University of Alberta

Energy Efficiency in Mobile Wireless Ad Hoc Networks

by

Chu, Man Fai (Tommy) ©

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
Fall 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

0-612-81381-9

Canada

University of Alberta

Library Release Form

Name of Author: Chu, Man Fai (Tommy)

Title of Thesis: Energy Efficiency in Mobile Wireless Ad Hoc Networks

Degree: *Master of Science*

Year this Degree Granted: Fall 2002

Tommy

20B Hong Shing Court,
Sun Hing Garden, Tai Po, N.T.,
Hong Kong

September 27, 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 Efficiency in Mobile Wireless Ad Hoc Networks submitted by Chu, Man Fai (Tommy) in partial fulfillment of the requirements for the degree of Master of Science.

Dr. Ioanis Nikolaidis

J. Harms

Dr. Janelle Harms

Dr. Marek Reformat

September 24th, 2002

Abstract

MANETs (Mobile Ad Hoc Networks) are wireless networks with no fixed infrastructure and with limited per-node energy reserves. This thesis targets the reduction of MANET energy requirements by controlling their transmission power. We take three approaches: (a) globally adjusting the transmission radius to a single value for all nodes while maintaining connectivity of the underlying graph, (b) adjusting the transmission radius on a per-node basis while making all nodes reachable by an arbitrary source, i.e., broadcast, and (c), determining energy efficient paths between source-destination pairs that satisfy node-destination traffic demands and per-node capacity constraints. We find, respectively, (a) several artefacts caused by the Random Waypoint (RWP) model, (b) ways to refine the wireless broadcast schemes in order to cope with mobility, and, (c), preliminary results of the complicated interplay between the wireless nature of the channel and the ability to satisfy the energy objectives while not violating the capacity constraints.

CONTENTS

LIST OF FIGURES

LIST OF SYMBOLS

General Graph Representations

E	A set of edges
V	A set of vertices
G(E,V)	A graph

Transmission Radius or Energy Consumption

α	The Loss Exponent for the Relation Between Energy and Radius
R	The Length of a Communication Radius for all nodes
r_{ij}	The Communication Radius from a Specific Node i to Another Node j

Mobility Model

N	Number of Nodes
Vm	Maximum Velocity
T	Average Pause Time
MaxX	The Width of a Rectangle Boundary
MaxY	The Height of a Rectangle Boundary

Node Density Analysis

2a'	Width of the Interior Rectangle
2b'	Height of the Interior Rectangle
2a	Width of the Exterior Rectangle
2b	Height of the Exterior Rectangle
β	The Width/Height Ratio Between The Interior and Exterior Rectangle

Broadcast Algorithms

BIP	Broadcast Incremental Power Algorithm
BLU	Broadcast Least Unicast-cost Algorithm
BLiMST	Broadcast Link Based Minimum Spanning Tree Algorithm
IS	Incremental Search Algorithm
DS	Decremental Search Algorithm

Network Traffics

τ_i	Received or Heard Traffic From Node i
ϖ_i	Forwarded Traffic From Node i
A_i	Originated Traffic From Node i
C	Capacity of Each Node

Chapter 1

Introduction

Cellular Networks and MANETs

Wireless technologies for support of voice communications emerged in the 1920s and, in the 1970s, the first cellular networks (also called wireless *infrastructured* networks) appeared. In terms of network architecture, cellular networks contain mobile nodes and *base stations*. Mobile nodes are portable devices, i.e., cell phones, equipped with wireless transceivers, and most of them are powered by batteries. In addition, a battery is a limited energy resource and, because of that, the autonomy of mobile nodes is also limited. The infrastructure part of cellular networks is the collection of the *base stations*. Base stations are fixed in location and they are connected each other, forming a network that connected by wire. Moreover, they act as gateways between the wireless mobile devices and the fixed (wireline) telephone network. The task of base stations is to receive/transmit any traffic from/to mobile nodes. Essentially, because the communication between base station can apply any routing algorithm in wireline network, the problem of finding a path from a mobile to another endpoint (a phone or another mobile) is reduced to that of setting up a connection between the corresponding base station ("near" the originating mobile) and the remote endpoint, and the process of terminating a call to a mobile is similar. In other words, the cellular infrastructure allows the reuse of the control mechanisms used in the wireline phone network. Certainly, a few control issues are still unique in the case of cellular networks. For example, the problem of *paging* (alerting a mobile of incoming phone calls) is an integral part of the expected functionality.

Each base station can only cover a specific area, called a *cell*. This limitation is sometimes by design but it always relates to the fact that wireless transmissions can only be received with relative reliability within a certain distance from the transmitter. The limitation of distance covered by base stations is one of the most pressing technological issues and no single network provides universal coverage. Mobile nodes can easily find themselves moving outside the coverage of one cell. Several different techniques, collectively known as *handoff* techniques, are used to transfer the control of a mobile device's connection from one base station to another base station in order to provide continuous connectivity despite the mobility. However, given sufficient distance from the closest base station, communication with the rest of the network may simply not be possible. Remote locations are generally neglected with respect to cellular connectivity. Furthermore, the base stations are, in principle, bottlenecks, since with the current extensive use of cell phones the number of channels available to a base station is much smaller than the possible instantaneous offered load, especially during peak demand, e.g., as in cases of emergencies.

In order to deal with the problem of scaling and extending the service area of wireless networks, an idea that appeared, first in research circles, and then in sporadic testbeds, was that of building a network exclusively of mobile nodes. According to this approach, the set of mobile nodes would form a "community" whereby a node not only acts as the origin or destination of traffic, but also forwards traffic on behalf of other nodes. The name associated with such networks is *multi-hop wireless networks*. No fixed base station infrastructure would need to be deployed, and the demand-based deployment of (mobile) nodes would be much more flexible in terms of extending the reach of the service area. While this approach eliminates the need for particular nodes to act as base stations, and assigns equal rights and responsibilities to all nodes, their coordination (e.g. for the sake of routing) is particularly difficult in light of the fact that continuous changes of the node locations continuously change the underlying topology of the network. Hence, it is not rare that routing algorithms applied in such networks are capable of finding redundant paths between source and destination to counter the effects of mobility that quickly render a discovered unusable path.

Along with the gradual reduction of prices for wireless transceivers, several factors motivated a fresh look into multi-hop wireless networks. Among them we can mention the gradual reduction of the cost and physical dimensions of mobile devices as well as the increasing congestion in cellular networks. Compared to their wireline cousins, wireless networks did not advance in terms of speeds (bit rates) as quickly, and hence the control of the available bandwidth was, is, and remains, an important issue. Finally, the transition towards a global packet-switched infrastructure, regardless of technology employed (be it wireline or wireless) meant that packet-switching principles of handling and forwarding traffic are to be used as well (including the wide use of legacy transport layer protocols, such as TCP). The name used to describe such networks in the past was *packet radio networks*. The current name by which the research in packet radio is known is *Mobile Ad-Hoc Networks (MANETs)* research. MANETs assume that nodes are able to automatically configure themselves without special roles assigned to particular nodes.

Similar to cellular networks' mobile devices, each MANET's node includes a transceiver, which allows the node to transmit packets to any node within its communication range. Moreover, the wireless medium is inherently a broadcast medium, i.e., one transmission is received by all nodes that happen to be in the vicinity of the transmitter. This particular property will be convenient when we study the problems of broadcast and multicast service. A node's transmission power defines the communication radius.

Thesis Motivation and Scope

The major research issue in MANETs, that has rightfully received extensive attention, is that of *routing*, finding and making use of a path from a source to a destination. The continuously changing position of nodes results in a fluid topology and presents a challenging context in which to find and maintain paths. However, routing is only one facet of the more complicated problem of *resource management*. It could be claimed that if objectives of Quality of Service (QoS) are ever to be met by MANETs (in a very similar way that QoS is expected out of wireline networks), more dimensions to resource management must be examined. Among the resource management issues we only mention four of considerable importance: (a) ability to reuse the available frequency spectrum by controlling the extent of transmissions (spatial reuse), (b) ability to reduce the energy consumption of devices, (c) ability to reduce the congestion by the uncoordinated nature of the nodes (typically handled by the Medium Access (MAC) Protocols), and (d) ability to reduce the interference between nodes (exemplified in interference-limited techniques, such as Code Division Multiple Access). We will focus on (b), which has an impact on both (a) and (d), and we will assume for the rest of the thesis that (c) is handled by a suitable MAC protocol.

Due to their small size (to encourage portability), nodes are powered by a restricted number of batteries. The result is a finite energy supply. In order to lengthen the autonomy of devices, reducing power consumption is desirable. As far as the lifetime of a network is concerned, smaller power consumption means extended lifetime, which could be translated to better QoS, if our QoS includes a concept of connection longevity. Thus, energy conservation in portable devices became a serious concern in recent years. In particular, we factor out the energy consumption of the user-oriented parts of a device (e.g. display, disk or memory units, etc.) and pay attention to the energy needed to operate the network interface card (the transceiver), which we will define as *operational cost.* That is, our primary focus as far as cost is concerned is the *energy* cost which reflects a measure of the energy consumption of the wireless transceiver. Both transceivers and their support software should be designed to minimize this cost.

For example, nodes can be programmed to stop transmitting and/or receiving for arbitrary time periods, which is called the *sleep period operation.* When a node does not receive and does not require any message forwarding, it can switch to sleep mode. Energy is conserved by refraining from sending or receiving during this specified period. Nevertheless, the ability to set the device into sleep period when it does not anticipate in any communication process requires close coupling with the data link layer protocol through a standardized interface, and the network layer routing protocol should be able to accommodate such periods without excessively adverse consequences, e.g. relay nodes that disconnect from the network frequently cause higher delay to a message. Instead of using the approach of idling the transceiver, we will focus on adjusting the transmission power as our primary tool for energy conservation.

Most proposed routing protocols do not adjust the transmitted power. Instead, all transmissions occur at the *maximum* power level. With the transmission power adjustment feature, nodes could in principle alter their transmission power level from almost zero to the maximum value. Nodes could decide on the best power level (as it happens e.g., in CDMA environments for interference reduction issues) on a packet-by-packet or timeslot-by-timeslot basis. What we imply by this is that regulation of transmitted power is already used in an interference control context. Why not use it in an energy conservation context then. Consequently, if not all nodes transmit at the maximum power, the *global energy consumption* (the total energy that all the nodes consume collectively) can be reduced. An open question is whether the global energy consumption can be minimized using fairly straightforward control algorithms. It is precisely this topic: *controlling the radio transmission power with the objective of reduction of global energy consumption*, that forms the main axis of the current thesis.

We will focus on three issues towards global energy minimization:

(a) Global reduction of transmission radius
(b) Per-node reduction of transmission radius for broadcast services.
(c) Per-node reduction of transmission radius for unicast services with known loads.

In particular, (a) is an "agnostic" approach, whereby the only objective considered is that of connectivity. That is, the radius of all nodes is adjusted to the same (small) value that maintains the connectivity, despite the mobility. However (a) does not consider any particular service scenario or node traffic demands, but just the need to maintain the underlying topology connected in the face of mobility. In contrast, (b) looks into how to adjust the transmission radius of *each* node so that a particular service (broadcast) occurs at minimum energy cost. Finally, (c) studies the load-dependent allocation of per-node transmission radius subject to bandwidth constraints and the natural spatial-reuse features present in wireless networks.

Thesis Organization

Chapter 2 provides a brief review of the features and problems in MANET energy conservation based on the control of transmission power. Key assumptions are introduced. A review of current literature related to the particular topic is given, but no attempt is made to cover the entire literature on MANET routing since most of it is not related to energy-related objectives or constraints. Chapter 3 embarks on the determination of the minimum transmission radius (for all nodes in a MANET) with the objective of maintaining connectivity with high probability, despite mobility. Clearly, the results are model-dependent, but what is found is the remarkable and counter-intuitive result that increased mobility results in higher connectivity. We present a proof of why we believe this is the case and how it relates to structural properties of the underlying graph in Random Waypoint (RWP) mobility simulations. Chapter 4 considers a per-node transmission radius assignment in order to minimize the global energy consumption in the context of broadcast service. We find the certain previous results for *static* wireless nodes are not satisfactory and that a new approach is necessary in order to maintain energy efficiency. This approach attempts to take corrective actions on-line to avoid re-calculating the transmission radii of all nodes in the system. Section 5 is just scratching the surface of the interrelated nature of constraints (in the particular case, bandwidth constraints) and energy efficiency. This time, per-node transmission power for unicast service is studied, given known traffic load matrices. Finally, Chapter 6 summarizes the major results of this thesis, and describes interesting research problems, especially as they relate to the material in Chapters 4 and 5.

Chapter 2

Background and Related Work

Basic Assumptions and Abstractions

In the remainder of this thesis a node's communication radius will be approximated by a circular area. Thus, we factor out the impact of shadow fading and other phenomena that can create severe time-dependent non-symmetric attenuation patterns around a transmitter. We note however, that the circular area assumption does not necessarily hurt the generality of the results and, at the same time, it is straightforward to imagine the interaction of nodes whose transmissions cover circular area (with the transmitter at the center). It is much more difficult to conceptualize their interaction if arbitrary coverage areas are assumed. Furthermore, all simulated mobility models known to us use the circular coverage assumption too.

Nodes are assumed to be able to receive data from a node if they are within its transmission radius. An abstract view of the underlying MANET topology can then be represented by a graph, G(*V*,*E*). *V* is a set of vertices that symbolizes each mobile node while *E* is a set of (directed) edges that determines the existance of a wireless link. If a node, i, is within the transmission radius of another node, j, then there exists a (directed) edge from j to i. One other assumption frequently made (unless otherwise specified) is that if all nodes transmit at the same power level and an edge from i to j exists, then, so does an edge from j to i (assumption of link symmetry). In this case, the links are called *bi-directional links*. However, if the transmission power of node *i* and node *j* are not the same, then it is possible to have the one direction and not the other. The links are accordingly called *uni-directional links*. Finally, it is perfectly possible to have no edge connecting to vertices (nodes) simply because they are far apart from each other.

By looking into the connectivity properties of the graph describing the topology of the MANET, we can determine whether a node is reachable by every other node. If this is true for all nodes, then the graph is said to be *connected*, i.e., a path exists from any node to any other node. Unfortunately, the continuous mobility of nodes means that the edges are not permanent; instead they are added or removed as per the mobility of nodes.

Hence, the connectivity and any other fundamental properties of the topology are time dependent.

Mobility Models

In MANETs, node movement is user/application dependent, and the assumption is that the user may be located at an arbitrary location, move towards an arbitrary direction, and move at an arbitrary (within reason) velocity. The abstract description of the collective movement of nodes is referred to as the *mobility pattern* or *mobility model*. Since mobility patterns influence the network topology dynamics, we naturally expect the use of a particular model in simulations to have immediate effects on the performance of any proposed routing protocol. A mobility model should reflect the real movement of nodes, and it should be independent of the routing (and other) protocols. In fact, there exist two types of mobility models. The first, *trace models* are mobility patterns collected from real life systems over a time period. Since MANETs have so far been used in military applications there is a very limited pool of existing systems to collect traces from. Moreover, they are not necessarily representative of other mobility scenarios. Thus the need for a second class, that of *synthetic models*, aims at representing the behaviour of mobile nodes without using any traces. Synthetic models attempt to make "reasonable" assumptions about the node movement without a particular application or existing system in mind. Hence, a straightforward circumstance to represent the node movement in mathematical notations is that nodes randomly move within a two dimensional space area.

The goal of finding the "right" mobility model becomes one of coming up with a plausible mobility model that possesses enough degrees of freedom to allow the stress-testing of any proposed MANET protocol. Different nodes should be allowed to move independently of each other, reflecting the degree of independence in the movement of different users. Among the existing mobility models, we note a number of synthetic models, such as Gauss Markov Model [1], and the Random Way Point (RWP) Model [2]. Their synthetic nature suggests that they are not necessarily replicating reality, but posses enough degrees of freedom (e.g., selection of direction of movement, selection of speed, etc.) that produces a sufficiently rich set of scenarios to use when exploring the effectiveness of protocols for mobile, and in particular for, MANETs. Other models, such as the Reference Point Group Mobility Model [3] attach a certain behavior to the nodes such that a *cluster*, i.e., groups of adjacent nodes, can form clusters as part of their dynamic behavior.

In order to have a model that can facilitate the generation of replicatable results from a simulation, *boundaries* (or cells) become an aspect of mobility models. The boundaries maintain a constant number of nodes moving within a restricted area, and hence the restriction affects the locality and movement of nodes. Simulated boundaries can be classified into four types, hexagonal cells, circular cells, rectangular cells, and boundless cells [4]. Hexagonal and circular cells are adapted from cellular networks, where they represent the relation of a (fixed) base station, usually placed at the center of the cell, and the freely roaming mobiles. In MANETs, on the other hand, the lack of a fixed infrastructure allows essentially an unrestricted topology of boundaries. Therefore, most simulations use a rectangular boundary for reasons of convenience.

The boundaries are essential to maintain a constant number of nodes in the model. Given the invariant number of nodes, one can relate the simulation results to the offered load which is captured (or related to) the number of users. For the invariant to hold, the boundaries can act as a *reflecting barrier* (nodes reflect with the opposite direction when they reach the boundary), a *non-reflecting barrier* (nodes warp around the boundary and instantly re-appear at the opposite boundary continuing their movement) or *an unreachable limit* (nodes are never reach the boundary by the predefined destination chosen within the boundary). The warped models could be extended to include the nodes on a sphere, mimicking the motion of users on the surface of the earth, but MANETs are considered to be operating in a substantially smaller size than a worldwide communication network. Furthermore, all kinds of boundaries only consider in two-dimensional boundary only, and in the literature, it does not exist any model that considers a higher dimension or barrier inside the boundary.

The model of choice in our study, and also the model used by the majority of researchers, is the Random Way Point (RWP) Model. The model is described by six parameters: N, is the number of independently moving nodes, capturing the number of mobile users participating in the simulation. R is the communication radius of each node. In Chapter 3, we will consider R to be the same for all nodes. In Chapters 4 and 5, we will assume per-node communication radii. The motion moreover depends on two parameters, Vm and T. Vm is the maximum velocity of a node, and T is the average pause time at a waypoint. The remaining two parameters, MaxX and MaxY, then will determine the two dimensions of (width and height) of the rectangular boundaries.

In the RWP model, nodes move independent of each other. Each node's movement consists of alternating periods: movement and pause periods. The simulation initializes the location of the nodes in a uniformly random fashion within the rectangle. For a single node, a destination point, called *waypoint*, is selected again in a uniform random fashion. A velocity is selected in a uniform random fashion in the range from 0 to Vm. The node moves with the selected velocity to the waypoint. When it reaches the waypoint, a random period of time (with mean T) is selected from an exponential distribution. The node pauses at the waypoint for this amount of time. After the pause, the node picks the next waypoint and velocity, starts moving, and the process continues indefinitely in this fashion. Hence, the boundary from the RWP becomes the unreachable limits since the pre-selected waypoint prohibits a node to travel beyond the rectangle boundary.

