INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

ProQuest Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

UMI®

University of Alberta

Energy Conservation in Mobile Networks

by

Ioannis Georgios Batsiolas ©

A thesis submitted to the Faculty of Graduate Studies and Research in partial
fulfillment of the requirements for the degree of Master of Science

Department of Computing Science

Edmonton, Alberta

Spring 2002

National Library of Canada
Acquisitions and Bibliographic Services
395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale du Canada
Acquisitions et services bibliographiques
395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

Our file Notre référence

0-612-69685-5

University of Alberta

Library Release Form

Name of Author: Ioannis Batsiolas

Title of Thesis: Energy Conservation in Mobile Networks

Degree: Master of Science

Year this Degree Granted: 2002

Permission is herby granted to the University of Alberta Library to reproduce single copies of this thesis and to lend or sell copies for private, scholarly or scientific research purposes only.

The author reserves all other publication and other rights in association with the copyright in the thesis, and except as herein before provided, neither the thesis nor any substantial portion thereof may be printed or otherwise reproduced in any material form whatever without the author's prior written permission.

2A 9009 112th str.
Edmonton, T6G2C5
Alberta
Canada

Wednesday, January 23, 2002

University of Alberta

Faculty of Graduate Studies and Research

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research for acceptance, a thesis entitled **"Energy Conservation in Mobile Networks"** by Ioannis Batsiolas in partial fulfillment of the requirements for the degree of Master of Science

Dr. Ioanis Nikolaidis
Supervisor

Dr. Pawel Gburzynski

Dr. Xiaoling Sun
External examiner

Date: 23/01/2001

Dedication

This work is kindly dedicated to my parents in humble acknowledgement for their emotional and financial support during this crucial stage in my life.

Abstract

Battery power is a very valuable commodity for mobile computing devices. Numerous techniques and proposal have been made that try to minimize the power consumption of every component of a mobile computing device, CPU and memory, storage, display and communications interface. This work focuses on the last one, the wireless network interface. We examine the problem of energy conservation in the popular context of wireless cellular networks as well as in the newly emerging one of mobile ad hoc networks (MANETs). In the former type of architecture, we investigate a transport layer solution to the energy conservation problem, as well as the addition of data link layer extensions that enable a base station to assist the wireless node in minimizing energy consumption. Our final proposal in cellular wireless networks achieves energy savings for a broad range of network conditions. In the latter type, ad hoc networks, we investigate the energy characteristics of the various network layers in MANETs and we explore a simple method for power conservation that is based on the popular 802.11. Simulation results show that a MANET with our simple power management scheme consumes about one third the amount of energy than without it.

Table of Contents

List of Figures

List of Abbreviations

ACK Acknowledgement.

AODV Ad hoc On demand Distance Vector.

ATIM Ad hoc Traffic Indication Map.

BS-SI-TCP Base Station enhanced SI-TCP.

CTS Clear To Send.

DSDV Destination Sequenced Distance Vector.

DSR Dynamic Source Routing.

EC-MAC Energy Conserving MAC.

IETF Internet Engineering Task Force.

I-TCP Indirect TCP.

LAN Local Area Network.

LAR Location Aided Routing.

LOTAR Location Trace Aided Routing.

MAC Medium Access control.

MANET Mobile Ad hoc Network.

NS2 Network Simulator 2.

OSI Open Systems Interconnect

PAN Personal Area Network.

PARO Power-Aware Routing Optimization.

TCP Transmission Control Protocol.

SI-TCP Selective Idling TCP.

RRP Route Request Packet.

RTS Request To Send.

RTT Round Trip Time.

TORA Temporally Ordered Routing Algorithm.

WWP Wave and Wait.

ZRP Zone Routing Protocol.

1. Energy Management techniques for Wireless Networks

1.1 Introduction

We often behold mobile computing devices with a sense of awe. They are small and stylishly designed; yet for their compact size they offer the user a full computing experience. More often than not, they are more powerful than their older desktop brethren. Continuous research and innovation in the field of microprocessors, storage, displays and operating systems have played a very important role in their success. Furthermore, the rapid growth of wireless networks, along with the introduction of a wireless interface in many mobile computing devices, has empowered them with new capabilities and opened new horizons to their use. Research visionaries, for example in the area of wearable computing [2], have already begun experimenting with the next generation of mobile computing devices, that promise a seamless integration of the real and the "virtual" world. Finally, it is a well-known fact that the market for mobile computing devices is expanding at a very fast pace, and many experts predict, that these little devices are poised to change the way we work and live our lives.

For all their advantages however, there are still several limitations that need be overcome before the aforementioned visions come to fruition. One of those is their reliance to battery power. In order to achieve mobility we have to rely on a portable power source, usually batteries. However, unlike microprocessors, memory, storage and the other microelectronic components that comprise a mobile computing device, batteries have not experienced the same leaps in performance and capacity. Unfortunately, Moore's Law does not apply to them [67, 68] and we are practically using the same technology as far as batteries are concerned as five years ago. The limited power supply represents a serious constraint and has affected the widespread adoption of mobile computing devices.

There are promising research initiatives, fuel cells are a notable example, however these revolutionary ideas are not ready for mass production and the end consumer yet.

Intelligent power management is touted as a technology that alleviates our limited power supply problem [1, 4, 5, 7, 9, 13, 23]. There have been numerous solutions and proposals regarding efficient power management, and the majority of them are focused on the microprocessor, storage or the display component. Our principal efforts are directed to the wireless network interface. Unquestionably, such an interface is a valuable addition to a mobile computing device. Without proper power management though, it can quickly become a source of grief, as it can shorten dramatically the device's lifetime. For example, it is reported that a wireless network card in a portable laptop computer can draw up to 15% of its power [5, 13]. This figure rises to 52% if the device is a handheld computer. In the absence of any power conservation strategies, it has been found that a wireless interface can dwindle the lifetime of a laptop's battery down to 45 minutes from 3 hours originally [7]. It is therefore unquestionable, that the application of efficient power management techniques at the wireless interface is a necessity.

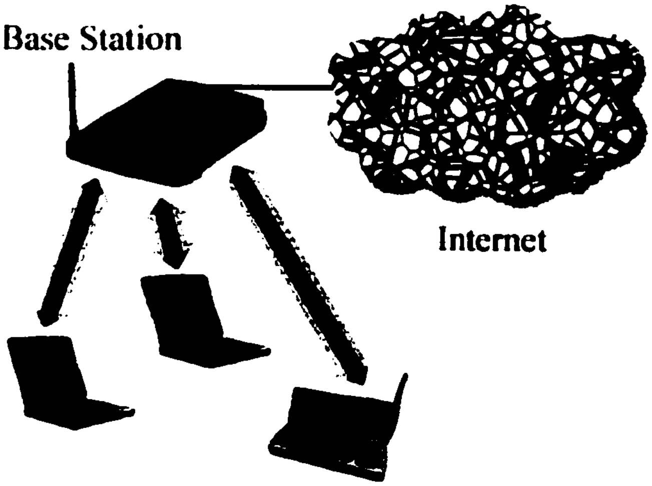


Figure 1.1 Cellular Network Architecture

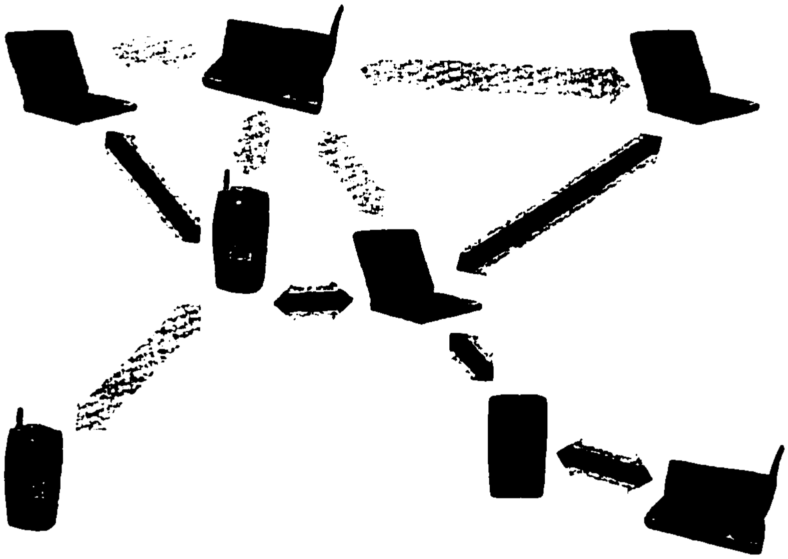


Figure 1.2 Example of an Ad Hoc Network

This thesis focuses on power conservation techniques for the wireless network interface of mobile computing devices. We examine this problem first in the context of cellular wireless networks and then in mobile ad hoc ones. In the former type of networks, we propose Selective Idling, an extension to the popular TCP protocol that aims to power down the network interface, and thus lessen energy consumption, when no activity is anticipated. Mobile ad hoc networks (MANETs) are considerably more complex than cellular ones. For this type of networks, we investigate a simple, IEEE 802.11 based method for minimizing the energy spent on data transmissions. Figures 1.1 and 1.2 illustrate the two different network architectures under consideration.

1.2 Previous Work in Wireless Cellular Networks

Power management techniques have been proposed and are presently in use by almost all components, both hardware and software, of a contemporary mobile computing device. Traditionally, the microprocessor, the hard drive and the display have been identified as the most power-hungry components. As such, most research efforts have been directed in finding ways to minimize their energy consumption. Let us examine for example a modern laptop. The display is usually considered as the most power-demanding component. A common policy that is implemented at the hardware layer, the software one or both, is to turn off the display during periods of user inactivity. Additionally, contemporary microprocessors are specifically designed to operate at a lower voltage and draw less power than their desktop equivalents. Extra intelligence allows them to monitor system usage and regulate their performance (and thus energy consumption) accordingly. For example, if a user is writing a document, the microprocessor may conserve energy by throttling down its clock speed. On the other hand, if the user is playing a movie, a task that requires increased computational resources, then the microprocessor may operate at its full capacity. Intel's SpeedStep technology [20] is a realization of the previous principles. In addition, a hard drive subsystem destined for use in a mobile environment may have multiple modes of operation, depending on the desired energy savings. Disks may spin at lower speeds during periods of infrequent disk access, or even be powered down during periods of extended user inactivity. Finally, some modern operating systems may assist the hardware-level power management techniques and work in synergy with them. This approach is exemplified by Microsoft's OnNow API extension [21]. Mobile applications built around it may offer hints and guidance to the power management subsystem, so as to achieve set energy saving goals while maintaining the end-user experience.

The problem of making wireless communications more energy conserving and efficient has been studied extensively in the past [1, 3, 4, 5, 7, 9, 12, 13, 18] [23]. Researchers have mainly focused their efforts on all of the lower layers of the OSI layer model and proposed several power management extensions and solutions. The seven-layer OSI protocol stack consists of (in a bottom-up approach) the physical, datalink, network, transport, application, session and presentation layers. Each layer provides a distinct number of services and functions to the upper ones and serves as a abstraction to any implementation details. This enables a modular design and facilitates interoperability between corresponding layers on different devices. Conventional protocol design practices try to adhere to this “layering principle” by keeping the services and functionality that each layer provides separate and not dependant on the implementation of any of the lower layers. However, we may have to break this principle when designing energy conserving protocols. The information that the lower layers may hide could be crucial in assisting the upper layers make the correct decisions and lessen energy consumption. Therefore, a “vertical” solution that spans across many layers of the OSI model may in fact be more energy efficient that one that strictly adheres to the aforementioned layering principle.

The central paradigm that permeates most of the energy conservation solutions is that one should try to minimize unnecessary transmissions and idle or turn off the transceiver unit altogether, when no communication activity is expected. Wireless connections often exhibit higher bit error rates than hard-wired ones (reported error rates are up to 10^{-3}, a stark contrast to figures for copper which are between 10^{-9} and 10^{-12}). Therefore, many research efforts have been directed towards making protocols that take this inherent unreliability of the wireless link into account and try to minimize unnecessary transmissions. There have been numerous proposals either in the data link or transport layers that try to alleviate the aforementioned problem [3, 4, 10, 11, 13, 18]. Being efficient, [8, 16] is vital but represents only one aspect of the problem. Additional gains in energy may be realized by idling the transceiver when no communication is expected. This particular idea, of idling down subsystems when they are not in use, has been successfully applied to most parts of a mobile computing device, such as the screen, microprocessor or the hard drive. However, its application to the network interface represents additional challenges, since when the latter is powered down, communication to the outside world is essentially cut off. This may have an adverse effect on the functionality of the device, or even result in a waste of resources by remote hosts trying to communicate with it. Power management decisions can be made in the data link, transport or applications layers. Furthermore, the use of a base station greatly aids in our efforts, and it often acts as a central point of coordination and/or as an intelligent proxy. Finally, in order to facilitate such strategies, modern network interface cards provide, in addition to transmit and receive modes, a sleep (or doze) one. While in this state the transceiver still consumes some power, albeit at least an order of magnitude less than the receive state. Its main advantage over having the receiver turned off completely is lower turnaround times.

1.2.1 Physical Layer

Power management decisions cannot easily be made at the physical layer. However, there is ongoing research in making it more efficient. A typical wireless network interface consists of RF circuits as well as modulations and coding systems. RF amplifiers for example, are still notoriously power hungry and inefficient [69, 70]. Fractal or sectored antennas represent another promising development, since they offer significantly higher gains than the traditional dipoles that are most commonly in use. Finally, research into the area of directional antennas [41] may yet provide us with additional gains in power consumption.

1.2.2 Data Link (and MAC) Layer

Contrary to their hard-wired cousins, wireless communication networks face additional challenges in the data link layer. Wireless networks have to contend with higher error rates due to fading, the exposed and hidden terminal problems as well as near and far effects. The hidden and exposed terminal problems are better illustrated in Figure 1.3 that shows a setup of three wireless nodes and their transmission radii. Node *B* can hear both *A* and *C*, however, *A* cannot hear C and vice-versa. A transmission from *A* to *B* may thus suffer a collision caused by *C*, who cannot detect if *B* is busy receiving *A*'s transmission. This is the hidden terminal problem (*C* is hidden from *A*). The RTS/CTS [15] mechanism employed by IEEE 802.11 represents an example of a mechanism specifically designed to alleviate the hidden terminal problem. Additionally, when node *B* transmits to *A*, node *C* cannot at the same time transmit to any other node as well, since it detects a "busy" medium due to *B*'s transmission. In this case, node *C* is said to be "exposed" to *B*. The exposed terminal problem suggests inefficient spatial and temporal reuse of the wireless medium. There are two possible solutions to the exposed terminal problem: One is for *C* to utilize a different frequency band, which may not always be available. The other one entails the use of sectorized or directional antennas in such a way that *C*'s transmission will not interfere with node *B*. Contention based MAC protocols, such as IEEE 802.11, HIPERLAN and ALOHA, suffer from collisions due to contention, especially in medium to high loads. From an energy standpoint, the increased number of collisions (and thus subsequent retransmissions), result in energy inefficiency.

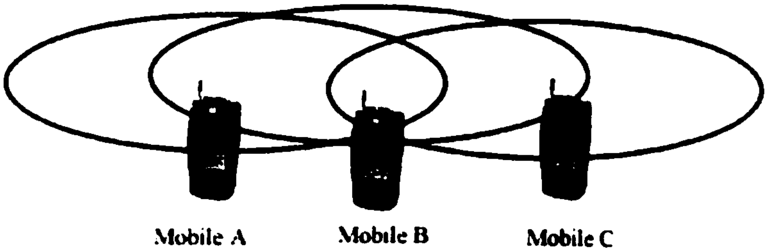


Figure 1.3 The hidden and exposed terminal problems

To combat this situation, special power-saving extensions have been proposed. IEEE 802.11 [15] for example, provides for power management techniques. Its scheme is much simpler in infrastructure-based networks, where coordination among the wireless hosts is achieved with the help of the access point, than in ad hoc wireless networks. In the former type of wireless LANs, a wireless host that operates in power-saving mode and wishes to enter "sleep" mode informs the access point of this decision. The access point acts from that point onward as a proxy and buffers any packets destined to the sleeping station. The latter one has to wake up periodically and listen to the beacon transmitted by the access point. Such a beacon informs the latent station if there are any packets destined for it. While these extensions do not increase the efficiency of the MAC protocol per se, they do give the opportunity to the mobile host to enter a sleep mode, when the access point informs it that it has no frames destined for it. However, this comes at a cost, as they introduce additional delays. Prediction of proper beacon intervals is yet another issue. In the absence of a fixed access point, IEEE 802.11's scheme becomes a lot more complicated, as wireless stations have to coordinate among themselves and buffer frames destined for stations that were sleeping. This strategy however faces scalability and applicability problems. In addition, 802.11 does not necessarily solve the problem of hidden/exposed terminal when used in a wireless multihop environment [65]. For example, let us consider the case that node *A* in figure 1.3 wishes to communicate with node *B*. If node *C* misses *B*'s CTS, it is unable to update its timers appropriately. Therefore it has the potential to attempt transmission at the same time as node *A* and thus cause a collision at node *B*. Possible causes for the loss of the CTS packet are collision with another packet at node *C*, or excessive errors due to wireless channel impairments.

The Energy-Conserving Medium Access Control Protocol (EC-MAC) [3] exemplifies infrastructure-based wireless network protocol design with efficiency and energy conservation in mind. The base station plays a pivotal role in EC-MAC and is responsible for beacon transmission, scheduling and slot allocation. Furthermore, it follows a fixed transmission frame scheme, which in turn enables each wireless station to know a priori and with precision when the next beacon will occur. The transmission frame itself is subdivided into specific phases: the reservation control phase, the new user phase, the schedule beacon and finally the downlink and uplink data phases. The reservation phase commences with a beacon transmitted from the base station, which dictates the transmission order of the wireless nodes during the remainder of this phase. In essence, a TDM scheme is followed and no collisions occur during the reservation phase. Following is the new user phase, during which new wireless nodes are allowed to register themselves with the base station using a contention based ALOHA scheme. At the end of these two rounds, the base station has enough information to calculate an optimal transmission schedule according to the requests it received. This schedule is then broadcast during the schedule beacon phase. In the remaining two phases, the wireless hosts know the exact point in time that they need to either transmit or receive data, and are able to power down their transceivers during the remainder of the time. EC-MAC has been shown to outperform IEEE 802.11 in terms of energy efficiency. However, to our knowledge, no international standard based on EC-MAC nor any EC-MAC based commercial products exist.

Bluetooth [19] is a wireless LAN technology, whose MAC layer also provides for power management. Primarily developed by Ericsson, it has received a lot of attention from the media in the past years and it touted as a single-chip, low-cost, radio based wireless network technology. The Bluetooth specification dictates three possible energy conserving power states: Park, Hold or Sniff state. A Bluetooth slave node can enter either one of those states depending on its expected traffic pattern. Park State allows for the greater energy savings, as the device releases its MAC address and idles its transceiver. It wakes up occasionally though and checks with the master device in order to retain synchronization and check for messages. If a device cannot afford the larger delay associated with the Park state, it can still save energy by entering the Hold state. In this state it does not release its MAC address and transmission can be resumed at no additional delay after exiting the Hold state. Finally, devices in Sniff state are still able to obtain some energy gains by reducing the rate at which they listen to traffic in the piconet.

An important point that must be made at this point is that we strive to achieve increased efficiency and reduced energy consumption by introducing centralized and more advanced medium access control. CSMA based schemes are simple yet inherently wasteful. However, a common realization is that additional information about the communications process is needed, in order to achieve optimal scheduling and make sound decisions about idling the network transceiver. Such information may be obtained from the upper layers, the transport and the application one. This strategy does constitute a violation of the layering principle, however this may be preferable to blind guesswork.

1.2.3 Network Layer

The cellular wireless architecture incorporates only one wireless hop; the one between the mobile end-node and its base station. As a result, there are no layer 3 decisions made at this point in the network and this layer has no impact in overall energy consumption. This is in stark contrast to mobile ad hoc networks (MANETs), which we examine later. In the latter type of networks, routing functions are a necessary service at each node and have a direct impact on both system performance and overall energy conservation. Proposals for reducing energy consumption at the network layer will be presented in the context of mobile ad hoc networks in following sections.

1.2.4 Transport Layer

Working at the transport layer makes for new venues in power management. Several schemes have been proposed that either enhance the efficiency of traditional transport layer protocols and tailor them for use in the wireless world, or replace them altogether. Other approaches call for intelligent idling of the network transceiver when no communication is anticipated.

The TCP protocol [42, 43] has become the de facto standard in wired networks when a reliable end-to-end connection is desired. Naturally, its use on wireless LANs is something to be anticipated. However, TCP was designed and fine-tuned for the characteristics of wired networks and it has been found to perform poorly in the wireless environment [16, 24, 25]. This problem is a direct consequence of the higher error rate often experienced in wireless transmissions. TCP is designed to fight congestion in the network by closing down its transmit window when packet losses occur. However, packets may be dropped due to poor wireless reception, resulting in retransmissions and unnecessary throttling down of TCP [8, 16]. This has a direct and negative impact on energy consumption, since retransmissions are handled by the wireless end-node and the duration of the communication exchange is prolonged.

Indirect TCP [44] is a split connection approach, which aims to address the aforementioned issues. According to this scheme, the wireless end-node communicates with the base station with a special version of TCP, which is adapted to the idiosyncrasies of the wireless link. The base station itself acts as an intelligent proxy and communicates directly with the remote destination node using standard TCP. Therefore, we have two separate TCP connections; one from the wireless node to the base station and one from the base station to the remote destination node. The latter one does not even notice the presence of the wireless node, for all intends and purposes it communicates directly with the base station. This approach can offer great performance gains over the traditional TCP. Its main disadvantage though is the loss of end-to-end TCP's semantics. The wireless host does not know whether its packets have in fact been successfully sent to the remote host. It is only aware that the base station has received them. Therefore, problematic situations may ensue, where the packets cached at the base station are never actually delivered at their destination, yet the mobile host is fooled into believing that they have. Such behavior however, is totally unacceptable for business or other mission-critical transactions.

