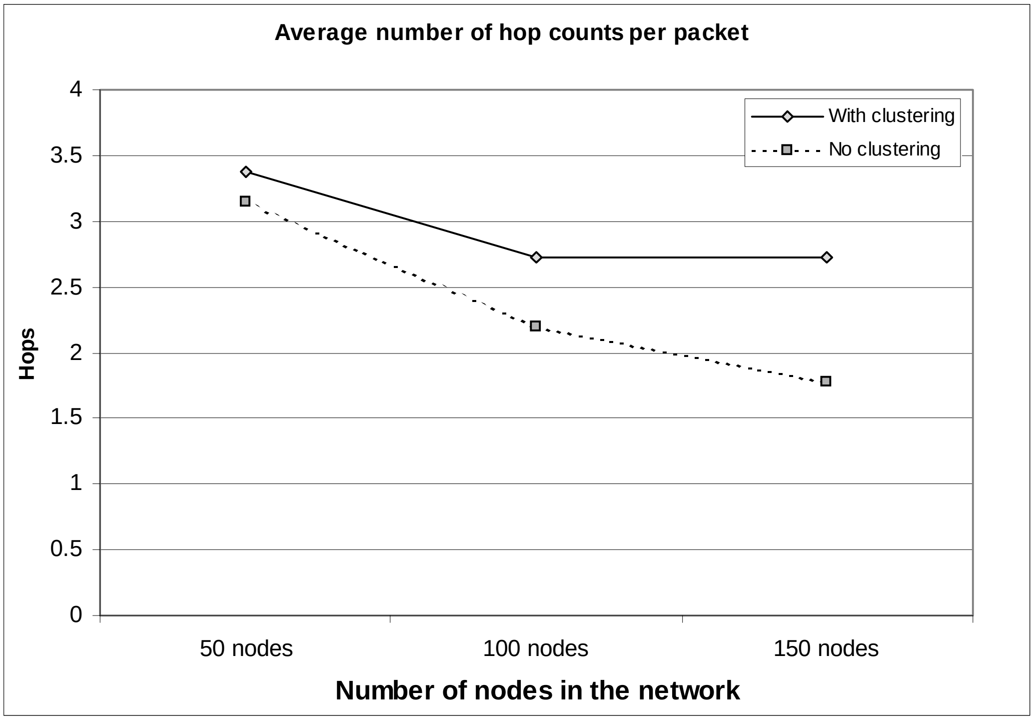


Figure 7-32. Hop counts comparison for the scalability study.

Figure 7-33 shows the percentage of routing and clustering overhead of all traffic received. The routing overhead of both CAMEN and non-CAMEN networks seems to increase linearly as the number of nodes increases. The routing traffic in the CAMEN network, however, increases at a slower rate. In both networks, the routing traffic is minimal when the network is sparse and increases rapidly as the network density increases.

Since CAMEN attempts to aggregate routing traffic and limits the number of active routers, the routing traffic should increase slowly. Figure 7-33, however, shows that this is not the case. As the network density increases, the network becomes congested, as also suggested by the end-to-end delay statistic. Such congestion leads to retransmission of routing information and changes in network connectivity. Both factors result in increased routing traffic. It is important to note that although the percentage of routing traffic increases, CAMEN is still able to deliver most of the traffic and its throughput remains relatively constant regardless of network density, as shown in Figure 7-30.

The non-CAMEN network generates a lot of routing information. At the extreme, this routing information amounts to almost all of the traffic flowing in the network. Clearly, such behavior is undesirable. In all scenarios, the networks with CAMEN enabled have a better performance than the networks with CAMEN disabled.

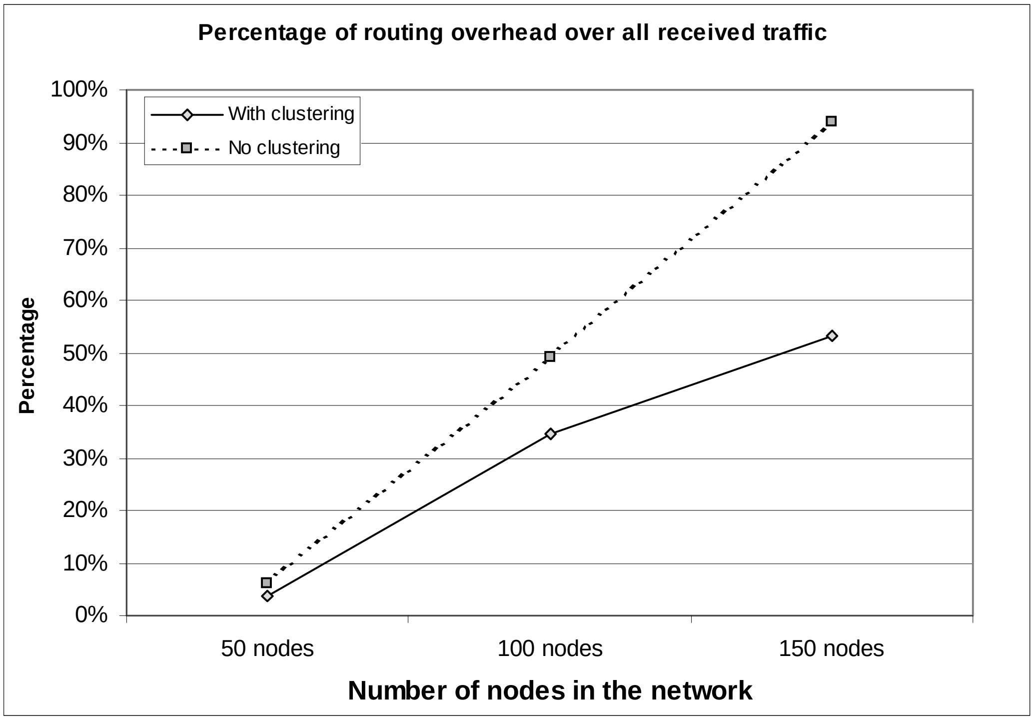


Figure 7-33. Percentage of routing overhead comparison for the scalability study.

Figure 7-34 shows the average retransmissions per packet statistic. The retransmission ratio is a direct result of network congestion. Since a packet is more likely to get lost in a congested network, the retransmission ratio is higher. The retransmission ratio for the CAMEN network increases slightly as the network density increases, and the retransmission ratio for the non-CAMEN network increases rapidly and becomes saturated in a dense network. As with earlier results, the CAMEN network has better performance than the non-CAMEN network.

Figure 7-35 shows the number of concurrent CDMA sessions for the entire network, which is small for the CAMEN network simply because there is no need for the extra connections and much higher for the non-CAMEN network because of all the extra routing traffic the network must handle in a dense network. The number of simultaneous connections slowly increases as the network density increases in the CAMEN network, partly because of the extra routing traffic and partly because of the increased congestion in the network. The same statistic, however, increases much more rapidly when the network density changes from 100 nodes to 150 nodes for the non-CAMEN network, indicating that the network is starting to become unstable. As previous results show, the CAMEN network has a better performance than the non-CAMEN network.

Figure 7-36 shows the power consumption of both networks. Again the CAMEN network degrades slightly as the number of nodes increases, whereas the non-CAMEN network becomes highly unstable when the number of nodes increases from 100 to 150 nodes. In all scenarios, the CAMEN network exhibits better performance than its counterpart.

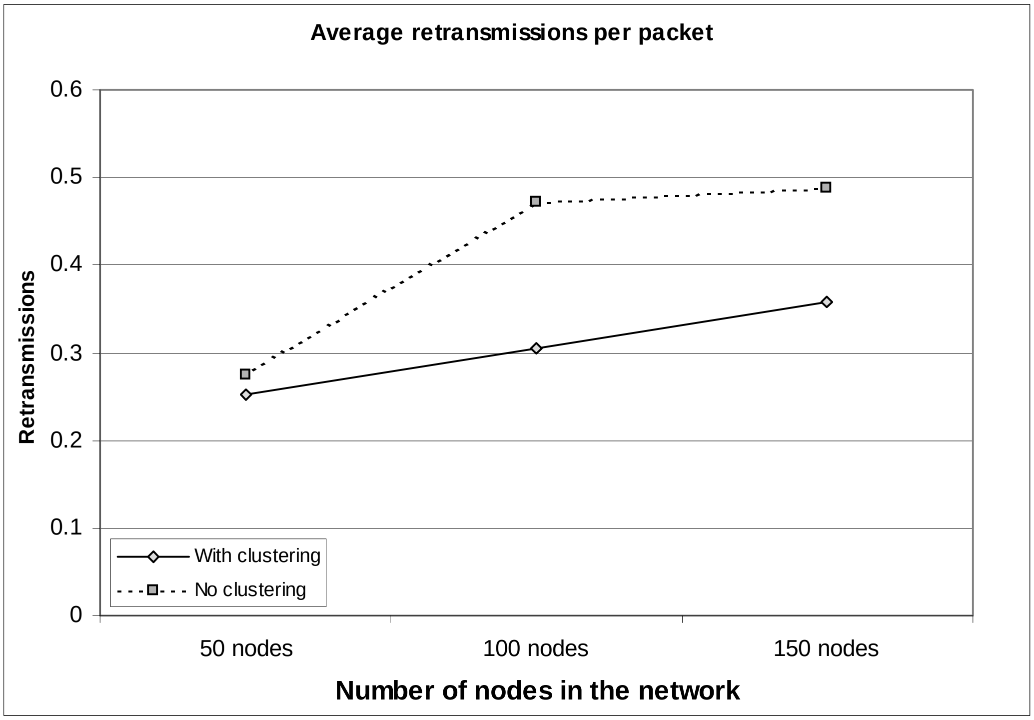


Figure 7-34. Average retransmission per packet comparison for the scalability study.

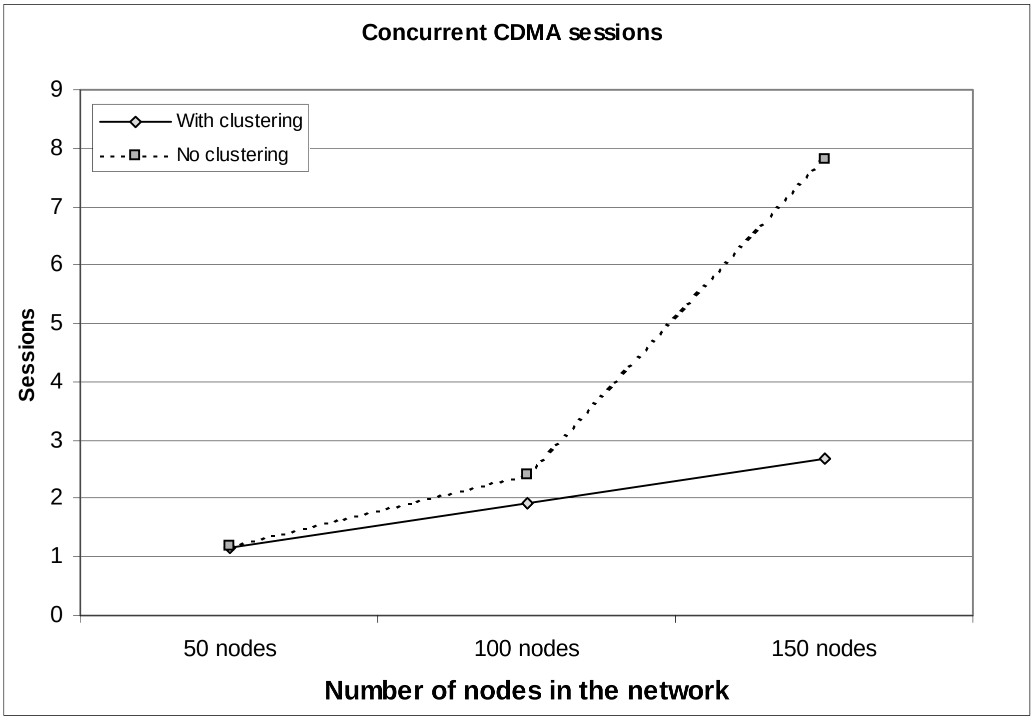


Figure 7-35. Concurrent CDMA sessions comparison for the scalability study.

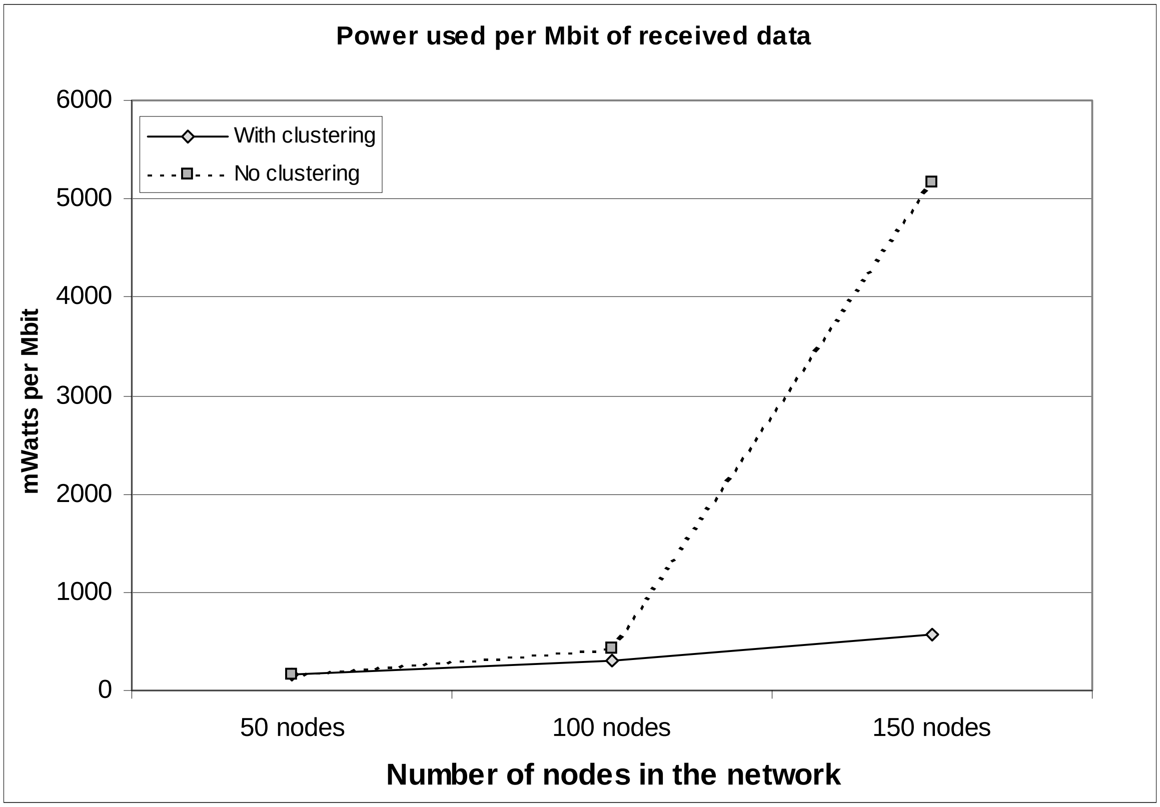


Figure 7-36. Power used per Mbit of received data comparison for the scalability study.

These simulation results confirm that CAMEN improves network scalability in all aspects studied. Although performance degrades as the network density increases, the degradation is graceful. As a reminder, all system components are optimized with CAMEN enabled, and any comparisons between enabling and disabling CAMEN represent a best case improvement and should be use cautiously.