RWP was selected in this study in compliance with the majority of researchers in the area. The common intuition that by maintaining the number of nodes invariant within the rectangle, the density of nodes is uniform throughout the rectangle is simply wrong. Boundary effects, i.e., differences in the node density due to the existence of boundaries, have been observed before [5], but without any quantification, [5] proposed an alternative mobility model instead. The presented research started with the assumption that boundary effects would not be particularly relevant or have a noticeable quantitative impact. As the next chapter points out, this was far from the truth.

Transmission Radius

From the standpoint of finding a path in the network, one may claim that the longer the transmission radius of all nodes, the better, because it will ensure there will be enough wireless links (edges in G) that allow the creation of paths from any source to any destination. A longer transmission radius reduces the number of hops that a packet needs to "traverse". However, by the same token, it increases the number of nodes that use (can hear) the same transmission. Collision avoidance schemes, such as Carrier Sense Multiple Access / Collision Avoidance (CSMA/CA), prohibit nodes from transmitting if other nodes are heard transmitting. Hence, the more nodes within reception range from a node, the fewer chances the node will have to transmit. In other words, larger transmission radii congest the network and decrease the overall performance, e.g. decrease the throughput [6,7] which also has an effect on the average delay necessary to deliver a packet. Consequently, larger radius implies smaller network capacity, because no effective use of the spatial reuse is made.

On the opposite direction, by decreasing the radius, we note that the dynamic topology of MANETs suggests that we increase the chances that the underlying graph is partitioned (not connected) with the slightest of movements. It is therefore not clear which radius is the smallest one that still can provide with high probability a connected network. In [6,7], it is suggested that the optimal transmission radius may not be determined by a fixed value for all nodes in the network. In practice, the optimal radius may constantly change due to mobility.

Kleinrock and Sliverster [7] have shown results regarding the radius such that the channel reuse is optimized, finding out in the process that the connectivity is indirectly related to the rule of *six neighbors*. That is, the underlying graph is very likely connected, if each node possesses a radius that allows it to reach at least six other stations. In particular, the paper demonstrates that the throughput of the network is optimal when the average degree of each node, i.e. number of (neighbors) mobiles that each transmission radius can cover, is equal to six with the assumption of uniform density throughout a 2-D space. We are interested in a version of this problem where mobility is assumed, and hence, we look into how to compensate for the mobility by allowing our transmission radius to extend beyond the absolutely necessary size to provide connectivity at a particular instant – so that connectivity can be ensured with high probability in the near future as well.

Energy Minimization Considerations

As far as the problem of finding a path from a source to a destination in MANETs is concerned, many routing protocols have been proposed, e.g. [8,9,10], and they are central on the ability to effectively forward traffic. Combining the problem of routing with energy efficiency, we find relatively few examples, e.g., [11,12,13,14], in MANETs' literature. These protocols operate by inspecting multiple paths and selecting among them the most energy efficient. Each of the candidate paths assumes that the nodes involved in it transmit at the maximum power level (rather than a particular nominal value). While well-matched to MANETs in one sense (multiple paths provide the source an alternative when some become unavailable due to mobility), the proposals are nevertheless not interested in adjusting the transmission power of each node on the path.

There are two ways to formulate energy conservation problems. One is to see them from the view of individual nodes. Another is to see them as global processes. According to the first view, it is tempting to try and minimize the energy consumption of a single node or particular nodes, or even the difference in consumption among all nodes (in order to "equalize" them). However, as a matter of their location and mobility, nodes could be forced to act as relays of the traffic of an unpredictable number of other nodes. Indeed, if we were looking into the minimization of the energy consumed by a single node, the answer would be that only traffic originating or destined for this node ought to be legitimate reasons of energy consumption – and it would also be the absolutely minimum possible energy consumed by the specific node. In other words, if we discount the nature of nodes to act as relays of other node's traffic, we reach an unrealistic minimum energy. Clearly, in order for the network to exist, all nodes must suffer the (collective) consequences of having to consume energy to forward other people's traffic. It is under this light that we consider as the primary issue of our study the *global* energy consumption. Per-node *objectives* are seen as secondary for a MANET environment. It is however still possible to consider the impact of per-node *constraints* (e.g. different available battery reserves) but we leave such constraints aside for the time being.

A consequence of global energy minimization is that certain nodes may be victimized, since their energy consumption is higher. Possible candidates to such "unfair" treatment include nodes that are in a particular position in the network topology and act as relays for many other nodes. Notably, our simulations do not provide cases where such victimization occurs. This is an artifact of the indistinguishable (in the statistical sense) nature of the nodes in the RWP model so that victimized nodes are randomly chosen. Furthermore, to maintain network connection has a higher priority than energy conservation in our consideration. Hence, if the node that forward traffic (the one and only one can) reduces its power level, it might disconnect the network. As a result, it decreases the routing performance.

Finally, we define the "cost" of transmission between two nodes in a wireless environment. In order for two nodes at distance r to directly communicate, the emitted power should be proportional to r^{α}, where α is called the *loss exponent* and it captures the effects of the path loss. Note that $2 \leq \alpha \leq 4$, i.e., the loss is more severe than the inverse square of the distance. The nature of the loss exponent suggests that there are cases where it is preferable to transmit using an intermediate node, instead of directly. For example assume nodes A, B, and C arranged in the given sequence on the same line. The distance from A to B is 10 units and from B to C is another 10 units. Assuming a loss exponent of 3, the direct transmission from A to C would involve $(10+10)^3=8000$ units of power. In the second case, A uses B as a relay, and B transmits to C. Clearly, $10^3+10^3=2000$ units of power are necessary for the data to be delivered to C. The difference suggests using an intermediate node is sometimes preferable to economize in energy. Of course, that's the case if B is between A and C. If it were in the opposite direction, it would not have made any sense to use it. Hence, the question is how to exploit intermediate nodes, if we do not know much about their location (and knowing the fact that they will not be at the same location for long).

The Wireless Broadcast Advantage

There are at least three different types of service in a data network depending on the number of recipients. They are *unicast*, *multicast*, and *broadcast* delivery, corresponding to one, multiple and all possible recipients in a network. In a wireline network, multicast and broadcast services typically build on top of the unicast service model, attempting to avoid transmitting multiple times the same packet on the same link. In wireless environments, the underlying service paradigm due to physical properties of the medium is *broadcast* because a single transmission is received by multiple nodes who are within the transmission radius of the sender. The other two services (unicast and multicast) are produced as restricted forms of a sequence of broadcasts. The underlying broadcast property is valuable in that a single energy cost (one transmission) can be amortized over a number of recipients.

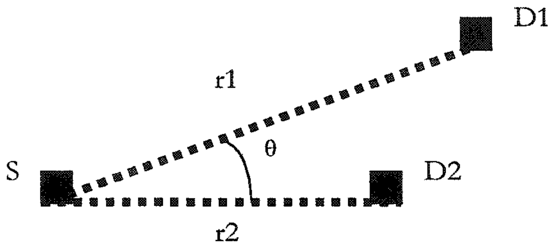


Figure 1: Example Configuration

Consider now Figure 1 (adopted from [15]). If S wishes to transmit to both destinations D_1 and D_2 (at distance r_1 and r_2, where $r_1 > r_2$) then it could either (a) transmit at power level proportional to r_1^{α} and the broadcast will be received by both, or (b) transmit at a power level proportional to r_2^{α} and let D_2 transmit to D_1 (i.e., let D_2 act as a relay node). It can be shown that the first option is preferable in terms of total expended energy when it satisfied the condition that is shown in Equation 1. The inequality certifies the energy consumption from a single transmission using r_1 is less than the energy consumption from two transmissions that using D_2 as a relay. (It uses the cosine laws to find the distance from D_2 to D_1).

$$\left(\frac{r_1}{r_2}\right)^{\alpha} - 1 < \left(1 + \left(\frac{r_1}{r_2}\right)^2 - 2\left(\frac{r_1}{r_2}\right)\cos\,\theta\right)^{\frac{\alpha}{2}}$$

Equation 1: Preferable Transmission Determination.

The property exploited in case (a), i.e., a single transmission from S that covers both D_1 and D_2 is also referred as the *wireless broadcast advantage*. In general, if an arbitrary node needs to transmit to n nodes, and assuming that the distances of the n nodes are sorted as $r_1 < r_2 < \ldots < r_n$, then it is enough to transmit at a power level proportional to r_n^{α} to cover all n nodes.

Undoubtedly, the energy efficient broadcasting is more complicated than the example given here. Typically, many more than just two destinations can be involved and therefore several distance and angle components need to be considered. We also note that a working assumption is that the angle θ is not known, although the distances are known. The lack of "direction knowledge" in the form of θ is linked to the assumption of omni-directional antennas. As for knowledge about the distance, although it may not be known exactly, technologies have been developed to determine distances with fairly high accuracy. In fact one does not need an exact knowledge about the physical distance but rather a one to one relation between transmission power levels and the nodes reachable by transmission at the corresponding power level. In the most rudimentary form, this information can be discovered from a sender by gradually increasing its transmitted power and inquiring at each step for the nodes receiving the signal to respond. Hence, from the perspective of a node, a mapping between transmission power and reachable nodes is possible.

As seen in the example, the broadcast problem is essentially a tree construction procedure (e.g. S with children D_1 and D_2 from Figure 1), hereafter called the *broadcast tree*. The idea is that all nodes receive the transmission of the root node. Some of the nodes receiving the broadcast act as relays and transmit the same message (the ones that received but do not transmit any further are called leaf nodes). A node receiving from a relay may also act as a relay and repeat the transmission even further, etc. The process continues until all nodes in the network have received the transmission from either the source or a relay. It is possible that the same node receives the transmission by more than one relay (or relays and the source). Such redundancy is not considered harmful. It is assumed that the payloads possesses relevant information (e.g., sequence numbers) in cases where duplicates detection is essential, i.e., it is a higher-layer issue. One characteristic of the broadcast tree is that the operation is initiated from a single source and all other nodes will involve in the operation. In Chapter 4, we will discuss more on constructing global energy conservation wireless broadcast trees.

Our attention to broadcast services is not accidental. Broadcasting messages is believed to be an effective way to deliver data in the wireless environment [16]. When all mobile nodes are involved in the communication process, i.e. broadcast operations are highly dependent in MANETs, the global power consumption becomes a concern. In this sense, broadcasting is a good stress-test of energy minimization schemes. Moreover, broadcasting is at the heart of many other services. For example *registration* (a station notifying all others about its location) and *paging* (a node in the network attempting to discover the location of a destination by flooding a request to the entire network) are essential services. Furthermore, many forms of flooding (i.e., broadcast of a message to the entire network) are at the heart of route discovery protocols, as e.g., in the DSR protocol. There is therefore ample reason to consider broadcasting as a sound test case. In fact, the static case (without mobility) of the problem called the Minimum Energy Broadcasting (MEB) problem, has been proven to be a NP Hard problem [17,18]. The new dimension added in this thesis is how heuristics that were proposed for MEB can be adapted to a mobile environment.

Constraints

Finally, the wireless medium brings about several other constraints and limitations. For example, wireless links have significantly lower capacity than links in a fixed network. This is mainly because the high bit rate error and the degradation of the signal when communicating through the air medium. Furthermore, because of the dynamic topology in MANETs, the network needs to maintain viable routes, whose discovery involves *control packet overhead*. Control traffic may become a non-trivial part of the overall load in the network if the mobility is fairly intense. Other constraints are constraints related to interference, constraints related to the maximum transmission power (or quantized level of transmission power) etc.

In this thesis, we will study communication radius adjustments, which aim at the minimization of global energy consumption. In chapter 5, we will introduce one constraint – that of bandwidth sharing between nodes whose transmission radii overlap. We will tacitly assume that sharing the bandwidth at any point in the network is doable in a near-perfect and fair fashion via a suitable MAC protocol. Notably, we will not introduce any particular MAC protocol. The lack of MAC protocol (or its model) should be seen as an attempt to provide results that assume ideal MAC protocol behavior (no collision and no overhead during the communication process), and hence, have a general value.

In the next chapter we start with the speculation that a certain transmission radius should exist for a MANET with specific mobility features (as captured by a mobility model) that is the smallest possible overall and guarantees with high probability that the network will remain connected, despite mobility.

Chapter 3

Critical Radius and the RWP Model

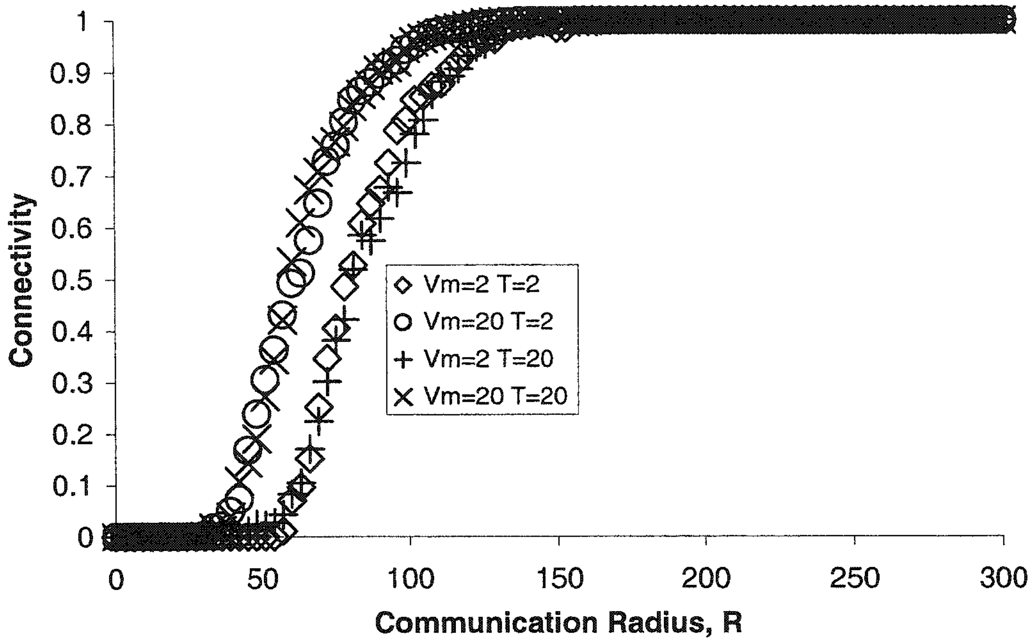
Critical Radius

By saying that a MANET is *connected*, we mean that the underlying graph is connected. Due to the dynamic nature of the edges, as nodes move within or out of range of each other, we will quantify *connectivity* as the fraction of time (a time-average quantity) that every node is reachable (in one or more hops) by every other node. To compute this metric, we will count whether the network is connected in each time unit, and the connectivity is equal to the sum of all time units that a network is connected over the total time that the network operates. Hence, a connectivity of 1 indicates that the underlying graph is continuously connected.

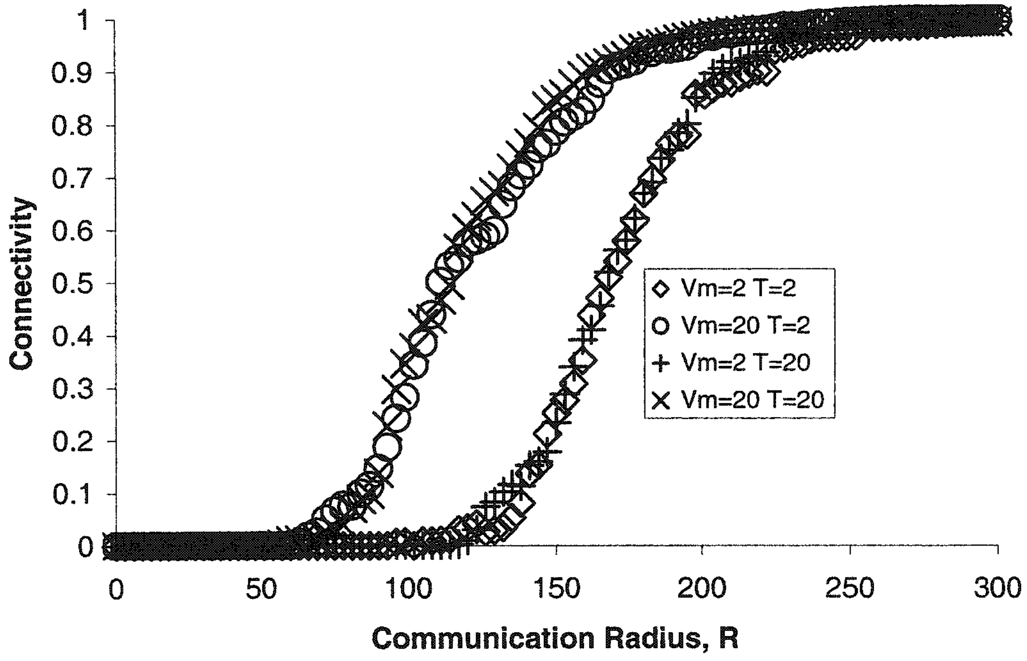
Trivially, in the RWP model, we can be certain that the graph is connected if the transmission radius of each node is equal to $\sqrt{MaxX^2 + MaxY^2}$ (considering the diagonal as the worst case radius from the boundary whose area is equal MaxX x MaxY square unit). However, considering bandwidth consumption, transmitting with the largest radius causes the diminishing of network resources [7], which results in a decrease of the routing performance. Our interest is to economize the energy spent on transmissions, and hence, a much smaller radius is desired.

To emulate MANET environment and to measure the performance metric, simulations should consists of two major properties – mobility and network topology. Mobility is user movements, which we use one of the synthesis mobility model, Random Waypoint Model (described as in Chapter 2) to generate the different scenarios. The topology is defined by the wireless links, which depends on the transmission radius. If a node inside ones communication radius, a wireless link is established, and in contrast, a wireless link is demolished when a node leaves ones communication radius. In this chapter, we consider the communication radius is adjusted globally for all nodes (we will consider different transmission radius for individual node in the chapter 4 and 5).

Each data point from the simulation results throughout this thesis is from the average of ten runs and it is conformed to the 90% of confidence interval. Connectivity results are shown in Figure 2. The simulations were carried out for N=50 mobiles in a 500 x 500 square or 500 x 1500 rectangular boundary. Different maximum speed and average pause time generated four scenarios, with Vm = 2 or 20 and T =2 or 20. Each data point is an average value of 10 simulations, and each simulation runs for 900 units of simulation time. The results indicate that the pause time does not affect connectivity much as speed. It is obviously the case that each curve can divide into three segments. First, the *subcritical* range where transmission radius is small and the graph remains disconnected. Then, we consider the connectivity transition from zero and one as the *critical* range. This indicates that the network is fragile, and where connectivity is not always the case. The *supercritical* range is the last segment of the curve where the connectivity is almost always 1. In other words, nodes are reachable from each other at all times while using any radius within this range. Having noticed the energy cost is equal to the communication raised to the power of the loss exponent, α, we would choose the lowest possible value, the critical radius, at the transition from the critical to the supercritical range in order to conserve energy while preserving connectivity.