Snoop [45] TCP tries to overcome I-TCP's disadvantages by leveraging the functionality and services provided by the base station. Additional intelligence is incorporated at the base station, which now actively 'snoops' into the traversing TCP flow. Its purpose now is mainly twofold. First, it tries to minimize the impact of losses on the wireless link on TCP. This is accomplished by caching packets destined to the mobile host and handling any retransmission requests locally, without forwarding them to the remote host. Secondly, it can alert the mobile node and request a packet retransmission whenever a packet drop occurs due to high bit error rate. This approach is completely transparent and TCP's end-to-end semantics are preserved. However, it is generally not as efficient as Indirect TCP.

Wave and Wait (WWP) [8] proposes a new reliable transport layer for mobile computing devices in place of TCP. Since transmissions are very expensive from an energy standpoint, it tries to minimize unnecessary ones by throttling down transmissions to just under calculated network congestion levels. WWP transmits data segments one after the other, in "waves" so to speak. The number of data segments that comprise a wave is determined by the wave level at which our mobile host operates. The underlying motivation of using waves is that the knowledge of the transmission pattern allows the receiver to better evaluate network congestion levels and direct the mobile host to be either more aggressive or conservative in its transmissions. Furthermore, WWP incorporates a probing mechanism, which allows it to monitor the network during periods of high congestion. Thus, it can identify windows of sufficiently low congestion opportunity which can be used for transmission with a high probability of success. WWP has been shown to outperform the most common variants of TCP and save energy used for transmission. However, it is a proprietary protocol and despite all its merits, it is unlikely it will replace TCP on the Internet.

1.2.5 Application Layer

Making applications power-aware and enabling them with features that allow them to aid in idling the wireless interface, provides for additional gains in battery lifetime. Indeed, applications are, or should be, in a position to judge and gauge their communication needs and patterns, and thus assist the underlying layers towards power management. It is argued that by enlisting the help of the application layer, predictions about the communication pattern can be made dynamically and with greater accuracy. For example, if the application is an email client, then the idle timers can be set to several minutes, as the application is configured to check for new messages every so often.

In "Application-Driven Power Management for Mobile Communications" [46], the authors propose a transport level solution, which can receive direct input and guidance about its idle timers from the overhead application layer. Its goal is to identify periods of network inactivity and suspend communications during them. This scheme employs a specialized transport layer protocol for communicating with the base station. The latter one plays a crucial role, as it acts as an intelligent proxy and buffer and assists in the making of proper scheduling decisions.

1.3 Our Proposal in Wireless Cellular Networks

Our approach, Selective Idling, represents an aggressive form of energy conservation designed for cellular wireless networks. We assume that the wireless host is connected to the rest of the network via a base station. The base station itself is hard-wired to the rest of the Internet and acts essentially as a gateway for the wireless segment of the network. Selective Idling builds upon TCP's congestion control mechanism and utilizes its round-trip time and round-trip time variance estimators to predict times of network inactivity in order to idle the receiver. Energy gains are accomplished by accumulating numerous short periods of time during which we power down the wireless interface. Initially, we assume no extra cooperation between the wireless end-node and the base station. However, later in our research, we explore and propose techniques and extensions to the data link layer, that enable the base station to assist and work in tandem with the mobile node towards the difficult goal of power management.

TCP's dominance in the Internet is a given fact. Therefore, one can only but expect its use on mobile computing devices as well, even though TCP is energy agnostic and not particularly well suited for the idiosyncrasies of wireless networks. In addition to TCP's prevalence, other factors that influenced us to work at the transport layer is the balance we may achieve between automation and power control. Application layer approaches such as [5, 46] enlist the help of application themselves in order to achieve energy savings. The pitfall in this approach is that the onus falls upon the application or the end-user himself to setup any parameter settings for energy conservation. Ultimately, this approach enforces the re-writing of applications (even legacy applications) to cope with the new interface – if efficient use of battery energy is to be made. Finally, our proposal may work in synergy with lower level solutions and achieve additional energy gains.

Our approach achieves energy savings when compared to standard TCP, however it also affects throughput. Furthermore, it depends crucially on the control of the maximum window size used by TCP and on accurate information of the congestion conditions in the network.

1.3 Mobile Ad Hoc Networks (MANETs)

1.3.1 Introduction

The area of Mobile Adhoc Networks (MANETs) [26] has received significant attention in recent years from the international research community. The term 'MANET' has been coined to describe an autonomous, wireless, multi-hop network, comprised of mobile, wireless nodes that operate in the absence of any supporting infrastructure. The IETF MANET working group was formed in 1997 and has since operated as a central point of coordination for protocol proposals in this area. Additionally, it has been trying to reach a consensus among the different proposals, and work towards establishing some accepted standards, albeit without a lot of success thus far.

As radical as it may sound, the very idea of autonomous, readily deployable networks that can operate without the need of any pre-existing infrastructure is not entirely new. Active research in this area has been under way since the 1970's (DARPA Packet Radio Network [47]), funded primarily by the United States' Department of Defense. It is easy to understand why MANETs are of particular interest to the military community. Communications are vital to battlefield coordination and greatly enhance the capabilities as well as the survivability of a combat unit. Furthermore, it is safe to assume a lack of pre-existing telecommunications infrastructure in an operational theater of war. Therefore, MANETs represent a very attractive technology, which is especially well suited to the communications needs of modern warfare. There has been significant research efforts invested in military MANETs and it is known that the US Army does possess MANET technology. Unfortunately, details and results about those research projects are related to the SURAN project [71, 70] and they are not readily available to the public but only to U.S. scientists working for the U.S. Department of Defense.

Plenty of opportunities exist for the application of MANETs in a civilian, commercial or industrial environment. One such example is nomadic computing. According to this paradigm, mobile professionals empowered with suitable wireless devices may form working groups on the fly and thus be able to work in an efficient collective and collaborative manner. In another possible scenario, the reach of the 'wired' Internet may be extended beyond the reach of base stations by the formation of wireless, multi-hop networks. Some intermediate nodes may act as routers for those nodes not readily within the base station's transmitter range and forward their traffic. Additionally, MANET technology is very well suited for applications in the automotive industry. Researchers envision 'smart' cars that are able to communicate on the road with one another and exchange useful information, e.g. receive alerts that an accident has occurred further down the road and the vehicle should decrease speed. Finally, the recent emergence of wearable computing and Personal Area Networks (PANs) [2], present yet another opportunity for MANET deployment.

1.4 Unresolved Problems in Mobile Ad Hoc Networks

The vision of Mobile Ad Hoc Networks is one of "Untethered Connectivity". According to the IETF MANET charter [26], "*the intent of the newly formed IETF manet working group is to develop a peer-to-peer mobile routing capability in a purely mobile, wireless domain. This capability will exist beyond the fixed* *network (as supported by traditional IP networking) and beyond the one-hop fringe of the fixed network*". Nevertheless, before this "holy-grail" of MANETs is reached, there are several issues that need to be adequately addressed.

The very first problem that needs to be solved (and which has received the majority of attention so far) is that of routing. One cannot overstress the importance of a proper network layer solution. Without a sound, efficient and robust underlying routing algorithm, a mobile ad hoc network simply cannot exist. Nodes that participate in a MANET operate in a resource-constrained environment; they rely on low-bandwidth wireless interfaces that have a limited transmit radius. In order to obtain full coverage and communicate with distant nodes, a multi-hop path has to be found through the network and maintained for the duration of the communication. This effort is compounded by node movement, which often causes links to fail as nodes move out of transmitter range.

In addition to the above routing question, there exist additional network layer issues that have not been adequately addressed yet. It is widely assumed that MANET nodes are going to be IP-based. The task of assigning IP addresses and prefixes to them however is still undefined. Existing routing proposals simply assume that nodes come "preconfigured" with suitable IP addresses. Furthermore, IP-aggregation techniques that are used routinely to conserve router lookup table space and are commonly employed in hard-wired networks do not lend themselves well to MANETs. This may not be a problem for networks comprising of a few hosts, however, it may have a negative impact on implementations comprising of hundreds or thousands of wireless nodes. In those cases, the wireless nodes may need to maintain and search through very large routing tables, an action that places additional strain on the limited computational resources of mobile computing devices.

The scarcity of available bandwidth represents yet another critical problem that civilian MANETs have to contend with. The available radio frequencies are a very valuable resource and as such, they are strictly regulated by national authorities worldwide. Current MANET proposals and implementations suggest the use of the 2.4 GHz ISM band, or the free ones at the 5 GHz range. Since these frequency bands are essentially unregulated, there are no guarantees of their availability at a particular geographic location. For example, the recent high growth of IEEE 802.11 and IEEE 802.11b networks has already saturated the ISM band in some areas [66].

The limited bandwidth availability has also a negative impact on the speed of the wireless interface. Recent IEEE 802.11a products may promise speeds up to 100Mbits [66], however this number depends upon proximity to the sender and overall network contention. In addition, reliable wireless multi-hop connections are known to suffer in a MANET environment. TCP flows are severely hampered by packet losses incurred due to the increased error rate of the wireless links [16, 24], as well as those caused by link failures due to node mobility [25]. The issue of TCP performance in MANETs is a very important one and remains yet unaddressed. The solutions proposed so far call for cooperation between the routing layer and the TCP agents. Their goal is to prevent TCP from interpreting link failures as excessive network congestion and seriously throttling down its throughput. Even with such techniques in place, the shared medium and multihop aspect of MANETs has been found to affect dramatically TCP performance. For example, it has been experimentally shown that TCP's throughput falls to less than 10% of the nominal link capacity after five hops [25]. The end result is that applications operating in a MANET environment may offer a totally different user experience than their desktop counterparts.

It has already been stressed, that battery power is a valuable commodity for mobile computing devices. In our study of cellular wireless networks, we presented several solutions to the energy conservation problem. However, the same problem cast in the context of mobile ad hoc networks requires additional consideration. It is assumed that nodes in a MANET have to act as routers and forward other nodes' traffic, otherwise communication may be unfeasible. Such behavior is expected from "well-behaving" nodes. From an energy point of view however, nodes are expected to spend their most precious commodity for the sake of foreign nodes. This is a fine point and subject to further deliberation. There are currently no mechanisms in place to identify misbehaving nodes, which may choose to conserve their own energy reserves by not forwarding traffic, or try to take advantage of surrounding nodes by offloading their traffic onto them. Even if we assume that all nodes are "well-behaving", power management techniques developed in the cellular wireless architecture are not readily extendable to MANETs. Mispredictions can be very costly in this environment. Repercussions may range from missing traffic and causing retransmissions to occur (thus offsetting any energy gains), to causing network segmentation and communication failure. Schemes for the cellular wireless architecture often employ the services of the base station in order to achieve coordination and a certain degree of protection from network unpredictability. Without the benefit of a centrally overseeing base station though, idling down the network interface in a mobile ad hoc network becomes a risky proposition.

The aforementioned notion of misbehaving nodes raises additional concerns about the security aspects of mobile ad hoc networks. "Do we trust intermediate nodes to forward our traffic?" "How do we know ascertain the identity and credentials of a foreign node, since there is no trusted third party that we can consult in a MANET?" "How do we protect ourselves from misbehaving nodes that try to inject false routing information to the network?". Such questions, as well as many others concerning security, need to be carefully examined and answered before any realizations of mobile ad hoc networks come to be. Encryption is touted as a possible solution, however it comes at the cost of additional computational resources and increased complexity [74]. Furthermore, due to the network's ad hoc nature, nodes are still vulnerable to man-in-the-middle attacks, even when encryption techniques are used.

Finally, there has been a plethora of protocol proposals, especially about the routing problem, submitted to the IETF MANET group. Nevertheless, the standardization process is moving along at a very slow pace and this hinders the deployment of MANETs.

1.5 Previous Work in MANETs

1.5.1 MANETs' Routing protocols

Routing protocols in mobile ad hoc networks can be classified in three general categories: proactive (or table-driven), reactive and hybrid. Proactive routing protocols (e.g. TORA [49], DSDV [48]) try to maintain a complete overview of the network topology and obtain routing information for all nodes, even those that may be inactive. Reactive routing protocols (e.g. DSR [27], AODV [28]) on the other hand, try to minimize the loss of bandwidth resources caused by the exchange of redundant routing messages, and establish routes only on demand, i.e. between active pairs only. Finally, hybrid approaches (e.g. ZRP [51]) make use of both techniques, depending on the local context. It has been shown that reactive routing protocols generally outperform the proactive ones and we will not therefore concern ourselves with the latter. This section will introduce some of the most prevalent routing schemes for mobile ad hoc networks.

DSR [27] works by injecting the whole list of nodes through which a packet must pass in its header. Therefore, intermediate nodes need not have up-to-date routing information; only the source needs to know the complete hop sequence to the destination. Two mechanisms are integral to the operation of DSR, route discovery and route maintenance. The former is invoked when node **A** wishes to communicate with a node it has currently no route to, node **B**. **A** then initiates a Route Request Packet (RRP) that is flooded through the network. Route request packets carry the list of nodes visited and when a node with a route to **B** is visited, a Route Reply packet is sent back to **A**. The route discovery process is depicted in figure 1.4. The RRP packet from node **A** passes through nodes **D** and **C** and this sequence of nodes is recorded in each RRP. Finally, when it reaches its destination node **B**, a Route Reply packet is sent back to node **A** following the reverse sequence of nodes and containing the newly constructed path **A→D→C→B**.

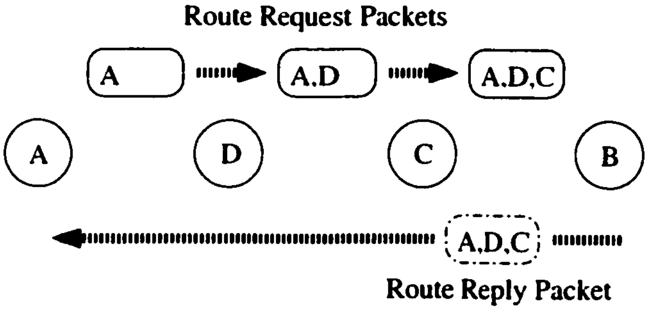


Figure 1.4 Example of DSR's route discovery process

This approach also prevents the creation of routing loops because nodes can examine the list of intermediate nodes in a packet's header. Additional gains can be made, if each node visited operates in promiscuous mode and examines the RRPs that pass through it. It may then effectively learn routes to other nodes in the network. Intelligent caching can thus be employed in order to reduce even further the adverse effects of flooding. However, it is questionable if mobile nodes should operate in promiscuous mode; energy and/or security concerns may render such an mode unfeasible. The route maintenance mechanism is used when **A** detects that the route it was using to **B** is not valid anymore. When such an event occurs, source node **A** is notified by the node that detected the link breakage with a Route Error packet. **A** can then try another route to **B**, if one exists in its cache, or initiate another route discovery to **B**. DSR also provides a mechanism for salvaging any packets that are already in transit. The node that experiences that link break may search its cache and modify the source routing information in any packets it receives from **A** to **B**, provided of course that it does have another route available. If it doesn't, it simply drops the packets and sends back a Route Error packet to **A**.

AODV [28] is another popular reactive routing protocol for MANETs. It employs a mechanism for route discovery and maintenance similar to the one used by DSR, and in addition, it uses hop-by-hop routing and sequence numbers as the Destination Sequenced Distance Vector (DSDV) protocol. When node **A** needs to connect to node **B**, a Route Request message is broadcast. This message is then flooded through the network, causing the nodes through which it passes to update their information for **A** and create a reverse route from itself to **A**. Route request messages also carry the most recent sequence number for destination **B** that **A** is aware of. If a node receiving this message knows of a route to **B** and its sequence number is equal or greater to the one indicated in the message (thus indicating a "fresher" route), then this node sends a Route Reply message back to the originating host **A**. As route reply messages travel back to **A**, intermediate nodes update their routing tables accordingly. In this way, a hop-by-hop path from **A** to **B** is formed and the creation of routing loops is prevented. Link failures are detected with the help of the underlying MAC layer and trigger the transmission of a link failure notification to all active neighbors that were using this link. Eventually, this notification will reach the originating source A, which in turn will initiate a new Route Request broadcast. Multicast routing can also be performed by AODV. If a node wishes to join a multicast group, it broadcasts a Route Request message, which has the destination IP address of the group and a flag that indicates its intention to join the group. A node that receives this message responds with a Route reply message if it already a member if the multicast group and has a fresh enough sequence number for it. The source may get several such responses from nodes participating in the group. It then chooses the one that is closest to it (in terms of hop counts) and notifies that node to direct multicast traffic to it as well.

The Location Trace Aided Routing (LOTAR) protocol [55] represents a hybrid approach to routing. Similar to LAR, [56], it assumes that each node has knowledge of its position, speed and the global time. This could be accomplished through the use of GPS receivers. LOTAR introduces a location table (LT), which stores location, speed and time information about its neighbors. This table is exchanged periodically and assists in route discovery, which is performed on demand. The information in the LT table is used to predict the location of the destination node and thus effectively "tighten" the search area of the route discovery process. LOTAR incorporates additional intelligence in the form of flow handoff heuristics. It tries to predict when an active path will break, when intermediary nodes are moving out of transmitter radius for example, and initiates a flow-handoff mechanism prior to the path being broken. Therefore, it successfully minimizes route discoveries and their associated costs, as well as maintains a high throughput, even in the case of highly mobile nodes.

ZRP, [51], is yet another hybrid approach. Its main motivation is that purely proactive protocols incur a high tax on the network when performing new route queries. Such requests are flooded through the entire network, and may result in "broadcast storms" [30]. ZRP introduces the concepts of a Routing Zone and Zone Radius. A node's Routing Zone consists of all its neighbor nodes that are at most Zone Radius hops away from it. ZRP utilizes a table-driven approach, the Interzone Routing Protocol (IERP) [57], within each zone and a reactive routing protocol, the Intrazone Routing Protocol (IARP) [58], for discovering routes to destination nodes beyond the boundaries of the zone in question. When such a route is needed, Route Request Packets are sent to the border nodes of each zone. In this fashion they propagate across different zones until the desired route is discovered.

1.5.2 Energy conserving routing protocols

The aforementioned routing solutions are totally energy agnostic. Route selection is based on traditional, lowest hop-count criteria. This choice makes sense for optimizing throughput and minimizing latency, however it may prove costly from an energy consumption point of view. In "Power-Aware Routing in Mobile Ad Hoc Networks [14]", the authors propose the use of power-aware metrics for path selection. Contrary to traditional routing metrics that aim to minimize the hop count or end-to-end path delay, power-aware metrics place battery power as the most important resource. Depending on the chosen policy, the routing protocol tries to minimize the energy consumed per data packet or to avoid nodes with low energy reserves. Additional policies might be maximizing the time to network partition or minimizing the variance in node energy levels. Noteworthy energy savings are achieved when these metrics are used in lieu of shortest path ones. However, the results presented are for static random graphs, and it is not clear how node mobility may affect this approach.

PARO [33], short for Power-Aware Routing Optimization, attacks the same problem from a different angle. A critical observation that the authors make is that received power is proportional to the inverse of the squared distance between transmitter and receiver. Therefore, it may be sound from an energy standpoint to reach a distant node via a longer path of many short hops, than following a short path of long hops. In light of this realization, the authors propose ways to select and maintain routes that minimize overall transmit power. However, this method depends crucially on the underlying physical model used for adjusting the transmission power.

Adaptive Energy-Conserving Routing [39], introduces two algorithms designed for power conservation in MANETs that work on top of existing proactive routing protocols. The first one, the Basic Energy-Conserving Algorithm (BECA), utilizes application and routing layer information and tries to minimize the time we keep the transceiver powered up. Using such higher-level information, it is possible to power down nodes that are not part of any active paths and periodically turn them on again to accommodate possible routing changes. The second one, the Adaptive Fidelity Energy-Conserving Algorithm (AFECA), is based on the observation that in densely populated ad hoc networks, there are many neighbor nodes, which may form alternative paths. Therefore, a node may increase the time it has its wireless interface powered down, since there are many neighboring nodes that may accommodate routing requests.

Finally, Tassiulas and Chang [23] propose a class of routing algorithms that are designed to maximize network lifetime, as opposed to minimizing individual packet transmission energy. The proposed algorithms take into account the remaining battery power of each node and thus effectively "spread" traffic across the whole network and avoid "overtaxing" any single path. Although the ideas and results presented are most interesting, the MAC has been abstracted away and the authors assume point-to-point, collision free links. Additionally, nodes are assumed to be static or very slowly moving.

1.5.3 Datalink Layer Proposals

Most mobile ad hoc network simulations assume the use of IEEE 802.11 radios. Given its widespread popularity and commercial availability, this is a sound choice. However, being a contention-based protocol it is generally very expensive in terms of energy consumption. The IEEE 802.11 [15] standard provides for power management controls while operating in ad hoc mode, through the use of Ad hoc Traffic Indication Maps (ATIMs). Since MANETs operate without the benefit of fixed supporting infrastructure, wireless nodes need to buffer packets themselves when operating in power-management mode, because their destination node is not awake all the time. Buffered packets are advertised during a contention period when all nodes are awake. The destination nodes are announced in the ATIMs. Therefore, nodes not found in the ATIMs are able to sleep during the following transmission cycle. However, the standard does not specify or make any recommendations about how to implement the aforementioned policy.

PAMAS [31] extends the original MACA protocol by adding a signaling channel. This addition enables nodes that are in the transmitter radius of an active node to determine periods of time that is safe for them to power down. Furthermore, it assists in negating the hidden and exposed terminal problems. PAMAS claims impressive energy savings over contention-based protocols, which may reach 70% in some cases. However, PAMAS does not address the problem of minimizing the energy consumption of nodes that are idle and not overhear any ongoing transmissions themselves.

1.6 Our Proposal in MANETs

The routing problem has received the most attention so far in MANETs. However, we are not going to concern ourselves with this problem since there have already been numerous proposals about it. We concentrate our efforts on studying the performance and energy-consumption characteristics of a typical, simulated MANET. We employ the most popular routing protocols, AODV and DSR, in order to quantify which operations are the most energy expensive and whether the mobility characteristics have any significant impact on them. Additionally, we comment on the usefulness or not of IP-compatibility in the context of MANETs from the energy conservation perspective.