7.7 Summary

Many simulations and experiments have been conducted to study the characteristics of both the ADIM-NB and CAMEN protocols. It was shown that both the ADIM-NB and CAMEN protocol perform better that their counterparts, the ADIM-NB/B protocol and the non-CAMEN network. Each algorithm contributes differently to the various aspects of the overall performance. Collectively, the network performance is substantially increased. The improvements from each algorithm of the ADIM-NB protocol were studied extensively. Their impact in different types of networks and environments is assessed. In the most optimal scenario, the ADIM-NB protocol provides as much as a 150% improvement in throughput over a regular ADIM-NB/B MAC protocol.

CAMEN also substantially improves the network scalability by aggregating nodes into clusters and reducing the number of routers in the network. It is able to maintain the same performance level when the number of nodes is tripled, whereas the non-CAMEN network exhibits several signs of instability.

Chapter 8. Conclusions

This research primarily attempts to explore the capabilities, strengths, and weaknesses of using multiple spreading codes in a single subnet and the associated performance trade-off between data rate and transmission ranges in a wireless mobile *ad hoc* network environment. The secondary goal is to optimize a medium access control protocol and an *ad hoc* routing protocol to take advantage of several new link-level techniques. This research also attempts to take advantage of all major opportunities that exist in the operating environment.

In this research effort, the network-level performance improvement from a radio equipped with two link-level techniques, an adaptive DSSS MAC protocol for a non-broadcast multiple access medium (ADIM-NB), and a four-user multi-user interference cancellation technique is presented. Through its environment characterization techniques, the ADIM-NB protocol improves several aspects of the wireless mobile *ad hoc* network performance, including throughput, delay, stability, and power consumption, using spread-spectrum multiple access and four different adaptive algorithms. A clustering algorithm for mobile *ad hoc* networks (CAMEN) is designed to augment the ADIM-NB protocol. With the ADIM-NB protocol in mind, CAMEN discourages the use of broadcast messages, supplements the ADIM-NB protocol's functionality at the network level, and improves the network scalability by aggregating nodes into clusters.

Simulation models have been developed and simulated to verify performance improvements as well as to provide a means to perform trade-off analysis. This chapter concludes the work performed during this dissertation and highlights significant achievements and extensions to the existing body of knowledge.

8.1 Summary

Chapter 1 provided an introduction to the motivation to develop ADIM-NB, a highly adaptive MAC protocol for a direct-sequence spread-spectrum multiple access environment. With the explosive popularity of personal wireless communication systems, there is an increasing need for a scalable, power efficient, and tightly integrated communication device that can adapt itself to different situations. To achieve those desirable features, a hierarchical routing protocol, CAMEN was developed to extend the ADIM-NB protocol's capability and functionality to the network layer. It also described the intended operating environment, which is the mobile *ad hoc* network, and the ultimate goals of this research.

A review of several important MAC protocols, *ad hoc* routing protocols, and link-level techniques were presented in Chapter 2. The popular medium access technique for contention-based networks, carrier sensing multiple access (CSMA), was described. Two wireless MAC protocols that are based on the CSMA technique were briefly discussed. The industry standard IEEE 802.11 wireless LAN uses CSMA with collision avoidance. The floor acquisition multiple access technique with pause and jamming tries to accomplish the same goal with a slightly different implementation. The baseline wireless routing protocol in which CAMEN is built upon was then discussed, along with two underlying core algorithms. Chapter 2 also discussed several active research topics related to wireless mobile *ad hoc* networks, including two MAC protocols and four routing protocols. Since the ADIM-NB protocol strives to take advantage of recent link-level technologies, the nature and characteristics of those technologies were also documented. Chapter 2 concluded with discussion of a well-known problem in a wireless *ad hoc* network environment, the hidden terminal problem, and a well-known problem in the direct-sequence spread-spectrum environment, the near-far problem. Although the ADIM-NB protocol does not manage to completely eliminate either of them, their severity is greatly reduced.

Chapter 3 presented the objectives of this research as well as the methodology employed to attain these objectives. Using a systematic approach to computer systems performance evaluation [Jai91], this chapter outlined the necessary steps to successfully complete an experimental evaluation of a networking system. Beginning with a problem definition that included bounding assumptions for the network, it delineated the scope of the research. The methodologies to conduct the experiment as well as to quantify the system performance were presented along with a high level overview of the operation of ADIM-NB.

A formal description of the problem was presented in Chapter 4, which also described important features and motivations for the opportunistic ADIM-NB MAC protocol. Inside the ADIM-NB MAC protocol, the baseline ADIM-NB/B algorithm establishes a unique communication channel for each packet by means of an RTS/CTS negotiation sequence. Every communication session in the network shares the same frequency spectrum using a different spreading code in a direct-sequence spread-spectrum (DSSS) CDMA system. The ADIM-NB/PG algorithm dynamically changes its processing gain by observing the current signal strength during the negotiation process. Secondly, the ADIM-NB/PC algorithm attempts to equate the received power from various sources to reduce the severity of the near-far problem, enhance the effectiveness of the PIC algorithm, and reduce the power consumption to an absolute minimum. This is achieved using the same principle as a regular cellular telephone’s power control algorithm. The third feature, the ADIM-NB/FE adaptive algorithm, dynamically changes the FEC coding rate based on the BER of the incoming packets. It is designed to enable acceptable performance in the presence of a varying number of jamming nodes and interference levels and to eliminate the need to manually adjust the communication parameters. Finally, the ADIM-NB/MU algorithm enables multiple simultaneous connections among multiple neighbor, to encourage multiplexing in a spread-spectrum environment and to increase effectiveness of the multi-user interference cancellation algorithm. All adaptive algorithms are relatively independent from each other and can be enabled and disabled separately.

Chapter 5 described the proposed hierarchical routing protocol CAMEN, the clustering algorithm for mobile *ad hoc* networks. It began with a formal description of the problem, important features, and motivation for the routing protocol. The motivation for CAMEN is to address new behaviors and features of the ADIM-NB protocol, augment the ADIM-NB protocol’s functionality at the network layer, and improve network scalability. CAMEN is based on a clustering concept. It aggregates several nodes within the same physical area into a cluster. A subset of nodes is then selected to be routers, which will dynamically create and maintain the backbone route. The other nodes remain passive by neither routing packets nor advertising their connectivity. All nodes within the same cluster can share a single route entry in the routing table, resulting in a smaller routing table. The reduced number of routers also results in a smaller number of nodes that have to advertise routing information and exchange route update messages, which significantly decreases the amount of routing traffic and greatly reduces the dependency of the routing protocol on the radio broadcast functionality. Additional features include the provision of a secondary route when the primary route fails, detection of a redundant cluster, delayed route removal algorithm, and quick response to set up a replacement route, among others. With the ADIM-NB protocol in mind, CAMEN communicates a significant number of operating parameters to the ADIM-NB protocol, and vice versa. This information includes SNR, BER, collision rate, and error correction coding rate at other nodes. In addition to the detailed specifications, examples of its operation were also provided throughout the chapter.

The simulation environment was presented in Chapter 6. This chapter discussed the simulation of a wireless network. The entire wireless MAC and routing protocols are modeled via simulation including the ADIM-NB MAC protocol, the CAMEN routing protocol, and the underlying DSSS physical layer. OPNET Modeler®/Radio was identified as the simulation tool used to construct, simulate, and analyze the system performance. Various aspects of the simulation experiments were discussed, including simulation model parameters and performance metrics. The methodology underlying the simulation design, simulation fidelity and appropriate simulation duration as well as the technique to ensure consistent results were discussed. The chapter concluded with a discussion of model verification and validation.

Chapter 7 presented the results of the simulation. It was shown that the adaptive ADIM-NB MAC protocol performed better than the non-persistent CSMA MAC protocol and the non-adaptive ADIM-NB/B MAC protocol in various network congestion states. The majority of the results in Chapter 7 characterized the ADIM-NB MAC protocol in various environment conditions. First, because the ADIM-NB protocol consists of four individual adaptive algorithms, and each has a different level of interaction with the other algorithms, the impact of each adaptive algorithm and their correlation are quantified. It was shown that each adaptive algorithm affects different performance metrics differently as is briefly summarized in Section 8.2. Collectively, the network performance is substantially increased. The improvements from each algorithm of the ADIM-NB protocol are extensively studied. The ADIM-NB/FE algorithm was put through a jamming scenario to determine its effectiveness in an environment with a dynamic interference. The performance of the ADIM-NB protocol was then collected using a large number of environmental conditions and traffic characteristics to characterize the protocol behavior and performance across various conditions. By combining all experiments together, the influence of each environmental factor on each ADIM-NB adaptive algorithm is then quantified. Chapter 7 concluded with a comparison between the hierarchical routing protocol, CAMEN, and its non-hierarchical counterpart. CAMEN also substantially improves network scalability by aggregating nodes into clusters and reducing the number of routers in the network. CAMEN is able to maintain the same performance level when the number of nodes triples, whereas the non-CAMEN network exhibits several signs of instability.

8.2 Research Findings

The simulation results show that by using a unique CDMA spreading code in each packet, as in the ADIM-NB/B MAC protocol, the network throughput can be doubled from a non-persistent CSMA MAC protocol in a moderately to heavily loaded environment. It is also shown to be more stable than the CSMA MAC protocol. As an additional benefit, by adaptively changing communication parameters using the ADIM-NB MAC protocol, the network throughput increases by another 50% to 100% over a system with fixed communication parameters in the same type of environment. The delay statistic also shows the same amount of improvement. Although not by the same factor, the ADIM-NB protocol also shows significant performance improvements in terms of network connectivity and reductions in the number of retransmissions and in routing traffic overhead.

Table 8.1 shows the average improvements for four different performance metrics from each ADIM-NB adaptive algorithm over the ADIM-NB/B MAC protocol. The ADIM-NB/PG algorithm provides the greatest improvement. Although the ADIM-NB/PG algorithm comes at the expense of additional overhead from the RTS/CTS negotiation sequence, this overhead significantly improves the protocol’s stability and capacity. In a congested network, the RTS/CTS negotiation sequence leads to the loss of a small RTS packet rather than a larger data packet.

Table 8.1. Summary of Improvements from Each ADIM-NB Algorithm.

	ADIM-NB/PG	**ADIM-NB/MU**	**ADIM-NB/FE**	**ADIM-NB/PC**
End-to-end network throughput	31%	5%	3%	-1%
Power consumption per data delivered	-33%	-4%	-4%	-17%
End-to-end delay	-10%	-2%	-1%	0%
Percentage of data received	21%	3%	2%	-1%

Although the ADIM-NB/PC algorithm slightly decreases network throughput and the percentage of data delivered, it significantly decreases the power consumption. The benefit of using the power control algorithm seems to outweigh the penalty associated with it. There are two primary reasons that the ADIM-NB/PC algorithm does not improve the network throughput. First, the locations of all transmitters and receivers in an *ad hoc* network are completely random, making it difficult to maintain constant received power at any particular node by just controlling the transmit power. Secondly, since the multi-user PIC algorithm can mitigate any non-drastic differences in the received power from multiple stations, controlling the transmitted power does not result in a big improvement in the reception statistics.

Although by different factors, both the ADIM-NB/MU and ADIM-NB/FE algorithms improve every aspect of the system performance. In a special jamming scenario, the ADIM-NB/FE algorithm is shown to improve the network throughput by up to 68%, depending on the interference power. Because the design objective of the ADIM-NB/FE algorithm is to allow it to operate in an environment with dynamic interference, rather than a throughput increase in a static environment, it performs well in such a scenario while still maintaining the same performance level in a static scenario.

The amount of offered traffic, the different types of traffic, and the control channel speed also have some impact on the network performance. Node speed, movement pattern, and network density, on the other hand, do not have a large impact on the network performance. The effects of each adaptive algorithm or each environmental factor seem to be relatively independent from one another. However, there are some cases in which the improvement from one algorithm depends on the availability of another algorithm or on certain network characteristics. Such cases were discussed in Sections 7.4 and 7.5.

The hierarchical *ad hoc* routing protocol CAMEN also substantially improves network scalability by aggregating nodes into clusters and reducing the number of routers in the network. It is able to maintain the same performance level when the number of nodes triples, while the non-CAMEN network exhibits several signs of instability and can only deliver one-third of the traffic delivered by CAMEN.

8.3 Contributions

The performance of an adaptive processing gain algorithm or ADIM-NB/PG is an active research topic. However, most studies have focused on the link-level or subnet-level performance, not on application-level performance in an *ad hoc* network environment as studied in this research. The algorithm’s impact at the application layer can be significantly different from those at the link layer. In addition, its improvement also depends on various network topologies and traffic characteristics, as demonstrated in Chapter 7. The findings of this study will, it is hoped, enable future work that relies on application-level information, such as a resource reservation protocol or adaptive applications that can adapt their response to changing network capacity.

A traditional *ad hoc* routing protocol usually relies on the broadcast ability of the radio. This assumption can be a significant limitation in functionality for some highly capable radios that employ CDMA techniques. For these radios, having to operate in a broadcast mode severely impairs their functionality. A radio with multi-user interference rejection capability, for example, is unable to fully exercise its sophisticated decoding algorithm in a strictly broadcast environment. The ADIM-NB and CAMEN protocols address this problem by limiting the amount of broadcast information to a minimum level and encouraging usage of multiple connections, each with a different spreading code. As a result, such a radio will be able to fully exercise its functionality in an *ad hoc* network environment. Furthermore, by assigning a distinct spreading code for each session, the network can be more secure than the one that relies on a single spreading code.

The half-duplex nature of the wireless mobile *ad hoc* network makes it difficult to negotiate multiple simultaneous on-demand connections, therefore limiting the performance of a multi-user receiver-capable radio. The ADIM-NB protocol attempts to overcome this limitation by synchronizing the channel setup process among multiple nodes so that a single node can simultaneously receive multiple transmissions from multiple sources and, therefore, fully utilize its multi-user receiver capability.

Although several recent studies propose to use a clustering technique to reduce the amount of routing information in the wireless mobile *ad hoc* network environment, most of them still require the participation of a regular node in the routing algorithm, either in the route establishment process or the packet forwarding task. These nodes still have to process a significant amount of information, route packets, periodically transmit connectivity information, or respond to a route request. In addition, few of them, if any, use a routing protocol based on a distance-vector algorithm, which is more suitable to the nature of the clustering algorithm. The CAMEN protocol attempts to completely isolate a regular node from the routing algorithm yet still allows the node to participate if necessary. The CAMEN protocol does not mandate a regular node to process any routing information, participate in the route construction process, or route packets. Consequently, a regular node can be much simpler and use a minimum amount of battery power. The dynamic clustering algorithm also allows a regular node to be promoted to a routing node when it is necessary to maintain the network connectivity and allows a routing node to be demoted when it is appropriate.

8.4 Potential Extensions of Current Research

This section describes possible extensions of the current research. Although notable, the work presented in this dissertation represents only a small fraction of the challenges presented by wireless networking. During the course of this research, additional areas of investigation have presented themselves, but have fallen outside the scope of this effort. Many extensions have both potential positive and negative features, and the combined benefit is unclear without extensive investigation.

8.4.1 Collision

All stations in the network operate independently from each other, which reduces complexity and increases stability. A cooperative approach similar to the FAMA-PJ [FuG95b] algorithm can be applied to provide additional protection against the hidden-terminal problem. With FAMA-PJ, stations that hear either an RTS or a CTS packet, to indicate the start of the data transmission, from other stations are requested to pause for an appropriate amount of time before attempting transmission. However, the FAMA-PJ algorithm has limited utility in a multi-user environment because each node now has the ability to simultaneously set up multiple connections. Moreover, this technique requires stations to send a CTS packet over the control channel, which will degrade the CTS success rate to that of an RTS and will increase the congestion probability in the presence of a slow control channel.