(a) MaxX=500, MaxY=500



(b) MaxX=500, MaxY=1500

Figure 2: Connectivity as a function of the transmission radius for 50 nodes in 4 different speed/ pause time scenarios inside (a) a 500 x 500 square boundaries, and (b) 500 x 1500 rectangle boundaries.

From the results presented here it is clear that while connectivity is insensitive to the pause time, and hence the critical radius is dependent on the speed. However, a moment's reflection on the results suggests that we need a smaller critical radius when the average velocity is higher! The observation is counterintuitive. We would expect at higher velocities that either the connectivity is lower – due to the more frequent movement of the neighboring nodes, or, more intelligently, the connectivity is the same regardless of speed because the average number of nodes within a nodes coverage (hence the cardinality of the set of neighbors) would be the same (nodes that move away from the set of neighbors get replaced quickly by other nodes moving into the set of neighbors). The two trends would cancel each other out.

At this point we need to explain why at low speeds, the graph remains disconnected for a certain transmission radius; however, under the same transmission radius, it gains in connectivity with higher mobility. In the case of the 500 x 1500 rectangle, in order to achieve the same connectivity of 50%, the nodes in the low speed scenario must have the radius, which is at least 100 meters more than the high-speed scenario. Given that energy costs in wireless transmission are related to the 3rd or even 4th power of the distance, attaining a distance increment of 100 meters involves a substantial energy penalty. Based on earlier observations about the boundary effects, we would expect that the node density within the rectangle is not uniform, which would explain possible problems on connectivity. However, it is not clear if the boundary effects would result in increase or decrease of connectivity. To this end, we need to determine the node density of the RWP model.

Node Density of the RWP Model

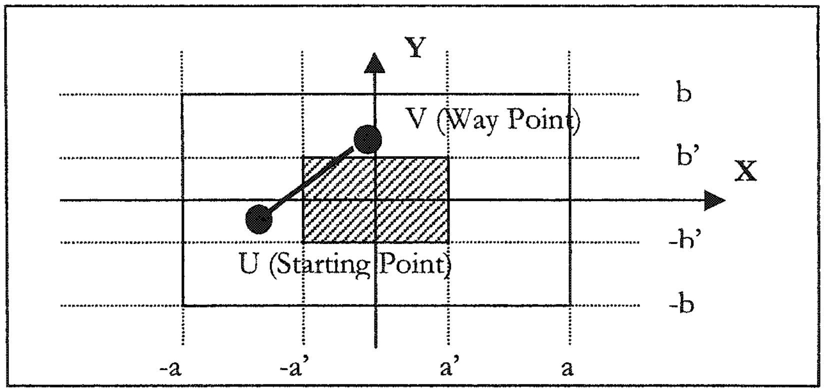


Figure 3: Configuration of rectangular boundaries used in the analysis.

In this section, we prove the non-uniform node density distribution or RWP, and then investigate its effect on connectivity. First, we consider a rectangle boundary with 2a units width and 2b units height, with its center at the origin of the two-dimensional coordinate system, as shown in Figure 3. In order to study the difference in density at the interior, we define an interior rectangle with its center also placed at the origin, with 2a' units width and 2b' units height, i.e., $a'/a = b'/b = \beta$ where $0 < \beta \leq 1$. Therefore, β becomes a scaling factor of the interior rectangle. Under the uniform density hypothesis, we would expect: $\Pr[\text{node} \in \text{interior rectangle}] = P_{int} = 4a'b'/4ab = \beta^2$. However, in the following proof, we will find out: $P_{int} \geq \beta^2$, which suggests a non-uniform density instead. To simplify the analytical derivation, we introduce the notion of quadrant as the rectangles of boundary outside the interior rectangle. The possible locations of quadrants are shown in Figure 4. First, we will concentrate on quantifying the properties of a node being located in quadrant I, Q_I. The area that covers the remaining area will be called quadrant II, Q_{II}.

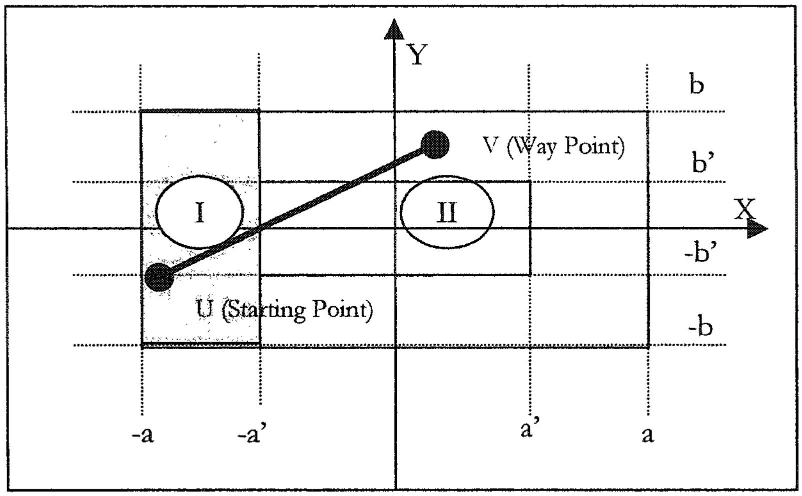


Figure 4: Quadrants divided from the boundaries. U represents a node's original location, and V represents a nodes destination waypoint.

Proof: The node density is not uniform

Assumptions:

1. Movement is generated using RWP model.
2. Each node's movement is independent of the rest.
3. Waypoints and velocities are uniformly randomly generated from the given ranges.
4. Pause times are generated from an exponential distribution with a given mean.
5. Pr[node ∈ interior rectangle] = β^2 under the uniform density hypothesis

Probability of a point chosen from a quadrant:

$$P((x,y)\in Q_I \mid (-a \le x \le a)\wedge(-b \le y \le b))$$

$$\frac{\text{Area of Quadrant I}}{\text{Total Area}} = \frac{2(a-a')b}{4ab} = \frac{1}{2}(1-\beta)$$

$$P((x,y)\in Q_{II} \mid (-a \le x \le a)\wedge(-b \le y \le b))$$

$$\frac{\text{Area of Quadrant II}}{\text{Total Area}} = \frac{2(a+a')b}{4ab} = \frac{1}{2}(1+\beta)$$

Consider now the sequence of two successive waypoints are generated, which are, by definition, independent. In the following text, we will denote as (U_x, U_y) the origin waypoint and as (V_x, V_y) the destination waypoint.

Various Types of Movement

In addition, there will be four types of movement in relation to the quadrants. The first two types of movement are nodes simply moving without leaving quadrant I or without leaving quadrant II. Also, the node may move from quadrant I to quadrant II, and vice versa. Hence, we will denote the probability of selecting each type of movement using the notation of P_i, where i symbolizes each type of the movement.

Probability of selecting each type of movement

$$\mathrm{P}_1 = P((U_x,U_y)\in Q_I)\bullet P((V_x,V_y)\in Q_I)$$
$$=\left(\frac{1}{2}(1-\beta)\right)\bullet\left(\frac{1}{2}(1-\beta)\right)=\frac{1}{4}(1-\beta)^2$$

$$\mathrm{P}_2 = \mathrm{P}((\mathrm{U}_x,\mathrm{U}_y)\in Q_\mathrm{I})\bullet \mathrm{P}((\mathrm{V}_x,\mathrm{V}_y)\in Q_\mathrm{II})$$
$$=\left(\frac{1}{2}(1-\beta)\right)\bullet\left(\frac{1}{2}(1+\beta)\right)=\frac{1}{4}(1-\beta^2)$$

$$\mathrm{P}_3 = P((U_x,U_y)\in Q_{II})\bullet P((V_x,V_y)\in Q_I)$$
$$=\left(\frac{1}{2}(1+\beta)\right)\bullet\left(\frac{1}{2}(1-\beta)\right)=\frac{1}{4}(1-\beta^2)$$

$$\mathrm{P}_4 = P((U_x,U_y)\in Q_{II})\bullet P((V_x,V_y)\in Q_{II})$$
$$=\left(\frac{1}{2}(1+\beta)\right)\bullet\left(\frac{1}{2}(1+\beta)\right)=\frac{1}{4}(1+\beta)^2$$

Expected time of spent in quadrant I (for each specific movement)

Now, we will obtain the expected time of a motion type spent in quadrant I, which is the percentage of time expected to spend in quadrant I. For the four types of movement defined in the above, we will refer E_i as the probability of a node being in quadrant I for each defined (i.e. the type of movement, i, ranges from 1 to 4) movement. In addition, we will denote d' as the expected distance in the movement of quadrant I, d as the expected total distance of the path, V is the expected velocity, and T is the expected pause time. To derive the expected time in quadrant, we use the expected distance of the movement in that quadrant divided by the expected velocity. Because of assumption 2 and 3, for E_2 and E_3, the expected path will be the line, joining from cross diagonals of quadrant I and the cross diagonals of quadrant II. Thus, d' is 1/2(a-a') and d is a. The summary of their derivations is as follows:

$E_1=1$, since the node movement is in quadrant I only.

$$\mathrm{E}_2=\frac{\mathit{Time\ Spent\ in\ Quadrant\ I}}{\mathit{Total\ Time}}=\frac{\mathit{Time\ Spent\ in\ Quadrant\ I}}{\mathit{Expected\ Moving\ Time}+\mathit{Expected\ Pause\ Time}}$$

$$=\frac{\frac{d'}{V}}{\frac{d}{V}+T}=\frac{d'}{d+VT}=\frac{a-a'}{2(a+VT)}$$

$$\mathrm{E}_3=\frac{\text{Time Spent in Quadrant I}}{\text{Total Time}}=\frac{\text{Time Spent in Quadrant I}}{\text{Expected Moving Time}+\text{Expected Pause Time}}$$

$$=\frac{\frac{\mathrm{d'}}{\mathrm{V}}+\mathrm{T}}{\frac{\mathrm{d}}{\mathrm{V}}+\mathrm{T}}=\frac{\mathrm{d'+VT}}{\mathrm{d+VT}}=\frac{\frac{\mathrm{a-a'}}{2}+\mathrm{VT}}{(\mathrm{a+VT})}=\frac{\mathrm{a-a'+2TV}}{2(\mathrm{a+TV})}$$

$E_4=0$, since both waypoints are outside quadrant I.

Expected time of a node spent in quadrant I:

$$P_{Q_I}=\Pr[node\ \in\ Q_I]=\sum_{i=1}^{4}(P_i\bullet E_i)$$

$$=\frac{1}{4}(1-\beta)^2\bullet 1+\frac{1}{4}(1-\beta^2)\bullet\frac{a-a'}{2(a+VT)}+\frac{1}{4}(1-\beta^2)\bullet\frac{a-a'+2TV}{2(a+TV)}+\frac{1}{4}(1+\beta)^2\bullet 0$$

$$=\frac{1}{4}(1-\beta)^2+\frac{1}{4}(1-\beta^2)\left(\frac{a-a'+TV}{a+TV}\right)$$

$$=\frac{1}{4}(1-\beta)\left(2-(1+\beta)\left(\frac{a'}{a+TV}\right)\right)$$

$$\therefore P_{Q_I} = \frac{1}{4}(1-\beta)\left(2-(1+\beta)\left(\frac{a'}{a+TV}\right)\right)$$

Equation 2: Probability Of A Node in Quadrant I (I)

Because of symmetry, it can apply the same reasoning to the remaining three quadrants that can take the role of quadrant I, i.e. Figure 5.b, c, d. Note that the same result applies to Figure 5.c, so let us use the following notation: P_X = Pr [node $\in Q_I$] for Figure 5.a, c, while P_Y = Pr [node $\in Q_I$] for Figure 5.b, d, where P_X and P_Y are identical except for a change in the relevant variables, which give us:

$$P_Y = \frac{1}{4}(1-\beta)\left(2-(1+\beta)\left(\frac{b'}{b+TV_Y}\right)\right)$$

Equation 3: Probability Of A Node in Quadrant I (II)

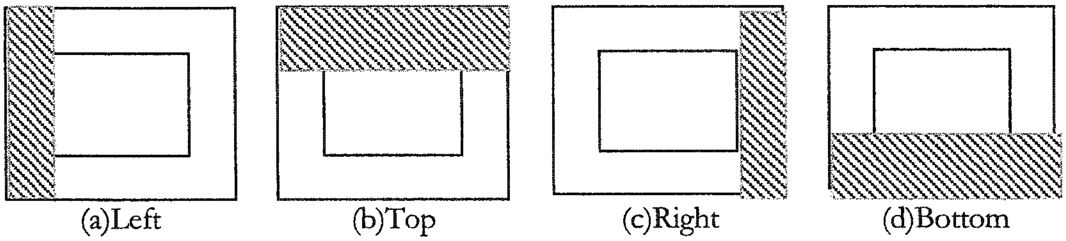


Figure 5: Four setting for Q_I in the shaded area.

The area in the boundary (the difference between interior and exterior rectangle) can be therefore approximated by $2P_X + 2P_Y - 4P_X P_y$ which expresses the reduction necessary because of the doubly summed area at the corners. Now, considering P_X and P_y are independent since they are functions of random variables that we already approximated as independent of the selection of successive waypoints and whose projections on X and Y is approximated as being independent). Hence, the following illustrates the derivation of this approximation, which we will denote it as P_{ext}.

Expected time spent outside the shaded area:

$= 4 \times \Pr[\text{node} \in Q_I] - 4 \times \Pr[\text{node} \in \text{the corner}]$.

$= 4 \times P_{QI} - 4 \times P_{QI} \times P_{QI}$

$= 4 \times P_{QI} \times (1 - P_{QI}) = 4 \times P_{QI} \times P_{QII}$

Hence, we may find the expected time that a node spent outside of the shaded area in Figure 3. Using the above equation, we left with P_{QII}, the expected time for a node spending on the larger rectangle or quadrant. With the similar method to find P_{QI}, Table 1 summarizes the results we obtain, which is $P_{QII} = \frac{1}{4}(1+\beta)\left(2+(1-\beta)\left(\frac{a'}{a+TV}\right)\right)$.

Type (U,V)	Probability of Line P_{type}	Expected time of this line E_{type}	Overall Expected Time $P_T E_T$
1 (Q_I,Q_I)	$\frac{1}{4}(1-\beta)^2$	0	0
2 (Q_I, Q_{II})	$\frac{1}{4}(1-\beta^2)$	$\frac{a-a'+2TV}{2(a+TV)}$	$\frac{1}{4}(1-\beta^2)\left(\frac{a-a'+2TV}{2(a+TV)}\right)$
3 (Q_{II}, Q_I)	$\frac{1}{4}(1-\beta^2)$	$\frac{a-a'}{2(a+TV)}$	$\frac{1}{4}(1-\beta^2)\left(\frac{a-a'}{2(a+TV)}\right)$
4 (Q_{II}, Q_{II})	$\frac{1}{4}(1+\beta)^2$	1	$\frac{1}{4}(1+\beta)^2$
Overall Expected Time $P_{QII} = \sum P_T E_T = \frac{1}{4}(1+\beta)\left(2+(1-\beta)\left(\frac{a'}{a+TV}\right)\right)$			

Table 1 Summary of the derivation of $E(Q_{II})$.

Let X be β, and Y be $\frac{a'}{a+TV}$

$4 \times P_{QI} \times P_{QII}$

$$= 4\left[\frac{1}{4}(1-X)(2-(1+X)Y)\right]\left[1-\frac{1}{4}(1-X)(2-(1+X)Y)\right]$$

$$
\begin{aligned}
&= \frac{1}{4}[(1-X)(2-Y-XY)][4-(1-X)(2-Y-XY)] \\
&= \frac{1}{4}[(1-X)(2-Y-XY)][4-(2-Y-XY-2X+XY+X^2Y)] \\
&= \frac{1}{4}[2-Y-2X+X^2Y][2+Y+2X-X^2Y] \\
&= \frac{1}{4}[2-(Y+2X-X^2Y)][2+(Y+2X-X^2Y)] \\
&= \frac{1}{4}[4-(Y+2X-X^2Y)^2] \\
&= 1-\frac{1}{4}(Y+2X-X^2Y)^2 \\
&= 1-\frac{1}{4}(Y(1-X^2)+2X)^2 \\
&= 1-\frac{1}{4}\left(\frac{a'}{a+TV}(1-\beta^2)+2\beta\right)^2
\end{aligned}
$$

Expected time of a node in the shaded area

The probability of a node being in the shaded area of Figure 3 (interior rectangle) becomes $P_{int} = 1 - 2P_X - 2P_Y + 4P_XP_{y.}$ In the case of rectangular boundaries (a=b, a'=b', $V_X=V_Y=V$), the above expression can be simplified as $P_{int} = 1 - 4P + P^2$, which it gives:

$$
P_{int} = \frac{1}{4}\left(\frac{a'}{a+TV}(1-\beta^2)+2\beta\right)^2
$$

Equation 4: Probability of a node in the interior of a center

To explain that the node density in the interior rectangle is greater than or equal to the uniform density hypothesis, we have to demonstrate the following inequality is hold and finally show that the node density is not uniform.

$$
\begin{gathered}
P_{int} \geq P_{int} \text{ (from the uniform density hypothesis)} \\
\frac{1}{4}(\gamma(1-\beta^2)+2\beta)^2 \geq \beta^2, where \;\; \gamma = \frac{a'}{a+TV} \\
(\gamma(1-\beta^2)+2\beta)^2 \geq 4\beta^2 \\
(\gamma(1-\beta^2)+2\beta)^2 - 4\beta^2 \geq 0
\end{gathered}
$$

$$\left(\left(\gamma\left(1-\beta^2\right)+2\beta\right)+2\beta\right)\left(\left(\gamma\left(1-\beta^2\right)+2\beta\right)-2\beta\right)\geq 0$$

$$\left(\gamma\left(1-\beta^2\right)+4\beta\right)\left(\gamma\left(1-\beta^2\right)\right)\geq 0$$

$$\because \left(\gamma\left(1-\beta^2\right)\right)\geq 0 \qquad \because 0<\beta\leq 1 \ \wedge \ 0<\gamma\leq 1$$

$$\because \left(\gamma\left(1-\beta^2\right)+4\beta\right)\geq 0 \because 0<\beta\leq 1 \ \wedge \ 0<\gamma\leq 1 \ \wedge \ \left(1-\beta^2\right)$$

$$\therefore \mathrm{P}_{\mathrm{int}} \geq \mathrm{P}_{\mathrm{int}} \text{ (from the uniform density hypothesis)}$$

Node density analysis

As we show the density in the interior is always greater than or equal to what is expected under the uniform density hypothesis. Now, we will examine the asymptotic behavior of $\mathrm{P}_{\mathrm{int.}}$ Specifically:

1. $P_{slow} = \lim\limits_{V\to 0} P_{\mathrm{int}} = \lim\limits_{T\to 0} P_{\mathrm{int}} = \frac{1}{4}\left(\beta\left(1-\beta^2\right)+2\beta\right)^2 = \frac{\beta}{4}\left(3-\beta\right)^2$

Equation 5: Limit for low velocities of the probability of node in the interior.