Power management in MANETs is considerably more complex than in cellular wireless networks. Most power conservation techniques in the latter type of networks depend on a base station for coordination and management. Therefore, they are inapplicable in MANETs, where the lack of any supporting infrastructure is given. We illustrate a simple energy conservation approach to 802.11, which maintains the semantics of the collision avoidance mechanism but attempts to use as little energy as possible for the data transfers. The real-world applicability of our proposal is related to the ability to include a suitable model for the propagated signal strength. Our chosen model has been shown to be accurate enough when operating in an outdoor environment and having line of sight [62]. By extension, knowledge of a model that fits the operating environment of hosts, could be equally well be used with the same objective.

We also examine the performance of the underlying MAC layer in a typical MANET simulation. In an effort to make our simulations more realistic, we examine the impact of extended carrier sense to the energy characteristics of our network. The extended carrier sense captures scenarios whereby the receiver sensitivity extends far beyond the vicinity reachable by the transmitter. We conclude that under such conditions, a typical MANET operating under 802.11 will suffer severe performance penalties due to increased collisions. Such observations raise serious questions about the viability of 802.11 based MANETs and provide substantial evidence that alternatives to 802.11 should be sought after if MANETs, and power-efficient MANETs in particular, are to be realized. We conclude with future research plans along the lines of remaining work towards realistic solutions for energy conservation in mobile network.

2. Selective Idling: Experiments in Transport Layer Energy Conservation

2.1 Introduction

Part of the challenge facing mobile computing is the need to provide support for existing applications and protocols while maintaining energy consumption to a minimum in order to achieve maximum autonomy. All mobile device components contribute to the overall energy consumption, including the CPU, the memory, any mechanically-driven storage device (such as hard drives), the screen and screen backlight, the network transceivers etc. From the networking perspective, we are concerned with the particular protocols (and protocol features) that allow the reduction of consumed energy at the network interface. Specifically, in this paper, we are interested in ways in which the transport layer protocol, possibly in synergy with other mechanisms, can participate in conserving the energy consumed at the network interface devices. It should be observed that most wireless network interface cards provide the option of deactivating the interface while maintaining it ready for re-activation. We will call such a state, the *idling* state (some authors call it the "doze mode"') which should not be confused with powering the device down. Idling still consumes some energy but much less than when transmitting or receiving data. Our objective is to keep the interface idle for the longest possible time without hurting the transport layer protocol performance.

In this study we consider a cellular architecture. That is, the first hop from the mobile to the network is a wireless link. The wireless link connects the mobile to a *base station*. Since base stations are aware of their particular position along the path towards the mobile, they are assumed to be capable of providing particular protocol features and possible extensions to the data link protocol used to communicate with the mobile. In the study, initially, no particular data link protocol features (and hence, no base station participation) is assumed. Subsequently, we will add data link extensions (and base station participation) to assist in the energy conservation task. The second assumption made in this paper is that a remote peer in the fixed part of the network uses a legacy transport layer protocol to communicate with the mobile. Specifically, it will use either UDP or TCP. Due to its definition as unreliable service, UDP will not be our concern. However, the performance of TCP is important because both the delay and the loss experienced by a TCP flow have a direct impact on the flow's throughput. Web page and file transfers, e-mail retrieval etc. are among the applications that almost always use TCP as the transport protocol. The particular assumption we make is based on the fact that remote peers in the fixed network are unlikely to support special protocols for communication with mobile endpoints. Hence, we take a view contrary to [5] and [8], in that the overhead of any mechanisms for energy conservation falls exclusively on either the mobile alone, or on the combination of mobile/base-station. At the same time, the end-to-end semantics of TCP are fully observed between the mobile and the fixed network peers. TCP's energy consumption behavior depends on its use of congestive losses as the means to regulate its window. That is, TCP includes the mechanism that produces congestion by increasing its window and overflowing the router buffers as the means to identify the extent to which it can expand its window. The congestion is subsequently relieved by suitable adjustments of the window size. However, the damage is already done in terms of retransmissions necessary to recover the lost packets. The retransmissions cost additional energy consumption. Moreover, the anticipation to receive ACKs for transmitted data in order to further expand its window, implies that a TCP flow must be at all times ready to receive ACKs for the data is sent out. In a mobile environment, this is equivalent to being ready in a "receive" mode all of the time. We question whether this is a good idea, especially if the data is sent out in bursts, and the ACKs are received, similarly, in bursts. Moreover, the cumulative ACK nature of TCP's ACKs, should suggest that a loss of an earlier sent ACK can be successfully compensated by an ACK sent later and received successfully. Thus, some loss of ACKs can be tolerated.

We will see in the following how the loss of ACKs can be the result of intentionally idling the transceiver interface.

If we investigate the activity of a TCP connection we can identify periods of time during which no transmission or reception is anticipated to occur. As a motivating example we can use the scenario whereby a mobile node sends a data packet and waits for its acknowledgement (assuming that the current window size permits only one packet to be transmitted, as is the case of the beginning of TCP's *slow-start* phase). Depending on the exact round-trip time (RTT), the mobile's transceiver could be idling until just before the anticipated reception of the acknowledgement. Generally speaking, the RTT is not known in advance. However, a task of significant complexity that is already handled by TCP is precisely the RTT measurement and the derivation of reasonable RTT estimates which are subsequently used in the calculation of timeouts.

In this paper we demonstrate how the RTT estimates calculated by TCP can be used to drive a mechanism that selectively idles the transceiver, in order to conserve energy. We will call this modification, *Selective-Idling TCP* (SI-TCP) Clearly, the choice of idling for a particular duration may not be correct, leaving the possibility that (a) either the idling causes the transceiver to lose an ACK, or, (b) it normal transceiver operation is reinstated earlier than necessary (thus not conserving as much energy as it could). The point is that the dynamics of the RTT calculation impact directly the achievable energy savings.

The remaining of the paper is structured as follows. In Section 2.2, we position the mechanisms proposed in this paper with respect to the protocol stack. In Section 2.3 we describe the basic operation of Selective Idling TCP (SI-TCP) without any assistance from the base station. In Section 2.4 we provide simulation results of SI-TCP that demonstrate its behavior. We conclude that certain anomalous conditions at high loads can be corrected to an extent with the additional support of a dedicated data link protocol between the base and the mobile. The necessary extensions are described in Section 2.5. In Section 2.6 we summarize the simulation results with the extensions involving the base station. In Section 2.7 we draw our conclusions and suggest avenues for further research.

2.2 Architectural Considerations

An issue that arises in the study of energy conservation with respect to the network component of a mobile device, is the location in the protocol stack where energy-related decisions are to be taken. We summarize here the four particular positions where energy conservation actions can be taken and which ones fall within the scope of our proposed scheme.

2.2.1 Application Layer

In this case, the user application running on the mobile is totally responsible for the energy conservation choices. An application is allowed to invoke primitives that directly activate or idle the interface devices. While this approach holds some promise, in that applications decide on the exact energy conservation policy, it cannot handle complications that arise from the interaction of applications with transport layer protocols. For example, an application should not be allowed to shut down the interface if there are still active TCP flows. That is, TCP may wait for and acknowledge packets of a flow (possible duplicates routed on a different path) even when the corresponding connection is unilaterally terminated by the application. In order to provide the correct endpoint delivery semantics towards the remote end, the TCP entity on the mobile will continue responding to packets from the remote peer even though, as far as the application running on the mobile is concerned, the connection has been terminated. Thus, the application layer is not in the best position to decide when to idle an interface. In this study we do not consider application-driven energy conservation.

2.2.2 Transport Layer (API)

In order to avoid the complications due to the direct involvement of the applications in handling the energy states of the interfaces, the transport layer API can be extended to include energy management *suggestions* ("hints") directed by the application. The difference here is that the application has no direct control over the interface states. Only the protocol stack (and the transport layer in particular) can control the interface states. For example, a close operation on a TCP socket can now include code that forces the interface to idle once all activity of this flow is finished and no other TCP flows are active. This approach avoids complications but is a large-grained approach, since it allows energy conservation decisions on a per-flow basis. The large granularity of energy conservation means that if the communicating endpoint is not fast (low throughput) or even non-responsive, then the interface remains activate even though, as far as the flow is concerned, no progress (packet exchange activity) occurs.

2.2.3 Transport Layer (internal)

In order to avoid the large grained decisions at the transport layer API, the alternative taken in this study is to implement energy conservation decisions on a per-packet basis. That is, the interface can be idled based on per-packet decisions. An example is the anticipation for an ACK as a result of a transmitted data packet. An ACK can be expected to be received at around the time the corresponding data packet was transmitted plus the RTT delay measured most recently. During this period of time, the interfaces can be idled. This strategy introduces the probability that the mobile will be idling when the ACK is forwarded from the base to the mobile, and hence the ACK could be lost. As long as the risk is probabilistically low, the impact of the strategy on the throughput ought to be minimal. Moreover, if the base station participates actively in scheduling the delivery of ACKs to the mobile (for example, via a protocol that establishes mutually agreeable windows of time during which the mobile's interface will remain active) then the potential for losing an ACK is eliminated, as long as the ACK makes it to the base station.

2.2.4 Data Link (and MAC) Layer

By far, the data link and MAC layer has received most of the attention in terms of energy conservation (see for example [1, 3]. The emphasis is not accidental, since contention protocols are inefficient in terms of consumed energy at high loads (where successive collisions are likely). Protocols with slotted frame structures are generally performing better in terms of consumed energy for data delivered but introduce coordination overheads. While certain data link layer protocols avoid transmission collisions, and can therefore be used to control the time spent transmitting, they are not in a position to determine the instants when a mobile is going to *receive* data or ACKs. For example, when a slotted protocol is considered, the interface of the mobile must be active whenever an opportunity to receive data appears, regardless of whether data are eventually received or not. In fact, in contention protocols, the time during which a packet may be received extends to the entire time period with the exception of the time when the same mobile is transmitting. The question that is addressed indirectly in this study is the extent to which the data link layer can be used to convey control information between base and mobile in order for the mobile to idle even though it anticipates packets (incl. ACKs) to be received.

Note that we have not listed energy conservation actions inside the network layer. This is consistent with our scope of cellular architectures only, where mobiles do not participate in routing. However, Mobile Ad-Hoc Networks (MANETs) present further opportunities to include energy conservation policies in the network layer, since they rely on routing over multihop wireless. Our approach is to consider Transport Layer energy conservation as well as extensions to the Data Link Layer that can assist the Transport Layer energy conservation.

2.3 Selective Idling

When instantiated, a TCP connection is unaware of the RTT it should expect to experience. After a short transient, which uses an initial large, thus forgiving, RTT estimate, RTT measurements, *M*, are taken using the fact that acknowledgements are almost immediately generated upon arrival of a packet at the remote endpoint. The smoothed RTT estimator, *R*, is thus updated:

$$R = aR + (1-a)M \tag{2.1}$$

where a is typically set to 7/8.The effect of interfering cross traffic in the Internet "cloud" is evident as increased variability of the RTT. The variation in the round-trip time of packets is accounted for in the timeout, *RTO*,

$$RTO = R + 4D \tag{2.2}$$

where *D* is a smoothed metric of the variation, defined as:

$$D = \beta D + (1-\beta)\,|M - R| \tag{2.3}$$

where β is typically 3/4.

2.3.1 ACK-Reception Interval

In the scheme presented in this paper, upon sending a TCP packet at time *now*, if the last measured round-trip time was *M*, and the last smoothed variance was *D*, then the transceiver will be set to idle until the timepoint *now* + *M* - *cD*. It will subsequently be activated for the interval:

$$[now + M - cD,\quad now + M + cD] \tag{2.4}$$

where c is a constant (in the following it will be assumed $c = 1$). That is, the transceiver will be energized up again with the intention of receiving the ACK corresponding to the transmitted packet. One interval (4) is maintained for each packet sent. The intervals may overlap, and when they do, it is possible that they force the transceiver to be continuously active. We note that instead of *M*, in the interval calculation (2.4) we could instead use *R*. While we have experimented with both versions, using *M* allows for a more responsive adaptation of the "reception intervals" than *D* does, however their performance does not differ significantly.

If ACKs are lost as a result of the wrong prediction of the corresponding ACK-reception interval, TCP may incorrectly infer a packet loss and reduce its current window (as per the operation of TCP-Tahoe). Therefore, our original attempt to conserve energy may in fact cost more energy because of retransmissions caused by lost ACKs. Fortunately, the cumulative nature of TCP acknowledgements aids in restricting the extent to which an occasional lost ACK harms the performance of the system.

Finally, in order for the ACK-reception interval to be calculated for the first packets, we adopt a simple bootstrapping technique whereby the intervals are set initially to [*now* + 0.7 *M*, *now* + 1.3 M]. Moreover, when timeouts and duplicate-ACKs trigger the congestion recovery algorithm of TCP, the transceiver is activated continuously until an acknowledgment is received, in order to minimize the impact that selective-idling has on the congestion recovery process.

2.4 SI –TCP Simulation Results

The results presented in this section are based on a configuration with cross-traffic added to the bottleneck link between the endpoints of the observed TCP connection. The link and width for all links was 1 Mbit/sec. Low values for cross-traffic intensity capture an underloaded network and high values a highly-loaded (to almost saturated) network. The cross-traffic followed an ON/OFF model with exponential ON and OFF periods. The average ON period is 0.5 sec. The average OFF period is also 0.5 sec. The ON/OFF model attempts to capture the aggregate load of several combined TCP flows, hence the "bursty" nature of the model. The cross-traffic is non-cooperating in the sense that it does not adapt to congestion conditions. The maximum receiver window is denoted by *w* and is measured in units of 1000 byte packets. All data packets are of the same size of 1000 bytes. The propagation round-trip time is 300 msec (except where noted). The communication between base and mobile is considered to be reliable, although in principle it can be assumed to be unreliable and the same scheme can be applied with suitable modifications. Finally, the various simulations were run for a total of 10,000 seconds.

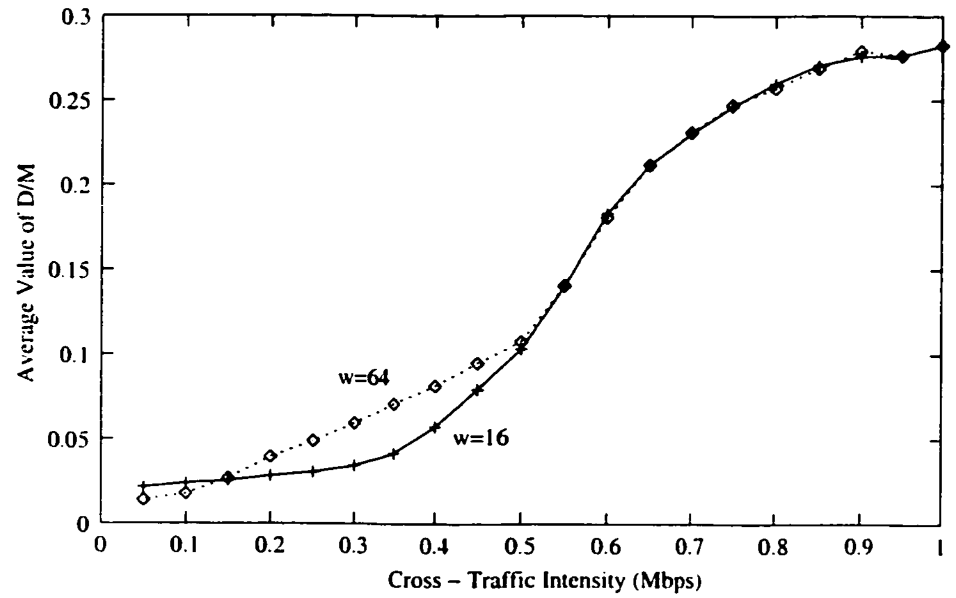


Figure 2.1 Average of the ratio between the observed variance estimator, *D*, and the measured RTT, *M*, for variable network load (receiver window, *w*, set to 16 and 64 packets)

As a first experiment, we consider Figure 2.1 where the relation of *D/M* versus *w* illustrates the nature of the ACK-reception interval. At low loads, the variance is limited if the window size is limited. The reason is that the larger window sizes result in larger bursts being sent by TCP, this inflating the variance of the delay experienced by the TCP connection itself. Thus, we anticipate limited value to the selective idling scheme if the window size is larger. The case is illustrated in Figure 2.2.a. The relative energy consumption in Figure 2.2.a compares how much energy is consumed (as active transceiver time) to successfully send a certain amount of data by SI-TCP versus the same data being sent using plain TCP-Tahoe. A ratio of 1.0 indicates identical energy consumption between SI-TCP and TCP-Tahoe. A lesser value leads to savings. For example, for w = 16 and a bottleneck link utilization of around 30% (representing 300 kbps of interfering traffic out of 1000 kbps available at the bottleneck link) SI-TCP can send the same amount of data as TCP-Tahoe would, despite keeping the interface active for only 50% of the time. Figure 2.2.b suggests that in lower loads, the throughput achieved by SI-TCP is not much different than that achieved by plain TCP-Tahoe. Hence, at low loads, SI-TCP not only delivers the data consuming less energy but the data transfers are accomplished at the same rate as TCP-Tahoe.

Not all scenarios lead to energy savings for SI-TCP. In fact, all points in Figure 2.2.a above the line corresponding to 1.0 consume more energy in SI-TCP that in plain TCP-Tahoe. The primary cause is the increase of the delay variance due to the network load. That is, at high loads not only there are no energy savings, but also more energy is expended in retransmitting packets for which the ACKs were not received because the mobile was idling.

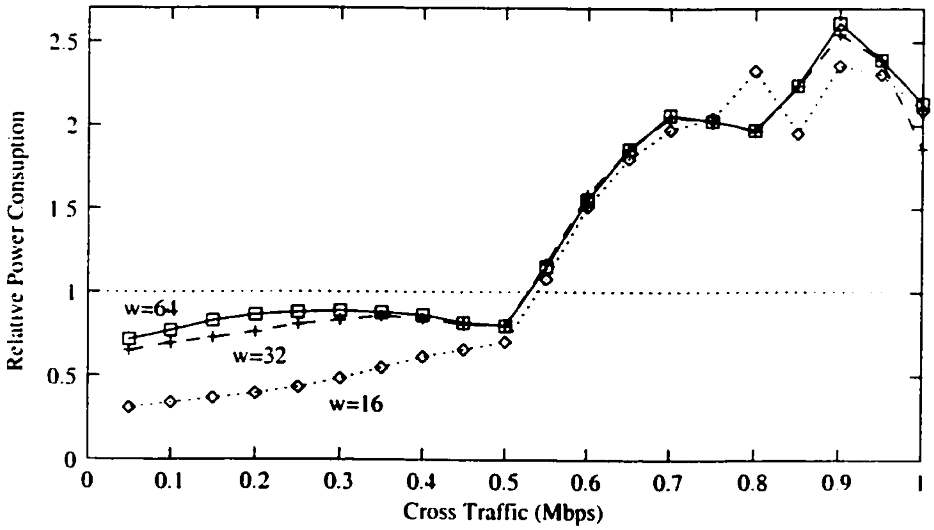


Figure 2.2.a Energy consumption of SI-TCP relative to TCP-Tahoe

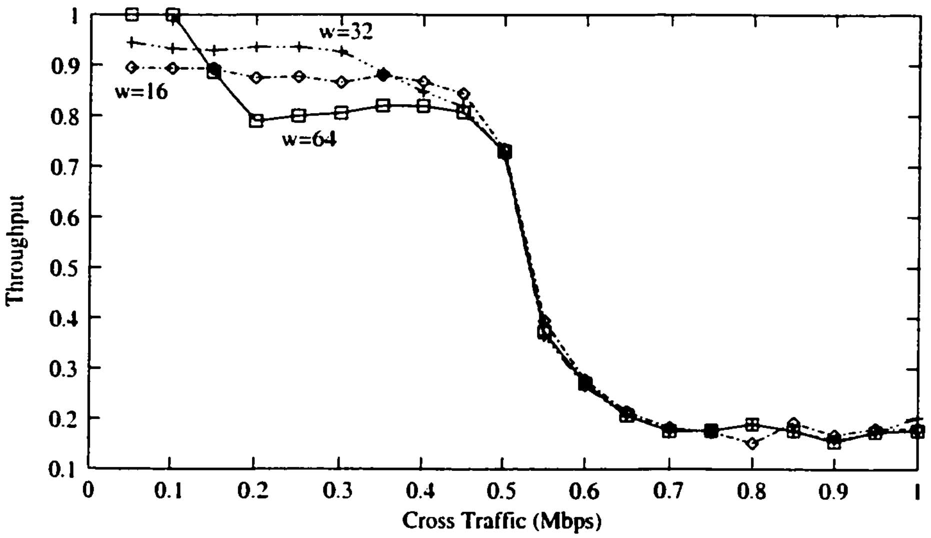


Figure 2.2.b Throughput of SI-TCP relative to TCP-Tahoe

To better explain the results at high loads, consider Figures 2.3.a and 2.3.b where the average time spent in timeouts and the number of timeouts to send the same amount of data as TCP-Tahoe is indicated. What can be clearly seen is that at high loads, the timeouts are frequent but each one of them is also longer. To put it in different words, we have two alternatives at high loads neither of which helps the energy conservation task. The first is to consider a large variance, and hence keep the transceiver active for longer periods of time when activated. The second is to attempt to keep the active time of the transceiver short but then, increase the probability to lose ACKs and subsequently enter the timeout and retransmission state during which our strategy is to keep the transceiver continuously active to recover from the timeout as soon as possible. In both cases, energy is consumed rather than conserved. In the following section we present modifications to the scheme that avoid the pathological energy consumption scenarios of SI-TCP at high loads.