Nevertheless, it is possible that additional performance gain can be obtained by increasing the level of coordination among several nodes in the same physical area. Each node can maintain a database for each of its neighbors. Using that database, it is possible to determine the period in which any particular neighbor is unable to receive any connection requests or the period in which its neighbor is expecting to receive connection requests. Implementing a time-slotted system would also increase the level of coordination among various nodes, which may result in an additional performance gain for the system.

8.4.2 Quality of Service (QoS) Extension

The clustering technique underlying CAMEN has the primary purpose of improving scalability. Because of the independence among sessions and the unsynchronized nature of the network, QoS can be provided as an additional benefit with a reasonable effort. This QoS support can be a pre-arrangement that instructs each user to periodically use a pre-negotiated spreading code to start a new session because a new session does not disrupt other existing sessions. Any reasonable transmission speed can be guaranteed to each node to provide different classes of service. This technique also eliminates subsequent negotiation overhead for a long session. Further integration between the MAC and routing protocols would allow the creation of permanent virtual circuit (PVC) connections that automatically reroute themselves when links get disconnected. Several other approaches to address QoS can be done by using different minimum received power, processing gains, timer values, or frame sizes for each class of service. QoS is just one example of how the independence of sessions in a CDMA system makes it easier to extend the protocol’s functionality. QoS support, however, must be carefully designed to preserve the scalability and stability of the protocol because of the impacts of added state information and coordination.

8.4.3 Situations that Degrade Performance

Three problems remain unsolved: bottleneck nodes, hidden terminals, and the near-far problem. Although the severity of these problems has been greatly reduced, the problems are not eliminated. Additional investigation is required to further reduce the effect of these problems or to eliminate them altogether. This includes further study of scenarios that intensify the effect of these problems, which may suggest possible solutions. Other less severe problems should be identified during the same process.

8.4.4 Non-Overlapping Cluster Problem

If two regular nodes separate cluster heads, they cannot communicate with each other via these two nodes. This problem arises from the fact that nodes do not talk to each other directly, even though there is a physical path between them. Consider the example shown in Figure 8-1.

Because the route advertisement containing cluster information will be made only by a cluster head, node B does not know of the existence of the D cluster and node C is not aware of the A cluster. For node B to set up a connection with node C, node C has to inform node B about the D cluster. This basic information is required, no matter what strategies are chosen to set up the B-C connection. For node C to propagate this information, it must periodically advertise its connectivity information. Initially, node C is a regular node. Therefore, this argument requires all regular nodes to periodically broadcast their connectivity, which contradicts the previous assumption that a regular node does not participate in any routing activities.

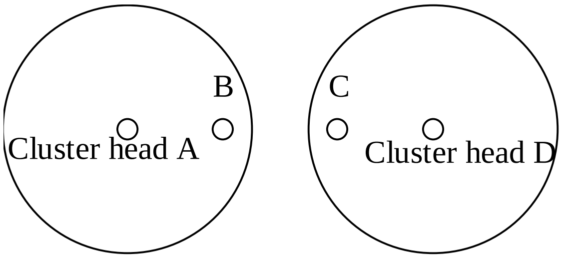


Figure 8-1. Example of the non-overlapping cluster problem.

After a certain duration, it is possible that nodes B and C will be aware of the situation and connect these two clusters together, especially if one of them is already a gateway to another cluster. If node B actively processes the distance vector routing information from node A, it may find out that node A uses more than three hops to connect to node D and node C is a member of cluster D. With both pieces of information, node B can establish a connection with node C to link two clusters together. However, the current protocol does not require the member nodes, B and C, to process any routing information. Consequently, this connection will not occur. Instead, CAMEN encourages a greater number of clusters than necessary to reduce the probability of this situation.

8.4.5 Smart Antennas

One concern that has arisen is CAMEN's applicability in an environment where antenna pattern does not follow a geometric shape or varies rapidly. Such a problem becomes increasingly important with the availability of the smart antenna because the antenna pattern can vary rapidly over time. Another related problem is the existence of a unidirectional link.

Although CAMEN can handle irregular antenna patterns as long as the pattern changes slowly and all links are bi-directional, it does not perform well when the pattern changes rapidly. An additional algorithm to isolate such temporary changes in connectivity, so as to minimize the changes in network topology, is necessary. Perhaps the most challenging problem is that the same algorithm should react quickly to any permanent changes in the network. Some possible solutions include a tightly integrated protocol stack that allows the routing protocol to control its antenna pattern. Any extended receiver-related information such as global positioning system (GPS) data, current antenna pattern, or the direction of the communicating party would be helpful to the routing protocol to distinguish temporary changes from the permanent changes in the network topology.

8.5 Conclusions

This dissertation investigated the impact of the adaptive medium access control protocol in a non-broadcast multiple access environment, or ADIM-NB. It focused on a wireless mobile *ad hoc* network environment using a radio equipped with multi-user receiver interference cancellation functionality. The ADIM-NB protocol establishes a unique communication channel for each packet and dynamically changes its communication parameters by observing the current environment condition. The strength of the ADIM-NB MAC protocol results from its underlying CDMA technology, which employs an RTS/CTS negotiation sequence to set up unique communication parameters for each session. This technique improves adaptation, flexibility, and stability of the overall system. At the network layer, CAMEN, a hierarchical extension to a wireless mobile *ad hoc* routing protocol, improves the network scalability, exploits new features introduced by the ADIM-NB protocol, and reduces the dependency of the routing protocol on radio broadcast functionality.

Traditionally, a DSSS CDMA radio is ill-suited to the MANET environment because of the near-far problem. Instead, a frequency hopping spread-spectrum CDMA radio using some form of centralized access control is more desirable. With the availability of multi-user interference rejection enabled radios, in conjunction with specially designed MAC and routing protocols, a DSSS CDMA radio becomes a viable alternative for a MANET environment.

Although primarily targeted to a tactical wireless mobile *ad hoc* network using DSSS radio, the techniques and protocols developed do not limit themselves to this environment. Other wireless communications systems that utilize the dynamic nature of the communication channel can benefit from this research as well.

Chapter 9. Bibliography

BaE81	Dennis J. Baker and Anthony Ephremides, “The Architectural Organization of a Mobile Radio Network via a Distributed Algorithm,” *IEEE Transactions on Communications*, Vol. COM-29, No. 11, pp. 1694-1701, November 1981.
BaR99	Rusty Olen Baldwin, “Improving the Real-time Performance of a Wireless Local Area Network,” Ph.D. Dissertation, Bradley Department of Electrical and Computer Engineering, Virginia Polytechnic Institute and State University, Blacksburg, VA, May 1999. (Available at http://scholar.lib.vt.edu/theses/available/etd-062499-102718/).
BaW94	R. L. Bagrodia and Liao Wen-Toh, “Maisie: a Language for the Design of Efficient Discrete-Event Simulations,” *IEEE Transactions on Software Engineering*, Vol. 20, No. 4, pp. 225-238, April 1994.
BeG92	D. Bertsekas and R. Gallager, *Data Networks*, Prentice Hall, Upper Saddle River, New Jersey, 1992.
BeG98	J. Behrens and J. J. Garcia-Luna-Aceves, “Hierarchical Routing Using Link Vectors,” *Proceeding of IEEE INFOCOM 98*, Vol. 2, pp. 702-710, 1998. (Available at http://www.cse.ucsc.edu/research/ccrg/publications/jochen.infocom98.pdf).
BeL97	D. A. Beyer and M. G. Lewis, “A Packet Radio API,” *IEEE MILCOM 97*, Vol. 3, pp. 1261-1265, 1997.
BGT93	C. Berrou, A. Glavieux, and P. Thitimasjshima, “Near Shannon Limit Error-Correcting Coding and Decoding: Turbo-codes (1),” *Proceeding of the IEEE International Conference on Communications*, Vol. 2, pp. 1064-1070, 1993.
BMJ98	Josh Broch, David A. Maltz, David B. Johnson, Yih-Chun Hu, and Jorjeta Jetcheva, “A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols,” *Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom’98)*, pp. 85-97, 1998. (Available at http://www.monarch.cs.cmu.edu/monarch-papers/mobicom98.ps).

BMT98	R. Bagrodia, R. Meyer, M. Takai, Chen Yu-An, Zeng Xiang, J. Martin, and Song Ha Yoon, "Parsec: a Parallel Simulation Environment for Complex Systems," *Computer*, Vol. 31, No. 10, pp. 77-85, October 1998.
CBC95	Carlos R. Cunha, Azer Bestavros, and Mark E. Crovella, "Characteristics of WWW Client-Based Traces," Computer Science Department, Boston University. (Available at http://www.w3.org/WCA/Reports/1998-01-PDG-answers.htm.)
CBW99	Neiyer S. Correal, Michael Buehrer, and Brian D. Woerner, "A DSP-Based DS-CDMA Multi-user Receiver Employing Parallel Interference Cancellation," *IEEE Journal on Selected Areas in Communications*, Vol. 17, No. 4, pp. 613-630, April 1999.
CoM85	George R. Cooper and Clare D. McGillem, *Modern Communications and Spread-Spectrum*, McGraw-Hill, Inc., New York, New York, 1985.
DHM98	Du Yonggang, Herrmann Christoph, May Peter Klaus, Hulyalkar N. Samir, and Evans David, "Wireless ATM LAN With and Without Infrastructure," *IEEE Communications Magazine*, Vol. 36, No. 4, pp. 90-95, April 1998.
DSY99	Kim Dongkyun, Ha Seokjae, and Choi Yanghee, "Cluster-Based Dynamic Source Routing Protocol in Wireless Ad-Hoc Packet Radio Network With Variable-Sized Clusters and Variable Transmission Ranges," *1999 IEEE 49th Vehicular Technology Conference*, Vol. 3, pp. 2441-2445, 1999.
Esc95	Julio Escobar, "Run-Away Dynamics of CDMA Channels in an Adaptive Packet Radio Network," *Wireless Networks*, Vol. 1, No. 1, pp. 37-46, 1995.
Fal99	K. Fall, "Network Emulation in the VINT/NS Simulator," *Proceedings IEEE International Symposium on Computers and Communications*, pp. 244-250, 1999.
FuG95a	C. L. Fullmer and J. J. Garcia-Luna-Aceves, "Floor Acquisition Multiple Access (FAMA) for Packet-Radio Networks," *ACM Computer Communication Review*, Vol. 25, No. 4, pp. 262-273, October 1995. (Available at http://www.cse.ucsc.edu/research/ccrg/publications/chane.sigcomm.pdf.)
FuG95b	C. L. Fullmer and J. J. Garcia-Luna-Aceves, "FAMA-PJ: A Channel Access Protocol for Wireless LANs," *Proceedings of The First Annual International Conference on Mobile Computing and Networking*, pp. 76-85, 1995. (Available at http://www.cse.ucsc.edu/research/ccrg/publications/chane.mcn95.pdf).
GaM97	J. J. Garcia-Luna-Aceves and S. Murthy, "A Path Finding Algorithm for Loop-Free Routing," *IEEE/ACM Transactions on Networking*, Vol. 5, No. 1, pp. 148-160, February 1997. (Available at http://www.cse.ucsc.edu/research/ccrg/publications/shree.ton.pdf).

Gar89 J. J. Garcia-Luna-Aceves, “A Minimum-Hop Routing Algorithm Based on Distributed Information,” *Computer Networks & ISDN Systems,* Vol. 16, No. 5, pp. 367-382, May 1989.

Haa97 Z. J. Haas, “A New Routing Protocol for the Reconfigurable Wireless Networks,” *1997 IEEE 6th International Conference on Universal Personal Communications,* Vol. 2, pp. 562-566, 1997.

HaP97 Shinsuke Hara and Ramjee Prosad, “Overview of Multicarrier CDMA,” *IEEE Communication Magazine,* Vol. 35, No. 12, pp. 126-133, December 1997.

IEE97 Editors of IEEE 802.11, “Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Draft Standard 802.11, P802.11/D6.1,” Institute of Electrical and Electronics Engineers, New York, 1997.

IEE99 Editors of IEEE 802.11, “Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Std 802.11, 1999 Edition,” Institute of Electrical and Electronics Engineers, New York, 1999.

IET00 Internet Engineering Task Force, 2000. (Available at http://www.ietf.org).

ITU99 Editors of ITU-R, “Physical Layer Standard for cdma2000 Spread Spectrum Systems, TIA PN-4694 3GPP2 C.P0002-A, Draft Baseline Version,” International Telecommunication Union, Geneva, Switzerland, October 1999.

Jai91 R. Jain, *The Art of Computer Systems Performance Analysis*, John Wiley and Sons, Inc., New York, 1991.

JLT99 M. Jiang, J. Li, and Y. C. Tay, “Cluster Based Routing Protocol (CBRP) Functional Specification,” draft-ietf-manet-cbrp-spec-01.txt, August 1999 (work in progress).

Joh94 David B. Johnson, “Routing in Ad Hoc Networks of Mobile Hosts,” *Proceedings of the IEEE Workshop on Mobile Computing Systems and Applications*,” pp. 158-163, December 1994.

KaA97 B. Kahne and P. Athanas, “Stream Synthesis for a Wormhole Run-Time Reconfigurable Platform,” *7th International Workshop on Field-Programmable Logic and Applications, FPL '97,* pp. 101-110, 1997.

Kle76 Leonard Kleinrock, *Queueing System, Volume 2, Computer Applications*, John Wiley & Sons, Inc., New York, 1976.

KlS87 Leonard Kleinrock and John Silvester, “Spatial Reuse in Multi-hop Packet Radio networks,” *Proceedings of the IEEE,* Vol. 75, No. 1, pp. 156-167, January 1987.

LiG97a	Chunhung Richard Lin and Mario Gerla, “Adaptive Clustering for Mobile Wireless Networks,” *IEEE Journal on Selected Areas in Communications,* Vol. 15, No. 7, pp. 1265-1275, September 1997.
LiG97b	Chunhung Richard Lin and Mario Gerla, “Asynchronous Multimedia Multihop Wireless Networks,” *Proceedings of IEEE INFOCOM '97*, Vol. 1, pp. 118-125, 1997.
MDM97	Barry E. Mullins, Nathaniel J. Davis, IV, and Scott F. Midkiff, “An Adaptive Wireless Local Area Network Protocol That Improves Throughput via Adaptive Control of Direct-sequence Spread-spectrum Parameters,” *Mobile Computing and Communications Review*, Vol. 1, No. 3, pp. 9-16, September 1997.
MIL97	MIL 3, Inc., “OPNET Modeler Manual,” Washington, DC, 1997.
MLA98	J. G. Markoulidakis, G. L. Lyberopoulos, and M. E. Anagnostou, “Traffic Model for Third Generation Cellular Mobile Telecommunication Systems,” *Wireless Networks*, Vol. 4, No. 5, pp. 389-400, 1998.
MuG96	S. Murthy and J. J. Garcia-Luna-Aceves, “An Efficient Routing Protocol for Wireless Networks,” *ACM Mobile Networks and Applications Journal,* Vol. 1, No. 2, pp. 183-197, October 1996. (Available at http://www.cse.ucsc.edu/research/ccrg/publications/shree.winet.pdf).
Mul97	Barry E. Mullins, “CATER: An Opportunistic Medium Access Control Protocol for Wireless Local Area Networks,” Ph.D. Dissertation, Bradley Department of Electrical and Computer Engineering, Virginia Polytechnic Institute and State University, Blacksburg, VA, June 1997.
Pax94	V. Paxson, “Empirically Derived Analytic Models of Wide-Area TCP Connections,” *IEEE/ACM Transactions on Networking*, Vol. 2, No. 4, pp. 316-336, August 1994.
PeR95	Paul Petrus and Jeffrey H. Reed, “Time Dependent Adaptive Arrays,” *IEEE Signal Processing Letters*, Vol. 2, No. 12, pp. 219-222, December 1995.
PMC98	V. D. Park, J. P. Macker, and M. S. Corson, “Applicability of the Temporally-Ordered Routing Algorithm for Use in Mobile Tactical Networks,” *IEEE MILCOM 98,* Vol. 2, pp. 426-430, 1998.
Pos81a	J. Postel, “Internet Protocol - DARPA Internet Program Protocol Specification,” RFC 791, USC/Information Sciences Institute, September 1981.
Pos81b	J. Postel, “Transmission Control Protocol - DARPA Internet Program Protocol Specification,” RFC 793, USC/Information Sciences Institute, September 1981.