2. $P_{fast} = \lim\limits_{v\to\infty} P_{\mathrm{int}} = \frac{1}{4}\left(0+2\beta\right)^2 = \beta^2$

Equation 6: Limits for high velocities of the probability of node in the interior.

The first limit, $\mathrm{P}_{\mathrm{slow}}$, suggests that slow moving mobiles, or nodes do not pause for any amount of time, the probability of nodes being in the interior square is $\frac{\beta}{4}(3-\beta)^2$, which is larger than the uniform density case (top line of Figure 6). The second limit indicates that only extremely fast moving mobiles result in scenarios that confirm the uniform hypothesis. Note that we do not seek to find a limit for a pause time to infinity since it only suggests that nodes remain in their location forever.

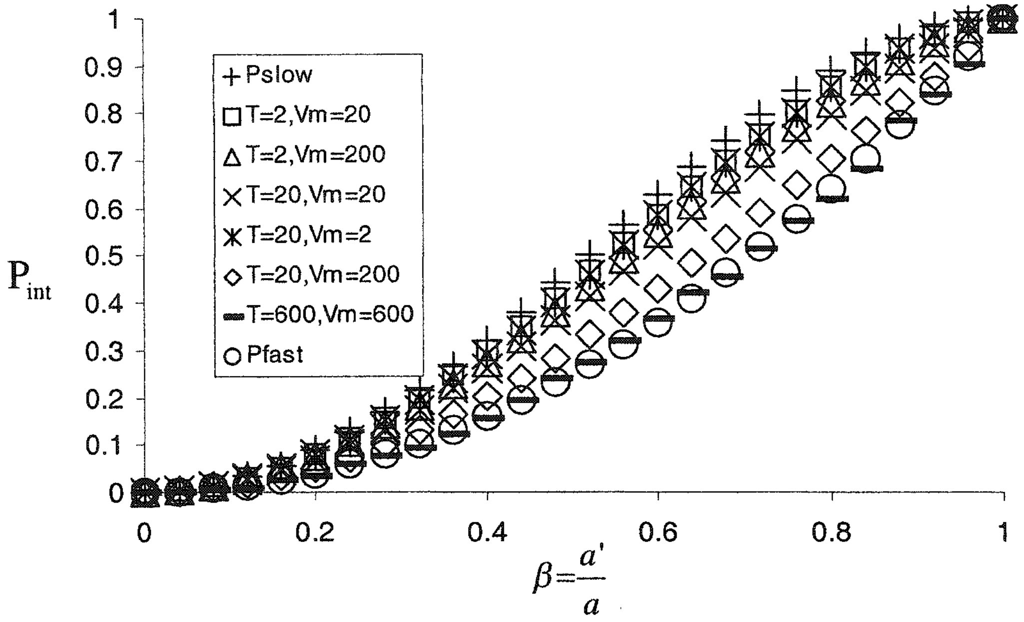


Figure 6: Density of interior rectangle as a function of $\beta=\frac{a'}{a}$ for the two asymptotic cases and for assortment of simulation with various maximum velocities, Vm, and average pause times T.

Figure 6 compares the results from the analysis with those RWP model simulations within a boundary of 500 x 500 square unit. The probability that the mobile is in the interior rectangle is bounded between P_{slow} and $P_{fast,}$ and it confirms with our analysis that the curve tends to P_{fast} when the speed is higher. With speed of 600 units per unit of time, it reaches asymptotically the uniform density P_{fast}. Obviously 600 distance units per unit of time is enough for a mobile to cover most distances within the square in less than a unit of time (the longest distance is approx. 700 from the diagonal). In most MANET's literature, meters per second are the unit of speed, and typically, the maximum speed is constrained within a narrower range, such as 2m/s as lower speed and 20m/s as higher speed. However, our simulation shows that the node density is not uniform even with velocities between 0 and 200 m/s. Besides, our analysis is unable to distinguish between configurations whose Vm x T is the same. As we may find from Figure 6, one set of result (Vm=2 and T=200 and Vm=20 and T=20 gives the products of Vm and T of 400) indicates that they are not entirely the same. With a higher pause time, it will shift the curve closer to the result of the uniform density.

Connectivity Under Non Uniform Node Density of the RWP Model

The result of non-uniform node density, at first appears to confirm our suspicion that there is a velocity-related artifact in the node density of RWP. Looking closer, the results still contradict intuition. If the density is higher in the interior of the rectangle at lower speeds, shouldn't connectivity be also higher at lower speeds? To explain the observed behavior, we must consider the dynamics of connectivity that are influenced not only by the density of the nodes but by their average speed as well.

At low speeds a node that is near the boundary will remain there for a longer period of time, compared to a higher velocity scenario. If the graph was not connected to begin with due to nodes that remain close to the boundary and away from the rest, then it will stay so for a *longer* period of time if the speed of the nodes, and hence of the nodes near the boundary, is low. Vice versa, at high speeds, the nodes near the boundary can "escape" faster. Thus, at high speeds, both more uniform density and higher connectivity is achieved.

In other words, at low speeds, a node near the boundary will remain there for a long period of time. If the graph is not connected because this mobile is far away from the rest, it reduces the connectivity if the speed is low. In contrast, at high speed, nodes near the boundary can get away faster, but by the same token, they do not stay near the center for longer periods of time. It turns out that as far as performance of connectivity is concerned, it is more important to be able to quickly move away from the isolation of being close to an edge of the rectangle boundary. In the rest of this chapter, we will study some of the artifacts that relate to the connectivity of the underlying graph.

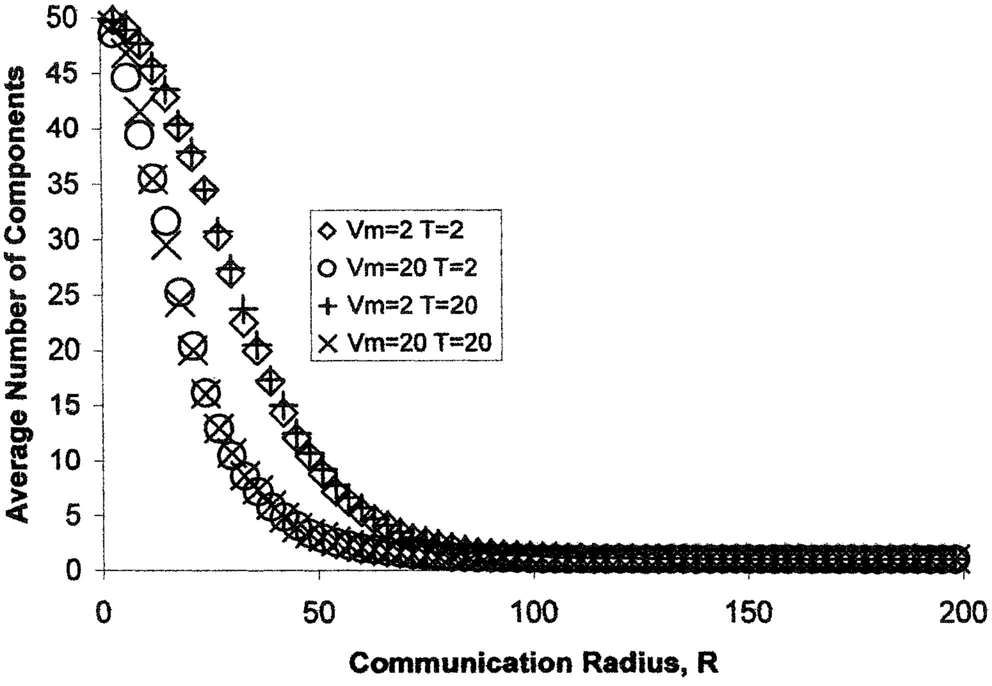
Connected Components: Sizes and Lifetimes

The lack of connectivity can be further elaborated upon by studying the average number of (connected) components that are present in the underlying connectivity graph. If our speculation that the nodes in the periphery are less connected is true, we would expect that in low speed simulations the average number of "small" clusters (clusters = groups of nodes forming a connected component) of nodes (existing mostly at the periphery of the rectangle) is larger than the average number of clusters at high speeds. .

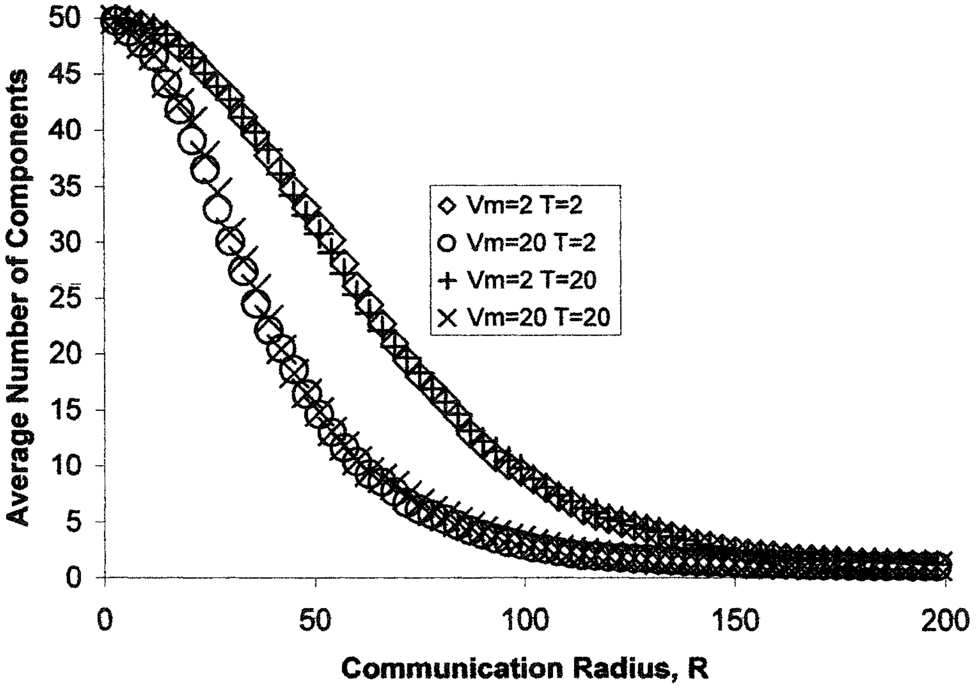
Figure 7 plots the time average of the number of components for the same simulation parameters as before and confirms our expectations. Clearly, a longer periphery, as well as area, result in increased number of connected components. A larger area (in the case 1500x500) which dilutes the density of the nodes together with the fact that there is a longer periphery near where slow moving nodes could be `"trapped". For the same transmission radius, the number of components at low speeds is indeed larger than at high speeds. As the transmission radius increases, the connected components merge together towards a single connected component. What is known as critical radius, i.e., the smallest radius that allows connectivity is highly dependent on the speed of the mobiles and the particular boundary.

The observation is not particularly comforting, as far as engineering a system is concerned. Even if we have a fairly good estimation of the speed of the users that we can use in our models, we can still not pinpoint the proper critical radius with the aid of simulations. We need a good understanding of the density of users. RWP is neither a good baseline case (no uniformity) nor expressive enough (e.g., capable of representing region--dependent densities).

One is tempted to interpret connectivity results as related to the 6--neighbor principle of the seminal paper by Kleinrock and Slivester. However, we do not have a uniform density throughout the modeled 2-D space, whereas we are interested to know how far from this uniform density assumption the RWP model is, noting that even in the case of uniform density inside the rectangle, the boundary nodes may possess fewer, on average, than six neighbors and still be connected, because of the fact that most other nodes are in the interior and none across the rectangle boundary. Hence, the impact of the boundaries is non-trivial especially when its interaction with the speed of the mobiles is considered.

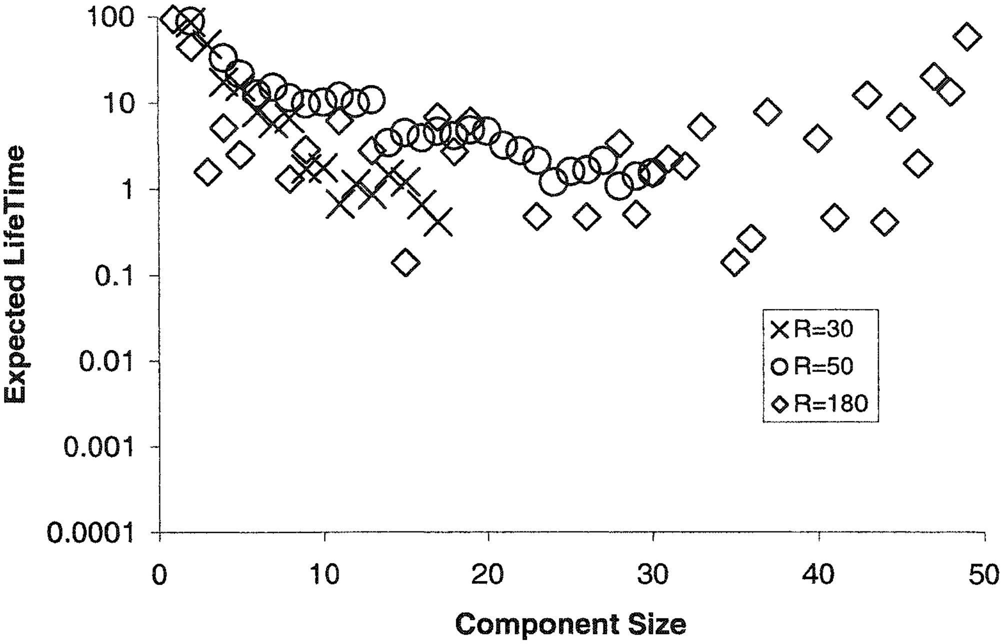


(a) MaxX = 500, MaxY = 500

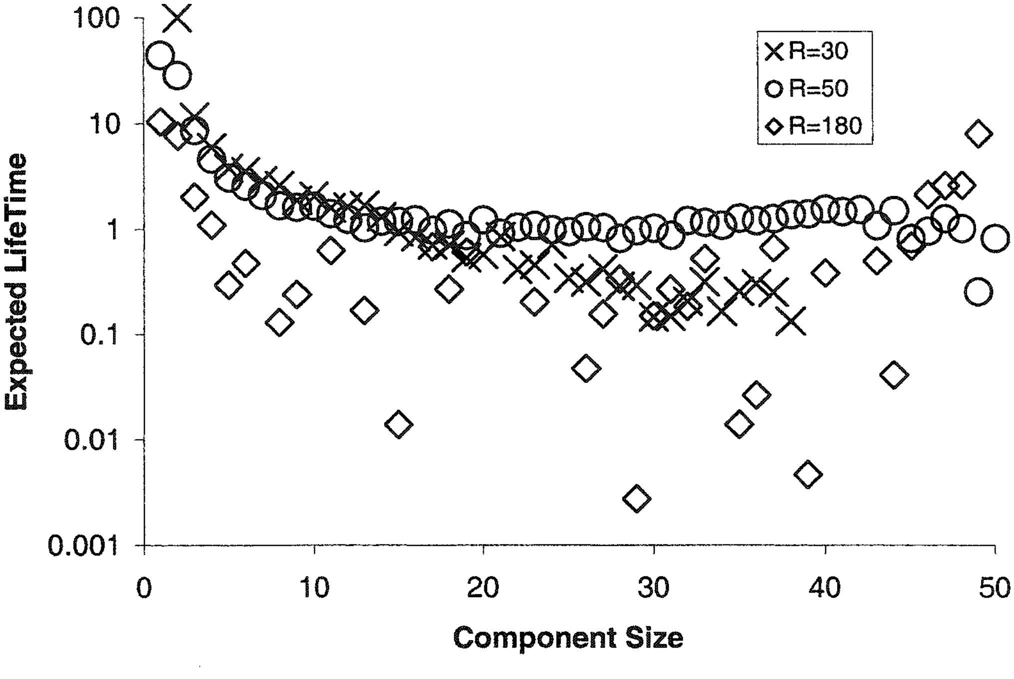


(b) MaxX=500, MaxY=1500

Figure 7: Average number of connected components (in the time average sense) corresponding to the connectivity results of Figure 2.



(a) Vm = 2



(b) Vm = 20

Figure 8: Average lifetime of connected components in two scenarios, with (a) Vm=2 and (b) Vm=20. Both senarios are sinumlated with a 500 x1500 rectangle, 50 nodes, average pause time is 2s, and three different transmission radii: 30, 150 and 180 units.

With regard to routing algorithms though, an important aspect is whether the connected components are fairly long--lived, as one can hope that the source and destination within the same component will be able to maintain a path, even though the entire graph may not be connected. To discover whether connected components persist, we conducted a series of simulations for a specific RWP boundary of 1500x500 and two different speeds of 2 and 20 meters per second. The radius of transmission was set to 30, 140 and 180 meters. The distances represent, respectively, a subcritical radius for both speeds, a near-critical radius for high speeds but subcritical for low speeds, and, eventually, a critical radius for high speeds but near-critical for low speeds. The results are summarized in Figure 8.

The first, and most impressive, conclusion from these simulations is that it is extremely unlikely we will encounter medium--sized components. As Figure 8 clearly indicates (note the logarithmic scale on the y--axis), for 50 nodes, it is unlikely that we will encounter components of sizes that are larger than 10 nodes and less than 40, because, when formed, they last only for a short time. What is more fascinating is that the observed property does *not* depend on the transmission radius.

When medium--sized components are formed, they usually last for a very short period of time, regardless of the transmission radius. If we also consider the results presented in Figure 7.b, where a large number of components exists for small transmission radii, we are left with one essential interpretation: when the underlying graph is not connected, the graph includes a large number of very small components. These small components are frequently isolated nodes or pairs of nodes. In addition to the multiple small components, there *may* exist a large component *if* the transmission radius allows its formation. For example, for R=140 and R=180, the large component exists, while for R=30 it does not. Again, this property is independent of the speed.

Looking close into Figure 8 we discover two more interesting properties. First, the distribution of the lifetimes is, in essence, bimodal. The two modes are in the small (less than 10) and large (more than 40) number of nodes per component. The transmission radius, R, regulates where the mass of the distribution falls: the lower or the higher end. For small (sub-critical) radii, the mass falls naturally at the low values of the number of nodes.

For near-critical radii, the mass seems to be split between the large and very small component sizes. When the radius moves to critical, and subsequently, super-critical value, the bulk of the density is shifted, to the large component sizes. Eventually, if we keep increasing the radius, the entire mass will migrate to the largest possible connected component: the entire graph. The behavior suggests that a form of the *giant component* often associated with properties of random graphs [19] present for near-critical to super-critical radii in the dynamic topology graphs of MANETs under the RWP model.

A final observation drawn by Figure 8 is that the higher speed leads to a shorter lifetime of the small components. This is by far the clearest evidence to suggest that it is the mobility of nodes near the boundaries that matters the most and provides high speed mobility scenarios with a better connectivity behavior.

Implications on Routing Algorithm Evaluation

We can look into the above results from the perspective of a routing algorithm, and its objectives. In MANETs, the primary objective of routing is to determine a path between a source and a destination. A secondary objective is, if several paths exist, to determine the *least cost* path. Quick location/construction of a path is more important than optimality because the flooding algorithms associated with the discovery of paths, are expensive in terms of message overhead. Hence, caching of partial paths [20] is used as the means to avoid a recurring discovery cost. However, no guarantee exists whether these cached paths are of any value and, if they are, for how long.