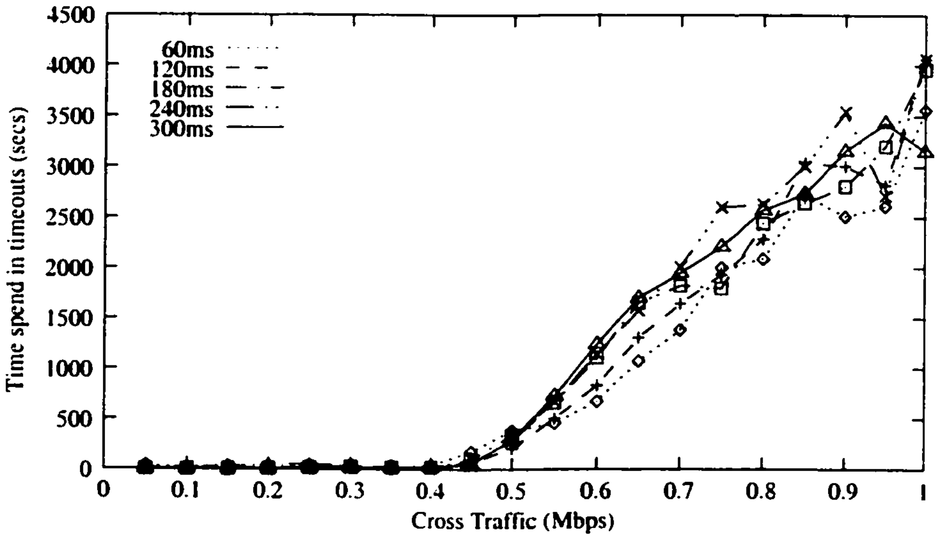


Figure 2.3.a Total time spent in timeouts by SI-TCP for different end-to-end propagation delays

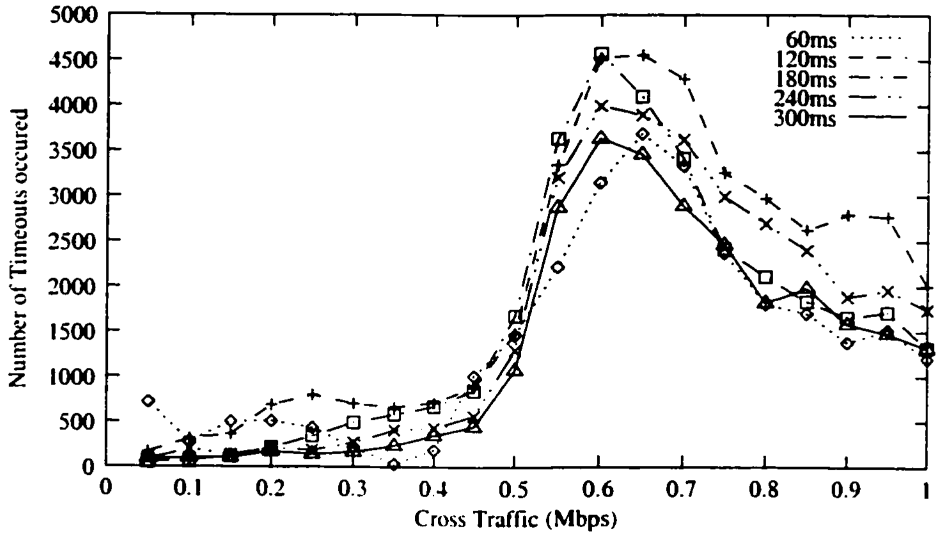


Figure 2.3.b Total number of timeouts experienced by SI-TCP for different end-to-end propagation delays

2.5 Base-Station Enhanced SI -TCP (BS-SI-TCP)

The original version of SI-TCP suffers greatly when due to increasing interfering traffic our RTT estimates are incorrect and, as a result, ACKs are lost. One way to address this problem is through the use of an intelligent base station that is aware of the energy conservation efforts of the SI-TCP agent and cooperates with it closely. To this end we have incorporated the following strategies at the base station and the mobile host:

Packet Time-stamping at the mobile host.

Each packet that leaves the SI-TCP and passes through the base station carries information about the estimate that our agent made and an announcement of the time interval during which the mobile will be active in anticipation of the corresponding ACK. Therefore, the base station is able to reconstruct in detail the mobile's activation schedule.

ACK-delaying at the base station.

When an ACK arrives at the BS, it is either sent immediately if the mobile interface is active, or delayed until the time that the mobile's interface will be activated again. Using a BS in this capacity, we have no lost ACKs due to erroneous RTT measurements. At most, we have ACK delays. Furthermore, when the SI-TCP agent at the mobile times out and retransmits a packet, this indicates to the BS that the agent has turned itself on again and ACK transmission may continue towards the mobile may resume. Note that, as in the previous experiments, the BS is assumed to be always active, since it does not face energy consumption limitations.

ACK-compression at the base station

In addition to ACK-delaying, the BS employs two more heuristics that further push the performance envelope of BS-SI-TCP. These methods are ACK compression and ACK time-stamping. Our simulations have shown that when utilizing only ACK-delaying at the BS, there are times when several ACKs are delayed at the BS waiting for the mobile receiver to be activated. When this event finally occurs, they are transmitted to the mobile and cause the transmission of new packets.

ACK Time-stamping at the base station.

We noticed two unwanted side-effects of using only ACK-delaying. The first is that the time ACKs were delayed is added to the RTT calculation at the mobile resulting in incorrect RTT estimates. We mitigate this problem by having the BS timestamp the ACKs at the moment of their arrival at the BS. This timestamp is then relayed to the mobile as part of the data link layer payload that encapsulates the corresponding ACK. The SI-TCP agent at the mobile uses this timestamp instead of the actual arrival time of the ACK to the mobile in order to calculate the RTT. That is, the additional delay from base station to mobile is effectively eliminated from the RTT measurement.

The burst of delayed ACKs from the BS results also in bursty transmissions from the mobile. Due to layering constraints though, our original SI-TCP implementation passed all the new packets to its data link layer without knowing the exact time that they would actually be transmitted. We further increase the accuracy of RTT estimates by violating the layering principle at the SI-TCP agent. The BS-SI-TCP agent is able to peek into the link layer's queue and determine the exact moment of transmission of each packet.

The second drawback to our approach manifests itself mostly at high loads. When several ACKs get delayed at the BS, they have usually suffered different congestion, and experienced different delays. Therefore, new packets that are generated by SI-TCP because of these ACKs have very different RTT estimates, all within one window of data. However, we can capitalize upon the cumulative nature of those ACKs and have the BS keep only the more recent one. Of course we do not know which RTT sample is actually the best or the most accurate, but at such extreme interfering cross traffic load, any valid guess of the RTT is as good as the next one. Furthermore, ACK compression affects the way TCP opens up its congestion window. When it does occur, TCP increases its window in a less aggressive way. Note however that we queue up to three ACKs with the same sequence number, in order to avoid violating the operation of the fast retransmit mechanism which is now present in most TCP implementations ("triple duplicate ACK").

2.6 BS-SI-TCP Simulation Results

By employing all the techniques listed in the previous section we are able to significantly improve the performance of SI-TCP, especially at high loads. In addition to performance improvements, the inclusion of an intelligent BS in our scheme also allows us to solve the reverse flow problem, where the mobile is on the receiving end. This is because, in addition to a schedule for the delivery of ACKs, the base station can establish a periodic activation schedule for the mobile to ensure that data and connection requests are delivered to the mobile from peers at the fixed part of the network.

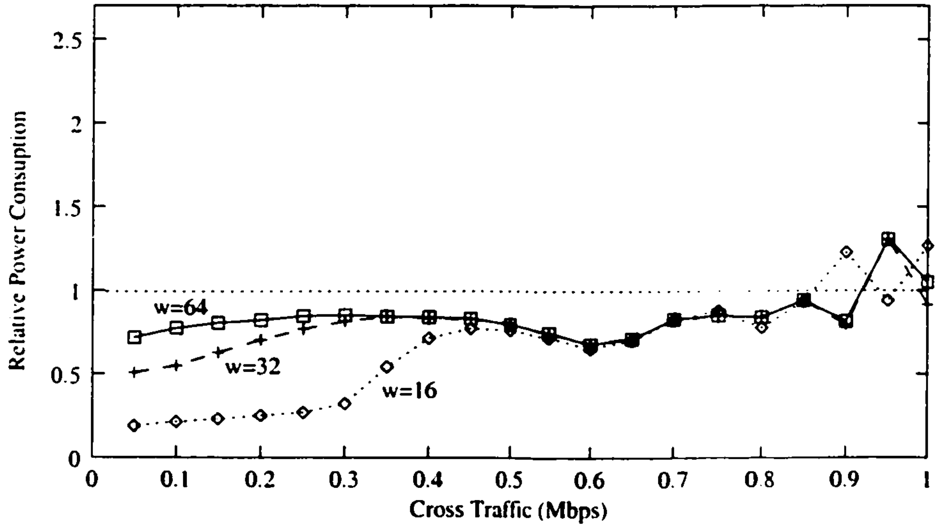


Figure 2.4.a Energy consumption of BS-SI-TCP relative to TCP-Tahoe

Figure 2.4.a depicts the total energy consumption of BS-SI-TCP relative to TCP-Tahoe and it should be compared to Figure 2.2.a. BS-SI-TCP clearly outperforms the original version of SI-TCP. Whereas Figure 2.2.a depicts a rapid deterioration

of SI-TCP with increasing interfering cross traffic (esp. when exceeding 0.5 Mbps), BS-SI-TCP appears to not be performing any worse (at least in terms of statistically important difference) compared to the energy consumption of plain TCP-Tahoe under heavy congestion. In addition, BS-SI—TCP offers energy savings at even higher loads, up to 0.85 Mbps compared to 0.5 Mbps.

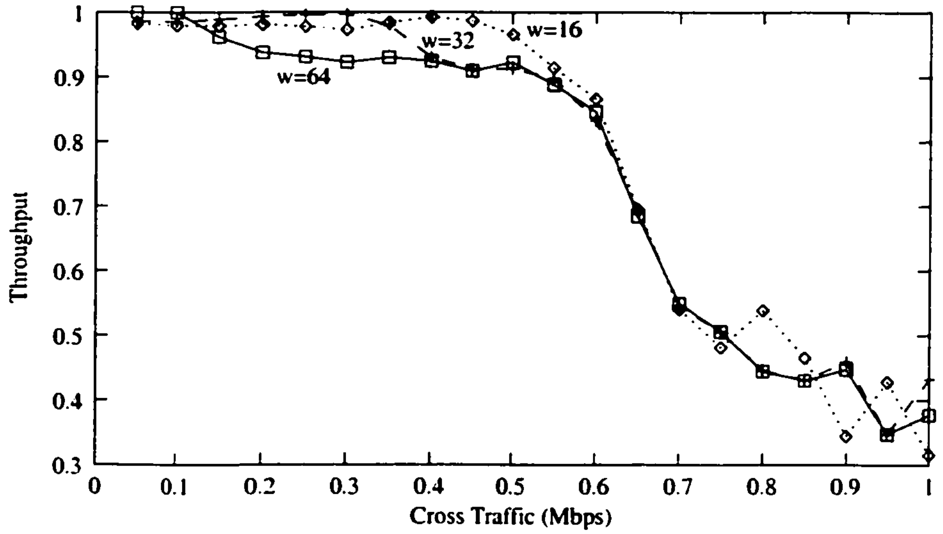


Figure 2.4.b Throughput of BS-SI-TCP relative to TCP-Tahoe

Figure 2.5.b illustrates the positive effect of ACK-delaying at the BS. A significant number of timeouts are kept short because an ACK was present at the BS which was sent immediately when the timeout at the mobile occurred and the retransmission took place. Contrasting Figure 2.5.b to Figure 2.3.b one can notice that the total number of timeouts occurring in BI-SI-TCP is significantly lower to the one of plain SI-TCP. This gain is a direct consequence of obtaining better RTT estimates and of the because of the ACK-delaying that occurs at the BS. As Figures 2.4.a and 2.4.b show, our energy consumption and throughput characteristics are better than those of the original SI-TCP. However, under extreme congestion, we still may expend slightly more energy than TCP-Tahoe. We have found that our RTT estimates in this case are no better than an "educated guess" and exhibits a large variance.

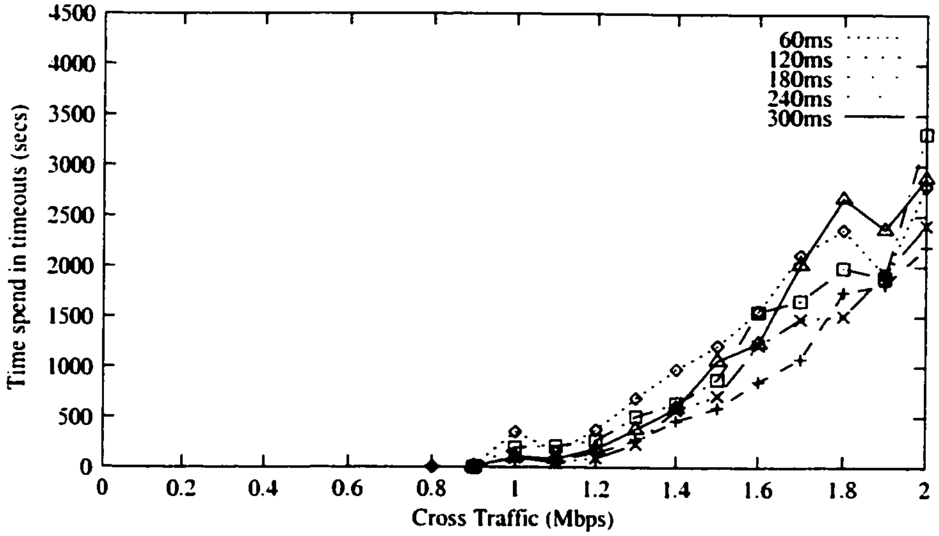


Figure 2.5.a Total time spent in timeouts by BS-SI-TCP for different end-to-end propagation Delays

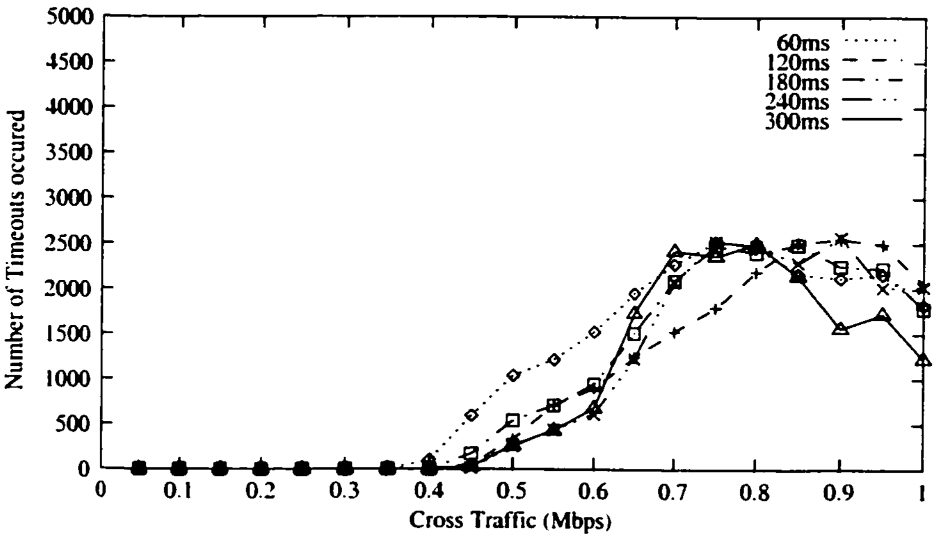


Figure 2.5.b Total number of timeouts experienced by BS-SI-TCP for different end-to-end propagation delays

2.7 Conclusions

Our simulations show that, with the added complexity of an intelligent base station, an energy conservation scheme can be developed without the need to completely replace TCP. The new scheme is energy efficient compared to plain TCP-Tahoe over a wide variety of interfering traffic. However, under extreme loads, it may underperform. Note that the question becomes one of scheduling when TCP connections should be established. Is it really worthwhile to establish a connection from a mobile when the fixed network is congested?

It appears that the need for retransmissions alone should discourage the establishment of a connection altogether, regardless whether it is plain TCP-Tahoe or SI-TCP or BS-SI-TCP. Hence, instead of replacing TCP, maybe the best approach is to schedule the TCP connections based on the congestion level of the network, by first probing the network. Then, once established, use SI-TCP or BS-SI-TCP to improve the energy consumption. To this extent, we are currently investigating schemes such as the one presented in [8]. Whereas [8] replaces TCP by a protocol that introduces a form of probing, we are instead considering a protocol that probes the congestion condition of the network concurrently with active TCP connections and starts, suspends or resumes the TCP connections according to the network congestion conditions.

Note that we have tried to avoid interfering with the TCP semantics of the endpoints, even though to achieve a certain level of performance, the protocol stack layers at the mobile and base station had to be collapsed. Still, the fixed network peer is not influenced by the changes we introduce.

The simulations only depict cases where the sender of the data is the mobile and the remote node is acknowledging. The reverse operation (data sent from remote to mobile) needs further techniques if selective idling is to be supported. The operation of the selective idling for data received by the mobile increases the experienced delay for the data transmissions originating at the fixed network peer, thus influencing the calculation of RTT at this remote peer. It is likely that the remote sender will misinterpret the increased RTT as an opportunity to increase the window size. It is therefore important that the receiver window (now at the mobile) must be set to a value that discourages the fixed sender to introduce more data in the bit pipe, that, inevitably, will accumulate at the BS. Indeed this is a concern when we consider the fact that the BS serves multiple mobiles and has to provide the appropriate buffering for all the traffic destined to its mobiles.

3. Energy Management in Mobile Ad Hoc Networks

3.1 Introduction

Mobile Ad Hoc Networks (MANETs) [26], consist of wireless, mobile nodes that form a multihop network without the assistance of any outside or pre-existing infrastructure. As such, MANETs represent a very complex operating environment and require sophisticated routing techniques in order to achieve and maintain communication. The most fundamental problem in MANETs, the routing one, has already been extensively studied. While there is yet no consensus about any specific routing protocols, we chose to work with two of the most promising ones and focus our efforts on energy conservation. Similar to wireless cellular networks, energy conservation is of paramount importance to MANETs as well. However, given their intrinsic characteristics we cannot readily apply the approach we followed in the previous chapter and are forced to seek new venues.

Our study commences with a detailed presentation of our tools, namely the Network Simulator 2, in section 3.2. We present the various modules that comprise our simulator and discuss their implementation, advantages and disadvantages as well as how they affect our results. Section 3.3 presents our analysis of MANET simulations. Initially we examine the energy breakdown of the various modules and identify their relative contribution to energy consumption. Following in section 3.4, we present our power management proposal for MANETs. Our simulation results show an approximately threefold decrease in the amount of energy required for unicast packet transmissions. Additional gains can be also observed in the amount of energy expended in receiving packets. Subsequently, we examine the behavior of MANETs when operating with very sensitive receivers in section 3.5. Such high sensitivity results in the receiver having a far greater sensing range than its effective transmission radius. The effects of this asymmetry are studied for MANET simulations with and without our energy management proposal. Our findings show that IEEE 802.11 performs very poorly in such an environment and becomes a serious detrimental factor. Finally, section 3.6 presents our conclusions and venues for future work.

3.2 Anatomy of a MANET Simulation

3.2.1 Network Simulator 2 (NS2)

NS2 [6] is a very popular and powerful discrete event simulator aimed at networking research. It dates back to 1989, when it first appeared as a variant of the REAL [63] simulator. NS2's primary advantages are its impressive array of network protocols and functions that span all layers of the protocol stack, as well as a modular design philosophy, which enables rapid configuration and setup. Furthermore, NS2 is available to the academic community at no charge and, most importantly, it has an active, worldwide user base.

For all its advantages however, programming in NS2 and extending its capabilities presents significant challenges. NS2 implements the split-language programming paradigm. It is build around C++ and Object TCL, the former being used mainly for computationally intensive tasks, whereas OTCL for setup and configuration. This duality however lends to complexity and a very steep learning curve. Furthermore, there are no guarantees about the validity of the simulator objects and validation is a major concern. This is particularly true for simulations of mobile ad hoc networks, since there is no real-world data against which one could compare the output of NS2.

Simulation of mobile ad hoc networks in NS2 is based primarily on CMU's mobile extensions [59]. NS2 tries to build a very detailed virtual world, where there is comprehensive support for the physical, datalink, network and application layers. Figure 3.1 depicts the architecture of a wireless node in NS2. In the following subsections, we will present with greater detail the various components of our simulator.