RaS98	R. Ramanathan and M. Steenstrup, “Hierarchically-Organized, Multihop Mobile Wireless Networks for Quality-of-Service Support,” *Mobile Networks & Applications*, Vol. 3, No. 1, pp. 101-119, 1998.
RoM50	A. L. de Rosa and M. Rogoff., Sec. I (Communications) of Application of Statistical Methods to Secrecy Communication Systems, Proposal 946, Federal Telecommunication Lab., Nutley, NJ, August 28, 1950.
Ron96	Zhigang Rong, “Simulation of Adaptive Array Algorithms for CDMA Systems,” M.S. Thesis, Bradley Department of Electrical and Computer Engineering, Virginia Polytechnic Institute and State University, Blacksburg, VA, September 1996.
Roo96	Rooftop Communications Corporation, “C++ Protocol Toolkit,” 1996. (Available at http://www.rooftop.com/rnd.shtml#CPT).
SAI99	SAIC, Inc., “SEAMLSS Project,” San Diego, California, 1999. (Available at http://www.seamlss.com).
SDB98	R. Sivakumar, B. Das, and V. Bharghavan, “Spine Routing in Ad Hoc Networks,” *Cluster Computing*, Vol. 1, No. 2, pp. 237-48, 1998.
SHA98	S. Swanchara, S. Harper, and P. Athanas, “A Stream-Based Configurable Computing Radio Testbed,” *Proceedings of the IEEE Symposium on FPGAs for Custom Computing Machines*, pp. 40-47, 1998.
Skl88	Bernard Sklar, *Digital Communications*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1988.
SmM00a	Natt Smavatkul and Scott F. Midkiff, “Inter-Organization Model Development and Evaluation Using SEAMLSS-Lite.” *OPNETWORK 2000*, August 28-September 1, 2000, Washington, DC, 6 pages. (Available at http://www.opnet.com/opnetwork2000/).
SmM00b	Natt Smavatkul and Scott F. Midkiff, “Impact of High-End Radio Functionality in Wireless Mobile Ad Hoc Networks,” *IEEE Workshop on Multiaccess, Mobility and Teletraffic for Wireless Communications*, December 3-6, 2000, Duck Key, Florida, 12 pages, to appear.
SRI98	SRI International, “GloMo Home Page,” May 1998. (Available at http://www.glomo.sri.com).
Sta94	William Stallings, *Data and Computer Communications, Fourth Edition*, Macmillan, New York, New York, 1994.
StT89	J. S. Storey and F. A. Tobagi, “Throughput Performance of an Unslotted Direct-Sequence SSMA Packet Radio Network,” *IEEE Transactions on Communications*, Vol. 37, No. 8, pp. 814-823, August 1989.

Swa98 Steven F. Swanchara, “An FPGA-Based Multi-user Receiver Employing Parallel Interference Cancellation,” M.S. Thesis, Bradley Department of Electrical and Computer Engineering, Virginia Polytechnic Institute and State University, Blacksburg, VA, July 1998.

TIA95 Telecommunications Industry Association, “Mobile Station-Base Station Compatibility Standard for Dual-Mode Wideband Spread-spectrum Cellular System,” TIA/EIA/IS-95-A, Washington, D.C., May 1995.

Tob87 F. Tobagi, “Modeling and Performance Analysis of Multihop Packet Radio Networks,” *Proceedings of the IEEE*, Vol. 75, No. 1, pp. 135-155, January 1987.

TsH98 David Tse and Stephen Hanly, “Effective Bandwidths in Wireless Networks with Multi-user Receivers,” *IEEE INFOCOM’98*, Vol. 1, pp. 35-42, 1998.

VaW97 M.C. Valenti and B.D. Woerner, “Variable Latency Turbo Codes for Wireless Multimedia Applications,” *Proceeding of International Symposium on Turbo Codes and Related Topics*, pp. 216-219, September 1997. (Available at http://lamarr.mprg.ee.vt.edu/documents/symp97.pdf).

ZMR96 Nevena Zecevic, Milap Majmundar, and Jeffrey H. Reed, “Interference Rejection Techniques for Direct-Sequence Spread-Spectrum CDMA Single-User Detection: A Comparative Study,” Submitted for review to *IEEE Transactions on Signal Processing*, 1996.

Appendix A. Simulation Model Descriptions

A.1 Simulation Model Overview

A.1.1 Models Layout

Figure A-1 presents a high-level overview of the radio model. The modules on the left are standard SEAMLSS components while the modules on the right represent the Virginia Tech radio model. The module net_interface serves as an interface between these two sections. The solid lines are packet streams and the dashed lines are statistical wires. The Virginia Tech radio model supports either four simultaneous transmissions or four simultaneous receptions in half duplex mode. As a result, there is one set of wires per transceiver.

A.1.2 Transmitter and Receiver Models

A.1.2.1 Virginia Tech Radio Transmitter and Receiver Models

Table A.1. Transmitter and Receiver Models.

Name	Comments
ant_tx	A standard isotropic antenna
radio_rx	Four channels radio receiver model in which all channels share the same characteristics except spread-spectrum spreading code.
radio_tx	Four channels radio transmitter model in which each channel shares the same characteristics except spread-spectrum spreading code.

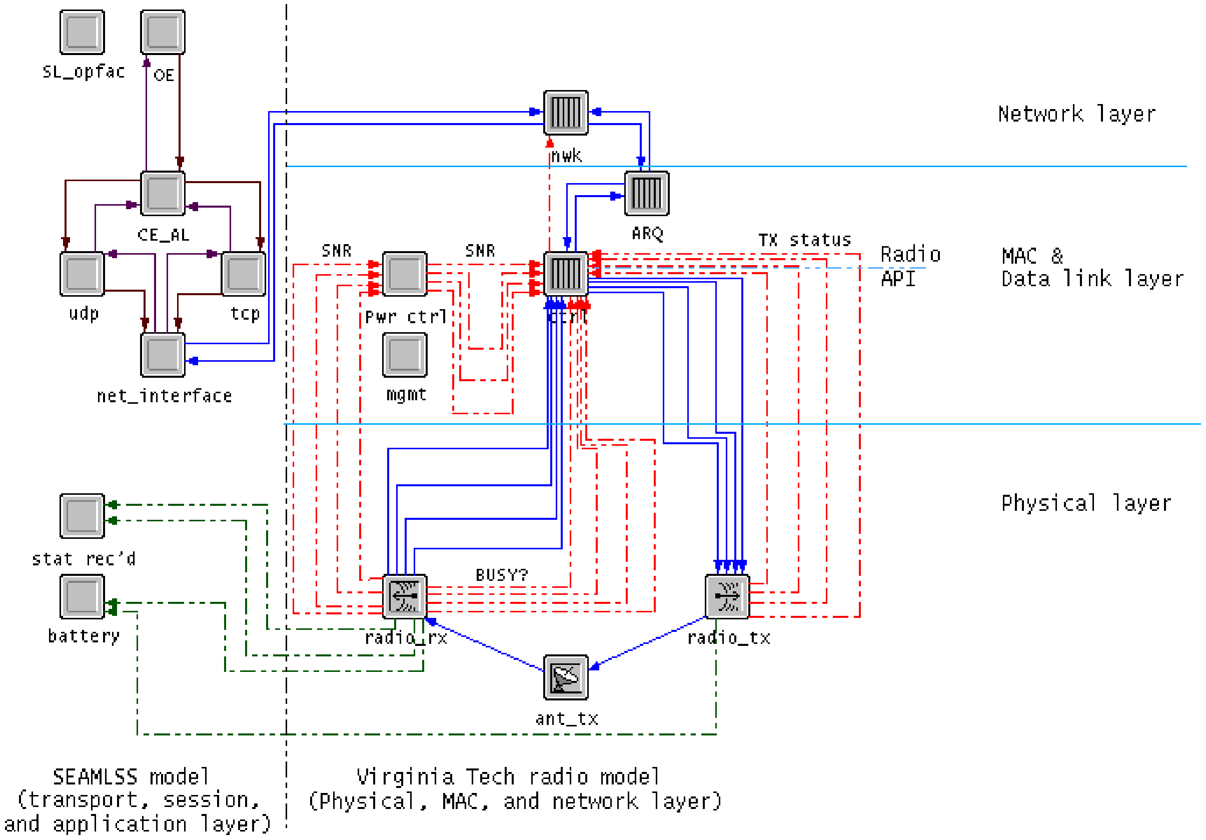


Figure A-1. Node model.

A.1.3 Process Models

A.1.3.1 Virginia Tech Radio Process Models

Table A.2. Process Models.

Name	Location	Comments
g_nwk	Lower network layer (nwk)	Networking protocol that forwards IP packets and maintains routing table
g_arq	Upper MAC layer (ARQ)	Automatic retransmission and acknowledgement
g_ctrl_mul	Lower MAC layer (ctrl)	MAC protocol that sits on top of the GloMo radio API to the control channel accessing mechanism and data transfer process
g_pwr	Physical layer (Pwr ctrl)	This module processes OPNET's SNR statistic for the g_ctrl_mul process
g_mgmt	Physical layer (mgmt)	This module simulates random node movement (Disable in SEAMLSS).

A.1.3.2 Virginia Tech Radio Interfacing Process Model

Table A.3. Interfacing Process Model.

Name	Location	Comments
seamlss_VT_IF	Upper network layer (net_interface)	This module provides module interface, address assignment, application level statistics, and IP packet encapsulation and decapsulation.

A.1.4Model Interfaces

A.1.4.1Packet Formats

The following table contains a list of all formatted packets that are exchanged between the various processes and different nodes in the Virginia Tech radio model. In addition to these formatted packets, several unformatted packets are used throughout the model. These unformatted packets are documented directly in the model code. Detailed descriptions for each field and data type of the formatted packets are given in Section A.2.

Table A.4. Packet Format.

Name	Description
g_phy	This is the lowest level packet and the only packet format that is transmitted over the air. It is compatible with the IEEE 802.11 DSSS MAC physical frame format.
g_ctrl	This is the MAC logical frame format and contains addressing information, packet type, sequence number, and piggybacked ACK. This packet is an extension of the IEEE 802.11 DSSS MAC logical frame format. All higher layer packets are encapsulated within this packet format.
g_ctrl_RTS	Encapsulated in g_ctrl, this packet carries channel allocation request (RTS) messages.
g_ctrl_CTS	Encapsulated in g_ctrl, this packet carries channel allocation acknowledgement (CTS) messages.
g_mux	This is a general-purpose multiplexing packet and has only two fields, protocol number and data.
g_nwk	Internal packet for network -> MAC communication, used in place of the ICI
g_nwk_b	Internal packet for MAC -> network communication, used in place of the ICI
g_nwk_bc	Routing and topology information
g_nwk_fwd	This packet format is used by network layer for non-local destination, requesting that this packet be forwarded to its final destination.
g_nwk_rte_upd	This packet format is used by network layer to initialize the routing table of a newly discovered neighboring router.

A.1.4.2ICI Formats

This model does not use ICI.

A.2 Model Detail Descriptions

A.2.1 Physical Models

A.2.1.1 Antenna

The simulation model assumes a standard isotropic antenna.

A.2.1.2 Radio Transmitter

A physical radio transmitter with four channels is used. All channels have the same parameters except the spread-spectrum spreading code. Default parameters are shown in the table below. All transceiver pipelines are standard OPNET pipeline models.

Table A.5. Radio Transmitter Parameters.

Description	**Value**
Default data rate	64 Kbits/s
Packet format	g_phy
Bandwidth	1,024 MHz
Minimum frequency	2,000 MHz
Transmit power	0.01 Watts
Modulation	BPSK

A.2.1.3 Radio Receiver

A physical radio receiver with four channels is used. All channels have the same parameters except the spread-spectrum spreading code. Default parameters are shown in the table below. There are three custom transceiver pipeline models. These models are used to facilitate model development by adding animation and trace information. They do not modify the original functionality.

Table A.6. Radio Receiver Parameters.

Description	Value
Default data rate	64 Kbits/s
Packet format	g_phy
Bandwidth	1024 MHz
Minimum frequency	2,000 MHz
Processing gain	12 dB
Modulation	BPSK
Error correction capability	0.0745 error/bits - assuming (255,123) Linear Block Code (subject to change)
Power model	gp_power – standard OPNET model plus code to draw animation.
Error model	tr_error – standard OPNET model plus ODB trace information.
Ecc model	gp_ecc – standard OPNET model plus code to clean up animation.

A.2.2 Process Models

A.2.2.1 MAC Module (g_ctrl_mul)

A.2.2.1.1 Model Scope and Limitations

The MAC module implements a Medium Access Control scheme to allow multiple users to share the same frequency spectrum by using CDMA. It provides two primary services: broadcast and unicast transmissions. There is a limited multicast ability.

A.2.2.1.1.1 Broadcast Service

This service sends data out to the control channel. Every station within the transmission range can receive this packet. This data is sent without any type of coordination with other nodes or any flow control mechanisms enforced. Broadcast service can be reliable with the help of the retransmission or network modules.

A.2.2.1.1.2 Unicast Service

Upon higher layer packet arrival, the originating node sends an RTS packet to the destination node. This RTS packet contains source address and channel identification, i.e., spreading code, information that the source desires to use for data transmission. The addressee observes the SNR of the received RTS packet to determine the appropriate transmission rate for data, stores that information into a new CTS packet, and sends it back to the source via the channel indicated in the RTS packet. The source then sends the actual data to the destination at the specified data rate and channel.

A.2.2.1.1.3 Collision

Since this MAC protocol operates in unsynchronized and uncoordinated fashion, collision is bounded to occur, especially for RTS and broadcast packets. To combat collisions, the chip rate should be at least four chips per bit to provide an additional 6 dB processing gain in the event that a collision does occur. The collision rate of RTS packets is usually the highest. The CTS collision rate is lower, and the data collision rate is the lowest (approximately 10% of the RTS collision rate). By using the RTS/CTS exchange sequence, a performance gain is obtained from the reduction in the data lost rate because, when a collision does happen, the majority of lost packets are RTS and CTS packets rather than data.