What we understand now by the connectivity measurements is that if a node is isolated in a small component, according to RWP, it will remain there for some time. A higher speed is helpful in reducing the lifetime of such small components, but is not sufficient, since the transmission radius also controls the distribution of which component sizes will be more prevalent. In order for cached paths from a previous path discovery operation to be valuable, nodes adjacent in the cached path must still be adjacent when the path is examined. The intuition is that at high speeds, this may not be as likely to happen as it is at low speeds.

A couple more possibilities exist. First, if the path to the destination does not currently exist, it will probably not exist in the near future. This is because either the source or the destination was isolated into a separate component. How much time should be expended into trying cached paths should be weighed against what is the typical lifetime of the components. In the case of high speeds, it is likely that a new path will exist in a few seconds to the destination, *but* it's likely not going to resemble any of the paths that were discovered and cached during the last discovery phase because all intermediate nodes move quickly. In the case of low speeds, it is likely that the component will exist for some time, rendering pointless successive attempts to use cached paths pointless. However, if and when connectivity is established again, the cached paths may still be feasible (since the nodes did not move that far apart).

Summary

Despite the fact that we have been unable to determine the way in which a single global transmission radius for an entire MANET can provide both immunity to mobility and minimum energy cost, we discovered along the way the importance of, and certain artifacts due to, the particular mobility model, RWP. What gets in the way of formulating a single radius for the entire MANET is a velocity-dependent non-uniformity of the mobile node density in RWP. We subsequently made several observations regarding the connectivity of the underlying graph. The observations also have implications on the quality and effectiveness of the paths that MANET routing algorithms produce. Rather than propose what would be the ideal algorithm for MANET routing in a RWP model, we have to accept that RWP is a synthetic model, and optimizing routing algorithm designs to fit a synthetic model is not necessarily a good idea. However, our work opens the discussion on whether it would be much more accurate to compare MANET routing algorithms with respect to the underlying graph properties rather than the "macro" parameters of average speed and boundary dimensions. The primary quantitative expression of the underlying graph is the connectivity, as the fraction of time that the network is connected. Another, equally important metric is the distribution of the component sizes, and their relative lifetimes. Routing algorithms can then be seen as being able to correctly detect whether a partition of components has occurred and what this event means to the paths (partial or complete) that have been already calculated. We plan to continue the examination of MANET routing algorithms along these lines with the long term objective of establishing quantitative results that link MANET routing to random graph theory.

Chapter 4

Energy Efficient Broadcast

Energy Efficient Broadcast in Static Wireless Networks

As shown in chapter 2, the wireless property brings about two special features: (a) the ability of a spatial reuse of the same frequency spectrum for the communication between different nodes at the same time when they are placed sufficiently apart to avoid interference, and (b) the inherent underlying broadcast paradigm, where a single transmission can be simultaneously received by a number of receivers in the vicinity of the transmitter. Broadcasting is an essential building block for protocols that utilize a form of discovery process, as in the case of route discovery/advertisement [21].

In this chapter, we are working on developing algorithms for Minimum Energy Broadcast (MEB) in MANETs. Although similar work has been presented in [16], the results were reported for static wireless networks (a MANET without mobility) only. We take special interest in the case of mobile networks, and MANETs in particular. In principle, one could cope with the changing distances between nodes by applying the previously reported algorithms continuously. However, the complexity of such algorithms as well as the fact that they require substantial communication between nodes renders them ineffective for continuous execution. Instead, we consider that they will run only occasionally (specifically: periodically) where the rest of the time simple "localized" algorithms are taking care of the broadcast tree re-arrangements. In essence, we study energy minimization techniques for wireless broadcast in MANETs by considering a mixture of (a) a periodically invoked tree construction algorithm and (b) real-time on-demand re-arrangement algorithms (between the tree construction instants).

Hence, we will first present three previously proposed algorithms, and two additional algorithms of our design for the tree construction, as well as three new schemes for tree re-arrangement to compensate for mobility. In presenting the tree construction algorithms, we note that although they are frequently expressed (as we will also do here) in an edge-wise fashion, the underlying costs are produced subject to the loss exponent α and the wireless advantage. That is, the cost of the constructed tree is shown in

Equation 7, where R is the set of the relays (including the source), C_i, for a relay i is the set of children the relay is supposed to cover (the covered children are the ones whose edge has been selected as per the corresponding algorithm), and r_{ij} is the communication radius from node i to node j.

$$\sum_{i\in R}\left(\max_{j\in C_i}\left(r_{ij}\right)\right)^{\alpha}$$

Equation 7: Total Energy Cost for Broadcast Tree Construction.

It is the minimization of the cost in the above equation that we seek. Abstracting the problem, what we would like to determine is nodes that will act as relays and the transmission power of each relay so that all the nodes in the network will receive, at least once, the transmission. The broadcast energy minimization problem introduces the objective that the sum of the energy expended in the relay transmissions (over all the relays) is minimized. The problem has been shown to be a computationally hard problem [18]. Given the need for a tree as the by-product of construction process, it is not strange that previously proposed algorithms, [16], are related to Minimum Spanning Tree (MST) construction, namely:

Broadcast Link-Based MST (BLiMST)

```
BLiMST( Input   : C(V,E): Adjacency Matrix, P: Power Consumption Matrix , S: Source of the
                  tree
        Output : G($V'$,$E'$): Broadcast Tree)

    i, j, u, v :        Temporary variables to store the node number
    min:                Temporary  variable to store the minimum energy
    R :                 Boolean array of size |V | indicates whether node u is being reached.
    AllReached( R ) : A Function return true if all the value of  boolean array R  is true, false
                        otherwise.

    for $\forall$ u $\in$ V
      do R[ u ] $\leftarrow$ true
    R[ S ] $\leftarrow$ true
    $V' \leftarrow V' \cup$ S
    while(!allReached(R))
        do    i $\leftarrow \infty$
              j $\leftarrow \infty$
          min $\leftarrow \infty$
```

```
for $\forall u \in V$
   for $\forall v \in V$
      if ( $\exists (u,v) \in E \mid R[u] \wedge !R[v] \wedge P[u][v] < min$ )
         $i \leftarrow u$
         $j \leftarrow v$
         $min \leftarrow P[u][v]$
$E' \leftarrow E' \cup (i,j)$
$V' \leftarrow V' \cup \{i\} \cup \{j\}$;
$R[j] \leftarrow$ true
```

Psuedocode 1: Broadcast Link-Based MST (BLiMST).

BLiMST is essentially the Minimum Spanning Tree (MST) algorithm applied to a completely connected graph where the cost of each edge is set to r_{ij}^{α}. One such algorithm is Prim's algorithm, which exhibits polynomial time complexity ($O(N^3)$, i.e., if there are N nodes participating in the network, the algorithms employs N^3 ordered number of steps to create the broadcast tree.). In addition, BLiMST does not exploit the wireless advantage, and hence its results are not particularly impressive in the case of static wireless networks [16]. However, the advantage of BLiMST is that it can be implemented in a distributed manner, and the constructed tree will remain the same as long as the wireless links are maintained.

Broadcast Least-Unicast-cost (BLU)

```
BLU ( Input : C(V,E): Adjacency Matrix, P: Power Consumption Matrix , S: Source of the tree
      Output : G(V',E'): Broadcast Tree)

   v : Temporary variable to store a node number.
   ShortestPath(C(V,E), P, S, v, Path) : A function to compute the shortest path. It
                     takes 4 inputs - the adjacency matrix indicates direction between
                     nodes, the power consumption matrix indicates the cost, the
                     source node, and the destination node respectively. The result
                     will be generated with a set of edges that can create a path, which
                     it takes minimal cost from the source to destination node.
   for $\forall v \in V$
      do ShortestPath(C(V,E), P, S, v, Path)
         $E' \leftarrow E' \cup$ Path
```

Psuedocode 2: Broadcast Least-Unicast-cost (BLU).

BLU is the union of all the shortest paths to all the nodes starting from the source. The shortest path in this context takes the energy as the cost, i.e., each edge will contain a cost of r_{ij}^{α}. When the topology changes due to mobility, the algorithm also guarantees to find the shortest energy path to each destination; nevertheless, BLU is also unable to exploit the wireless advantage to minimize the overall energy consumption.

Broadcast Incremental Power (BIP)

```
BIP( Input  : C(V,E): Adjacency Matrix, P: Power Consumption Matrix , S: Source of the tree
     Output : G($V'$,$E'$): Broadcast Tree)

   i, j, u, v :       Temporary variables to store the node number.
   min:               Temporary  variable to store the minimum energy.
   R :                Boolean array of size |V | indicates whether node u is being reached.
   CP:                An array of size  |V | indicates the current transmission power.
   AllReached( R ) :  A Function return true if all the value of  boolean array R  is true, false
                      otherwise.

   for $\forall u \in V$
     do R[ u ] $\leftarrow$ true
        CP[ u ] $\leftarrow$ 0
   R[ S ] $\leftarrow$ true
   $V' \leftarrow V' \cup S$
   while(!allReached(R))
      do    i $\leftarrow \infty$
            j $\leftarrow \infty$
          min $\leftarrow \infty$
          for $\forall u \in V$
             for $\forall v \in V$
                if ( $\exists (u,v) \in E \mid R[u] \wedge !R[v] \wedge P[u][v] - CP[u] < min$ )
                   i $\leftarrow$ u
                   j $\leftarrow$ v
                min $\leftarrow$ P[ u ][ v ]  - CP [ u ]
          $E' \leftarrow E' \cup (i, j)$
          $V' \leftarrow V' \cup \{ i \} \cup \{ j \}$;
          R[ j ] $\leftarrow$ true
          CP[ i ] $\leftarrow$ max( CP[ i ], P[ i ][ j ])
```

Pseudocode 3: Broadcast Incremental Power (BIP).

BIP is the first algorithm studied to exploit the wireless broadcast advantage. It proceeds in the same fashion as Prim's MST algorithm, however the costs involved at each step are the *incremental* costs to reach off tree vertices from vertices already in the tree. For a node in the tree that is not yet a relay, the incremental cost represents the cost of making this node a relay and transmitting at enough power to reach the off--the--tree node. For a node that is already a relay, the incremental cost represents the cost to modify (extend) the range of the transmission to include the off--the--tree node. Hence, the edge selected each time is the one that requires the minimum energy increase.

The comparison of BIP, BLU and BLiMST indicated, as it would be expected, clear advantage for BIP in [16]. However, the fact that the results were restricted to static networks does not provide us with evidence of how they will perform on a mobile network. As we note earlier, the objectives of energy efficiency are somewhat antithetical to the existence of multiple paths, and hence the viability of the underlying topology to remain connected subject to mobility.

In addition to the three above, we consider two additional simple tree construction algorithms:

Incremental Search (IS)

```
IS( Input   : C(V,E): Adjacency Matrix, P: Power Consumption Matrix , S: Source of the tree
    Output : G(V',E'): Broadcast Tree)

    i, j, u, v :          Temporary variables to store the node number.
    min:                  Temporary  variable to store the minimum energy.
    CP:                   An array of size |V | indicates the current transmission power.
    AllReached( R ) : A Function return true if all the value of  boolean array R  is true, false
                          otherwise.
    L:                    An array of size |V | indicate the current level of a node i pointing to
                          SC[ i ].
    SC:                   A matrix of size |V |x |V|, each row store the power level and the
                          destination for a node.
    SortIncreasing(A): A function to sort the array A in an increasing order of the cost.
    AllReachableFrom(S,G ) : A Function return true if S can reach all other nodes in the graph
                          G, false otherwise.

    for ∀ u ∈ V
      for ∀ v ∈ V
        do SC[ u ][ v ]. cost ← P[ u ] [ v ]
           SC[ u ][ v ]. destination ← v
    for ∀ u ∈ V
      do SortIncreasing(SC[ u ])
         L[ u ] ← 1
         CP[ u ] ← 1
    while( ! allReachableFrom( S, G) )
      do  min ← ∞
          i ← ∞
          j ← ∞
```

```
for $\forall u \in V$
  do if( $min < SC[u][L[u] + 1].cost - CP[u]$ )
      then $min \leftarrow SC[u][L[u] + 1].cost - CP[u]$
           $i \leftarrow u$
           $j \leftarrow SC[u][L[u] + 1].destination$
if( $i = \infty$ )
  then $E \leftarrow \varnothing$ and return
else
  $E \leftarrow E \cup (i, j)$
  $L[i] \leftarrow L[i] + 1$
```

Psuedocode 4: Incremental Search (IS).

Following in the steps of BIP, IS is a modified MST algorithm that admits edges that offer the minimum additional cost (in terms of transmission radius adjustment). Initially all nodes are assumed to be relays but with a radius of zero (A node is considered a leaf, if at the end of the algorithm its transmission radius is still zero) and subsequently edges are added that would result in the minimum incremental cost. In this sense, the algorithm is a version of Kruskal's MST whereby a forest of trees eventually is connected to one single tree by adding an edge at a time. The difference in the case of IS is that cycles are not a concern because each node require to transmit one time only, and that the costs of edges at any iteration represent the additional cost on top of the cost already expended. The algorithm terminates when there exists a path from the root to all other nodes.

Decremental Search (DS)

```
DS ( Input  : C(V,E): Adjacency Matrix, P: Power Consumption Matrix , S: Source of the
                tree
     Output : G($V'$,$E'$): Broadcast Tree)

   CP:              An array of size $|V|$ indicates the current transmission power.
   SC:              A matrix of size $|V| \times |V|$, each row store the power level and the
                      destination for a node.
   L:               An array of size $|V|$ indicates the current level of a node i pointing to
                      $SC[i]$.
   valid:           An array of size $|V|$ indicates the source node is being checked.
   i, j, u, v :     Temporary variables to store the node number.
   max:             Temporary variable to store the maximal energy.
   SortDecreasing(A): A function to sort the array A in a decreasing order of the cost.
   AnyValid(A):     A function return true if any truth value in A is true, false otherwise.
   AllReachableFrom(S,G ) : A Function return true if S can reach all other nodes in the graph
                      G, false otherwise.
```

```
for $\forall$ u $\in$ V
   do for $\forall$ v $\in$ V
      do SC[ u ][ v ].cost $\leftarrow$ P[ u ][ v ]
         SC[ u ][ v ].destination $\leftarrow$ v
G $\leftarrow$ C
for $\forall$ u $\in$ V
   do SortDecreasing(SC[ u ])
valid[ u ] $\leftarrow$ true
L[ u ] $\leftarrow$ 0
CP[ u ].cost $\leftarrow$ SC[u][ L[ u ] ]
while ( AnyValid(valid ) )
   do  i $\leftarrow \infty$, j$\leftarrow \infty$ , max $\leftarrow$ 0
       for $\forall$ u $\in$ V
          do if ( valid[u]$\wedge$ max < CP[ u ] -SC[ u ][L[ u ].cost + 1 ] )
               then  i $\leftarrow$ u
                     j $\leftarrow$ SC[ u ][L[ u ] + 1 ].destination
                     max $\leftarrow$ CP[ u ] -SC[ u ][L[ u ] + 1 ]
       if (i = $\infty$ $\vee$ j= $\infty$ )
        then  E $\leftarrow\varnothing$ $\wedge$ return
        else G' $\leftarrow$ G \( i , j )
             if ( AllReachableFrom(S,G') )
               then  G $\leftarrow$ G'
                     L[u] $\leftarrow$ L[ u ] + 1
               else  valid[ u ] $\leftarrow$ false
```

Psuedocode 5: Decremental Search (DS).

The approach of DS is almost the inverse of IS, in that we assume that all nodes are relays and initially transmit at the maximum radius (the one reaching the furthest node). This allows us to start with a completely connected graph, and work along the lines of eliminating edges (hence restricting the transmission radius) of the relays. In every iteration, we remove the edge whose removal would result in the largest cost savings. This top--down approach continues until we discover that any further edge removal would result in a disconnected graph, i.e., there would no longer be a path from the source to one or more of the remaining nodes.

Broadcast Tree Rearrangement Schemes

The contribution of this chapter comes out of the interaction of the tree construction algorithms, and the tree maintenance algorithms. Indeed the algorithms for tree construction exhibit complexity of $O(N^2)$, or worse. In a mobile environment where decisions have to be taken almost in real-time, re-running the tree construction algorithms is inefficient. We therefore consider the occasional run of tree construction algorithms, with tree maintenance algorithms applied between tree construction epochs. The principle will be to try to "repair" the tree as it becomes unusable (when a node does not receive an acknowledgment from the destination by the timeout) due to the node mobility by performing low-complexity "local" actions. Hopefully, the resulting energy efficiency of the constructed tree is maintained by the repairs until the next tree construction instant. Specifically, we will consider three approaches to tree maintenance:

Naïve Scheme

```
Naïve [for node i ]
    CP:                   An array of size | V | storing the power level that can reach each node.
    SortIncreasing(A): A function to sort the array A in an increasing order of the cost.
    AquireStatus(Nodes,Power): Sending messages to all the nodes in the Nodes list using
                          Power as the transmission power.
    RecieveAck(Nodes ,TIME):    The function return true node I receive the
                          acknowledgmentsfrom all nodes in Nodes by TIME.

    SortIncreasing(CP[ i ])
    while(true)
        do AquireStatus(Children, CP[ L ]);
           if( ! RecieveAck(Children ,TIMEOUT))
              then L ← L +1
                   E ← CP[ L ]
```

Psuedocode 6: Naïve Update Scheme.

As the name suggests, this scheme is meant to provide a very basic approach to maintain the tree without any clever operations. Namely, even if the nodes move, we will attempt to maintain the tree exactly as it was before. That is, the same children will have to be covered by the same parent. Due to the mobility, much more efficient ways may exist to reach a node that has moved away from the area covered by its relay (possibly by another relay) but in the naïve scheme we chose to ignore this case for a quicker response. That is, the relays adjust continuously their power to reach *at all times* their initial children.

Re-parenting re-arrangement Scheme

```
Re-parenting [ for node i ]
    CP:             An array of size |V | storing the power level that can reach each node
    Children:       An Boolean array of size |V| indicates which node is node i 's child.
    SortIncreasing(A): A function to sort the array A in an increasing order of the cost.
    AquireStatus(Nodes,Power): Sending messages to all the nodes in the Nodes list using
                    Power as the transmission power.
    RecieveAck(Nodes ,TIME):   The function return true node I receive the
                    acknowledgmentsfrom all nodes in Nodes by TIME.
    EnquireParent( Node, Parent): Send a message to the parent and enquire the incremental
                    power need to reach the Node.
    ReceiveCost(Parent, TIMEOUT, IncrementalCost): Wait for the parent's control message
                    indicates the power need to increase.
    ReParent(Parent, Node, Power):  Send a message to Parent and indicates that Node
                    becomes their children by increase its power  by Power more.

    SortIncreasing(CP[ i ])
    while(true)
      do AquireStatus(Childen, CP[ L ]);
          if( ! RecieveAck(Childen[ i ] ,TIMEOUT))
             then EnquireParent( Childen[ i ], Parent)
                     ReceiveCost(Parent, TIMEOUT, IncrementalCost)
         if( CP [ L+1] – CP [ L ] > IncreamentalCost)
            then L ← L +1
                 E ← CP[ L ]
         else Reparent(Parent,Children[ i ], IncrementalCost)
                 Children \ Children[ i ]
```

Psuedocode 7: Re-parenting Update Scheme.