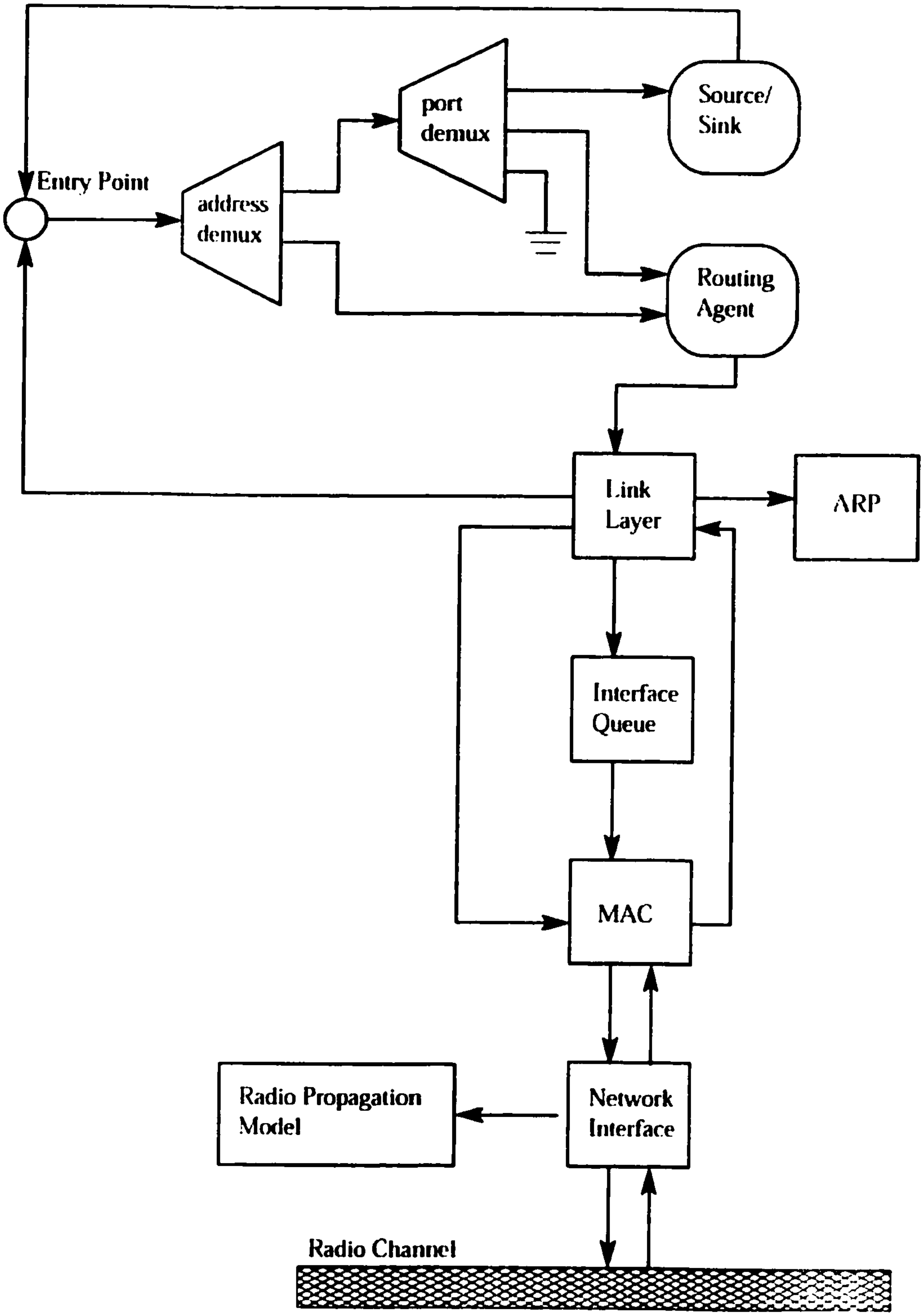


Figure 3.1 NS2's wireless node architecture

3.2.2 Physical Layer

Simulations in hard-wired networks very seldom have to deal with this layer. One can assume with a high degree of confidence that an electrical signal suffers too little degradation to be of importance while it is propagated down the wire. However, this hypothesis does not hold for wireless transmissions as well.

In its most basic form, the radio propagation model is based on the Friis Free-Space [52] attenuation model. Free Space however is a purely theoretical concept. It is used to denote a space that is void of all matter. The Friis model is used when real-world conditions approximate free space with sufficient accuracy. Therefore, equation 3.1 can be used to predict the received power P_R at the receiving interface when there is a clear, unobstructed line-of-sight path between the transmitter and the receiver, in the absence of all interference:

$$P_R = \frac{P_T \cdot G_T \cdot G_R \cdot \lambda^2}{(4\pi d)^2} \tag{3.1}$$

P_T denotes the transmitted signal power, G_R and G_T are the receiver and transmitter antennas' gains respectively, λ is the wavelength and finally d is the distance between receiver and transmitter. It is important to notice that under this model power attenuates in proportion to the inverse of the square of the distance, d.

Friis model depends on the assumption that we operate in conditions close to free space one. However, such a hypothesis is often unrealistic and we obtain optimistic power estimates through Friis's model, especially for far distances. In order to better represent real world conditions we take into account the signal reflected by the ground (assuming that our mobile nodes operate on flat terrain). The Two-Ray Ground model [52] employed is given in the following equation:

$$P_R = \frac{P_T \cdot G_T \cdot G_R \cdot h_T^2 \cdot h_R^2}{d^4} \tag{3.2}$$

In this equation, h_T and h_R are the height of the transmitter and receiver antennas respectively. According to this model, reception power is proportional to the inverse of the fourth power of the distance between sender and receiver, d.

Our simulations employ a complex model for the radio signal propagation. Friis model is used for distances up to a crossover point, and the two-ray ground model is used for greater distances. The crossover point occurs at the distance where both Friis and Two-Ray Ground models output the same received power. Finally, the radio interface was modeled after the 914 MHz Lucent WaveLan card [64]. Given this card's radio characteristics and the aforementioned propagation models, figure 3.2 shows the minimum transmission power required for reception at a given distance.

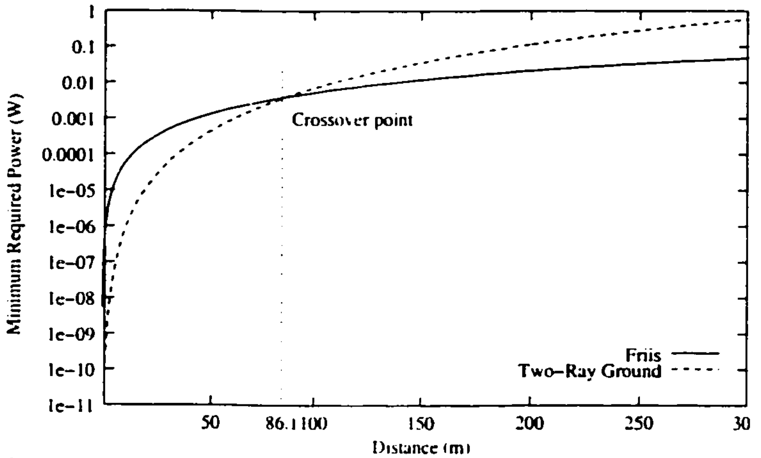


Figure 3.2 Power Requirements predicted by Friis and Two-Ray Ground models

A missing element in the physical layer is support for channel impairments due to blocking, shadowing and short term or long term fading. In our simulations, we assume that signal power attenuates only as per our propagation model and there are no additional errors due to any channel impairments or outside interference. This hypothesis does have a negative impact on the realism of our "virtual, simulation world", however it also subtracts from its complexity. Finally, all radio propagation models are highly empirical and work well for only a given type of environment (e.g. free space, indoor environment, urban metropolis with a certain degree and type of structures, etc). The employed complex propagation model has experimentally been shown to be accurate enough for wireless nodes operating on a large flat area with unobstructed line of sight and lack of outside interference in [53].

3.2.3 IEEE 802.11 based MAC Layer

NS2 offers a complete and accurate implementation of IEEE 802.11 [15] working in DSSS mode. Currently, there is support for operation under the Distributed Coordination Function (DCF) only. In addition all timer constants have been assigned typical values for operation at 2Mbps. Transmission of all unicast data packets, regardless of their size, is preceded by an RTS/CTS exchange, and correct reception is followed by an ACK. IEEE 802.11's RTS/CTS exchange is employed in order to minimize collisions and alleviate the hidden and exposed terminal problems. However, it should be noted that in a multi-hop wireless environment, a successful RTS/CTS exchange does not guarantee collision-free reception at the receiver. A station that has missed the CTS and cannot sense the sender's data transmission may still cause collisions and data packet loss at the receiver. The ACK packet that follows a successful data packet reception attempts to remedy this situation. If the sender does not receive this acknowledgement, it assumes packet loss at the receiver and simply retransmits the lost packet.

The implementation of 802.11 offers support for basic channel capture as well as both physical and virtual carrier sense. Channel capture may occur while a node is in the process of receiving a packet. If another packet arrives during that time, the signal strengths of the two are compared. If the reception power of the first one is 10dB greater than that of the second one, then we have "channel capture" by the first packet and no collision occurs. However, we examine each possibly colliding packet individually, whereas, in real world conditions, each one contributes to the noise floor level. As a result, a single packet may not cause a packet drop, but many, low power packets may. Virtual carrier sense takes place at a node after successful reception of an RTS or a CTS packet. The node then updates its Network Allocation Vector (NAV) according to the information in the control packet and defers channel access. Physical carrier sense on the other hand implies that the node detects a "busy" medium when accessing the wireless channel state because it overhears a transmission from another node.

The choice of IEEE 802.11 as the underlying MAC layer reflects the huge popularity that this standard enjoys as well as current market availability. While there have been numerous wireless MAC protocols proposed that outperform it, only 802.11 has received strong commercial and industrial backing so far. Therefore, most wireless network interface cards on the market today are 802.11 based and this trend is likely going to continue in the future. Bluetooth [10] is poised as a contender to 802.11, especially in the wireless LAN market. However, given its short transmission radius it is not well suited for operating in a MANET environment.

3.2.4 Network Layer

ARP

The ARP module implementation follows closely the one present in FreeBSD 4.3. Its purpose is to map IP addresses to MAC layer ones. ARP is commonly used in hard-wired networks for routing packets within the same IP subnet. Even though we make full use of the TCP/IP suite of protocols in our simulations and assume that it will also be used in real-world implementation of MANETs, we believe that ARP represent a feature of hard-wired networks that we could certainly do away with in MANETs. One could assume that nodes participating in MANETs come preconfigured with both the IP and MAC addresses of all nodes. We modified the original ns2 code so that packet snooping is performed on all packets received by the network interface and thus each node's ARP cache is always up to date about its neighbors. This modification results in no ARP packets ever being transmitted and a very slight increase (in the order of 1 to 3%) in overall throughput and packet delivery ratio. Although the exclusion of the ARP module bears little impact on our results, we feel it makes for better and more efficient MANET design.

DSR

The DSR module used in our simulations is based on the original CMU extensions to ns2 [59]. In particular, its features and cache implementation follows the description presented in [60]. Packets are passed to the routing agent from the overlay application and are "stamped" with the source-route to the destination. If such a route does not exist then the packet is cached and a new route query is sent.

AODV

We did not utilize the version of AODV that came with the original ns2 distribution, but rather the one available from the protocol's principal maintainers. This version was used in [61] and it is highly optimized. Initial comparative results have shown that it offers significant performance improvements over the one supplied with the standard NS2 simulation.

Both routing protocols depend on feedback from the underlying datalink layer in order to detect link failures. 802.11 tries to retransmit a data packet or RTS up to three times. In case it does not succeed, the packet is dropped and the routing layer is informed that the link in question is broken and a path discovery process is initiated.

3.2.5 Topology, Connectivity and Mobility Models

Given the inherent complexity of MANETs, an implementation of a realistic simulation is a difficult proposition on its own. This goal is further compounded by the fact that there are no real-world applications of mobile ad hoc networks yet (with the exception of the U.S. military, perhaps, [71]). Researchers have focused exclusively on solving the fundamental problems of mobile ad hoc networks, rather than working on real-world applications of them. A crucial question that has not found an adequate answer yet concerns the practical use of mobile ad-hoc networks. In the absence of a well-defined operational environment however, one can only attempt to make educated guesses about end-user connectivity and mobility behavior.

Our simulations take place on a flat, rectangular area of 1500 x 500 m^2. The reasons behind our choice of the particular area were twofold. Given our maximum transmission range of 250m, we would like to support multi-hop paths. At the same time, taking into account the computational resources at our disposal, simulating larger topologies was simply not practical. We constructed our connectivity and mobility scenarios based on the *random waypoint model* [38]. At the beginning of the simulation, each node is assigned a random starting position and destination point within our movement area, as well as a random constant speed for the duration of its transition. Nodes have to wait for a pre-specified amount of time, *pause time*, before commencing their transition. Destination points are chosen in a uniform fashion and the mobile's speed is also uniformly distributed between [0, *maxspeed*]. The maximum speed in our experiments was chosen to be 20 m/s. Upon arrival at its destination, the node remains stationary for *pause time* again and is subsequently assigned another random destination and speed. This movement pattern is repeated until the conclusion of the simulation. Figure 3.3 shows the connectivity of the resulting random graph as a function of the transmitter radius. We note that for our transmitter radius, the graph is almost always fully connected.

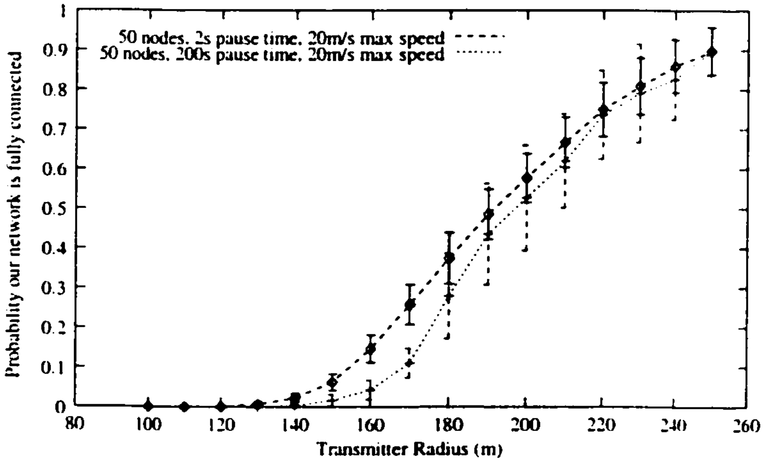


Figure 3.3 Probability our network is fully connected

An artifact of the random waypoint model that we observed is that nodes tend to spend most of their time concentrated in the center region of our topography. Even though the destinations points are uniformly distributed, the transition from one to another usually crosses the center and results in an increased concentration of nodes around the center. In order to test this hypothesis, we analyzed the behavior of our mobility model in two topographies of the same total area, a rectangle of 1500 x 500 m^2 and a square of 866 x 866 m^2. In both cases, we measure the probability a node is in a center rectangle or square that occupies an area of only 25% of the total. The speed of the mobile is uniformly distributed between 0 and 20 m/s. The results of this experiment for various pause times are presented in Figure 3.5

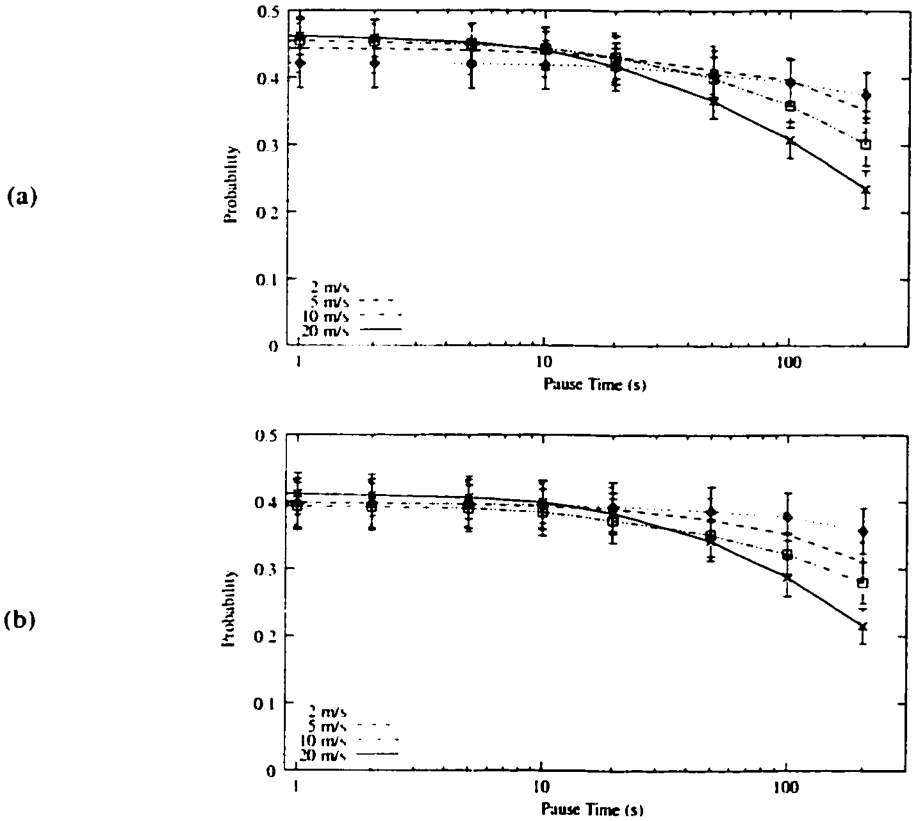


Figure 3.4 Probability a node is in a center region with an area 25% of the total area (a) square topography. (b) rectangle topography.

Since the center area under consideration is only 25% of the total, we would expect a node to show a probability of being in that area roughly around that number. However, the results presented in figure 3.5 show that in cases of high mobility and low pause times, nodes tend to frequent the center of our topography. Figure 3.5 depicts that in both cases a node is present in the center area under consideration with a probability a little over 40% instead of 25%. As the pause times increase, this probability drops and it approaches our expected value. This occurs because at such cases, the high pause times themselves and relatively high speed are more significant than the transition times.

Finally, simulations are run for 900 seconds of simulation time. Constant bit rate connections are initiated during the first 180 seconds and are then kept running for the duration of the simulation.

3.2.6 Simulation Parameters

The following table summarizes the main parameters employed in our simulations:

Simulation Parameters	
Routing Protocols	DSR, AODV
MAC	IEEE 802.11
Interface speed	2 Mbps
Interface type	Lucent WaveLan 914MHz
Transmit Power	218.8 mW
Receive Power	160 mW
Idle Power	150 mW
Traffic sources	10, 20, 30 or 40 CBR sources
Traffic type	UDP
Traffic intensity	4 packets per second
Data packet size	512 Bytes
Number of nodes	50
Speed	0 – 20m/s, uniformly distributed
Movement model	Random Waypoint Model
Pause times	1, 2, 5, 10, 20, 50, 100, 200 and 500 s.
Physical Model	Complex (Friis and Two-Ray Ground)

Nominal Transmitter Range	250 m.
Simulation Time	900 s.
Simulation Runs	10

Table 3.1 Main Simulation Parameters

Each experiment was performed 10 times, each one with different connectivity and movement scenario. The exact size of each packet, including all headers added by the various layers was used when computing the amount of energy expended on transmitting or receiving that packet. Finally, our figures show the average values as well as the 95% confidence intervals of all metrics under consideration.

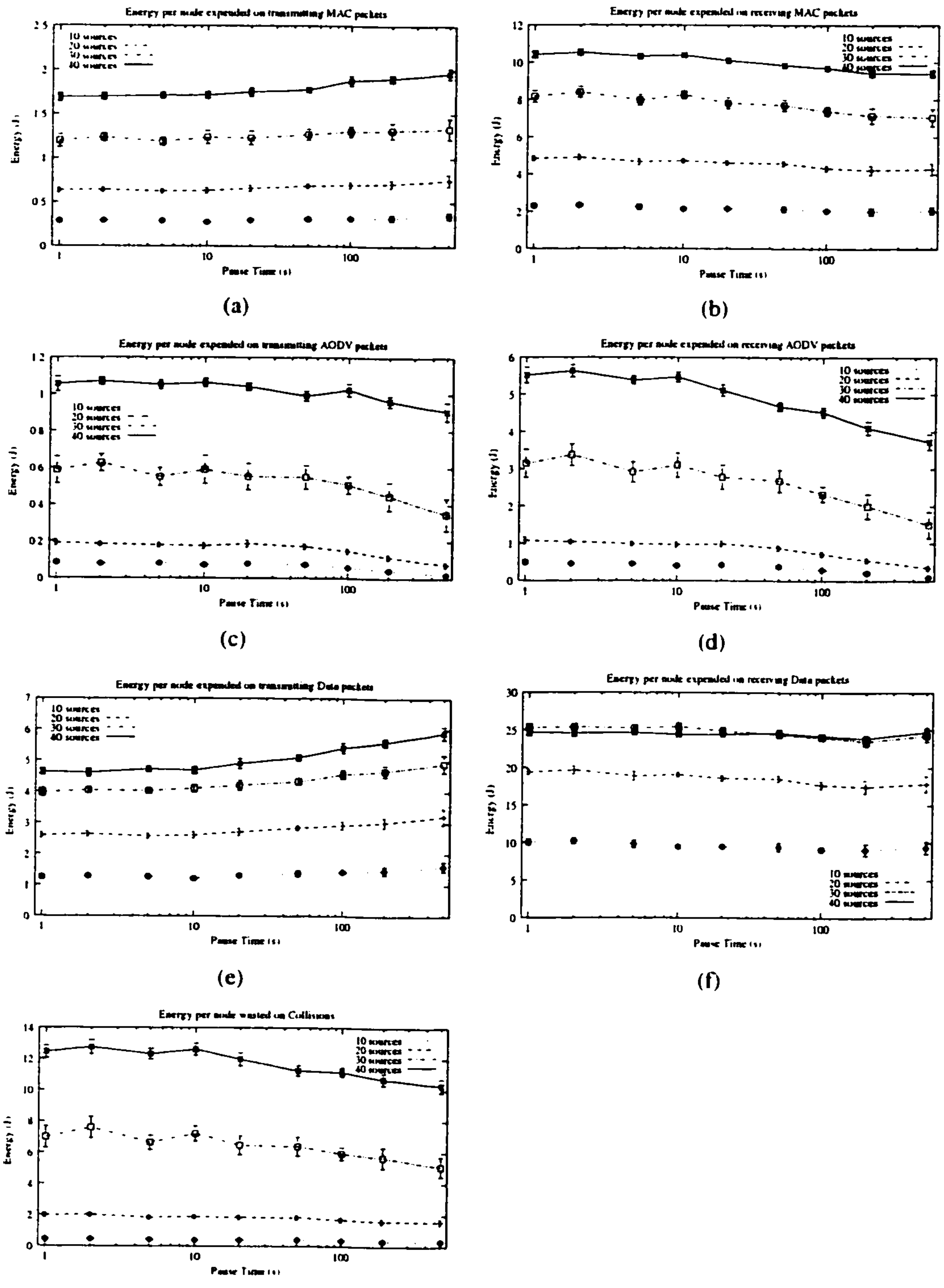
3.3 Energy Breakdown Analysis

We start our analysis by identifying the relative energy consumption of the various network layers. We have studied the energy characteristics of MANETs using both DSR and AODV. Both routing protocols exhibit the same overall behavior and trends. In addition, our findings show that AODV consistently outperformed DSR in our simulations in regard to our energy as well as performance metrics, a point that was also made in [61]. Therefore, we chose to report the results only for AODV. Our observations and conclusions hold true for DSR as well. Finally, the results of this section represent our reference case and form the basis of subsequent comparisons.