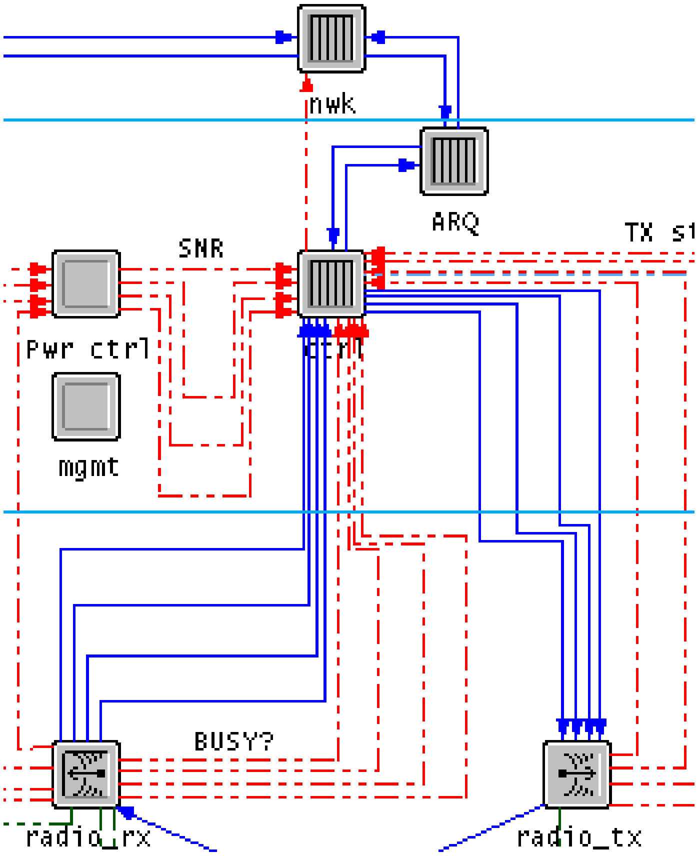


Figure A-2. Node model structure surrounding the MAC module.

A.2.2.1.1.4 Model Architecture

The following diagram depicts a typical node model using the process model defined for the MAC model suite. The g_ctrl_mul process resides in a module called *ctrl*, which connects to the receiver and a transmitter module via statistic wires and packet streams. It also has a statistic wire connection to the network module and packet stream connections with the retransmission module.

Other modules search for this processor module by name, so changing the processor name is not recommended.

A.2.2.1.1.5 Process Models

Table A.7. MAC Process Model.

Name	**Location**	**Summary**
g_ctrl_mul	ctrl module	Provides complete Medium Access Control ability, allowing multiple users to share the same frequency spectrum using CDMA and multi-user receivers

A.2.2.1.2 Model Interfaces

A.2.2.1.2.1 Packet Formats

Table A.8. MAC Physical Frame Format (g_phy).

Field	**Size (bits)**	**Descriptions**
Preamble	144	Packet acquisition and synchronization
Speed	8	Reserved (Modulation method)
Length	16	Reserved (Length of data packet in bytes)
Next frame	1	Reserved (Back-to-back frame)
Power control	2	Reserved (Decrease, increase, or keep the transmitted power)
Protocol	2	Reserved (MAC protocol ID)
Address	3	Reserved (Physical address)
CRC	16	Reserved for IEEE 802.11 compatibility
line_andid	0	Storage for OPNET animation handle
Data	inherit	Logical MAC frame

Table A.9. MAC Logical Frame Format (g_ctrl).

Field	Size (bits)	Descriptions
Frame control type	2	Type of payload (Management, control, or data)
Frame control subtype	4	Subtype within the specified type
Frame control other	10	Reserved for IEEE 802.11 compatibility
Source address	32	Source address
Destination address1	32	Primary destination address
Destination address2	32	Extra destination address for limited multicast and piggybacked ACKs
Destination address3	32	Extra destination address for limited multicast and piggybacked ACKs
Destination address4	32	Extra destination address for limited multicast and piggybacked ACKs
Dest addr adder	16	Reserved for IEEE 802.11 compatibility
Duration ID	16	Reserved for IEEE 802.11 compatibility
Sequence number	32	Unique number for each source address (used by the retransmission module)
Ack sequence no1	32	Piggybacked ACK
Ack sequence no2	32	Piggybacked ACK
Ack sequence no3	32	Piggybacked ACK
Ack sequence no4	32	Piggybacked ACK
Data	0 - 65485 bytes	Data payload size can be larger but the protocol will no longer be compatible with the IEEE 802.11 standard.
FCS	32	Reserved (Frame check sequence)

Table A.10. Frame Control Type and Subtype Used (Extensions to the IEEE 802.11 Specification).

Type	Subtype (base 10)	Description
1 (Control)	11	RTS
1	12	CTS
2 (Data)	0	Unreliable unicast
2	8	Unreliable broadcast
2	9	Reliable unicast with ACK request
2	10	ACK (since ACK is handled by an external module, it is considered data to the MAC module)

Table A.11. CTS Packet Format (g_ctrl_CTS).

Field	Size (bits)	Descriptions
Source address	32	Originating address of this packet
Destination address	32	Intended recipient of this packet
Data rate	32	Desired data transmission speed, in bits per second

Table A.12. RTS Packet Format (g_ctrl_RTS).

Field	Size (bits)	Descriptions
Source address	32	Originating address of this packet
Destination1	32	The first intended recipient of this packet
Data size1	32	Data size to be transferred
Spreading code1	64	Spreading code to be used for the corresponding recipient
Destination2	32	The second intended recipient of this RTS packet
Data size2	32	Data size to be transferred
Spreading code2	64	Spreading code to be used for the corresponding recipient
Destination3	32	The third intended recipient of this RTS packet
Data size3	32	Data size to be transferred
Spreading code3	64	Spreading code to be used for the corresponding recipient

A.2.2.1.2.2 ICI Formats

This process model has no ICI format.

A.2.2.1.2.3 Process Model Attributes

Table A.13. MAC Process Model Attributes.

Name	Description	Units	Default
Ethernet address	MAC address of the radio, also used as a network address		-1
Queue size	When the MAC module is not ready to transmit data, it stores that data into this transmission queue.	packets	5

A.2.2.1.2.4 Simulation Attributes

Table A.14. MAC Simulation Attributes.

Name	Description	Units	Default
Base data rate	A transmission speed for the control channel, which is used for all routing update and topology broadcast	Kbits/ second	64 Kbps
CTS timeout (`DEFAULT_WAIT_TIME`)	Waiting period after the (start of) RTS transmission and the (end of) CTS reception.	seconds	Twice the estimated round-trip time of RTS and CTS packets
Channel wait time	Waiting period after the start of current packet transmission and the attempt to transmit the next packet, to give other stations opportunities to initiate their transmissions	seconds	CTS timeout
Minimum SNR	Desired minimum SNR, used in combination with the observed SNR in the RTS packet to determine appropriate data transmission rate	dB	10 dB
Maximum backoff time	This is the maximum duration during successive packet transmissions. If a packet stays in the MAC module for more than one second, the ARQ module will timeout and treat that packet as lost. To avoid this, this value should be much less than one, but setting it too low will lead to an aggressive transmission.	second	1 - DEFAULT_WAIT_TIME - DEFAULT_WAIT_TIME_DATA.
Minimum data size	For a small packet, the overhead of the RTS/CTS exchange sequence may be too high to show any performance gain. If the data size is less than this value, it will be sent to the broadcast channel without any RTS/CTS exchange sequence. This variable must include MAC header. The reliability is not compromised because retransmission is handled separately by the ARQ module.	bits	1320 bit
Maximum data speed	This variable indicates the maximum speed for the data transmission, in multiples of the base data rate. It should be the power of two, up to the number of chips per bit.	times	4 times
Backoff rate	The backoff interval will be increased by this factor following an unsuccessful transmission (RTS or CTS message is lost).	times	1.1
Code rate	The error correction code rate must be between 0 and 1. The ECC threshold in a radio receiver must also be set to the appropriate value.		0.5
Single user mode	If this variable is one, the radio will operate with only one transceiver pair. If this variable is zero, the radio will use all transceivers.		0
Maximum data size	This attribute is not a limiting factor but rather a guideline to calculate various timeout variables in the system. Setting it too high will unnecessary increase the system delay.	bits	13000

A.2.2.1.2.5 Available Statistics and Reports

Table A.15. MAC Available Statistics and Reports.

Type	State Variable	Name	Description
Local	`rx_data`	RX data throughput	Number of user data received in bit at each node. It does not include header or control overhead of this layer.
Internal	`nd_hd`	Node heard	Source address of the most recent packet.
Disable	`tx_pk_count`	TX packet count	Data link layer: Number of packets transmitted.
Disable	`rx_pk_count`	RX packet count	Data link layer: Number of packets received.
Disable	`rx_raw_bit_cnt`	RX raw bit count	Data link layer: Number of bits received, include all headers at all levels.
Global	`gv_sending`	Concurrent session	Number of simultaneous transmissions from all nodes.
Global	`gv_packet_rx`	Global RX bit	Total number of bits received (user data payload only) at all nodes in the network. This includes broadcast traffic as well as unicast traffic.
Global	`gv_packet_tx`	Global TX bit	Total number of bits transmitted (user data payload only) at all nodes in the network. This includes broadcast traffic as well as unicast traffic
Global	`gv_tx_bcast`	Global TX bit (broadcast)	Total number of bits transmitted (user data payload only) at all nodes in the network. This includes only broadcast traffic
Global	`gv_tx_direct`	Global TX bit (not broadcast)	Total number of bits transmitted (user data payload only) at all nodes in the network. This includes only unicast traffic
Global	`gv_thru`	Throughput	Number of bits received at all nodes divided by the simulation time. This includes broadcast traffic as well as unicast traffic

A.2.2.1.2.6 Physical Interfaces

A.2.2.1.2.6.1 Incoming Interfaces

Table A.16. MAC Incoming Interfaces.

Neighbor	**Streams**	**Statistic Wires**
Power control module	None	These variables (four statistic wires, the signal to noise ratio of the most recent packet from the n^{th} receiver.) take into account the spread-spectrum processing gain. (`SNR_INSTAT0, SNR_INSTAT1, SNR_INSTAT2, SNR_INSTAT3`)
Transmitter module	None	Four statistic wires, the transmission status of the n^{th} radio (`TX0_BITSIZE_STAT, TX1_BITSIZE_STAT, TX2_BITSIZE_STAT, TX3_BITSIZE_STAT`)
Receiver module	Four input data streams from the n^{th} physical radio (`LOWER_LAYER_IN_STRM0, LOWER_LAYER_IN_STRM1, LOWER_LAYER_IN_STRM2, LOWER_LAYER_IN_STRM3`)	Four statistic wires, reception status of the n^{th} radio (`RX0_IDLE_STAT, RX1_IDLE_STAT, RX2_IDLE_STAT, RX3_IDLE_STAT`)
Retransmission module	Input data streams from the higher layer (`HIGHER_LAYER_IN_STRM`)	None

A.2.2.1.2.6.2 Outgoing Interfaces

Table A.17. MAC Outgoing Interfaces.

Neighbor	**Streams**	**Statistic Wires**
Transmitter module	Four output data streams to the n^{th} physical radio (`LOWER_LAYER_OUT_STRM0, LOWER_LAYER_OUT_STRM1, LOWER_LAYER_OUT_STRM2, LOWER_LAYER_OUT_STRM3`)	None
Retransmission module	Output data stream to the higher layer (`HIGHER_LAYER_OUT_STRM_1`)	None
Network module	None	One statistic wire, the source address of the current packet

A.2.2.1.2.7 Inter-Process Communications

At initialization time, the external module (SEAMLSS) must change the MAC module's model attribute *Ethernet Address* before the MAC module initializes. Failure to do so will result in an automatic address assignment, which will lead to inconsistent addressing. Other radio modules will also read this attribute for unified addressing (primarily for debugging purposes).

For every packet that is heard promiscuously, this process model will signal the network module via the statistic wire with the source address of the packet.

A.2.2.1.2.8 ODB Traces and Diagnostics

There are two macro definitions, DEBUG and DEBUG_EXT. Defining DEBUG to 1 will cause important trace information to be printed. Defining DEBUG_EXT to 1 will cause all debug information to be printed. In addition, the following ODB traces are defined for run-time trace information.

Table A.18. MAC Diagnostic Information.

Label	**Type**	**Descriptions**
ack	trace, breakpoint	ACK message handling mechanism
adapt	trace	Transmission speed calculation, and other adaptive aspects of the protocol
all	trace	All traces that are not api
api	trace	Radio API (changing radio parameters)
ber	trace	Bit error information (in transceiver pipeline)
cts	trace, breakpoint	CTS packet transmission and reception
pkt	breakpoint	Packet reception and transmission
received	breakpoint	Packet reception
rts	breakpoint	RTS packet transmission and reception
seq	trace	General packet and protocol sequence
state	trace	Finite State Machine information
tfci	trace	Link layer information
nsm	breakpoint	Reserved (Custom breakpoint)

A.2.2.1.2.9 External Files/Functions

Table A.19. MAC External Files/Functions.

Name	**Descriptions**
ctrl.ex.c	Most MAC functions unrelated to state variables or statistics.
API.ex.c	Radio API related entry points
TFCI.ex.c	Link layer encapsulation/decapsulation

A.2.2.1.3 Model Internal Structure

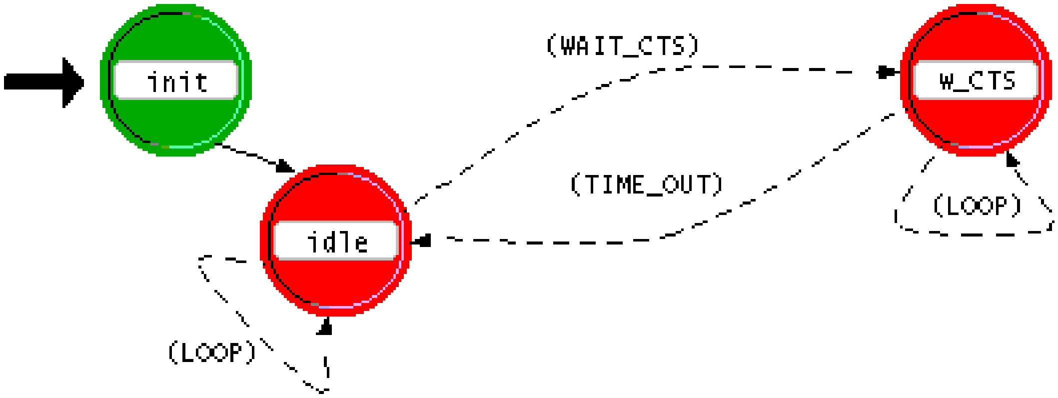


Figure A-3. MAC state transition diagram.

A.2.2.1.3.1 Multi-User Ability

The Virginia Tech MAC model supports up to four transmitters and receivers. The only limitation is that when transmission is taking place, no reception is possible (and vice versa) as in a typical half-duplex radio.

The spreading code for the first receiver is fixed to 1 (the control channel). For other transceivers, the spreading codes can be any value (stored in a 64-bit floating point variable). When the MAC module is ready to transmit a packet, it fetches up to three packets from its internal queue and assigns a unique spreading code for each of them. Then, it stores the destination addresses, spreading codes, and packet sizes in a single RTS packet, sends the packet via the control channel, and changes to the w_CTS state. The destinations, upon receiving the RTS packet, compare the addresses in the RTS packet with their address and, if they match, wait for a short period before sending a CTS reply. This short waiting period is necessary to allow RTS packets from other sources to arrive so this node can perform simultaneous receptions from multiple sources. If the destination receives more than one RTS packet, it will create one CTS packet for each RTS because each CTS must go through a separate channel. At the end of the waiting period, all CTS packets are transmitted simultaneously and the process will remain in the idle state. At the source, if all expected CTS packets have arrived or the waiting period is over, it will send out data packet(s) to the destination(s) and transition back to the idle state.