In re-parenting, the search is limited to the relay covering the node (i.e., the parent of the node) and the immediate parent of the relay (i.e., the grandparent of the node). Between the parent and the grandparent, we select the one that results in the least incremental cost. One could consider looking into children nodes also (siblings of the node) and suspend the node from them, but such operation could eventually change the number of relays. That is, "re-parenting" maintains the number of relays the same (or less -- in case a relay identifies no children assigned to it, it ceases acting as a relay until a child re--appears). This scheme exemplifies a "local" re-arrangement of small computational (and protocol) cost. A node needs to know only its relay (parent) and relay's relay (grandparent) node. The intuition is that if a node has moved away from its parent relay, (or the other way round) there is a chance to maintain the wireless broadcast advantage.

Roaming Leaf Scheme

```
RoamingLeaf [ for node i ]
    CP:              An array of size | V | storing the power level that can reach each node.
    Children:        An Boolean array of size |V| indicates which node is node i 's child.
    SortIncreasing(A): A function to sort the array A in an increasing order of the cost.
    AquireStatus(Nodes,Power): Sending messages to all the nodes in the Nodes list using
                     Power as the transmission power to check if it is still reachable.
    RecieveAck(Nodes ,TIME):   The function return true node I receive the
                     acknowledgmentsfrom all nodes in Nodes by TIME.
    EnquireNonLeafNode(Children[ i ]): Send a message to all the non-leaf node and enquire
                     the incremental power need to reach the Node.
    ReceiveCost(NonLeafNode, TIMEOUT, IncrementalCost): Wait for the non leaf node's
                     control message indicates the power need to increase to reach that child.
    Roaming(N, N1): Send a message to N and indicates that N1 becomes its child.

    SortIncrease(CP[ i ])
    while(true)
        do AquireStatus(Childen, CP[ L ]);
           if( ! RecieveAck(Childen[ i ] ,TIMEOUT))
             then EnquireNonLeafNode( Childen[ i ])
                  min ← ∞
                  newParent ← ∞
                  while( ReceiveCost(NonLeafNode, TIMEOUT, IncrementalCost))
                        do if ( IncrementalCost < min )
                             then min ← IncrementalCost
                                  newParent ← NonLeafNode
                           if( CP [L+1] – CP [ L ] > min)
                             then L ← L +1
                                  E ← CP[ L ]
                             else Roaming(NonLeafNode, Children[ i ])
                                  Children \ Children[ i ]
```

Psuedocode 8: Roaming Leaf Update Scheme.

In the Roaming Leaf scheme, a leaf node that moves out of its relay's coverage, determines the incremental cost from all possible relays and attaches itself to the relay with the minimum incremental cost. That is, a scheme that provides freedom to leaf nodes determines their most energy-efficient relay. However, relay nodes are re-arranged in accordance with the re-parenting scheme.

All three tree maintenance approaches can operate in the distributed fashion so that they can be implemented or embedded in a routing protocol, and they are summarized in the above psuedocode. We introduce the power level array P[i], which is in increasing order of possible transmission power levels, and we assume nodes will discover their possible power levels throughout the broadcast tree construction process. Another view of this power level array is the number of nodes that each power level can cover. Employing a lower power level, nodes will establish fewer wireless links, i.e., to fewer neighbors. Thus, P[i] covers less or equal number of nodes as P[i+1]. On the other hand, we will expect the naïve scheme will not operate as well as the re-parenting and roaming leaf rearrangement schemes. This is because the naïve scheme does not exploit the wireless broadcast advantages, and it relies on the "optimal" solution generated by the algorithm of creating MEB tree.

Evaluation

In this section we consider the simulation-based evaluation of the tree construction and tree maintenance schemes under the assumption of a Random Waypoint (RWP) model. The RWP model consists of 50 nodes moving in a 500 x 500 meters square area. The pause periods are exponentially distributed with a mean of 5 seconds. When moving, the velocity is selected in a uniform random fashion between 0m/s and 5m/s (for slow mobiles) and 0m/s and 50m/s (for fast mobiles). The simulations consist of mobiles that are either all slow or all fast. In the evaluation of the above algorithms, the source will be selected randomly. Since the RWP does not assign any particular role to any specific node, we expect that the selection of an arbitrary node as the source will have no effect in the results.

We set the execution of the tree update every 180 seconds (3 minutes). Assuming the naïve re-arrangement scheme, Figure 9 captures the energy cost as a function of time for an example part of the simulation (between simulated time 360 and 1440 seconds). All mobiles are slow moving (velocities in the range from 0 to 5 meters/sec). The first interesting property is that BIP, BLU and BLiMST are almost identical. Despite the fact that they produce different results on a static network, with BIP being the best, any effort to maintain the trees results in an increase of the energy cost by orders of magnitude (note that the y axis scale is logarithmic). Any advantage of these three schemes disappears quickly. Indeed, the fragility of the network comes from mobility and the fact that nodes use the near minimal energy to reach their children. Thus, BIP, BLU and BLiMST result in worse performance than what is achievable (in terms of connectivity as we remarked earlier) with the very simple-minded DS algorithm.

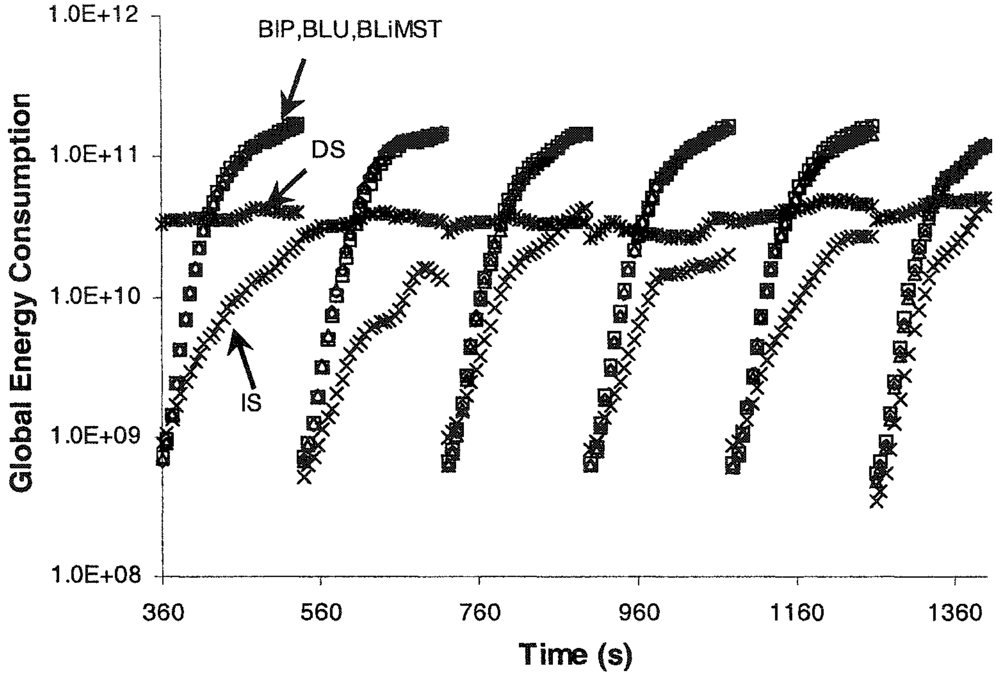


Figure 9: Naïve re-arrangemenet (5m/sec), period of 180s update the tree construction.

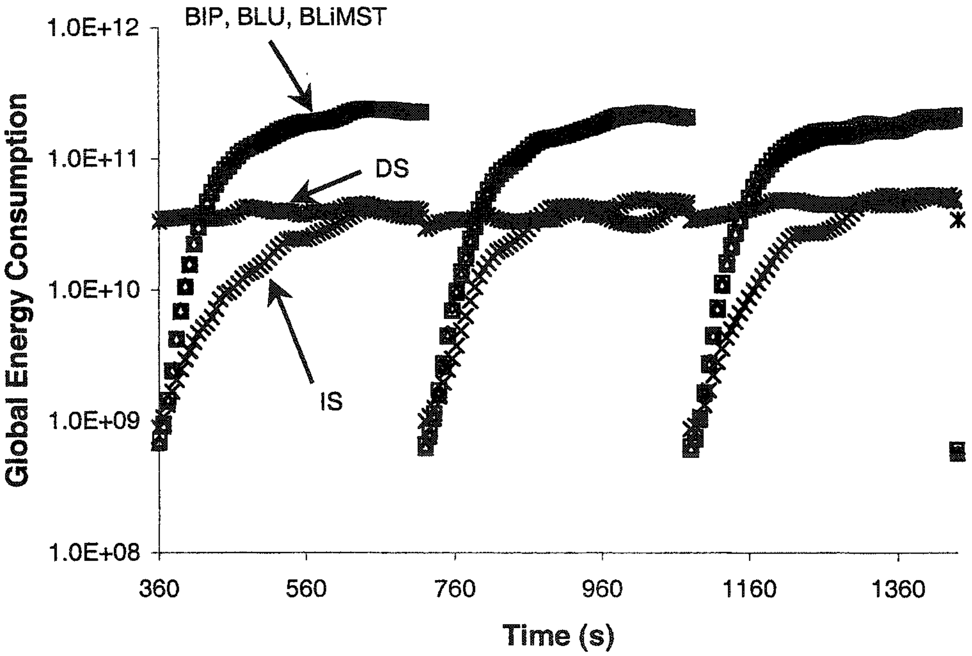
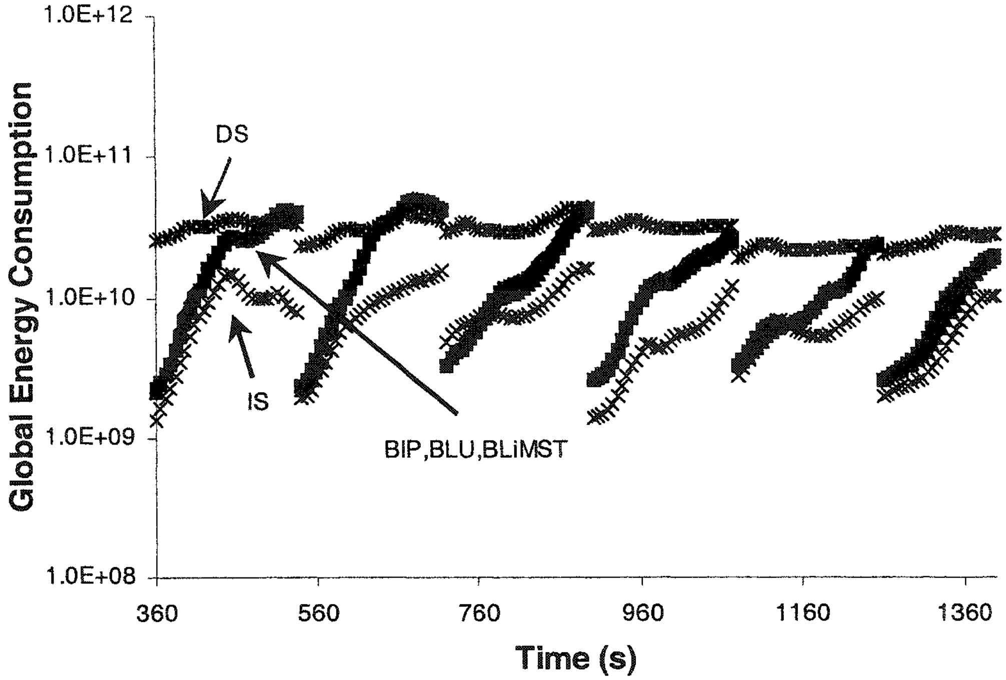
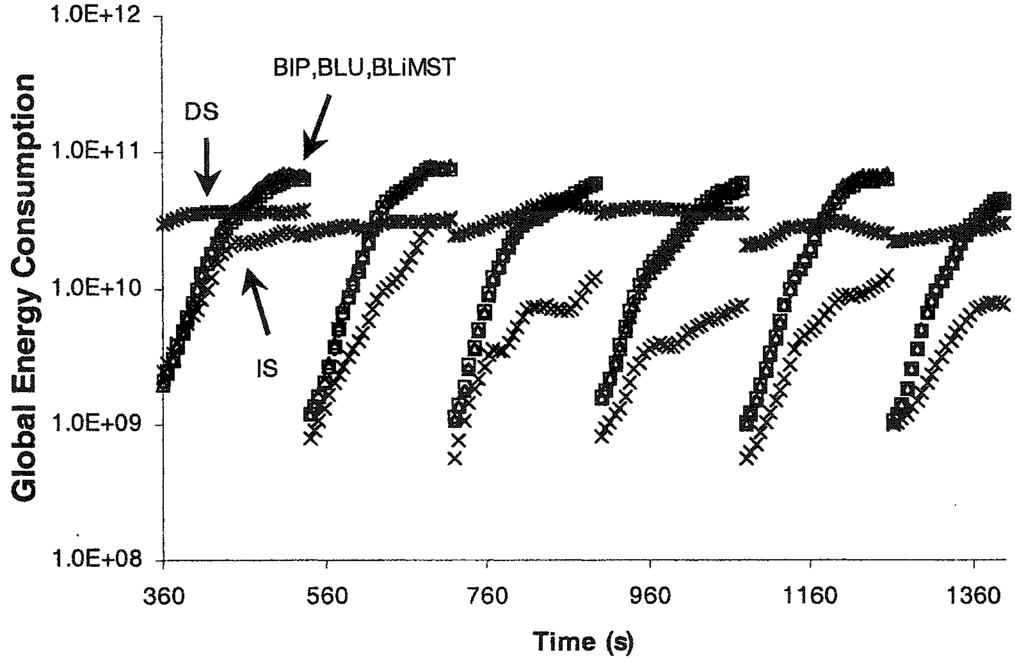


Figure 10: Same maxium speed and number of nodeas Figure 9, with period of 360s for the MEB tree constrution update.



(a)



(b)

Figure 11: Same scenarios as Figure 9, except the number of nodes, with scenario (a) N=10 and (b) N=20.

The global energy consumption results indicate the energy consumption of the broadcast operation at a specific time. With 95% confidence interval, results from each curve will cover up the figure, and each curve is formed by consecutive of data point where each point is an average of 10 simulations. Because simulations start with heuristic that may find the optimal solution, the energy use will be less comparing to the tree that we try to repair using any of the re-arrangement scheme throughout time. Hence, the results demonstrate several intervals that the energy gradually increases and finally converges, and for the reason that we attempt to look at the quality of tree throughout time, we will compare results by the energy consumption that before next update. For instance, Figure 9 illustrates the tree construction update is within 180s and the second update occurs at 540s, we consider the first value to look at is from 539s. By average all the energy consumption that before the next update in each interval, we can compare each heuristic performance (using corresponding tree maintenance scheme) with respect to mobility. Another approach to compare different curve is to consider the energy increase rate. That is, a flatter slope in each period of tree construction update produces a better result.

DS performs equally well (rather, equally badly) throughout the interval. This is because, in the process of eliminating the edges with the largest costs, it quickly runs out of options and leaves a good part of the edges in the resulting relay graph. Hence, even though some of these edges will increase in cost (distance) due to the mobility, some others will decrease as well. The cumulative effect is that the two trends cancel each other leaving the energy cost unchanged. More importantly though, because many edges have survived upon termination of the DS algorithm, there exist many different ways that a node is reachable because it receives the broadcast transmission from multiple relays due to the overlap of their ranges. One is not likely to give a second look to DS when compared against BIP, BLU and BLiMST on a static network, but it turns out to be insensitive to the changes caused by the mobility. It also appears that it results, in the long run, in better energy efficiency than BIP, BLU and BLiMST, assuming enough time has passed since the last tree construction phase.

IS performs by far the best. Even though it is generally slightly worse than BIP, BLU and BLiMST on static networks, its resulting tree cost does not deteriorate as quickly. The explanation behind this behavior is the fact that IS selects the smallest cost edges. Even though the mobility will undoubtedly result in the edge costs to mostly increase, their sum will still be a combination of small and slightly larger values. The "slightly" changes gradually to "significantly" as the time from the last tree computation increases. One observes from Figure 9, a tendency for the lines to converge, and to clarify that they indeed converge, we present in Figure 10 a longer tree construction period of 360 seconds (6 minutes) which is enough to see the resulting convergence. That is, IS converges to the performance of DS, while BIP, BLU and BLiMST converge all to a larger value (almost an order of magnitude larger). In addition, with different number of nodes participated in the network, it does not adversely affect the performance of different tree construction heuristics. Figure 11 illustrates the global energy consumption results for 10 and 20 nodes, comparing with Figure 9, the energy consumption is less, but the slope of each curve remains.

It is also worth looking into an extreme high--speed example where nodes can move with speeds up to 50 meters per second (Figure 12). In a sense, Figure 12 depicts a case where the location of nodes is almost independent of the node locations in the recent past. Such a scenario is unlikely to be encountered in reality but is able to give an idea of the performance under extreme uncertainty regarding the true location of nodes (within the limits of a boundary). The convergence of BIP, BLU and BLiMST as well as that of DS and IS is clear to see and it occurs fairly quickly. Since the location of nodes in the immediate future is almost completely unrelated to their position in the recent past, any distance measurement (hence "cost") on which the trees were initially constructed becomes quickly irrelevant (and plain misleading) to the true location ("cost") of a relay's children nodes. Similarly, the advantage of IS that comes from using the lower cost edges quickly disappears as the selected "smallest" edges quickly deteriorate towards the average edge cost. The performance becomes similar to that of DS. However, IS and DS enourage the presence of cycles in the underlying topology, suggesting that they are able to maintain connectivity despite topology changes without drastic changes to the required energy.