There are three types of packets being transmitted: MAC, Routing and Data packets. Figure 3.5 shows the energy per node in Joules that was expended for receiving or transmitting packets. A first observation that can be drawn from it is that even though packet transmission is almost 1.8 times more energy expensive than receiving a packet, significantly more energy is spend on receiving packets. In some cases, we spend up to ten times more energy receiving packets than transmitting. The reason for this behavior lies in the fact that all nodes within the transmitter's radius will hear and process its transmission, even though not all of them actually need to receive its packet. This problem becomes more severe in densely populated network topologies.

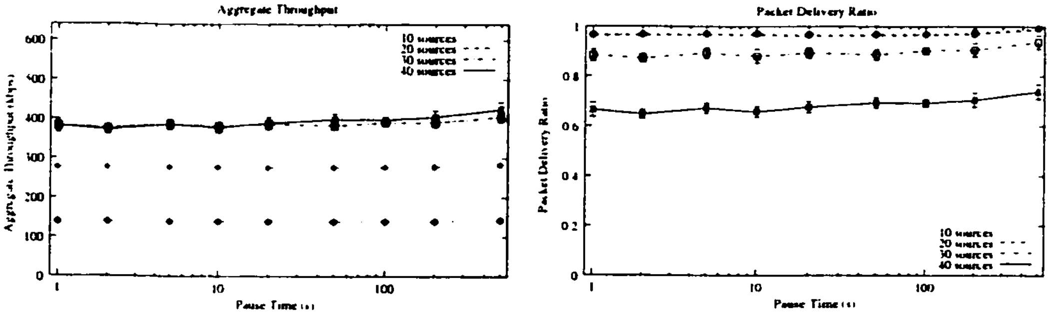
A second observation stemming from figure 3.5 is that energy expended on MAC packets exceeds that of Routing Packets. The effect of node mobility can be seen on figures 3.5.c, 3.5.a, and 3.5.e. The first one shows that as pause times become larger and thus link breakages occur less often, the energy expended on routing packets decreases. Furthermore, since we have longer lived paths, we are able to successfully deliver more data packets, and thus we notice an increase in the energy expended for MAC and data packets at high pause times in figures 3.5.c and e. When mobility increases however, so does the energy required for routing, as new route requests need to be flooded through the network more often. This affects negatively our throughput and packet delivery ratio, figure 3.5, because when a path breaks, packets already in transit get dropped and new packets have to wait for a new path to be found. In addition, figure 3.5.g shows that increased node mobility causes more energy to be expended on collisions, as opposed to low mobility. The increased routing activity is also to blame for this occurrence. New route requests are flooded through the network and they are broadcast packets. As such, the MAC layer transmits them at the first chance it gets, whenever it senses the medium to be free and makes no effort to mitigate the "hidden" terminal problem by employing RTS/CTS mechanism. As a result, collisions may occur at all neighboring nodes that happen to be in the process of receiving a packet.

Finally, figure 3.6.a and 3.6.b show the aggregate throughput and packet delivery ratio measured in our simulation. It is important to note that our network appears to be saturated at 30 sources. The addition of 10 more sources does not offer any significant change to overall throughput, which practically remains the same and at the same time, the packet delivery ration drops drastically. Even though the nominal capacity of our network is 2Mbps, it appears that our network cannot sustain more than 13Kbps per traffic source approximately.



(g)

Figure 3.5 Energy per node expended on (a) receiving MAC packets. (b) transmitting MAC packets. (c) receiving AODV packets. (d) transmitting AODV packets. (e) receiving Data packets. (f) transmitting Data packets. (g) Collisions



(a) (b)

Figure 3.6 (a) Aggregate Network Throughput. (b) Packet Delivery Ratio

3.4 A Simple Power Management Approach

3.4.1 Introduction

It was shown in section 3.2.2 that transmission power attenuates in proportion to the squared distance between sending and receiving nodes, when the Friis model holds. At far distances, power attenuates at an even greater pace, in proportion to the fourth power of the distance, equation 3.2. However, we always transmit with maximum power, regardless of the actual distance between sender and receiver. This represents an energy-expensive practice especially for nodes that happen to be close to each other. Furthermore, it is more costly for a node to transmit a packet than to receive one, always speaking from an energy point of view. Table 3.1 shows that the transmit to receive energy ratio for our particular WaveLan card is 1.76. [5, 7] report similar or larger ratios for different types or wireless network interface cards. Therefore, we focus our efforts towards minimizing the power required to transmit packets. This is accomplished by predicting the minimum power required for successful communication between the wireless nodes. We expect that this approach will also have a positive effect on the amount of energy spent overhearing packets, since fewer nodes now may be able to hear our data transmissions. A final motivating observation has to do with the prevalence of IEEE 802.11 as the only viable commercial MAC offering at the time of this writing. While it may not be the soundest choice for multihop ad hoc networks, its definitive domination of the cellular wireless market seriously hampers the development and adoption of alternative, competing MAC protocols for the non-existent yet MANET market.

Our approach comes in a stark contrast to the one we followed in the previous chapter about cellular wireless networks. We chose to operate between the MAC and physical layer for a variety of reasons. We can easily discard for the time being power management solutions at the application layer. There is currently no defined application framework for MANETs and although interest in them is peak, real-world applications of them have yet to materialize. Transport layer approaches, such our selective idling one appear no less problematic. TCP has been shown to behave poorly in wireless multihop environments in [24, 25, 65, 73]. Our own results in section 3.3 stipulate to these findings. Given a packet delivery ratio of 70% (figure 3.6.b), or conversely a loss ratio of 30%, we can safely conclude that had we employed TCP in our reference case, its overall performance would have been abysmal. This is the main reason why simulations of mobile ad hoc networks employ UDP instead of TCP. Furthermore, since MANETs assume a complete lack of supporting infrastructure, we cannot readily employ our TCP-based scheme, which employs the services of an "overseer" base station to assist in energy conservation. Routing layer solutions [33, 40, 54, 60] are certainly poised to play an important role in power management. However, we chose to work at an even lower layer and present an approach that is orthogonal to the choice of routing protocol. It is not meant to be a complete solution to the problem of energy conservation in MANETs but a first step towards this goal.

3.4.2 Power Management

Our approach capitalizes on the RTS/CTS exchange of IEEE 802.11. Since all unicast transmissions are preceded by the aforementioned exchange, we support that we can use it in order to facilitate our energy conservation scheme. Each successfully received CTS is in essence a power measurement and an indication of the distance between sender and receiver. Therefore, the transmission of unicast packets (data and unicast routing packets) that follow the CTS can be performed based on that power estimate. The RTS and CTS packets are transmitted at full power. This is a necessary requirement for the 802.11's virtual carrier sense scheme to work. Otherwise, we run the risk of increased collisions and congestion due to the hidden terminal problem. Finally, we expect that overall throughput will roughly be the same with the case without power management. Since we transmit the RTS and CTS packets at full power, we do not allow for spatial frequency reuse.

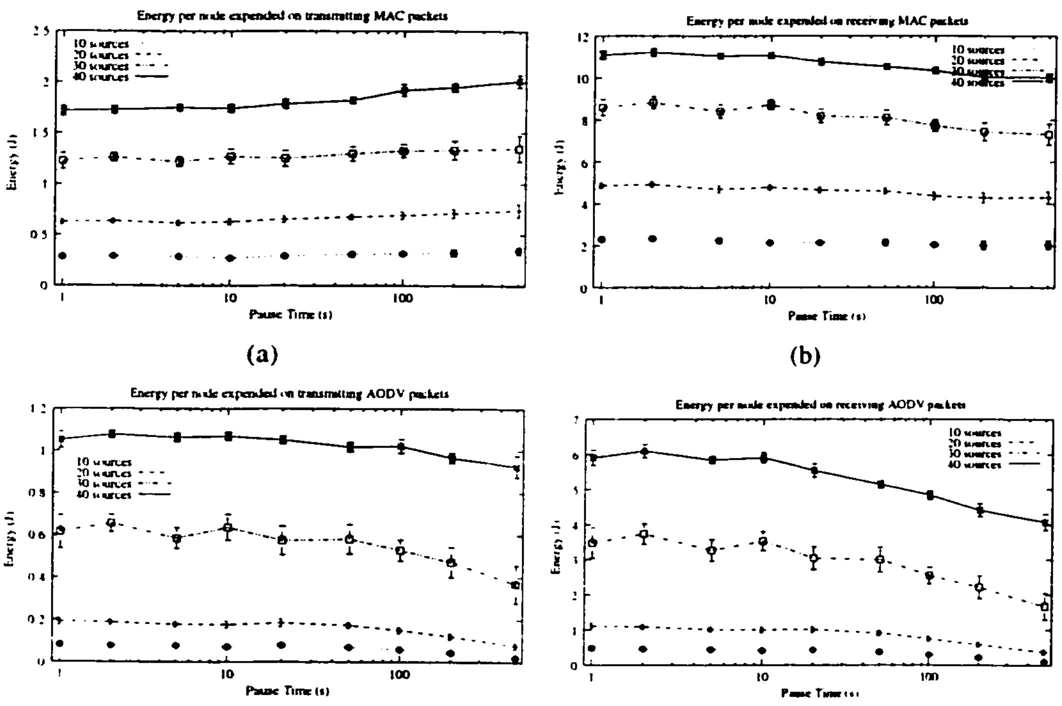
Our power management scheme is based on the hypothesis that we can predict the amount of power required to reach a neighboring node. Therefore, an accurate radio propagation model is essential. However, a requirement for a radio model that is 100% accurate is unrealistic. Radio propagation models are often empirical and depend on a variety of factors, which are often ill defined [52]. We assume that the radio transceiver is able to measure the power level at which an incoming packet was received. Following that measurement, we employ the complex radio propagation model, as described in section 3.2.2 in order to predict the distance between sender and receiver and subsequently the minimum power required to reach our next hop destination. The nominal distance, d_{nom}, is calculated using the following equation:

$$d_{nom} = \begin{cases} \left(\dfrac{0.2818 \cdot G_T \cdot G_R \cdot \lambda^2}{16\pi^2 \cdot P_R} \right)^{1/2} & \text{if } P_R > 0.003795\,\mathrm{W} \\ \left(\dfrac{0.2818 \cdot G_T \cdot G_R \cdot h_T{}^2 \cdot h_R{}^2}{P_R} \right)^{1/4} & \text{otherwise} \end{cases} \tag{3.3}$$

We then employ equations 3.1 or 3.2 in order to estimate the required power. The value of P_R is set to the reception threshold of our particular WaveLan card (3.752 10^{-10} W). Furthermore, in order to compensate for our model's inherent inaccuracy and account for node mobility, we added a fixed 10% cap to all our power estimates. The complex model has been found accurate enough when conducting experiments on flat terrain with clear line-of-sight [62]. A similar approach has been proposed in [33]. However, the authors propose a new routing protocol that breaks paths into ones comprised of more hops that are close to each other. We prefer shorter paths and choose not to further exacerbate the already poor performance of 802.11 in a multihop environment [65]. Furthermore, is spite of our transmitting the RTS and CTS packets at full power, our transmitting the actual data packets at reduced power may impede on the physical carrier sense performed by 802.11. It is possible that a node that missed the RTS/CTS exchange, and therefore has not updated its NAV timer, will try to transmit during our data packet transmission and cause a collision at the sender. This would not have been the case had we transmitted our data packet at full power, since physical carrier sense would have prevented the node in question from transmitting. Finally, another potential problem is possible interference with the route-shortening algorithm of the routing protocol. Both DSR and AODV incorporate a mechanism that allows a route to be shortened by one hop. For example, if our path takes us through nodes A, B and C and C can listen to the transmission occurring from a to B, then C man inform A to send directly to it. Current protocol implementations however examine only the data packets exchanged between hosts and since we now transmit with less power, the algorithm may not be triggered. However, this situation may be remedied by examining the MAC-layer packets as well.

3.4.3 Power Management Simulation Results

The results of our power management scheme are presented in figures 3.7 and 3.8. Comparing figures 3.8 and 3.5 we can see that the performance characteristics of our network have not been affected by our power management scheme. Figure 3.8 shows only a very slight decrease in the aggregate throughput and packet delivery ratio occurs at high loads of 30 or 40 nodes, in the order of 4%. Such performance penalties though are easily offset by the energy gain margins that we were able to achieve. Figure 3.8.c shows an almost threefold decrease in the amount of energy per node expended on transmitting data packets, when compared to the previous figure 3.5.c. MAC and routing layer packets are not affected by our scheme at all. This is contrasted between figures 3.7.a,b,c,d and 3.5.a,b,c,d, which show the energy spend on transmitting or receiving these type of packets. In addition to the decrease in the energy expended for transmitting packets, our power management scheme causes a decrease in the amount of energy expended in receiving packets as well as the energy expended on collisions. Figures 3.7.g and 3.5.g show that the amount of energy spent on collisions is decreased approximately by 35%. A decrease of similar magnitude is evident in figures 3.7.f and 3.5.f, which depict the amount of energy spent on receiving data packets. Such a decrease was expected and it owes itself to the fact the fewer stations in the senders neighborhood can now overhear its unicast transmissions.

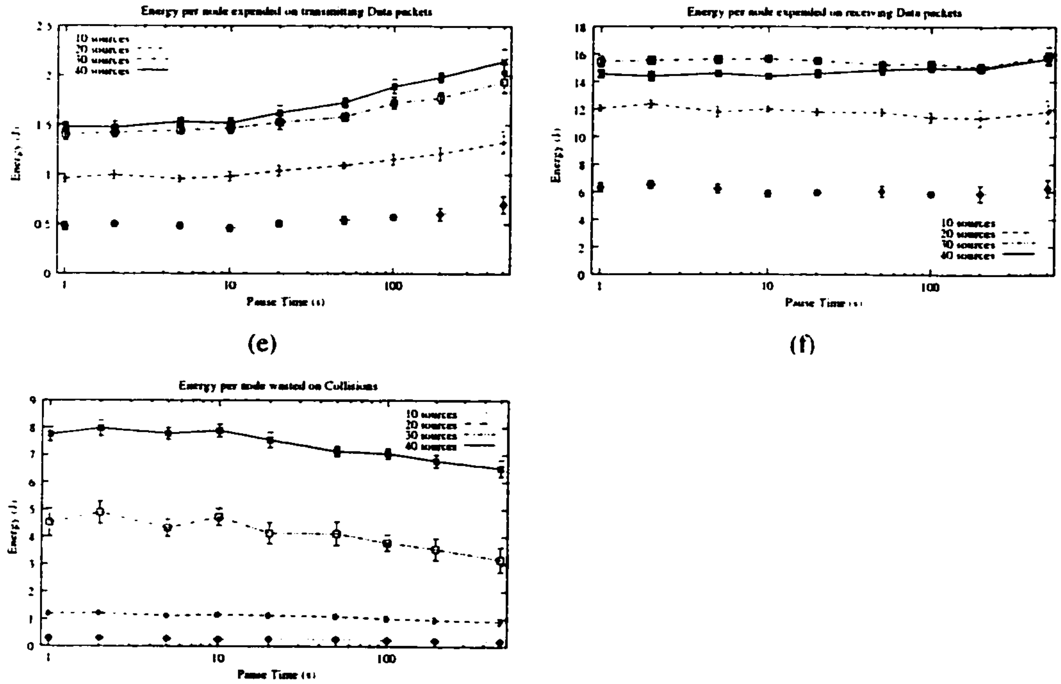


(a)

(b)

(c)

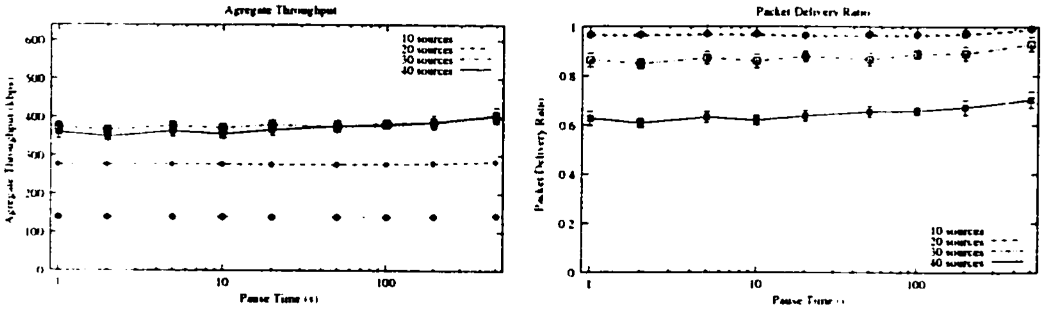
(d)



(e) (f)

(g)

Figure 3.7 Power Management: Energy per node expended on (a) receiving MAC packets. (b) transmitting MAC packets. (c) receiving AODV packets. (d) transmitting AODV packets. (e) receiving Data packets. (f) transmitting Data packets. (g) Collisions



(a) (b)

Figure 3.8 Power Management: (a) Aggregate Network Throughput. (b) Packet Delivery Ratio

3.5 Effect of Extended Carrier Sense

3.5.1 Introduction

A parameter that is usually not taken into consideration in MANET simulations is the fact that radios can sense a busy channel at a much lower power threshold than the one needed for successful packet reception. Radio receivers usually exhibit an increased sensitivity as far as carrier sensing is concerned. However, this behavior is usually ignored for the sake of simplicity and an equal reception and carrier-sensing radius is often assumed. It should be noted that other simulation packages, such as GloMoSim [53], employ no such mechanism but rather a single range for reception and carrier sense. In this chapter we will examine the behavior of MANETs when sensitive receivers are employed. Therefore, in our simulations wireless nodes can now sense a "busy" channel for a much larger radius than their actual reception range [65]. The reported values for our modeled WaveLan [64] cards are 3.753 10^{-10} W for correct packet reception and 1.559 10^{-11} W for the carrier sense threshold. Given that our threshold for channel capture is 10dB and assuming the Two-Ray Ground radio propagation model, the following figure shows our nominal transmission, interfering and extended carrier sense range:

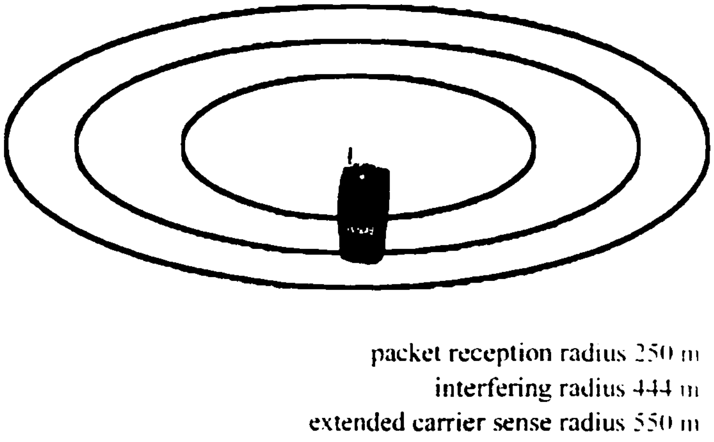


Figure 3.9 Packet reception, interfering and extended carrier sense radius for our modeled WaveLan radio cards

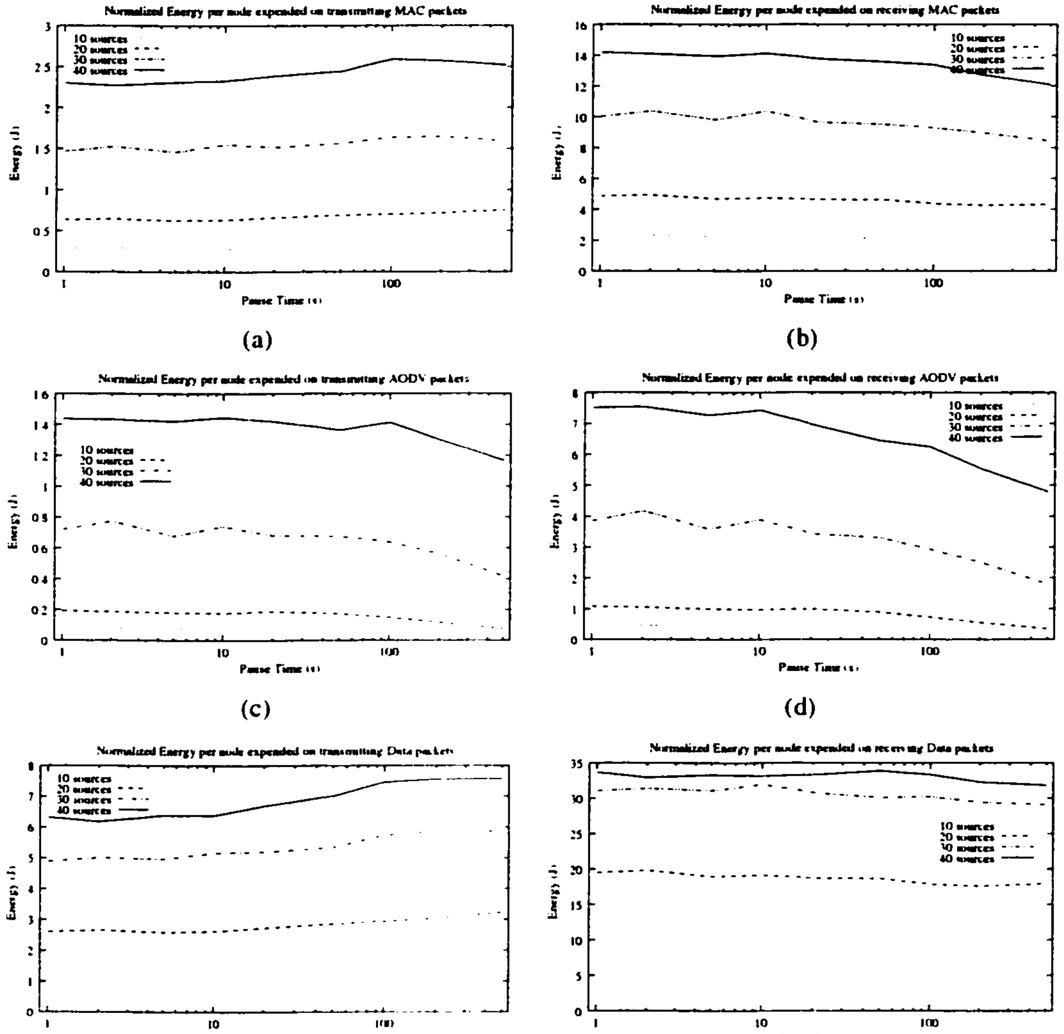
In order for a packet to be received correctly, the distance between sender and receiver has to be less or equal to the packet reception radius (250m). Nodes within the interfering and packet transmission radii are unable to successfully exchange packets, but may very well cause collisions. Such collisions may occur if the interfering transmission is 10dB (our card's capture threshold) or less weaker than the one being received at the receiver. The range of the interfering radius (444m) was computed for a power threshold of 10dB lower than our nominal packet reception threshold. Finally, transmissions taking place within the interfering and extended carrier sense radii cannot cause any collisions at all at the receiver. However, they are still strong enough for the wireless node to detect a busy channel and thus defer any transmissions.