A.2.2.1.3.2 WINGs Radio API Implementation

To ease the joint effort among several institutions co-operating in the GloMo project, the interface between radio hardware and MAC protocol is conforms to the GloMo radio API version 2.03. This interface is encapsulated in the MAC module for ease of implementation and maintenance. All function calls in the MAC module are made to the API, which will be forwarded to the lower layer entities for their intended tasks.

Table A.20. List of GloMo Radio API Commands Implemented.

Command	Description
RadioCmdReset	Equivalent to hardware reset
RadioCmdXmtPkt	Packet transmission

Table A.21. List of GloMo Radio API Variables Implemented.

Variables	Description
RadioVarVersion	Radio API version
RadioVarName	Radio name
RadioVarStatus	Device busy/idle, channel busy/idle
RadioVarMacAdr	MAC address
RadioVarBitRate	Transmission speed (bps)
RadioVarXmtPower	Transmitted power (Watts)
RadioVarRcvSignal	SNR of the incoming packet
RadioVarCode	Spreading code in use

Table A.22. List of GloMo Radio API Signals Implemented.

Signal	Qualifier option	Description
RadioSigRcvPkt	None	A complete packet has been received.

A.2.2.1.3.3 Transition Condition Descriptions

There are only three states in the finite state machine because the multi-user receiver's algorithm demands a highly integrated model to independently control each radio receiver.

Table A.23. MAC Transition Condition Descriptions.

Condition	Description
`WAIT_CTS`	After receiving data from the higher layer and transmitting the RTS packet, it transitions to the w_CTS state to wait for a CTS packet.
`TIME_OUT`	The waiting period is over. All CTS replies may or may not arrived at this node.
`LOOP`	A collection of multiple conditions that does not lead to changes in state. Please see the exit executives in the FSM for more detail.

A.2.2.1.3.4 Operating Condition

These are state variables and macros indicating current system operating conditions.

Table A.24. MAC Operating Condition.

Condition	Description
`CSMA`	System operates in CSMA mode, no handshaking sequence is required, only one transceiver is used
`CSMA_BUSY`	Status of the first radio receiver (busy or idle), if CSMA is TRUE
`CSMA_IDLE`	Status of the first radio receiver (busy or idle), if CSMA is TRUE

`TX0_EMPTY,`
`TX1_EMPTY,`
`TX2_EMPTY,`
`TX3_EMPTY`
	Status of the n^{th} radio transmitter (transmitting or idle)

`TX_EMPTY`	Return TRUE if and only if all radio transmitters are idle
`RX0_IDLE, RX1_IDLE, RX2_IDLE, RX3_IDLE`	Status of the n^{th} radio receiver (receiving or idle), receiving in non-CSMA mode implies that the spreading code is matched
`RX_IDLE`	Return TRUE if and only if all radio receivers are idle
`END_SIM`	End of simulation
`STAT_INT`	Statistic interrupt
`ARRIVAL`	Stream interrupt
`TIMEOUT`	Generic self interrupt
`TIME_OUT_WAIT_DATA`	Self interrupt indicating that the waiting period for a CTS packet is over
`IP`	Higher layer packet arrived
`LOWER_LAYER`	Lower layer packet arrived

A.2.2.1.3.5 Process State Definition

Table A.25. MAC Process State Definition.

State	Description
init	Performs all necessary initialization
idle	The protocol waits for data, RTS, or higher layer packet arrival
w_CTS	The protocol waits for one or more CTS replies

A.2.2.2 Network Module (g_nwk)

A.2.2.2.1 Model Scope and Limitations

This module is the core of the *ad hoc* routing protocol. It performs three primary tasks: cluster formation across the network, routing table updates, and packet routing.

A.2.2.2.1.1 Cluster Formation Process

Nodes are categorized into three groups: regular nodes (cluster members), cluster heads (physically located at the center of each cluster), and gateways (primary routers between cluster heads). After a node is activated, it will keep listening for an advertisement message from a cluster head for at least two advertisement intervals. If the advertisement arrives, the listening node sends a registration message to the cluster head (advertisement originator) registering itself to that cluster. The cluster head then adds the requesting node to its member list. If no cluster head's advertisement message is received during the waiting period, the newly activated node will declare itself a cluster head and begin sending an advertisement message to other nodes. This mechanism will ensure that, initially, there are no two cluster heads within the transmission range of each other. Additionally, it guarantees that all nodes will be within the transmission range of at least one cluster head.

This algorithm can be viewed as an automatic and dynamic selection of base stations in a cellular telephone network. Using these dynamically created base stations, other nodes can function just like a cellular telephone in a traditional cellular network.

A.2.2.2.1.1.1 Priority

To incorporate priority into the cluster head election process, the timeout values at each node are different from one another. Nodes with lower addresses tend to timeout earlier than those with higher addresses. This algorithm effectively favors nodes with lower addresses to be a cluster head.

A.2.2.2.1.2 Cluster Maintenance

A cluster head will periodically broadcast an advertisement message providing its connectivity and routing information. A regular node sends no routing message (except a registration request message if it has not already done so).

If a regular node misses several advertisement messages, indicating that it might be out of range of its cluster head, it will declare itself a cluster head and start broadcasting advertisement messages. To demote itself, a cluster head uses the information in an advertisement message from another cluster head to determine if it is a subset of that cluster head (if all of its members are also members of another cluster head). This algorithm tends to generate more clusters than necessary, which is desirable for two reasons: first, it provides redundant connections, which lead to a better fault tolerant system and more optimized routes; second, more clusters make the two gateways problem (a situation in which there is no single gateway to connect two cluster heads together, which leads to unconnected clusters) less likely to occur. In a situation in which the bandwidth and power requirement is more important than a full connectivity, a less aggressive cluster maintenance algorithm can be developed.

A.2.2.2.1.3 Route Generation and Maintenance

A regular node rarely participates in the routing process. It merely forwards packets to its cluster head for routing and does not generate or forward any routing information to any nodes.

The network backbone consists of interconnecting links among cluster heads and gateways. Cluster heads periodically generate and advertise routing information. Gateways, nodes that connect two cluster heads together, forward that information to the adjacent cluster head and passively update its routing table.

The route selection algorithm is based on a distance vector routing protocol with a path finding algorithm [GaM97]. There is only one route entry for each destination. In the case of route failure, it is possible to flood a local subnet by sending out packets to all nearby cluster heads in the hope that they will have the correct routing information. This technique should only be used for very important messages.

A.2.2.2.2 Model Architecture

The following diagram depicts a typical node model using the process models defined for the networking module. As shown in the diagram, the g_nwk process resides in a module called nwk and connects to a retransmission module and an interfacing module via packet streams. It also has a statistic wire connection from the MAC module.

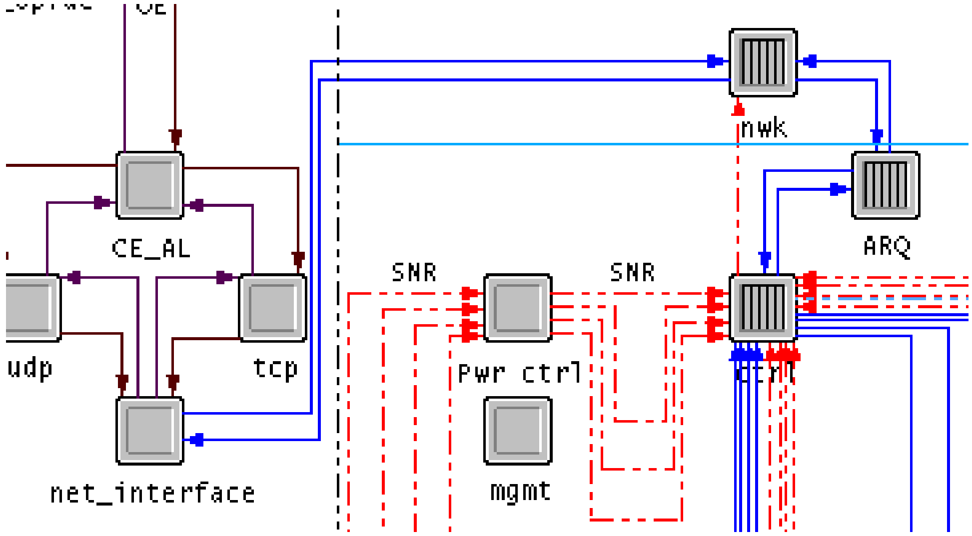


Figure A-4. Node model structure surrounding the network module.

A.2.2.2.2.1 Process Models

Table A.26. Networking Process Models.

Name	Location	Summary
g_nwk	nwk module	Provides multi-hop routing capability with standard IP address and packet format

A.2.2.2.3 Model Interfaces

A.2.2.2.3.1 Packet Formats

Table A.27. General Purpose Multiplexing Frame Format (g_mux).

Field	Size (bits)	Descriptions
Protocol	8	Protocol ID.
Data	inherit	Data payload.

Table A.28. Network Message Type.

Type	Descriptions
`CLS_DATA`	Generic packets for local destinations (one hop away)
`CLS_HD_BCAST`	Cluster information, originated from the cluster head
`CLS_REBCAST`	Propagated cluster information, forwarded by gateway
`CLS_REG_REQ`	Cluster member registration request
`CLS_FWD_NEED`	Generic packets for remote destinations (more than one hop away)
`CLS_RTE_TBL`	Routing table sent to newly discovered neighbor cluster
`CLS_ND_DOWN`	Special packet type used internally for communication from the retransmission module

Table A.29. Networking Frame Format, Used Internally (g_nwk).

Field	Size (bits)	Descriptions
Destination Address	32	Destination address
Broadcast	32	TRUE = broadcast, FALSE = unicast
Control subtype	32	MAC Frame Control Subtype (8 = unreliable, 9 = reliable)
Reliable	32	TRUE = reliable, FALSE = unreliable
Sequence number	32	Unique number for each source address for ARQ purpose (set by retransmission module)
Start time	32	Time this packet arrived at the retransmission module
Data	inherit	Data payload
Next	inherit	Packet aggregation for small packet (used internally)
Transmission rate	32	Reserved
Power	32	Reserved
Signal strength	32	Reserved
Interface	32	Reserved

Table A.30. Backward Networking Frame Format, Used Internally (g_nwk_b).

Field	Size (bits)	Descriptions
Source address	32	Originator of this packet
Sequence no	32	Unique number for each source address for ARQ purpose (set by retransmission module)
type	2	MAC control type
subtype	4	MAC control subtype
data	inherit	Data payload

Table A.31. Packet Forwarding Frame Format (g_nwk_fwd).

Field	Size (bits)	Descriptions
Destination address	32	Ultimate destination address
Data	inherit	Data payload

Table A.32. Cluster Head Advertisement Packet Format (g_nwk_bc).

Field	Size (bits)	Descriptions
Sequence number	32	A unique number for each source address with no relationship with the retransmission module
Sender address	48	
Cluster info	multiple of 32	A list of member addresses in this cluster, including the cluster head
Ack request	multiple of 32	A list of cluster heads that are requested to send acknowledgements to this advertisement
Ack reply	multiple of 64	A combination of cluster head addresses and their sequence numbers in which the sender acknowledges the reception of their advertisement messages
Node unreachable	multiple of 32	This is a list of nodes that are unreachable by the sender. Instead of using an infinity distance to indicate unreachable nodes, a separate entry is provided to reduce the amount of the information transferred
Update	inherited from g_nwk_rte_upd	A collection of zero or more route update entries

Table A.33. Route Update Entry (g_nwk_rte_upd).

Field	Size (bits)	Descriptions
Head	32	The destination cluster head
Distance	32	The distance from this cluster head to the destination cluster head
Member	multiple of 32	A list of member addresses in the destination cluster
Predecessor	32	The predecessor of the destination cluster head
Successor	32	The successor of the destination cluster head
Next	inherited	A pointer to the next route update entry

A.2.2.2.3.2 ICI Formats

This process model has no ICI format.

A.2.2.2.3.3 Routing Table Format

Each node maintains one routing table. Each entry has the following format.

Table A.34. Routing Table Entry Format.

Field	Size (bits)	Descriptions
Destination address	32	
Distance	32	Hop count to the destination
Predecessor	32	The second to last hop in the path to the destination
Successor	32	The next hop in the path to the destination
Marker	32	Additional information about this entry (cluster head, gateway, or regular node)

A.2.2.2.3.4 Process Model Attributes

This process model has no process model attributes.

A.2.2.2.3.5 Simulation Attributes

Table A.35. Networking Simulation Attributes.

Name	Description	Units	Default
Maximum node	Maximum number of node in the network, for generating random destination address (disable in SEAMLSS)		0

A.2.2.2.3.6 Available Statistics and Reports

Table A.36. Networking Available Statistics and Reports.

Type	State Variable	Name	Description
Global	`nwk_bit_tx`	BitTx	User data received from the higher layer plus a portion of routing information that is transmitted when a new neighbor cluster is found
State variable	`nwk_rte_tbl_tx`		Total number of bits transmitted (routing information) at all nodes in the network
Global	`nwk_bit_rx`	BitRx	Total number of bits received (user IP datagram) at all nodes in the network
State variable	`nwk_dat_bit_rx`		Total number of bits transmitted (user IP datagram only) at all nodes in the network
State variable	`nwk_rte_tbl_rx`		Total number of bits received (routing information) at all nodes in the network
Global	`nwk_bit_lost`	BitLost	Number of bits dropped that is resulted from no route to the destination
State variable	`nwk_tx_cnt`		Number of data and routing packets transmitted
State variable	`nwk_dat_tx_cnt`		Number of data packets transmitted
State variable	`nwk_hop_cnt`		Total number of hops that all packets traverse
State variable	`nwk_dat_hop_cnt`		Total number of hops that the data packets traverse
State variable	`nwk_delay`		Average end-to-end delay for each packet
State variable	`nwk_dat_delay`		Average end-to-end delay for each data packet

A.2.2.2.3.7 Physical Interfaces

A.2.2.2.3.7.1 Incoming Interfaces

Table A.37. Networking Incoming Interfaces.

Neighbor	Streams	Statistic Wires
Network interface module	Input data streams from the higher layer (HIGHER_LAYER_IN_STRM)	None
Retransmission module	Input data streams from the lower layer (LOWER_LAYER_IN_STRM)	None
MAC module	None	One statistic wire, source address of the current packet

A.2.2.2.3.7.2 Outgoing Interfaces

Table A.38. Networking Outgoing Interfaces.

Neighbor	Streams	Statistic Wires
Retransmission module	Output data stream to the lower layer (LOWER_LAYER_OUT_STRM)	None
Network interface module	Output data stream to the higher layer (HIGHER_LAYER_OUT_STRM)	None

A.2.2.2.3.8 Inter-Process Communications

At initialization, the network module will read the MAC module's Ethernet address attribute for unified addressing (debug information). For every packet that is heard promiscuously, the MAC process model signals this network module via a statistic wire with the source address of that packet.

A.2.2.2.3.9 ODB Traces and Diagnostics

There are two macro definitions DEBUG and DEBUG_EXT. Defining DEBUG to 1 causes important trace information to be printed. Defining DEBUG_EXT to 1 causes all debug information to be printed. In addition, the following ODB traces are defined for run-time trace information.

Table A.39. Networking Diagnostic Information.

Label	Type	Descriptions
nwk	trace	General information
head	trace	Cluster setup information

A.2.2.2.3.10 External Files/Functions

This process model uses no external file.

A.2.2.2.4 Model Internal Structure

There are two unforced states in the finite state machine representing the cluster setup and established phases. Another unforced state is solely for debugging purposes.