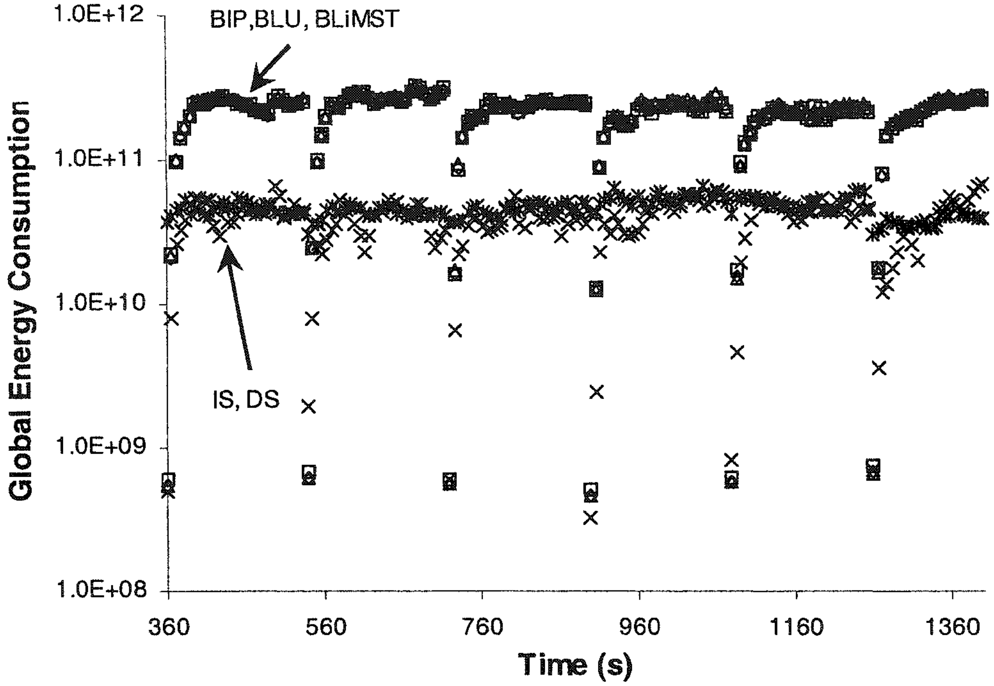


Figure 12: Naïve rearrangement (50 m/sec)

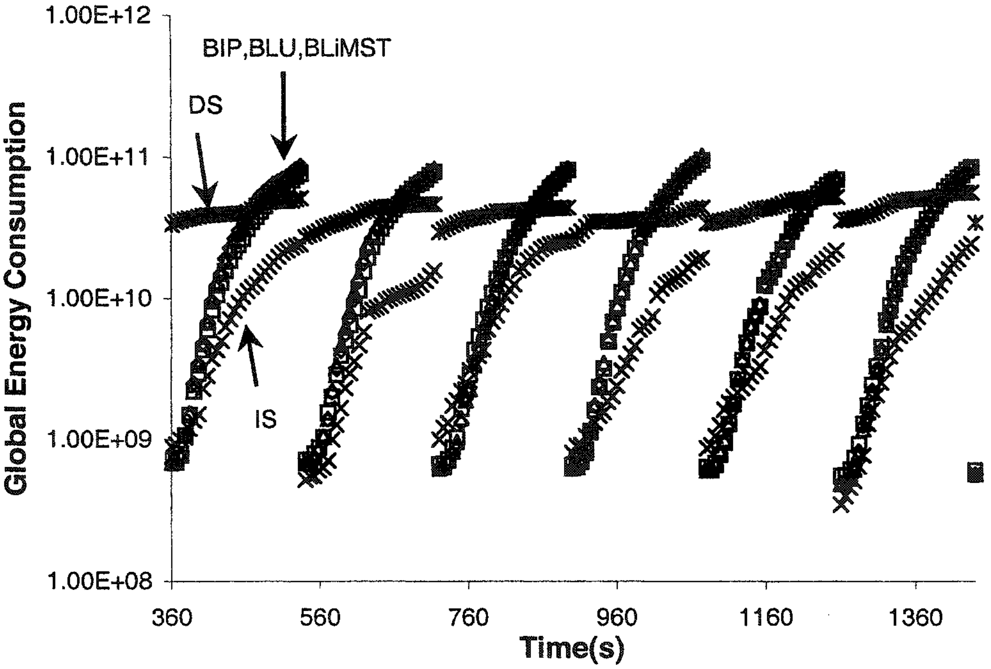


Figure 13: Re-parenting(5m/s).

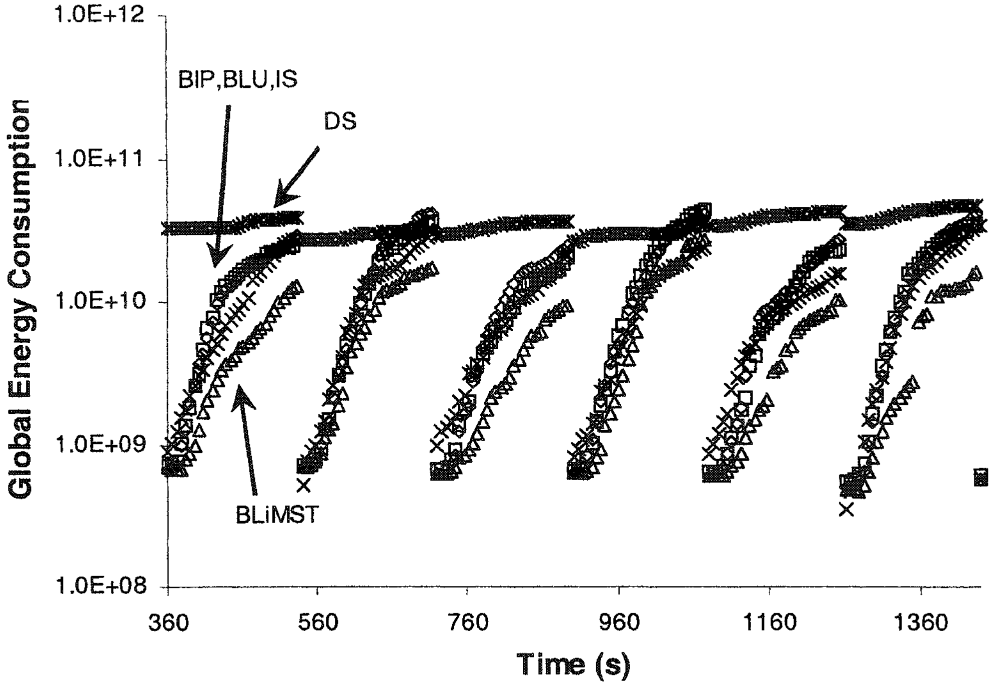


Figure 14: Roaming leaf (5m/s).

Two questions are evident: (a) can we avoid the degenerative impact of the naive scheme on the BIP, BLU and BLiMST, and (b) is IS the best overall scheme when it comes to mobile environments (at least for low speeds)? The answer to the first appears to be affirmative and to the second appears to be negative. As can be seen in Figure 13 and Figure 14, replacing the naïve scheme with either the Re--Parenting or the Roaming Leaf scheme results in substantial improvement. In fact, Re--Parenting has a rather limited impact compared to the Roaming Leaf. It provides a less abrupt slope for BIP, BLU and BLiMST, rendering them less sensitive to the elapsed time, but they still converge to a larger value than DS. It is with the introduction of the Roaming Leaf that the problem is taken care of, that is the convergence of the cost happens at a smaller value than DS. Note however that, in the process, the advantage of IS is lost relative to BIP, BLU and BLiMST. BLiMST is actually performing better than the rest. BLiMST, turns out, tends to use more relays than BIP or BLU, hence effectively hedging the total cost to more components (where each one of them may increase or decrease in cost but each is of small cost to begin with). The leaf nodes in the case of BLiMST are fewer, and due to the higher density of relays around them, they can always find a relay relatively close to them. Similarly high densities of relays relative to leaf nodes are true for IS but in this case, the initial tree of IS had an inflated energy cost compared to BLiMST.

To show that the results are not strongly dependent on the particular mobility model, we change the form of selecting waypoints to a non-uniform one in order to bias the density of the mobile nodes in the 2-dimensional space. We select half of the destination waypoints within the same quadrant (one fourth of) the square. The remaining half of the waypoints are selected uniformly from the remaining three quadrants. The results (from Figure 15 and Figure 16) are essentially identical as far as the relation of the schemes is concerned, although the total energy required is smaller since a larger fraction of the nodes is to be found in a quarter of the square space. Moreover, a large number of nodes is constrained in a smaller area, meaning that a few relays in the dense area can effectively cover a large fraction of the nodes, reducing the relative advantage that BLiMST (or any of the algorithms that tend to use more relays) can attain.

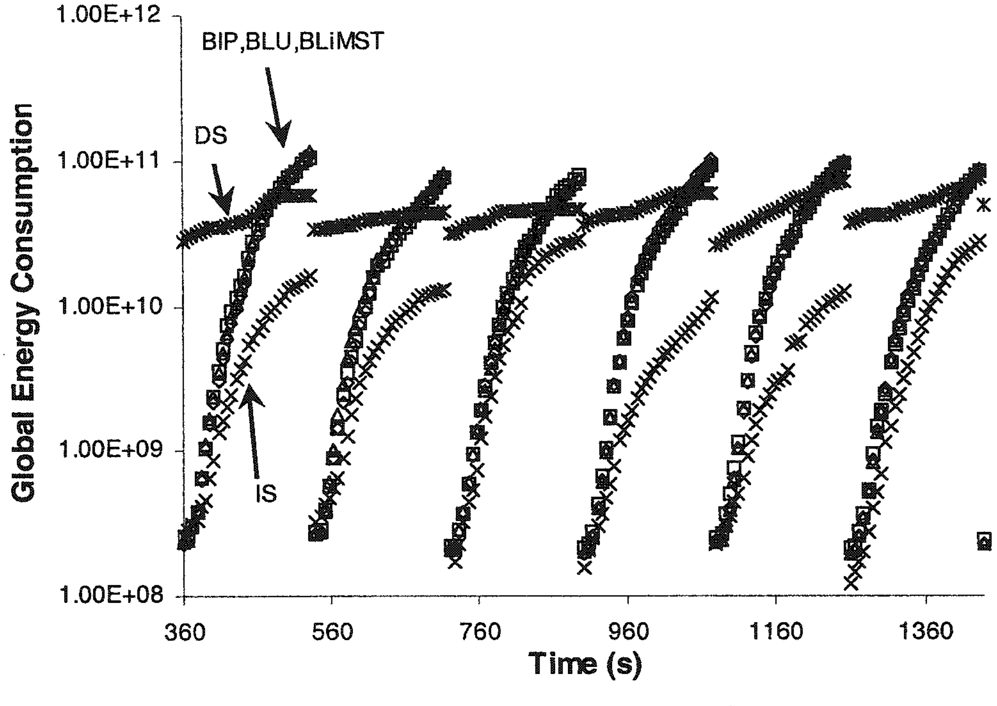


Figure 15 Re-parenting in biased density(5m/s).

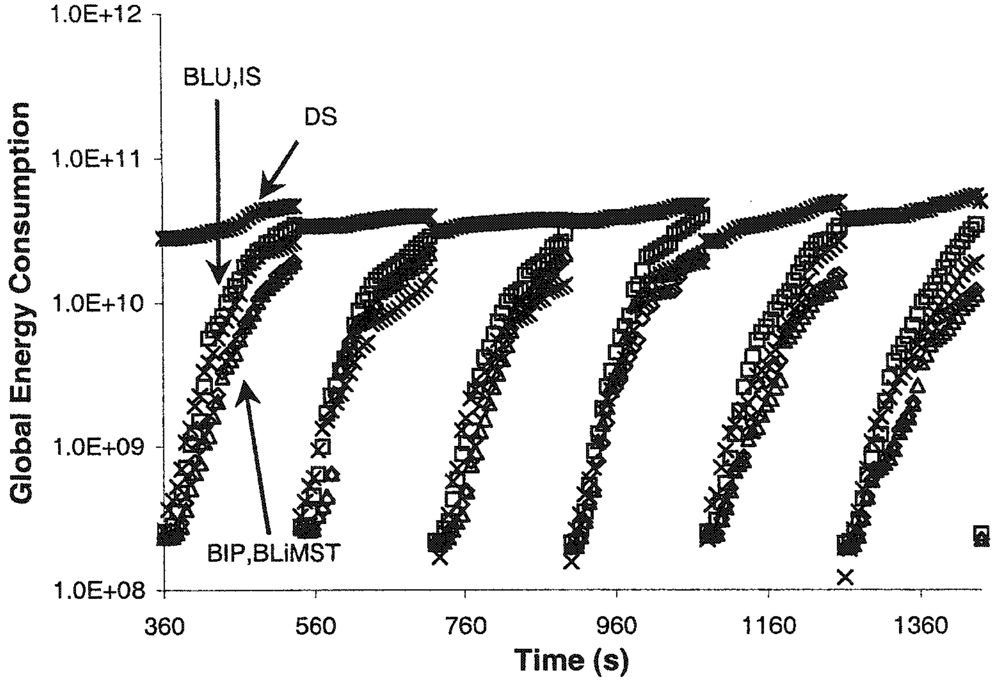


Figure 16 Roaming Leaf in biased density(5m/s).

Summary

An assortment of energy efficient broadcasting algorithms were studied in a mobile environment, along with two additional simple algorithms. It was found that the energy efficiency of some schemes in a fixed wireless environment came at the cost of fragile connectivity and that, in maintaining the connectivity, we had to either continuously execute the algorithms (which is computationally expensive) or develop real--time "repair" schemes. We proposed three such schemes and identified two that result in fairly efficient energy consumption. Summarizing the conclusions: if the naive scheme is employed for the maintenance of the broadcast tree, then a simple scheme such as Incremental Search (IS) provides a good energy efficiency compromise even if it does not produce the best efficiency in fixed networks. If instead, one of the original schemes, BIP, BLU and BLiMST is to be used, then they have to be augmented by a "smarter" tree maintenance scheme, such as the "Roaming Leaf" presented here. We recognize that the addition of such a scheme comes at the cost of additional protocol overheads, but given the "local" nature of the presented schemes, they are still preferable to invoke the tree construction.

What is clear by the results we have gathered so far is that we need a better way to characterize the constructed broadcast trees in order to link them to properties of energy efficiency in the phase of mobility. For example, there are incentives to keep the number of relays small (so that each corresponds to a larger set of children and hence remains relatively insensitive to the movement of a small subset of its children) and incentives to keep it large (so that the sum of the energy expended will be small due to smaller transmission radii employed). It is not clear how one should balance the two factors in a mobile environment. In particular, in the future we will examine algorithms that (a) during the tree construction phase allow explicit control over the number of relays and the length of a path from source to any destination (i.e., control on the breadth and depth of the tree), and (b), due to the movement of nodes, will control the number of relays as a function of the mobility trying to maintain the corrective actions always "local".

Chapter 5

Bandwidth Constrained Energy Efficiency

The Nature of Bandwidth Constraints

In this chapter, our focus remains on energy conservation, but instead of a wireless advantage, we examine briefly the flipside of the broadcast nature of wireless transmissions. Namely, if our intention is *not* to perform broadcast but unicast instead, we end up performing a sequence of transmissions for a packet to be forwarded from source to destination. Each of these transmissions is a local broadcast, that is, it is received by more than just the next hop along the path. As such, even though nodes outside the path from source to destination are uninterested in the specific packet (they will neither forward it, nor they are the intended destination), they will nevertheless "hear" it. Assuming the energy required to receive a packet is substantially small, at least compared to transmitting a packet, overhearing packets can result in a small energy penalty but, more importantly, it results in a congestion penalty. That is, the nodes that overhear the packet transmissions have to refrain from using the medium for their own purposes. The assumption is that the particulars of the medium access are handled by the MAC protocol and it is the MAC protocol that forces stations that overhear packets not destined or related to them, to stay silent during such transmissions.

Consider a specific node, *i*, that is within the range of several other nodes. Let us assume that these other nodes transmit traffic of τ_i bits per second total. The specific node is liable for forwarding some of this received traffic, ϖ_i bits per second (clearly $\varpi_i \leq \tau_i$). Finally, the node also originates some of the traffic (acts as a source) for a load of Ai bits per second. From the perspective of the node, the wireless medium provides a capacity of C bits per second. In order for the routes in the network to be feasible, the *capacity constraint* $\tau_i + \varpi_i + Ai \leq C$ must be satisfied for all nodes in the network. We note however, that any technique that attempts to minimize energy consumption, will attempt to minimize transmission radii. The result is that energy consumption reduction is expected to force τ_i to be reduced as well, except for the fraction that has to be absolutely (because of the topology) forwarded by node *i* which implies that the least traffic possible as seen from node *i* is $2\varpi_i + Ai \leq C$. Nevertheless, τ_i also reflects the fact that nodes are assigned a particular transmission range to ensure the connectivity of the network, or more specifically, the existence of paths from source to destination (if we care only about particular source-destination pairs and not about the connectivity of nodes that have no traffic to send or receive). Hence, too low a τ_i can result in a disconnected underlying topology graph.

For the sake of exposition, we will consider the problem of finding paths between source-destination pairs in *static* wireless networks. We will be given a traffic matrix which indicates the traffic load between each source-destination pair. The load is asymmetric (that is, the load from A to B is not necessarily equal to the load from B to A) and is non-zero for all source-destination pairs (but can be arbitrarily small). Each source-destination pair will subsequently have to be routed using multiple intermediate hops. If no capacity constraints are present, a source-destination pair can pick the lowest energy cost from source to destination using a conventional shortest path algorithm. That is, the shortest path algorithm can be applied on a graph in which the costs stand for the distance between the nodes raised to the loss exponent. For a single source-destination pair, and if no other traffic or bandwidth constraints existed, that would be the optimum solution.

Unfortunately, the general case of the problem (multiple source-destination demands, capacity constraints and wireless broadcast nature) is sufficiently complex to defy currently a simple answer. We note that the first two features (multiple source-destination demands, capacity constraints) render the problem a case of the *node-capacitated multi-commodity flow* class of problems. Unfortunately, the literature on the topic does not consider the broadcast nature of the wireless medium, and thus the fact that τ_i is not just a function of the traffic *intended* to be received by *i* but also of the traffic of other nodes that are "near" i – i.e., it depends on the geometric features of the topology. The overall problem appears to be developing to an extremely complex case, even without the presence of mobility.

The cost we computed is to measure the global energy consumption. After we construct the paths from all source-destination pair, we compute the maximum transmission radius for each node, and hence a transmission from the same node is considered using the same amount of power. As a result, the global energy consumption is the total transmission power required for each source-destination path.

While the rest of the chapter only scratches the surface of the problem to indicate the interplay between energy efficiency and bandwidth constraints, we point out that several varieties of the basic problem exist. For example, transmission and reception by a node are lumped into one capacity constraint only. Clearly, separate channels can be used for transmitting and receiving, with separate fixed capacities. Secondly, we assume knowledge of the traffic loads (between all source-destination pairs). If the loads are randomly generated and the traffic matrices present only knowledge of the *average* load, it is conceivable that although the average traffic indicated by the traffic matrix can be accommodated, the *instantaneous random load* cannot be. It is therefore little comfort to construct paths based on the average behavior. In such case, a quick fix, albeit at the cost of over-allocation resources, is that the traffic load matrices be transformed to represent peak demands (or close-to-peak demands between each source-destination pair), which guarantees that the instantaneous load is unlikely to exceed the values on which the paths were constructed to begin with.