3.5.2 Simulation Results

The negative effect in network performance of having a much larger carrier sense than reception radius can readily be seen in figure 3.11. There is a noticeable 27% decrease of the overall throughput, figure 3.11.a, when we compare it to our original simulations, figure 3.6.a. This decrease manifests itself in the cases of 30 and 40 sources, as the network reaches its saturation point faster. Similar observations can be made about the packet delivery ratio, figure 3.10.b. In the case of 40 sources in particular, it falls down to less than 50%, a 30% decrease when compared to figure 3.6.b.

Given the drop in the performance-related metrics we expect a similar behavior for our energy related ones. In order to facilitate proper comparison with our simulation results presented in sections 3.3 and 3.5, we had to normalize our energy consumption metrics. Figure 3.10 shows the energy expended by the different network layers had we transferred the same amount of packets as our reference simulations in section 3.3. Energy consumption in the case of 10 and 20 sources in figure 3.10 is almost identical to the one in figure 3.6, across all our energy metrics. This is expected, since the overall network performance in figures 3.10 and 3.6 is very similar. However, this is not the case for 30 and 40 sources.

We can readily observe an increase in the order of 40% approximately for all our energy metrics in figure 3.10 in comparison to figure 3.6. Clearly, extended carrier sense affects our simulations in a very negative way. It is responsible for increased collisions, network saturation and greater energy consumption. These results point to the inefficiency of IEEE 802.11 when operating in a multihop wireless environment. In this case, the simple RTS/CTS exchange mechanism cannot handle very well the exposed and hidden terminal problem resulting in serious degeneration of network performance.



(a)

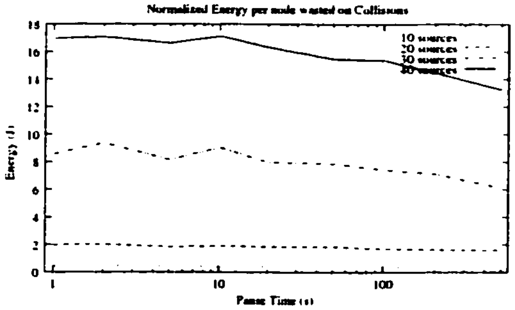
(b)

(c)

(d)

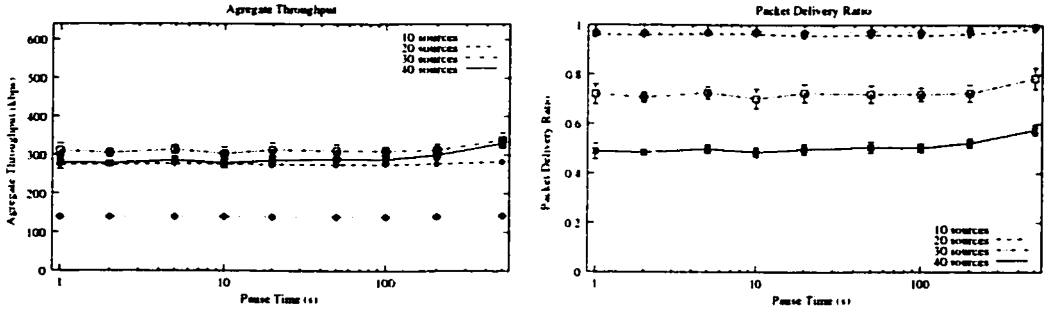
(e)

(f)



(g)

Figure 3.10 Extended Carrier Sense: Normalized Energy per node expended on (a) receiving MAC packets. (b) transmitting MAC packets. (c) receiving AODV packets. (d) transmitting AODV packets. (e receiving Data packets. (f) transmitting Data packets. (g) Collisions



(a) (b)

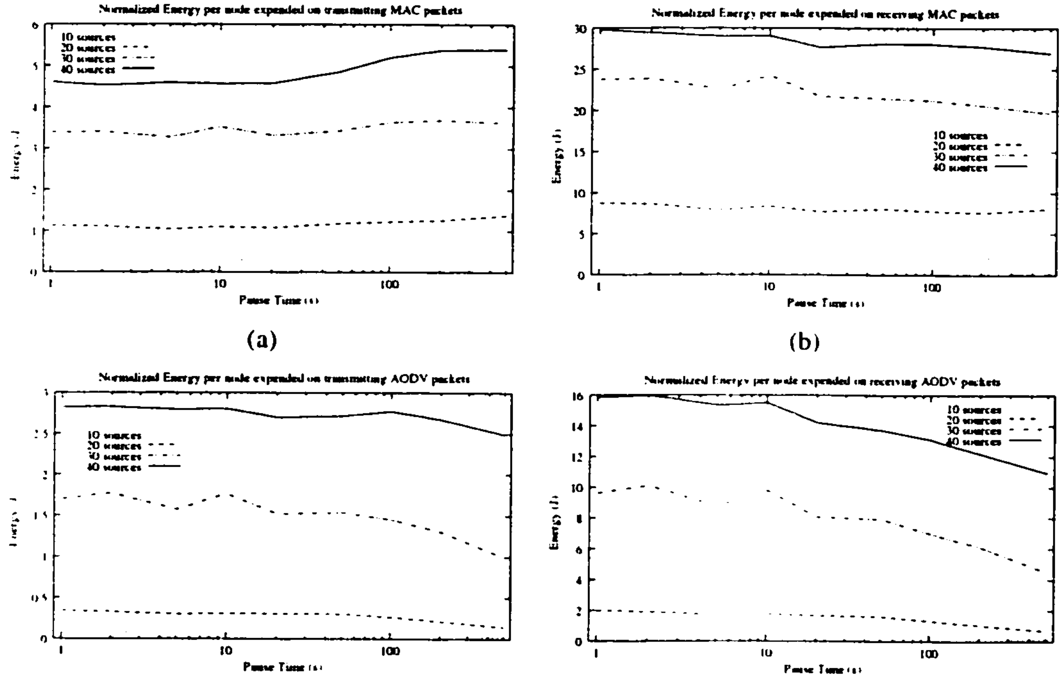
Figure 3.11 Extended Carrier Sense (a) Aggregate Network Throughput. (b) Packet Delivery Ratio

3.5.3 Effect of Extended Carrier Sense on Power Management

The previous section brought the effect of extended carrier sense to light and accentuated the inefficiency of 802.11 is such an environment. We now turn our attention to the effect of extended carrier sense on our power management scheme. Figure 3.13 shows that the aggregate network throughput, as well as the packet delivery ratio decrease by a wide margin. Only the results for 10 sources are similar to those of our reference case (section 3.3). When using 20, 30, or 40 stations, we experience so many collisions that our network becomes practically unusable. The maximum aggregate load it is able to sustain is only about 160

Kbps. Such low network performance takes a dire toll on total energy consumption. Figure 3.12 shows clearly that the normalized energy consumption of each one of the various sub layers is almost twice that of our reference. The only exception to the above is the total energy expended for transmitting data packets, figure 3.12.e, however, any slight energy gains in that case are readily offset by the losses in the rest of the cases.

Given our analysis of extended carrier sense in section 3.5.1, the results of this section come to little surprise. Figure 3.9 shows that the interfering range of our nodes is now extended to 444m, a 75% increase of our previous range of 250m. Since we are transmitting our data packets at almost the minimum required energy (plus 10%), simple RTS/CTS/ACK packets or any broadcast packets that are sent at full power from nodes up to 444m away have the a high probability of causing a collision.

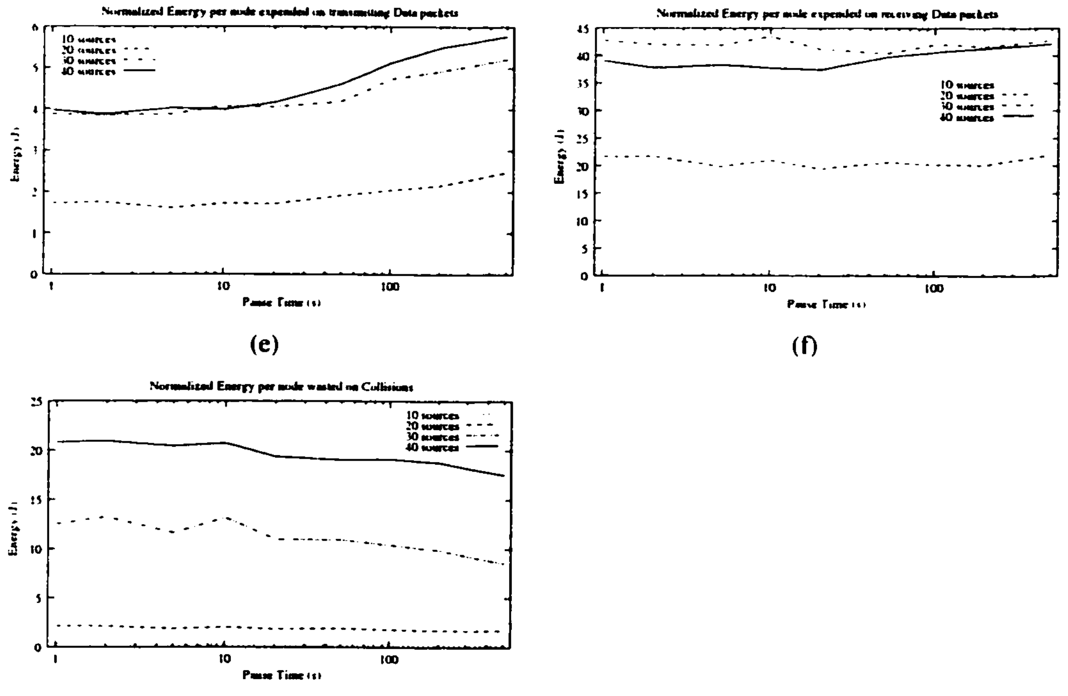


(a)

(b)

(c)

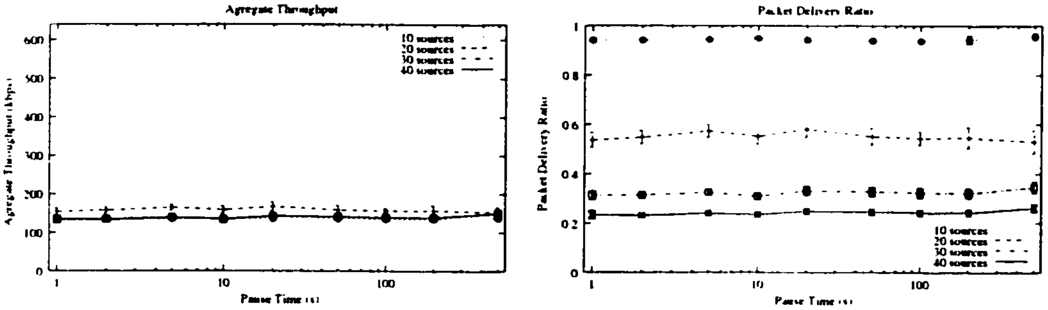
(d)



(e) (f)

(g)

Figure 3.12 Extended Carrier Sense and Power Management: Normalized Energy per node expended on (a) receiving MAC packets. (b) transmitting MAC packets. (c) receiving AODV packets. (d) transmitting AODV packets. (e receiving Data packets. (f) transmitting Data packets. (g) Collisions



(a) (b)

Figure 3.13 Extended Carrier Sense and Power Management: (a) Aggregate Network Throughput. (b) Packet Delivery Ratio

3.6 Conclusions

This chapter examined the energy characteristics of MANETs and proposed a simple power management approach to them. Our approach relies on the underlying radio propagation model for predicting the least power required to reach our destination node. We utilize the RTS/CTS packet exchange of IEEE 802.11 in order to obtain the necessary power measurements. Our simulations show that our simple power management technique manages to expend almost three times less the energy needed for unicast transmissions than our base reference case. Furthermore, since we transmit with less power, less nodes can hear our transmissions and additional energy gains can be observed in the amount of energy spent on overhearing packets. However, the greatest advantage of our approach is its orthogonality to the choice of a MANET routing protocol. Therefore, it works on the datalink layer alone, it can readily be applied along any routing layer solutions.

An important observation made in the analysis of the energy characteristics of MANETs is that the most energy in our simulations is spent on overhearing the transmissions of other nodes. Our power management approach can only help as much; further gains can only be realized if nodes idle themselves when they not be active. We are currently investigating such schemes that will direct inactive nodes to sleep and conserve energy. An example of the schemes under consideration is the ad-hoc power management functions of 802.11 [15]. Furthermore, the importance of routing solutions to power management should not be underestimated. Noteworthy proposals in static ad-hoc wireless networks [23, 36], employ energy -aware metrics and propose a class of algorithms that find the most energy -efficient routes or those that maximize the lifetime of the network. However, the effect of mobility on such solutions is the subject of future investigation.

Finally, we examined the detrimental effect of using sensitive receivers in mobile ad-hoc networks. Our main conclusion is that such behavior seriously impedes the mechanics of 802.11 and results in very poor throughput and increased energy consumption.

4. Final Thoughts

Questions about the importance of power management in mobile computing devices are rhetorical. It is simply essential that such techniques are developed and put to place. Our work examined the energy conservation problem from the perspective of the wireless network interface in two different operating environments, in cellular and in mobile ad hoc networks. This section will provide some concluding thoughts on the problems left still un-addressed and try to offer some insight for future work.

- In cellular wireless networks, we studied a transport layer approach that is based on TCP. Our initial goal was for a pure end-to-end solution that would not violate the layering principle and thus be readily applicable to real-world wireless applications. However, our findings provide evidence to the contrary, close cooperation among different layer entities is essential in order to be maximally efficient and energy conserving. To this extend, we had to utilize information gained at lower layers (and thus normally hidden away from TCP) at the mobile host, as well as employ the services of a closely cooperating base-station. It is important to note that we had to add a significant amount of intelligence pertinent to the inner-workings of TCP to the base station in order to successfully adapt to cases of high network congestion. A continuation of this trend brings the following question to the front: "Do we really need to use TCP when energy conservation is our primary concern?"

- The advent of WAP [75] may have a catalytic effect on future wireless devices. WAP is an industry developed set of standards aimed at enhancing the user-experience of a wireless device by providing fast, easy, reliable, efficient and secure network services to it. Such services are considered key to the future growth of m-commerce and Internet use on wireless devices. WAP operates above the MAC layer (or bearer services in WAP lingo) and replaces the traditional, Internet based, transport, session, presentation and application layers with ones engineered specifically for operation in a wireless environment. This represents a radical departure from the way of thinking of traditional approaches, which try to adapt the protocols used in the hard-wired Internet to the wireless environment. However, in spite of its strong industrial backing, WAP does not yet enjoy the wide adoption and success its proponents originally hoped for.

- However, regardless of opting for a solution based on Internet-based standards and protocols or choosing a completely new approach, such as WAP, it is evident that only through close synergy among the different layer entities can we achieve significant gains in energy. The "layering principle" dictates that protocols should not have functions that span across all layers. This approach provides an abstraction of the implementation details of lower details to the upper ones and makes for a modular instead of a monolithic design. Therefore, one could in principle take an implementation of the TCP stack and readily apply it to a wireless device. Such ease of use, simplicity and modularity are very powerful advantages in system design and proposals that violated the "layering principle" have often been shunned, [16]. However, we have come to the conclusion that isolated approaches that work only on one layer and are totally agnostic of the operation details of the lower layers cannot provide for a high degree of efficiency and energy conservation. In particular, the higher the layer a solution is positioned in, the more it needs to be aware of the inner workings of lower layers. Our approach for example, needs to peek into the datalink layer in order to determine the actual transmission times of packets. Therefore, it becomes evident that systems designed with energy -conservation in mind need to "bend" the layering rule and gather intelligence from all available layers.

- Mobile Ad hoc Networks are designed to operate in the absence of any supporting infrastructure. They represent a unique communication environment where each wireless node may act as a router and forward other nodes' traffic. Our power management solution in MANETs operates at a very low layer and tries to predict the least amount of power needed for reliable unicast transmissions. Besides its energy gains, a significant advantage it offers is its orthogonality to the choice of routing protocol. While we have employed only DSR and AODV, we expect similar gains for other routing protocols as well. A drawback of our approach is its reliance on an accurate radio propagation model. Nevertheless, there is no such model that works sufficiently well across all possible types of operating environments. In order for our proposal to be applied in real-world conditions, additional methods are needed whereby a wireless node tries to adapt the radio propagation model to real-world conditions.

- An additional insight for future research comes from the observation that nodes expend a lot of energy overhearing transmissions not intended for them. It would be highly desirable if nodes could power themselves down when they do not partake in any active communication. However, policies for idling nodes in a MANET are not trivial and may hide the potential of causing more harm in the long run than good. For example, since nodes are unaware of the topology of the network, such actions may result in network partition. In addition, a node cannot predict when another may wish to contact it, and may have to rely on its neighbors to provide temporary buffering. Therefore, a distributed solution for power management that allows for node coordination appears to be essential. To this end, we are currently investigating the energy saving potential of IEEE 802.11 with the use of ATIMs in a MANET environment.

- Finally, one of the greatest challenges left for MANETs is inventing applications that justify their cost and deployment. There is yet no clear and well defined operational environment for MANETs and most presented results are based on simulations alone or rudimentary experiments comprised of a handful of nodes. The lack of real-world experiments is an issue that needs to be addressed and such results will probably provide helpful insight to practical problems that are often ignored in simulations. Use of very sensitive receivers, as shown in section 3.5 is one such example, which may seriously impede on proper network operation. Simulations can only provide so much information. A real MANET testbed is necessary for studying the complex interactions of the various network layers and node mobility in detail.

References

1. P. Agrawal. Energy efficient protocols for wireless systems. In *Proceedings of the IEEE 9th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '98)*, pp. 564-569, Boston, Mass., September, 1998.
2. B. Rhodes, N. Minar, and J. Weaver. Wearable computing meets ubiquitous computing: Reaping the best of both worlds. In *Proceedings of The Third International Symposium on Wearable Computers (ISWC '99)*, pp. 141-149, San Francisco, CA, October 1999.
3. I. Chlamtac, C. Petrioli, and J. Redi. Energy-conserving selective repeat ARQ protocols for wireless data networks. In *Proceedings of the IEEE 9th International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC ' 98)*, pp. 836-840, Boston, MA, September 1998.
4. G. Hadjiyannis, A. Chandrakasan, and S. Devadas. A low power, low bandwidth protocol for remote wireless terminals. *Wireless Networks*, 4(1): 3-15, 1998.
5. R. Kravets, and P. Krishnan. Power Management techniques for mobile communication. In *Proceedings of MOBICOM '98*, pp. 157-168, Dallas, Texas, October 1998.
6. NS (version 2), UCB/LBNL/VINT Network Simulator. http://www-mash.cs.berkeley.edu/ns/, 2000.
7. M. Stemm, P. Gauthier, D. Harada, and R. Katz. Reducing power consumption of network interfaces in hand-held devices. In *Proceedings of the 3rd International Workshop on Mobile Multimedia Communications (MoMuc-3)*, pp. 130-132, Princeton, N.J., September 1996.
8. V. Tsaoussidis, H. Badr, and R. Verma. Wave & wait: an energy saving transport protocol for mobile IP-devices. In *Proceedings of IEEE ICNP '99*, pp. 301-308, Toronto, Ont., Canada, October 1999.

9. N. Bambos. Toward power-sensitive network architectures in wireless communications: concepts, issues, and desing aspects. *IEEE Personal Communications*, vol.5, no.3, pp. 50-59, June 1998.

10. P. Lettieri, C. Fragouli, M.B. Srivastava. Low power error control for wireless links. In *Procceedings of the Third ACM/IEEE International Conference on Mobile Computing and Networking 1997 (MobiCom '97)*, pp. 26-30, Budapest, Hungary, September 1997.

11. M. Zorzi, R.R. Rao. Energy-constrained error control for wireless channels. *IEEE Personal Communications*, vol. 4, (no.6), pp. 27-33, December 1997.

12. A. Chockalingam, Xu Weiping, M. Zoorzi. Energy efficiency analysis of a multichannel wireless access protocol. In *Proceedings of the IEEE 9th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '98)*, pp. 1096-1100, Boston, Mass., September, 1998.

13. H. Woesner, J.-P. Ebert, M. Schlager, A. Wolisz. Power-saving mechanisms in emerging standards for wireless LANs: the MAC level perspective. *IEEE Personal Communications*, pp. 40-48, vol.5, (no.3), June 1998.

14. S. Singh, M. Woo, C.S. Raghavendra. Power-aware routing in mobile ad hoc networks. In *Proceedings of MOBICOM '98*, pp. 181-190, Dallas, Texas, October 1998.