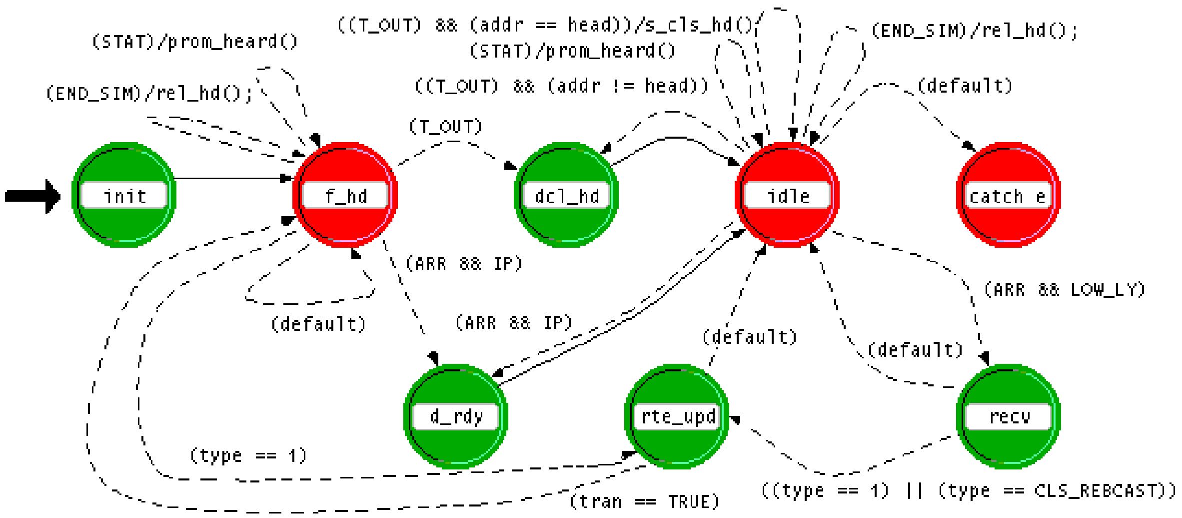


Figure A-5. Network model state transition diagram.

A.2.2.2.4.1 Transition Condition Descriptions

Table A.40. Networking Transition Condition Descriptions.

Condition	**Description**
`T_OUT`	Self interrupt (time out)
`ARR`	Stream interrupt
`IP`	Packet arrived from the higher layer
`type == CLS_REBCAST,` `type == 1`	Receiving cluster head broadcast or rebroadcast message
`addr == head`	This node is a cluster head.
`STAT`	Statistic interrupt
`END_SIM`	End simulation interrupt
`tran == TRUE`	Collection of multiple conditions that leads to changes in state

A.2.2.2.4.2 Process State Definition

Table A.41. Networking Process State Definition.

State	Description
init	Performs all necessary initialization
f_hd	This node just been initialized and is still searching for its cluster head.
idle	Waiting for packet arrival
d_rdy	This is data arrived from the higher layer, encapsulating the routing headers and forwarding the packet according to the information in the routing table.
dcl_hd	No existing cluster head was found. It declares itself a cluster head and broadcasts cluster information.
rte_upd	Routing information was received and the routing table is updated.
recv	The packet arrives from the lower layer and forwards it to the higher layer.

A.2.2.2.4.3 Operating Parameters

The following parameters are hard-coded into the process model, usually in the form of pre-processor symbolic constants. The exception is Address information, which is a state variable.

Table A.42. Networking Operating Parameters.

Name	Variable	Default value	Description
Address information	addr		A 32-bit integer that has been set to the same number as the Ethernet address in the MAC module
Advertisement interval	BCAST_INT	3 seconds	This is an interval in which the cluster head sends out a periodic advertisement, in seconds. In order to avoid synchronization, the actual broadcast period is randomized between 80% to 120% of this value.
Maximum Cluster information	MAX_CLS_INFO	20 clusters	This is a maximum number of cluster heads that a regular node can subscribe to. Nodes always subscribe to cluster heads within their transmission range.
Saved sequence number	MAX_SEQ_NO_ SAVE	50 entries	The number of sequence numbers in the incoming advertisement messages that a receiver will remember to avoid using old routing information
Whole routing table update period	WHOLE_TBL_ UPD	100 advertisement intervals	Periodically, a cluster head broadcasts the entire routing table to its neighbors to flush incorrect route information and update missing routes. This information does not propagate beyond those neighbors if their routing tables are already up-to-date.
Cluster head absence - startup		2 advertisement intervals	Minimum period in which a node will keep searching for cluster head after it was newly activated, actual value is a random variable depending on the priority of each node
Maximum timeout period	MAX_OUT_ TIME	15 seconds	A maximum value for the Cluster head absence startup variable
Cluster head absence - operation	HEAD_OUT_INT	3 advertisement intervals	A node will consider itself moving out of the cluster boundary if it misses a number of cluster head advertisements.

A.2.2.3 Retransmission Module (g_arq)

A.2.2.3.1 Model Scope and Limitations

This layer functions as a part of the MAC protocol. It provides reliable hop-by-hop data delivery. In particular, it assigns a unique sequence number to each packet that requests reliable transmission. When that packet safely reaches its destination, it generates an ACK message back to the source of this transmission. Using a unique sequence number allows duplicated packets to be detected (a duplicated ACK is sent but the packet itself is discarded). If no ACK message arrives within an estimated round-trip time, the ARQ module retransmits another packet until the number of retransmissions reaches its maximum. In that case, it notifies the higher layer regarding this failure and discards the packet.

A.2.2.3.2 Model Architecture

The following diagram depicts a typical node model using the process models defined for the retransmission module. As shown in the diagram, the ARQ module hosts a g_arq process and connects to the network module and MAC module via packet streams.

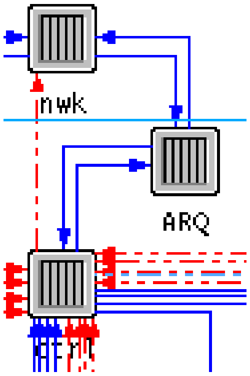


Figure A-6. Node model structure surrounding the retransmission module.

A.2.2.3.2.1 Process Models

Table A.43. ARQ Process Model.

Name	Location	Summary
g_arq	ARQ module	Provides reliable hop-by-hop packet transfer and detects packet duplication using a sequence number and ACK message

A.2.2.3.3 Model Interfaces

A.2.2.3.3.1 Packet Formats

There is no new packet format introduced by the retransmission module. It uses the same packet formats as the network module.

Table A.44. General Purpose Multiplexing Frame Format (g_mux).

Field	Size (bits)	Descriptions
Protocol	8	Protocol ID
Data	inherit	Data payload

Table A.45. Networking Frame Format, Used Internally (g_nwk).

Field	Size (bits)	Descriptions
Destination Address	32	Destination address
Broadcast	32	TRUE = broadcast, FALSE = unicast
Control subtype	32	MAC Frame Control Subtype (8 = unreliable, 9 = reliable)
Reliable	32	TRUE = reliable, FALSE = unreliable
Sequence number	32	Unique number for each source address for ARQ purpose (set by retransmission module)
Start time	32	Time this packet arrived at the retransmission module
Data	inherit	Data payload
Next	inherit	Packet aggregation for small packet (used internally)
Transmission rate	32	Reserved
Power	32	Reserved
Signal strength	32	Reserved
Interface	32	Reserved

Table A.46. Backward Networking Frame Format, Used Internally (g_nwk_b).

Field	Size (bits)	Descriptions
Source address	32	Originator of this packet
Sequence no	32	Unique number for each source address for ARQ purpose (set by retransmission module)
type	2	MAC control type
subtype	4	MAC control subtype
data	inherit	Data payload

A.2.2.3.3.2 ICI Formats

This process model has no ICI format.

A.2.2.3.3.3 Process Model Attributes

This process model has no process model attributes.

A.2.2.3.3.4 Simulation Attributes

This process model has no simulation attributes.

A.2.2.3.3.5 Available Statistics and Reports

Table A.47. ARQ Available Statistics and Reports.

Type	State Variable	Name	Description
Global	`arq_pkt_delay_hd`	Hop-by-hop delay	Hop-by-hop delay of each packet
Internal	`arq_ack_rx`		Number of acknowledgement packets received
Internal	`arq_ack_tx`		Number of acknowledgement packets transmitted
Internal	`arq_pkt_tx`		Number of packets transmitted, including both reliable and unreliable mode
Internal	`arq_pkt_rx`		This is the number of packets received, excluding duplicated packets. Because of broadcast messages, this value can be significantly greater than the number of packets transmitted.
Internal	`arq_pkt_delay`		Delay experienced by each packet
Internal	`arq_pkt_overflow`		Number of discarded packets because the maximum storage capacity has been reached
Internal	`arq_nd_unreach`		Number of undeliverable packets because they reach the maximum retransmission counts

A.2.2.3.3.6 Physical Interfaces

A.2.2.3.3.6.1 Incoming Interfaces

Table A.48. ARQ Incoming Interfaces.

Neighbor	Streams	Statistic Wires
Network Module	Input data streams from the higher layer (HIGHER_ IN)	None
MAC Module	Input data streams from the lower layer (LOWER_ IN)	None

A.2.2.3.3.6.2 Outgoing Interfaces

Table A.49. ARQ Outgoing Interfaces.

Neighbor	Streams	Statistic Wires
MAC Module	Output data stream to the lower layer (LOWER_ OUT)	None
Network Module	Output data streams to the higher layer (HIGHER_ OUT)	None

A.2.2.3.3.7 Inter-Process Communications

At initialization, the retransmission module will read the MAC module's Ethernet address attribute for unified addressing (debug information).

A.2.2.3.3.8 ODB Traces and Diagnostics

There are two macro definitions DEBUG and DEBUG_EXT. Defining DEBUG to 1 causes important trace information to be printed. Defining DEBUG_EXT to 1 causes all debug information to be printed. In addition, the following ODB trace is defined for run-time trace information.

Table A.50. ARQ Diagnostic Information.

Label	Type	Descriptions
arq	trace	General information

A.2.2.3.3.9 External Files/Functions

This process model uses no external file.

A.2.2.3.4 Model Internal Structure

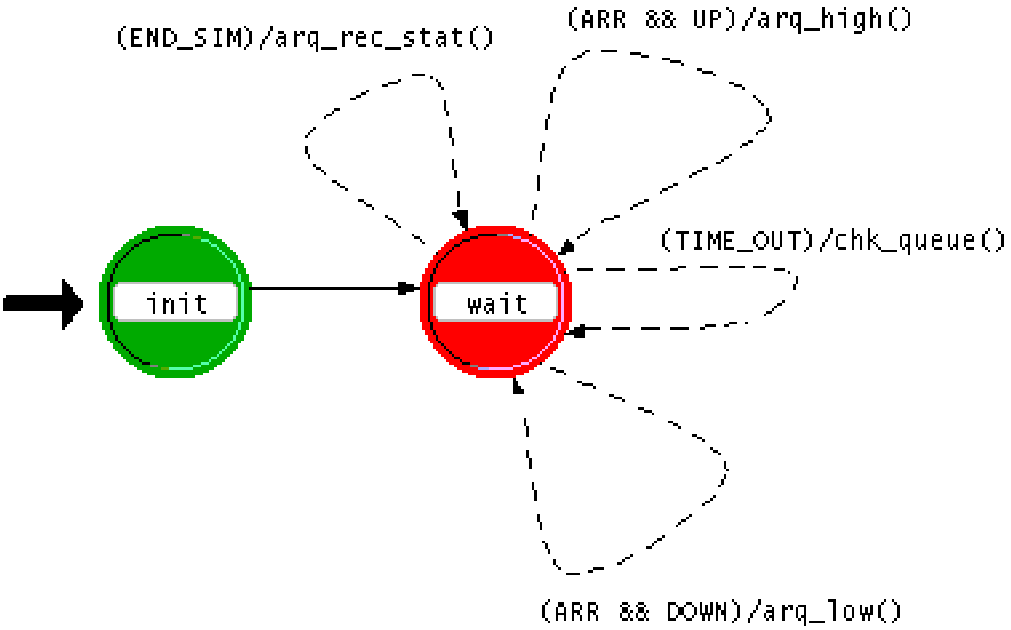


Figure A-7. Retransmission module state transition diagram.

A.2.2.3.4.1 Transition Condition Descriptions

Table A.51. ARQ Transition Condition Descriptions.

Condition	Description
`TIME_OUT`	Self interrupt
`ARR`	Stream interrupt
`UP`	Packet arrived from the higher layer
`DOWN`	Packet arrived from the lower layer
`END_SIM`	Simulation end

A.2.2.3.4.2 Process State Definition

Table A.52. ARQ Process State Definition.

State	Description
init	Performing all necessary initialization
wait	Waiting for packet arrival

A.2.2.3.4.3 Operating Parameters

The following parameters are hard-coded into the process model, usually in the form of pre-processor symbolic constants.

Table A.53. ARQ Operating Parameters.

Name	Variable	Default value	Description
Maximum retransmission attempt	MAX_RETRANSMIT	6 times	Maximum retransmission counts for each packet
Estimated round-trip time	RTT	1 second	Maximum allowable time from the moment that it sends packet to the MAC module until the ACK is delivered
Maximum sequence number saved	MAX_SEQ_NO_SAVE	50 entries	The maximum number of arriving sequence numbers that the receiver will remember to avoid packet duplication
Buffer size		50 packets	The queue size for storing unacknowledged packets

A.2.2.4 Power Control Module (g_pwr)

A.2.2.4.1 Model Scope and Limitations

The power control module plays a supporting role to the MAC module. It has the following functionality.

A.2.2.4.1.1 Filtering an Invalid Value

The OPNET simulator provides an SNR value for all received packets, including valid packets, noise packets, and packets that were just transmitted from this node itself. The only value of interest is the SNR of valid packets. This module filters out undesired values.

A.2.2.4.1.2 Retain Value

The SNR is valid during packet reception only and becomes invalid as soon as reception is complete. Thus, an algorithm to retain validity after reception is needed.

A.2.2.4.1.3 Take Processing Gain Into Account

The standard SNR value is taken before despreading the received signal. This module adds the effect of processing gain and supplies the MAC module with a new value that represents the SNR after despreading.

A.2.2.4.2 Model Architecture

The following diagram depicts a typical node model using the process model defined for the power control module. As shown in the diagram, the g_pwr process resides in a module called Pwr ctrl and connects to the network module and radio receiver module via statistic wires.

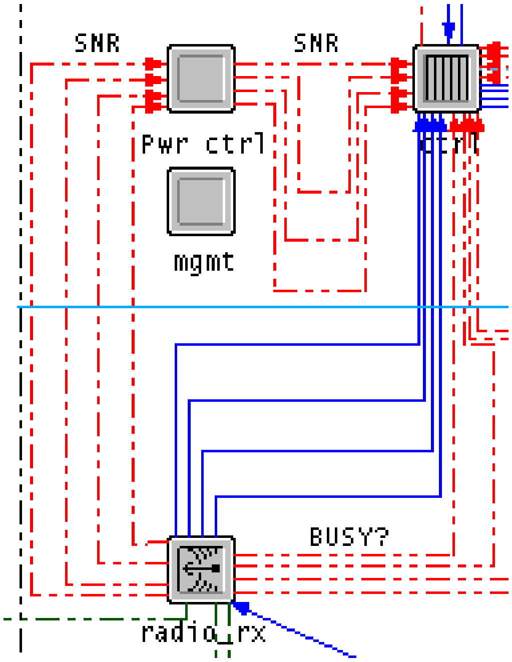


Figure A-8. Node model structure surrounding the power control module.

A.2.2.4.2.1 Process Models

Table A.54. Power Module Process Model.