15. IEEE. Wireless LAN Medium Access Control and Physical Layer Specifications. Standard IEEE 802.11, 1997.

16. H. Balakrishnan, S. Seshan, E. Amir, R. H. Katz. Improving TCP/IP Performance over Wireless Networks, In *Proc. 1st ACM Conf. on Mobile Computing and Networking*, pp. 2-11. November 1995.

17. K.M. Sivalingam, M.B. Srivastava, P. Agrawal. Low power link and access protocols for wireless multimedia networks. In *Proceedings IEEE Vehicular Technology Conference*, Phoenix, AZ, pp. 1331-1335, May 1997.

18. Jyh-Chang Chen, Krishna Sivalingam, Prathima Agrawal, and Shalinee Kishore. A comparison of MAC protocols for wireless local networks based on battery power consumption. In *Proceedings of IEEE INFOCOM 98*, pp. 150-157, 1998.
19. The Bluetooth Consortium, 2000, http://www.bluetooth.com.
20. Intel. Pentium III Processor Mobile Module: Mobile Module Connector 2 (MMC-2) Featuring Intel SpeedStep[tm] Technology, 2000.
21. OnNow and ACPI: Introduction and Specifications", http://www.microsoft.com/hwdev/onnow.htm
22. R. Castaneda and S.R. Das. Query localization techniques for on-demand routing protocols in ad hoc networks. In *Proceedings of Annual ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM)*, pp. 186-194, August 1999.
23. J. Chang and L. Tassiulas. Energy conserving routing in wireless ad hoc networks. In *IEEE INFOCOM*, pp. 22-31, March 2000.
24. M. Gerla, K. Tang, R. Bagrodia. TCP performance in wireless multi-hop networks. In *Proceedings of IEEE Workshop on Mobile Computing systems and Applications (WMCSA)*, pp.41-50, February 1999.
25. G. Holland, N. H. Vaidya. Analysis of TCP performance over mobile ad hoc networks. In *Proceedings of International Conference on Mobile Computing and Networking (MOBICOM)*, pp. 195-206, August 1999.
26. IETF Mobile Ad Hoc Networks (MANET) working Group, http://www.ietf.org/html.charters/manet-charter.html.
27. D. Johnson, D. A. Maltz and J. Broch. The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (Internet-Draft). *Mobile Ad-hoc Network (MANET) Working Group*, IETF, Mar. 2001.
28. C. E. Perkins, E. M. Royer, S. R. Das. Ad hoc On-Demand Distance Vector (AODV) Routing. *Mobile Ad-hoc Network (MANET) Working Group*, IETF, Mar. 2001.

29. S.-J. Lee, W. Su, J. Hsu, M. Gerla, R. Bagrodia. A performance comparison study of ad hoc wireless multicast protocols. In *Proceedings of IEEE INFOCOM*, pp. 565-574, March 2000.

30. S. Ni, Y. Tseng, Y. Chen, J. Chen. The broadcast storm problem in mobile ad hoc networks. In *Annual ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM)*, pp. 151-162, August 1999.

31. S. Singh, C. Raghavendra. PAMAS-power aware multi-access protocol with signaling for ad hoc networks. *ACM Computer Communications Review*, 28(3): 5-26, 1998.

32. C.-K. Toh. Associativity based routing for ad hoc mobile networks. *Wireless Personal Communications*, Vol. 4, no. 2, March 1997.

33. J. Gomez, A. T. Campbell, M. Naghshineh and C. Bisdikian. PARO: Power-Aware Routing Optimization for Wireless ad hoc Networks. In *IEEE 9th International Conference on Network Protocols (ICNP'01)*, Riverside, California, November 2001.

34. T. Simunic, L. Benini and G. De Micheli. Dynamic Power Management for Portable Systems. In *The 6th International Conference on Mobile Computing and Networking*, pp. 22-32, 2000.

35. J. Li, C. Blake, D. S. J. De Couto, H. I. Lee, R. Morris. Capacity of Ad Hoc wireless networks. In *The 7th International Conference on Mobile Computing and Networking*, pp. 61-69, 2001.

36. Q. Li, J. Aslam, D. Rus. Online power-aware routing in wireless Ad-hoc networks. In *The 7th International Conference on Mobile Computing and Networking*, pp. 97-107, 2001.

37. R. Morris, J. Jannotti, F. Kaashoek, J. Li, and D. S. J. De Couto. CarNet: A scalable ad hoc wireless network system. In *Proceedings of the 9th ACM SIGOPS European workshop: Beyond the PC: New Challenges for the Operating System*, Kolding, Denmark, September 2000

38. P. Johansson, T. Larsson, N. Hedman, B. Mielczarek and M. Degermark. Scenario-based performance analysis of routing protocols for mobile ad-

hoc networks. In *Proceedings of the fifth annual ACM/IEEE international conference on Mobile computing and networking*, pp. 195-206, Seattle, WA USA, August 15-19, 1999.

39. Y. Xu, J. Heidemann, D. Estrin. Adaptive energy-conserving routing for multihop ad hoc networks. *Research Report 527 USC/Information Science Institute*, October 2000.
40. J. Gomez, A.T. Campbell, M. Naghshineh C. Bisdikian. Conserving Transmission Power in Wireless Ad Hoc Networks In *Proc. 9th International Conference on Network Protocols (ICNP 2001)*, Riverside, California, November 11-14, 2001.
41. A. Nasipuri, Robert E. Hiromoto, Shengchun Ye. Performance evaluation of medium access control protocols using directional antennas in mobile ad hoc networks. In *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC'2000)*, 2000.
42. J. Postel. Transmission Control Protocol, *RFC 793*, September 1981.
43. M. Allman, V. Paxson, W. Stevens. TCP Congestion Control. *RFC 2581*, April 1999.
44. A. Bakre, B. Badrinath. I-TCP: Indirect TCP for mobile hosts. In *Proceedings of the 15th International Conference on Distributed Computing Systems (ICDS)*, pp. 170-177, Vancouver, Canada, 1995.
45. H. Balakrishnan, S. Seshan, R. Katz. Improving reliable transport and handoff performance in cellular wireless networks. *Wireless Networks*, vol. 1, No. 4, pp. 469-481, J.C. Baltzer, 1995.
46. R. Kravets, P. Krishnan Application-driven Power Management for Mobile Communication. In *Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, pp. 181-190, Dallas, TX, Oct. 1998.
47. R. Kahn et al. Advances in Packet Radio Technology. In *Proceedings of the IEEE*, vol. 66, pp. 1468-1496, November 1978.

48. C. E. Perkins, P. Bhagwat. Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers. In *Computers Communications Review*, pp. 234-244, October 1994.
49. V. Park, S. Corson. Temporally-Ordered Routing Algorithm (TORA). *Mobile Ad-hoc Network (MANET) Working Group*, IETF, July 2001.
50. S. Corson, J. Macker. Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations, *Mobile Ad-hoc Network (MANET) Working Group*, IETF, January 1999.
51. Z. Haas, M. R. Pearlman, P. Samar. Zone Routing Protocol (ZRP). *Mobile Ad-hoc Network (MANET) Working Group*, IETF, January 2001.
52. J.B. Andersen, T.S. Rappaport, S. Yoshida. Propagation Measurements and Models for Wireless Communication Channels. In *IEEE Communications Magazine*, vol. 33, no. 1, pp. 42-49, January 1995.
53. X. Zeng, R. Bagrodia, M. Gerla. GloMoSim: a Library for Parallel Simulation of Large-Scale Networks. In *Proceedings of PADS'98*, pp. 154-161, Banff, Canada, May 1998.
54. Ya Xu, J. Heidemann, and D. Estrin. Geography-informed Energy Conservation for Ad Hoc Routing. In *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking*, pp. 70-84. Rome, Italy, ACM. July 2001.
55. Kui Wu, Janelle Harms. Location Trace Aided Routing (LOTAR) in Mobile Ad Hoc Networks. In *Proceedings of 9th IEEE International Conference on Computer Communications and Networks (ICCCN)*, pp. 364-360, Las Vegas, NV, Oct. 2000.
56. Young-Bae Ko, Nitin H.Vaidya. Location-Aided Routing (LAR) in Mobile Ad Hoc Networks. In Proceedings of Mobicom '98 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking. pp. 66-75, Dallas, TX, October 1998.
57. Z. Haas, M. R. Pearlman, P. Samar. The Intrazone Routing Protocol (IARP) for Ad Hoc Networks. *Mobile Ad-hoc Network (MANET) Working Group*, IETF, June 2001.

58. Z. Haas, M. R. Pearlman, P. Samar. The Interzone Routing Protocol (IERP) for Ad Hoc Networks. *Mobile Ad-hoc Network (MANET) Working Group*, IETF, June 2001.
59. The CMU Monarch Project. The cmu monarch project's wireless and mobility extensions to ns. http://www.monarch.cs.edu.
60. J. Broch, D. A. Maltz, D. B. Johnson, Yih-Chun Hu , J. Jetcheva. A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols. In *Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking*, pp. 85-97, Dallas-Texas, October 1998.
61. S. R. Das, C. E. Perkins, E. M. Royer. Performance comparison of two on-demand routing protocols for ad hoc networks. In *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, pp. 3-12, March 2000. Tel Aviv, Israel.
62. D. A. Maltz, J. Broch, D. B. Johnson. Experiences Designing and Building a Multi-Hop Wireless Ad Hoc Network Testbed. Carnegie Mellon University, March 5, 1999.
63. S. Keshav. REAL: A Network Simulator. Technical Report 88/472, *Department of Computer Science*, UC Berkeley, 1988.
64. Lucent Technologies. WaveLAN. http://www.wavelan.com/.
65. S. Xu, T. Saadawi. Does IEEE 802.11 MAC Protocol Work Well in Multi-hop Wireless Ad Hoc Networks? *IEEE Communication Magazine*. vol. 39, no. 6, pp. 130-137, June, 2001.
66. Xilinx. The ABC's of 2.4 and 5 GHz Networking. White Paper. Available from http://www.xilinx.com/publictions /whitepapers/wp_pdf/wp148.pdf.
67. www.itworld.com. Batteries hold the key to Mobility. Available from http://www.itworld.com/nl/wireless_watch/08012001/.
68. [Duracell] Duracell. Duracell OEM specifications for AAA Alkaline baterries. http://www.duracellusa.com/Info/BatteryInfo/Big5/index.html.
69. J. Lorch. A complete picture of the energy consumption of a portable ocmputer. Masters Thesis, Computer Science, University of California at

Berkeley, December 1995. Available from http://golem.cs.berkeley. edu/~lorch/papers/.

70. V. Gutnik and Anantha P. Chandrakasan. Embedded Power Supply for Low-Power DSP. In *IEEE Trans. VLSI Systems*, vol.5, no.4, pp.425--435, December 1997.
71. J. Stevens, Spatial reuse through dynamic power and routing control in common-channel random-access packet radio networks, *SURAN Program Technical Note (SRNTN) 59 (1988)*. Available from the Defense Technical Information Center, www.disa.mil.
72. D. A. Beyer. Accomplishments of the DARPA Survivable Adaptive Networks SURAN Program. In *Proceedings of the IEEE Military Communication Conference (MILCOM)*, pp. 855-862, Monterey, California, 1990.
73. T. V. Lakshman , U. Madhow. The performance of TCP/IP for networks with high bandwidth-delay products and random loss. In *IEEE/ACM Trans. Networking*, pp. 336-350, vol. 5, June 1997.
74. L. Zhou and Z. J. Haas. Securing Ad Hoc Networks. In *IEEE Network Magazine*, vol. 13, no.6, pp. 24-30, November/December 1999.
75. WAP. WAP White Paper. Available from http://www.wapforum.com/.

Appendix A

This section provides the main TCL script used in NS2 for the MANET simulations. This is perhaps the main part of the NS2 simulation scripts. One additionally needs the movement as well as traffic files in order to drive the simulation. The format of those files is described in the NS2 manual.

Of special interest are the following parts:

1. the `report()` procedure, which does all the work on reporting the various statistics.
2. configuration of `CSThresh_` and `RXThresh_`
3. setting the flag `PreConfigureARP` that controls ARP module behavior and `Divine_Power_Prediction`, that governs our simple power management approach.

```
#================================================================
# Define options for our mobile nodes
# This part for is standard - it follows the examples provided
# with ns2
#================================================================

set opt(chan)               Channel/WirelessChannel
set opt(prop)               Propagation/TwoRayGround
set opt(netif)              Phy/WirelessPhy
set opt(mac)                Mac/802_11
set opt(ifq)                Queue/DropTail/PriQueue
set opt(ll)                 LL
set opt(ant)                Antenna/OmniAntenna
set opt(x)                  1500
set opt(y)                  500
set opt(ifqlen)             64
set opt(seed)               0.0
set opt(tr)                 /dev/null
set opt(nam)                test.nam
set opt(adhocRouting)       [ lindex $argv 0 ]
set opt(nn)                 50
set opt(cp)                 [ lindex $argv 1 ]
set opt(sc)                 [ lindex $argv 2 ]
set opt(stop)               900
set opt(nsources)           [ lindex $argv 3 ]

#================================================================
# Other default settings

Queue/DropTail/PriQueue set Prefer_Routing_Protocols    1

# unity gain, omni-directional antennas
Antenna/OmniAntenna set X_ 0
Antenna/OmniAntenna set Y_ 0
Antenna/OmniAntenna set Z_ 1.5
Antenna/OmniAntenna set Gt_ 1.0
Antenna/OmniAntenna set Gr_ 1.0
```

```
# Initialize the SharedMedia interface with parameters to make
# it work like the 914MHz Lucent WaveLAN DSSS radio interface
# Here is the interesting part: by controlling the RXThresh,
# and CSThresh we can play with the interface's carrier
# receive and sense radii
Phy/WirelessPhy set CPThresh_ 10.0
Phy/WirelessPhy set CSThresh_ 1.559e-11
Phy/WirelessPhy set RXThresh_ 3.652e-10
Phy/WirelessPhy set Rb_ 2*1e6
Phy/WirelessPhy set Pt_ 0.2818          ;# range is now 250m
Phy/WirelessPhy set freq_ 914e+6
Phy/WirelessPhy set L_ 1.0

# jbats - Enable the use of GOD !
# whether we use predetermined connectivity information or not
# if we do, the movement files have to include it ..
Phy/WirelessPhy set Use_GOD [ lindex $argv 4 ]

#==================================================================
# Main Program
#
==================================================================

# this procedure gets all the info we want at the end of the
# simulation. Normally, NS2 processes a trace file to extract
# statistics. However, we have offloaded this functionality to
# the main program itself and just use counters for our purposes.

proc report {} {
      global node_ opt cbr_ null_ lindex

      # get the netif instances for each node
      for {set i 0} {$i < $opt(nn) } {incr i} {
            set netif($i) [$node_($i) set netif_(0)]
      }
      set ArpPktTx 0
      set MacPktTx 0
      set DsrPktTx 0
      set AodvPktTx 0
      set DataPktTx 0
      set ArpTime 0
      set MACTime 0
      set DSRTime 0
      set AODVTime 0
      set DataTime 0

      # Collect the packet statistics, time spent transmitting
      # various packets
      for {set i 0} {$i < $opt(nn) } {incr i} {
            set ArpPktTx [ expr $ArpPktTx + [ $netif($i)
report_ArpPktTx ] ]
            set MacPktTx [ expr $MacPktTx + [ $netif($i)
report_MacPktTx ] ]
            set DsrPktTx [ expr $DsrPktTx + [ $netif($i)
report_DsrPktTx ] ]
```

```
            set AodvPktTx [ expr $AodvPktTx + [ $netif($i)
report_AodvPktTx ] ]
            set DataPktTx [ expr $DataPktTx + [ $netif($i)
report_DataPktTx ] ]
            set ArpTime [ expr $ArpTime + [ $netif($i)
report_ArpTime ] ]
            set MACTime [ expr $MACTime + [ $netif($i)
report_MACTime ] ]
            set DSRTime [ expr $DSRTime + [ $netif($i)
report_DSRTime ] ]
            set AODVTime [ expr $AODVTime + [ $netif($i)
report_AODVTime ] ]
            set DataTime [ expr $DataTime + [ $netif($i)
report_DataTime ] ]
      }
      # time is per node
      set ArpTime [ expr $ArpTime / 50.0 ]
      set MACTime [ expr $MACTime / 50.0 ]
      set DSRTime [ expr $DSRTime / 50.0 ]
      set AODVTime [ expr $AODVTime / 50.0 ]
      set DataTime [ expr $DataTime / 50.0 ]

      # output totally transmitted packets, time
      set fd [ open "total_tx_pkts" w ]
      puts $fd "$ArpPktTx $MacPktTx $DsrPktTx $AodvPktTx
$DataPktTx"
      close $fd
      set fd [ open "total_tx_time" w ]
      puts $fd "$ArpTime $MACTime $DSRTime $AODVTime $DataTime"
      close $fd

      # output time receiving packets
      for {set i 0} {$i < $opt(nn) } {incr i} {
            set mac($i) [$node_($i) set mac_(0)]
      }
      set ArpTime 0
      set MACTime 0
      set RoutingTime 0
      set DataTime 0
      for {set i 0} {$i < $opt(nn) } {incr i} {
            set ArpTime [ expr $ArpTime + [ $mac($i)
report_ARP_time ] ]
            set MACTime [ expr $MACTime + [ $mac($i)
report_MAC_time ] ]
            set RoutingTime [ expr $RoutingTime + [ $mac($i)
report_Routing_time ] ]
            set DataTime [ expr $DataTime + [ $mac($i)
report_Data_time ] ]
      }
      # time is per node
      set ArpTime [ expr $ArpTime / 50.0 ]
      set MACTime [ expr $MACTime / 50.0 ]
      set RoutingTime [ expr $RoutingTime / 50.0 ]
      set DataTime [ expr $DataTime / 50.0 ]
      set fd [ open "total_rcv_time" w ]
      puts $fd "$ArpTime $MACTime $RoutingTime $DataTime"
      close $fd
```

```
      set PktSent 0
      set PktRcvd 0
      for {set i 0} {$i < $opt(nsources) } {incr i} {
            set PktSent [ expr $PktSent + [ $cbr_($i) report ] ]
            set PktRcvd [ expr $PktRcvd + [ $null_($i) report-
received-packets ] ]
      }
      set fd [ open "delivery_ratio" w ]
      puts $fd "$PktSent $PktRcvd [ expr $PktRcvd / [expr
$PktSent * 1.0 ] ]"
      close $fd

      set Collisions 0
      set Discarded_packets 0
      set CollisionsTime 0
    for {set i 0} {$i < $opt(nn) } {incr i} {
      set Collisions [ expr $Collisions + [ $mac($i)
report_Collisions ] ]
      set Discarded_packets [ expr $Discarded_packets + [
$mac($i) report_Discards ] ]
            set CollisionsTime [ expr $CollisionsTime + [
$mac($i) report_Collisions_time ] ]
    }
      set CollisionsTime [ expr $CollisionsTime / 50.0 ]
    set fd [ open "MAC" w ]
    puts $fd "$Collisions $Discarded_packets $CollisionsTime"
    close $fd


}


#
# Initialize Global Variables
#

# create simulator instance

set ns_           [new Simulator]

# set wireless channel, radio-model and topography objects
set wtopo    [new Topography]

# create trace object for ns and nam
set tracefd [open $opt(tr) w]
set namtrace    [open $opt(nam) w]
# use new trace file format
#$ns_ use-newtrace
$ns_ trace-all $tracefd
#$ns_ namtrace-all-wireless $namtrace $opt(x) $opt(y)


# define topology
$wtopo load_flatgrid $opt(x) $opt(y)

#$wprop topography $wtopo
```

```
#
# Create God
#
set god_ [create-god $opt(nn)]

# Create channel #1
set chan_1_ [new $opt(chan)]

#
# define how node should be created
#

# global node setting

$ns_ node-config -adhocRouting $opt(adhocRouting) \
            -llType $opt(ll) \
            -macType $opt(mac) \
            -ifqType $opt(ifq) \
            -ifqLen $opt(ifqlen) \
            -antType $opt(ant) \
            -propType $opt(prop) \
            -phyType $opt(netif) \
            -topoInstance $wtopo \
            -agentTrace OFF \
            -routerTrace OFF \
            -macTrace OFF \
            -movementTrace OFF \
            -channel $chan_1_
#
#  Create the specified number of nodes [$opt(nn)] and "attach"
them
#  to the channel.

for {set i 0} {$i < $opt(nn) } {incr i} {
      set node_($i) [$ns_ node]
      $node_($i) random-motion 0          ;# disable random
motion
#     $node_($i) topography $wtopo
}


# jbats
# preconfigure the ARP module with all possible MAC addresses
for {set i 0} {$i < $opt(nn) } {incr i} {

      # jbats
      # preconfigure the ARP table
      set ARP_($i) [ $node_($i) return_ARPtable ]
      $ARP_($i) PreConfigureARP $opt(nn)
      # -jbats
}

# jbats
# configure the MAC to use DIVINE power prediction
for {set i 0} {$i < $opt(nn) } {incr i} {
                set mac($i) [$node_($i) set mac_(0)]
```

```
                $mac($i) Divine_Power_Prediction
        }

#
# Define node movement model
#
puts "Loading connection pattern..."
source $opt(cp)

#
# Define traffic model
#
puts "Loading scenario file..."
source $opt(sc)

# Define node initial position in nam

for {set i 0} {$i < $opt(nn)} {incr i} {

    # 20 defines the node size in nam, must adjust it according
to your scenario
    # The function must be called after mobility model is defined

    $ns_ initial_node_pos $node_($i) 20
}


# report
$ns_ at $opt(stop) "report"

#
# Tell nodes when the simulation ends
#
for {set i 0} {$i < $opt(nn) } {incr i} {
    $ns_ at $opt(stop).000000001 "$node_($i) reset";
}
# tell nam the simulation stop time
#$ns_ at  $opt(stop)    "$ns_ nam-end-wireless $opt(stop)"

$ns_ at  $opt(stop).000000001 "puts \"NS EXITING...\" ; $ns_
halt"


puts "Starting Simulation..."

$ns_ run
```