Name	**Location**	**Summary**
g_pwr	Pwr ctrl module	This module processes SNR statistic probes for the MAC module.

A.2.2.4.3 Model Interfaces

A.2.2.4.3.1 Packet Formats

This process model has no packet format.

A.2.2.4.3.2 ICI Formats

This process model has no ICI format.

A.2.2.4.3.3 Process Model Attributes

This process model has no process model attributes.

A.2.2.4.3.4 Simulation Attributes

This process model has no simulation attributes.

A.2.2.4.3.5 Available Statistics and Reports

Table A.55. Power Module Available Statistics and Reports.

Type	**State Variable**	**Name**	**Description**
Local		snr1	Filtered SNR for the 1st channel of the radio receiver
Local		snr2	Filtered SNR for the 2nd channel of the radio receiver
Local		snr3	Filtered SNR for the 3rd channel of the radio receiver
Local		snr4	Filtered SNR for the 4th channel of the radio receiver

A.2.2.4.3.6 Physical Interfaces

A.2.2.4.3.6.1 Incoming Interfaces

Table A.56. Power Module Incoming Interfaces.

Neighbor	Streams	Statistic Wires
Radio receiver module	None	Four statistic wires, signal to noise ratio of the currently received packet (SNR_INSTAT0, SNR_INSTAT1, SNR_INSTAT2, SNR_INSTAT3)

A.2.2.4.3.6.2 Outgoing Interfaces

Table A.57. Power Module Outgoing Interfaces.

Neighbor	Streams	Statistic Wires
Network Module	None	These variables (four statistic wires, signal to noise ratio of the most recent packet) take into account spread-spectrum processing gain.

A.2.2.4.3.7 Inter-Process Communications

All inter-process communications are performed via statistic wires.

A.2.2.4.3.8 ODB Traces and Diagnostics

There is one macro definition, DEBUG, which, when defined, causes all trace information to be printed.

A.2.2.4.3.9 External Files/Functions

This process model uses no external file.

A.2.2.4.4 Model Internal Structure

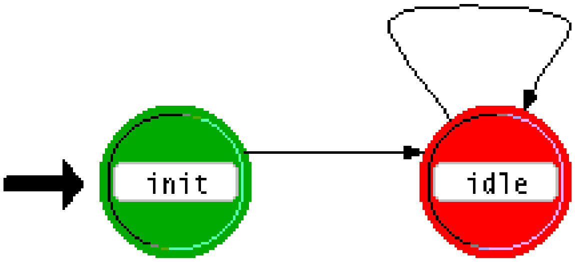


Figure A-9. Power control module state transition diagram.

A.2.2.4.4.1 Transition Condition Descriptions

There is no conditional transition for this model.

A.2.2.4.4.2 Process State Definition

Table A.58. Power Module Process State Definition.

State	**Description**
init	Initialization
idle	Waiting for statistic updates

A.2.2.5 Management Module (g_mgmt)

A.2.2.5.1 Model Scope and Limitations

This module is designed to simulate random node movement. Since this functionality is already provided by SEAMLSS, this module is disabled for the OpFac_VT model.

A.2.2.5.2 Model Architecture

A.2.2.5.2.1 Process Models

Table A.59. Power Module Process Models.

Name	**Location**	**Summary**
g_mgmt	mgmt module	Simulate random node movement

A.2.2.5.3 Model Interfaces

A.2.2.5.3.1 Packet Formats

This process model has no packet format.

A.2.2.5.3.2 ICI Formats

This process model has no ICI format.

A.2.2.5.3.3 Process Model Attributes

This process model has no process model attributes.

A.2.2.5.3.4 Simulation Attributes

This process model has no simulation attributes.

A.2.2.5.3.5 Available Statistics and Reports

This process model has neither statistic nor report.

A.2.2.5.3.6 Physical Interfaces

This process model has no physical interface.

A.2.2.5.3.7 Inter-Process Communications

This process model has no inter-process communications.

A.2.2.5.3.8 ODB Traces and Diagnostics

This process model has no diagnostic capability.

A.2.2.5.3.9 External Files/Functions

This process model uses no external file.

A.2.2.5.4 Model Internal Structure

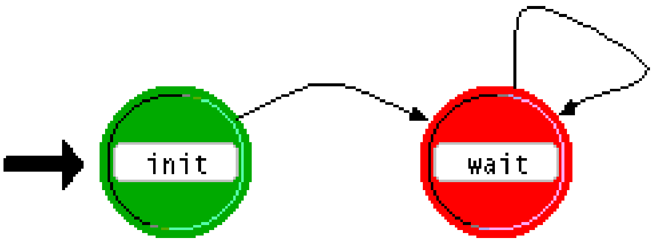


Figure A-10. Management module state transition diagram.

A.2.2.5.4.1 Transition Condition Descriptions

There is no conditional transition condition in this model.

A.2.2.5.4.2 Process State Definition

Table A.60. Power Module Process State Definition.

State	**Description**
init	Initialization
wait	Waiting for scheduled positioned update

A.2.2.6 Network Interface Module (SEAMLSS_VT_IF)

A.2.2.6.1 Model Scope and Limitations

This module is built on top of the SEAMLSS trans_phy_IF process model. No functionality is removed from the original model. The added functionality includes encapsulating the application layer's packets into IP datagrams and decapsulating the packet upon arrival of the IP datagram. In addition, it exports SEAMLSS addressing to other modules in the Virginia Tech radio model.

Because the modification is a transparent add-on, the documentation in this section describes the added functionality without referring to its original SEAMLSS functionality. By doing so, the SEAMLSS team can modify the original trans_phy_IF at will and still be able to add the interface functionality with ease. In addition, it is in the best interest for all parties to have the original developers document their module.

A.2.2.6.2 Model Architecture

The following diagram depicts a typical node model using the process model defined for the network interface module. As shown in the diagram, the net_interface module hosts the SEAMLSS_VT_IF process and connects to the network, tcp, and udp modules via packet streams.

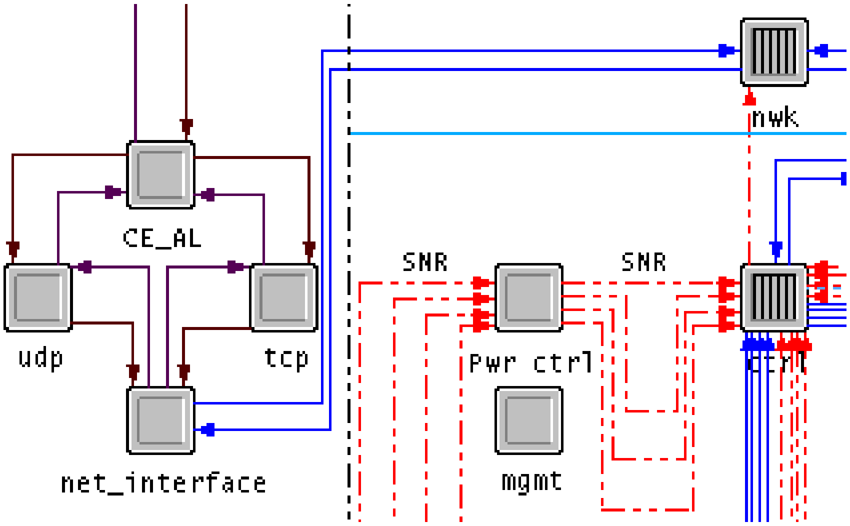


Figure A-11. Node model structure surrounding the network interface module.

A.2.2.6.2.1 Process Models

Table A.61. Interface Module Process Models.

Name	Location	Summary
SEAMLSS_VT_IF	net_interface module	This module encapsulates/decapsulates IP datagrams and exports SEAMLSS addressing.

A.2.2.6.3 Model Interfaces

A.2.2.6.3.1 Packet Formats

This process model uses standard IP datagram format.

A.2.2.6.3.2 ICI Formats

No additional ICI is used other than those used by SEAMLSS.

A.2.2.6.3.3 Commands and Indications

No additional commands and indications are defined other than those defined by SEAMLSS.

A.2.2.6.3.4 Process Model Attributes

Table A.62. Interface Module Process Model Attributes.

Name	Description	Units	Default
local node	Unified address string from the sdf2opnet.ef		
Net 0	Unified address string from the sdf2opnet.ef		
Net 1	Unified address string from the sdf2opnet.ef		
Net 2	Unified address string from the sdf2opnet.ef		
Net 3	Unified address string from the sdf2opnet.ef		

A.2.2.6.3.5 Simulation Attributes

This process model has no simulation attributes.

A.2.2.6.3.6 Available Statistics and Reports

Table A.63. Interface Module Available Statistics and Reports.

Type	State Variable	Name	Description
Global		Packet size	The size of user data (not including IP header) in bits
Global		Packet throughput	Throughput of each packet
Global		Packet transmission time	A duration between IP packet creation and packet arrival at the destination

A.2.2.6.3.7 Physical Interfaces

A.2.2.6.3.7.1 Incoming Interfaces

Table A.64. Interface Module Incoming Interfaces.

Neighbor	Streams	Statistic Wires
Network module	Input data streams from the lower layer	None
Transport module	Input data streams from the higher layer	None

A.2.2.6.3.7.2 Outgoing Interfaces

Table A.65. Interface Module Outgoing Interfaces.

Neighbor	Streams	Statistic Wires
Transport module	Output data stream to the higher layer	None
Network module	Output data stream to the lower layer	None

A.2.2.6.3.8 Inter-Process Communications

No additional inter-process communications are defined other than those defined by SEAMLSS.

A.2.2.6.3.9 ODB Traces and Diagnostics

In addition to those defined by SEAMLSS, the following ODB trace is defined for run-time trace information.

Table A.66. Interface Module Diagnostic Information.

Label	Type	Descriptions
OE	trace	General information

A.2.2.6.3.10 External Files/Functions

No additional external files are used other than those used by SEAMLSS.

A.2.2.6.4 Model Internal Structure

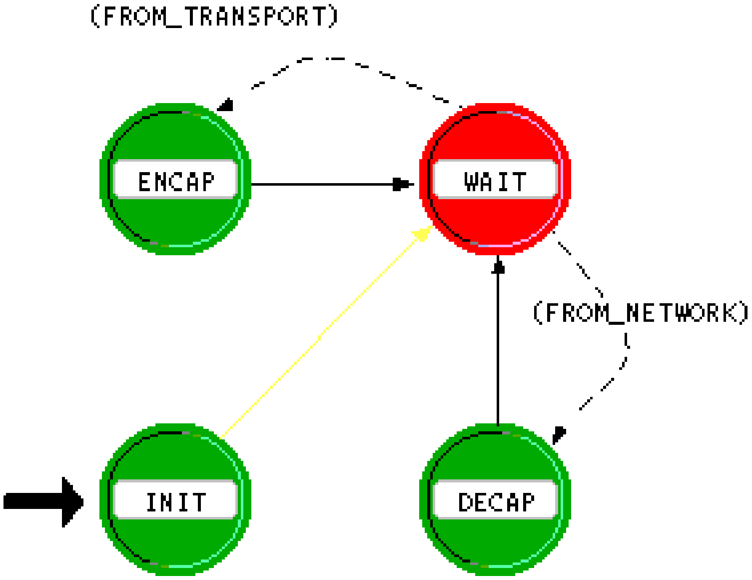


Figure A-12. Power control module state transition diagram.

A.2.2.6.4.1 Transition Condition Descriptions

No additional transition condition is defined other than those defined by SEAMLSS.

A.2.2.6.4.2 Process State Definition

No additional state is defined other than those defined by SEAMLSS.

Appendix B. Abbreviations

A	ACK	Acknowledgement message
	ADIM-NB	Adaptive DSSS MAC protocol for non-broadcast multiple access network
	ADIM-NB/B	ADIM-NB baseline
	ADIM-NB/FE	ADIM-NB with adaptive FEC coding rate algorithm
	ADIM-NB/MU	ADIM-NB with multi-user algorithm
	ADIM-NB/PC	ADIM-NB with adaptive power control algorithm
	ADIM-NB/PG	ADIM-NB with adaptive processing gain algorithm
	ALVA	Area-based link-vector algorithm
	API	Application program interface
	ARQ	Automatic repeat request algorithm
	ATM	Asynchronous transfer mode
B	BER	Bit error rate
	BPSK	Binary phase shift keying
C	CAMEN	Clustering algorithm for mobile ad hoc network
	CATER	Code adapts to enhance reliability
	CBRP	Cluster-based routing protocol
	CCK	Complimentary code keying
	CCM	Configurable computing machine
	CDMA	Code division multiple access
	CPT	C++ protocol toolkit
	CRC	Cyclic redundancy check
	CSMA	Carrier sensing multiple access
	CSMA/CD	Carrier sensing multiple access with collision detection
	CTS	Clear to send message
D	DBF	Distributed Bellman-Ford algorithm
	DBPSK	Differential binary phase shift keying
	DCF	Distributed coordination function
	DQPSK	Differential quaternary phase shift keying
	DSR	Dynamic source route protocol
	DSSS	Direct sequence spread spectrum

F	FAMA	Floor acquisition multiple access
	FAMA-PJ	FAMA with pause and jamming
F	FEC	Forward error correction
G	GUI	Graphic user interface
L	LAN	Local area network
M	MAC	Medium access control
	MAI	Multiple access interference
	MAISIE	University of California, Los Angeles’s simulation package
	MAN	Metropolitan area network
	MANET	Mobile ad hoc network
	MMWN	Multi-hop mobile wireless network
	MTU	Maximum transmit unit
O	OFDM	Orthogonal frequency-division multiplexing
P	PCF	Point coordination function
	PIC	Parallel interference cancellation
Q	QoS	Quality of service
R	RTS	Request to send message
S	SEAMLSS	GloMo program tool
	SEAMLSS-Lite	GloMo program tool
	SMTP	Simple mail transfer protocol
	SNR	Signal to noise ratio
	SSMA	Spread spectrum multiple access
T	TDM	Time division multiplexing
	TTL	Time-to-live
W	WDM	Wavelength division multiplexing
	WRP	Wireless routing protocol
Z	ZRP	Zone routing protocol

Vita

Nattavut Smavatkul
December 2000

EDUCATION

August 1996 – December 2000	*Virginia Polytechnic Institute and State University*, Blacksburg, VA. Doctorate of Philosophy, Electrical Engineering.
May 1991 – July 1996	*Ramkhamhaeng University*, Bangkok, THAILAND. Bachelor of Science, Business Administration.
August 1994 – December 1995	*California State University, Fullerton*, Fullerton, CA. Master of Science, Electrical Engineering,.
May 1990 – May 1994	*Chulalongkorn University*, Bangkok, THAILAND. Bachelor of Engineering, Electrical Engineering.
May 1988 – April 1990	*Triamudomsuksa School*, Bangkok, THAILAND.

EXPERIENCE

September 1998 – April 1999	Application Engineer OPNET Technology Inc., Washington D.C.
March 1993 – May 1993 and March 1994 – July 1994	Planning and Maintenance Engineer TelecomASIA, Bangkok, THAILAND.

PUBLICATION

Natt Smavatkul and Scott F. Midkiff, “Inter-Organization Model Development and Evaluation Using SEAMLSS-Lite.” *OPNETWORK 2000*, August 28-September 1, 2000, Washington, DC, 6 pages. (Available at http://www.opnet.com/opnetwork2000/).

Natt Smavatkul and Scott F. Midkiff, “Impact of High-End Radio Functionality in Wireless Mobile Ad Hoc Networks,” IEEE Workshop on Multiaccess, Mobility and Teletraffic for Wireless Communications, December 3-6, 2000, Duck Key, Florida, 12 pages, to appear.