Two Simple Algorithms

Successive Minimum Energy Paths

```
SuccessiveMinimumEnergyPaths (Input:  C(E,V): Adjacency Matrix,
                                      P: Power Consumption Matrix,
                                     TI: An Array of size |V| x |V|, T[i] contains
                                         source,
                                         destination and the traffic intensity;
                              Output: Ps: Minimum energy paths);

   i, j, u, v :          Temporary variables to store the node number.
   SortIncreasing(A): A function to sort the array A in an increasing order of the cost.
   ShortestPath(C(V,E), P, S, v, Path) : A function to compute the shortest path. It
                         takes 4 inputs - the adjacency matrix indicates direction between
                         nodes, the power consumption matrix indicates the cost, the
                         source node, and the destination node respectively. The result
                         will be generated with a set of edges that can create a path, which
                         it takes minimal cost from the source to destination node.
```

```
for $\forall u \in V$
  do C[ u ] $\leftarrow$ 1
     CP[ u ]$\leftarrow$ 0
SortIncreasing(TI)
for $\forall i \in$ TI
   do G'(V',E) $\leftarrow$ C
      for $\forall u \in V$
         do if (C[ u ] < 0)
              V'\ u
      while(true)
           do if (! ShortestPath(G,P, TI[ i ].source, TI [ i ].destination, Path) )
                then  Ps $\leftarrow \varnothing$
                      return
              for $\forall$ (u , v) $\in$ Path
                do C[ u ] $\leftarrow$ C[ u ] – TI[ i ].cost
                   C[ v ] $\leftarrow$ C[ v ] – TI[ i ].cost
              valid $\leftarrow$ true
              for $\forall u \in V$
                   do if (C[ u ] < 0)
                        V'\ u
                        valid $\leftarrow$ false
              if (valid)
                 Ps[ i ] $\leftarrow$ Path
                 break
              else  for $\forall$ (u , v) $\in$ Path
                   do C[ u ] $\leftarrow$ C[ u ] + TI[ i ].cost
                      C[ v ] $\leftarrow$ C[ v ] + TI[ i ].cost
```

Psuedocode 9 : Successive Minimum Energy Paths

In the first algorithm we create one (for one source-destination pair) minimum energy path at a time. Such path construction can employ the shortest path algorithm, e.g. the Bellman Ford algorithm, with the transmitting energy as the cost. We continue adding paths, until a node exceeds the capacity. We remove the node where the capacity was exceeded from consideration and rerun the shortest path. It is still possible that the new path will influence the load of the removed node, due to the nearby nodes relaying its traffic. Subsequently, we remove the nearby nodes from consideration and re-run the shortest-path algorithm. The process continues until a path can be found that does not exceed the capacity constraints of any of the nodes it traverses through and of any of the nodes that overhear its transmission. Therefore, the order of path construction is critical to the outcome. We also note that certain traffic load matrices are simply infeasible –

they cannot be accommodated, because one or more source-destination paths cannot be established. Note also that the solution is not optimal in terms of global energy consumption. It merely attempts to reduce the global energy consumption.

All Pairs Minimum Energy Paths

```
AllPairMinimumEnergyPaths (Input:  C(E,V): Adjacency Matrix,
                                   P: Power Consumption Matrix,
                                   TI: An Array of size |V| x |V|, T[ i ] contains
                                       source,
                                       destination and the traffic intensity;
                          Output:  Ps: Minimum energy paths);

  i, j, u, v :          Temporary variables to store the node number.
  SortIncreasing(A): A function to sort the array A in an increasing order of the cost.
  ShortestPath(C(V,E), P, S, v, Path) : A function to compute the shortest path.  It
                        takes 4 inputs - the adjacency matrix indicates direction between
                        nodes, the power consumption matrix indicates the cost, the
                        source node, and the destination node respectively. The result
                        will be generated with a set of edges that can create a path, which
                        it takes minimal cost from the source to destination node.
  AllPairShortestPaths(C(V,E),P, Ps): A function to compute the all pair shortest paths. It
                        takes 2 inputs – the adjacency matrix indicates the graph, and the power
                        consumption matrix indicates the cost. The results will be stored in Ps,
                        all the minimal energy paths of all the combination node pairs.

  for $\forall u \in V$
     do C[ u ] $\leftarrow$ 1
        CP[ u ] $\leftarrow$ 0
  SortIncreasing(TI)
  AllPairShortestPaths(C, P ,Ps)
  for $\forall i \in TI$
     do for $\forall (u, v) \in Ps[ i ]$
          do C[ u ] $\leftarrow$ C[ u ] –TI[ i ]
             C[ v ] $\leftarrow$ C[ v ] –TI[ i ]
  for $\forall i \in TI$
     do G(V',E) $\leftarrow$ C
     while(true)
        do valid $\leftarrow$ true
         for $\forall (u, v) \in Ps[ i ]$
           do if (C[ u ] < 0 | | C[ v ] < 0)
                V' \ u
              valid $\leftarrow$ false
         if(valid)
           break
         else
            for $\forall (u, v) \in Ps[ i ]$
               do C[ u ] $\leftarrow$ C[ u ] +TI[ i ]
```

```
        C[v] ← C[v] + TI[i]
    if(!ShortestPath(G,P,Path))
      return
    else
      Ps[i] ← Path
```

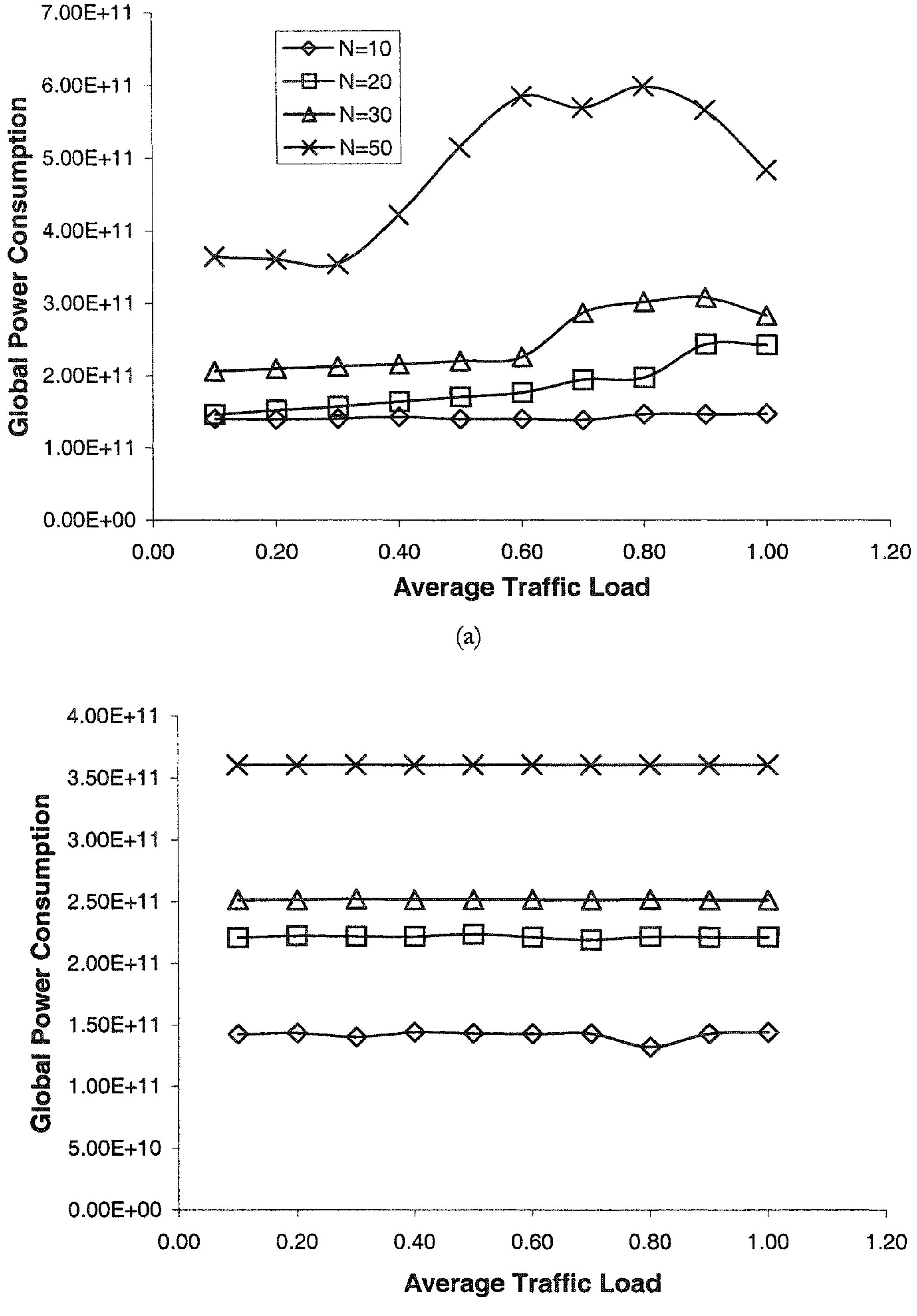
Psuedocode 10 : All Pairs MinimumEnergy Paths

This algorithm is based on the idea of all-pairs shortest paths algorithm such as the Floyd-Warshall algorithm, which runs in $O(N^3)$ time. To employ the algorithm while minimizing the global energy consumption, we associate the energy instead of the distance to the cost of the graph edges. Subsequently, we examine the nodes where the bandwidth constraint is violated. We remove from such nodes the loads (source-destination paths) in decreasing order of traffic intensity, until the nodes no longer exceed the constraints. The source-destination loads that were removed, are recalculated on different paths, from which we exclude the nodes that were originally found to be congested. The process continues until no nodes handling or overhearing the traffic of the re-allocated paths exceed the available capacity. This approach differs from the previous one in that the order of calculating paths no longer matters, but the traffic volume between source and destination does (the higher it is the more likely it will have to get re-routed because it congests the network). Again, it is totally within reason to end up with an infeasible configuration.

Simulation Results

The simulations were conducted with randomly placed nodes within the familiar rectangular area but no mobility model was used. The nodes remain static. The bandwidth capacity is assumed to be equal to 1 and all bandwidth quantities are subsequently normalized with respect to the bandwidth capacity. The traffic load matrix, T_{ij}, was produced in a random fashion. Specifically, each element of T_{ij}, is uniformly randomly generated to be a demand between 0 and L/2(N-1) (where L is a parameter controlling the "global" load over all nodes, expressed as a number from 0 to 1, and N is the number of nodes). The particular formula guarantees that the sum of traffic originating from source i is less than ½ which guarantees that the load of another nodes due to forwarding the load of this source-destination pair is going to be less than 1. Note however that the restriction on the random generation of the traffic matrices is not sufficient to avoid infeasible scenarios.

The scenarios simulated span the parameters N=10, 20,30, and 50, L=0.2, 0.4, 0.6, 0.8, and 1. For the first algorithm, the construction of minimum energy paths, the sequence of individual path construction is according to the node number order (which is thus unrelated, i.e., unordered with respect to the traffic intensity). Then, we tweak the first algorithm by forcing the path construction according to decreasing traffic demand. We also simulate the second algorithm. The simulations results are demonstrated in Figure 17 (for the two version of the first algorithm) and Figure 19 for the second algorithm.



(a)

(b)

Figure 17: The global energy consumption for Successful Minimum Energy Path Algorithm with constructing each path according to (a) the node number order, and (b) the traffic intensity order corresponding to the average traffic load.

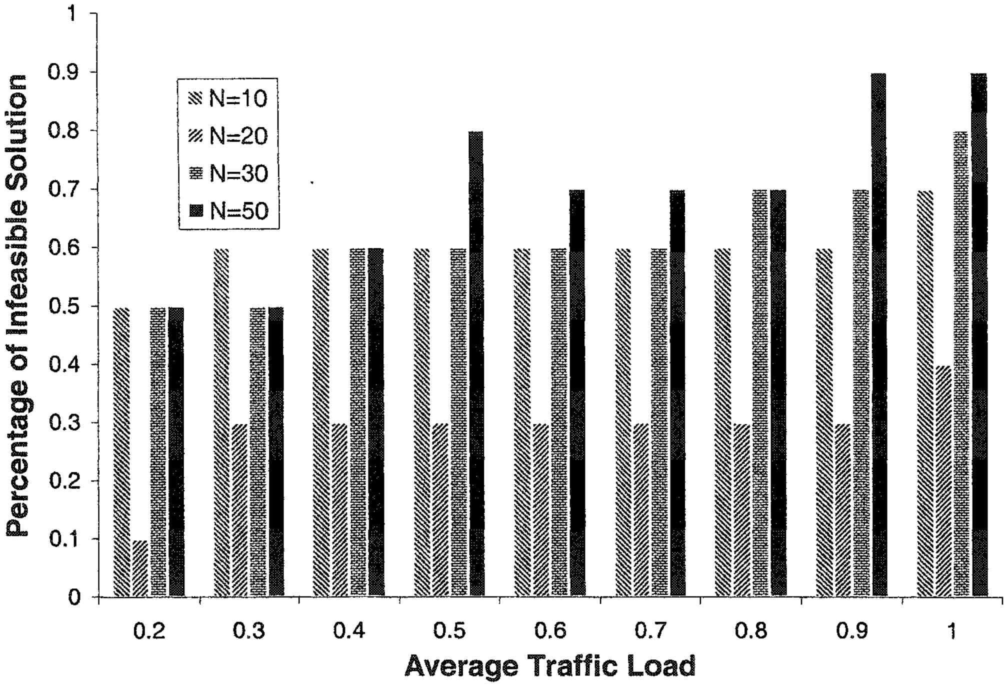


Figure 18: Percentage of infeasable solution corresponding to the Figure 17 (a) setting.

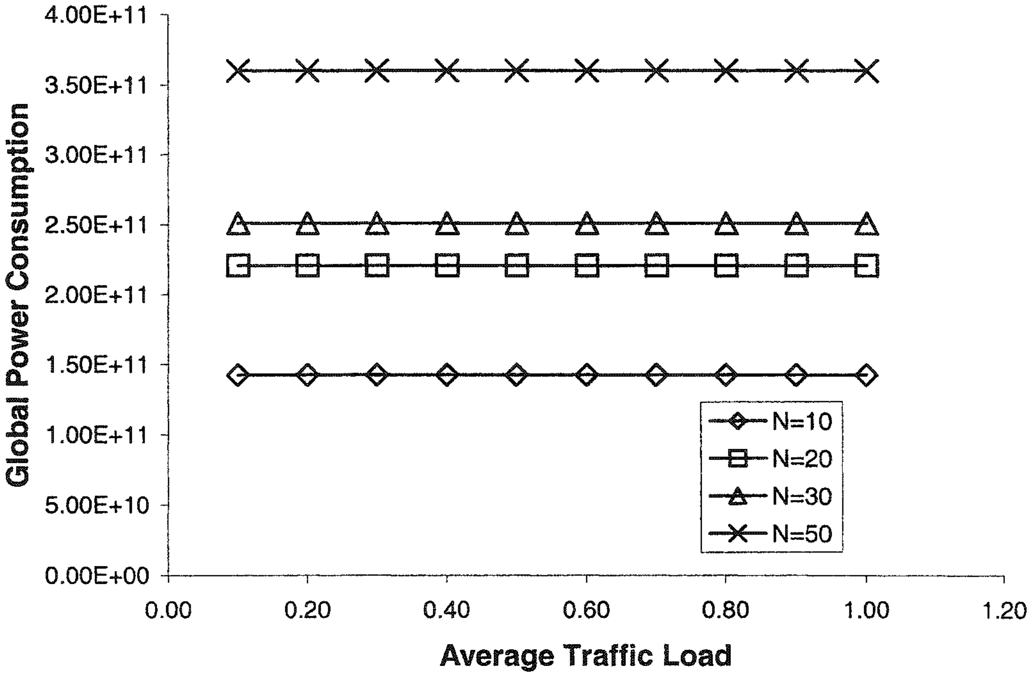


Figure 19: Simulation results for All Pair Minimum Energy Paths.

The results are expressed as a relation between global energy consumption and the average traffic load. Our intuition would suggest that the more the load, the more the nodes that end up violating their respective constraints, the longer the paths to avoid such congested nodes, hence, the more the energy require. This situation is partly what happens in Figure 17.a. However, the graph shows that the energy consumption drops when the load is near 0.9. The figure is misleading in this respect because what is missing is the fact that several of the runs that correspond to the point at 0.9 and higher resulted in infeasible scenarios. Figure 18 demonstrates the ratio of infeasible solution corresponding to the result generated from the successive minimum energy paths algorithm with unordered path construction. With 50 nodes, the number of infeasible solution is around 90% of the run, which is extremely high.

Hence, the energy consumption is smaller but a larger number of configurations are eventually found to be infeasible. A more definite result is the one that suggests that the energy consumption increases as the number of nodes increases. True, each additional node provides an opportunity for paths to be split along a longer path where the sum of energy required over the entire path is lower than with fewer intermediate hops, but this cannot counter the fact that additional nodes produce a higher node density and a higher probability that the transmissions will congest other nodes, and hence it restricts paths from being available. The unavailability of a path therefore has a direct impact on the ability of conserve energy by using it.

Figure 17.b and Figure 19 provide more promising results (less sensitive to the load). Essentially the one-by-one construction with decreasing demands and the second algorithm provides similar results. An explanation, albeit not complete with these preliminary results are due. First, the network can handle the traffic load which is greater than 1, or, second, these two algorithms are the optimal solution to the problem. The second is quite unlikely since we lack a complete grasp of the computational complexity of the particular problem. The first option could be true because of simultaneous transmission from two or more nodes without interfering with each other. Consider the case of the two (which can be generalized to more) highest source-destination load demands. If we are successful in accommodating them, there is a good chance that the

smaller demands can be accommodated too, possibly using longer paths as well. The longer paths imply smaller energy consumption, and hence forcing the small demands to take such paths is in retrospect a good energy-wise approach. We are therefore concluding that the network is able to handle comfortably a load greater than 1, which suggests successful spatial reuse of the wireless medium. Finally, note that again at high loads (0.9 and more) the results presented here are the average of the *feasible* runs. A number of traffic matrices produced result again in infeasible configurations.

Summary

In this chapter, we considered the bandwidth constraint problem. The wireless broadcast advantaged that we exploited in the previous chapter becomes now a serious "disadvantage" in that it congests nearby nodes that have no purpose or interest in forwarding specific unicast traffic. The corresponding energy optimization becomes even more complicated due to the need to satisfy the bandwidth constraints as seen by each node. The particular formulation of the problem assumes the existence of an ideal MAC protocol. It is up to the optimization process to determine if the loads are infeasible, cannot be carried given the bandwidth constraints, and terminate rather ungracefully if that is indeed the case. We note that a simple approach to avoid declaring the particular scenario as infeasible is to split the traffic between the same source-destination pair along multiple different paths. We do not use this technique, but instead try to find algorithms that will declare certain configurations as infeasible but along the way they will not overly compromise the energy consumption. We find that algorithms that are based on ranking the load of connections hold a better promise towards this end. However, we are still far from a complete understanding and clear formulation of the underlying problems and tradeoffs.

Chapter 6

Conclusions

This thesis presents the work carried out towards the reduction of consumed energy in a MANET environment by controlling the transmission radius of each participating node (via either one radius for all nodes, or using individual per-node radii). The global energy optimization problem has to respect several of the constraints that exist either due to the nature of the medium (finite bandwidth, broadcast nature etc.) or for the sake of the particular service (connectivity, broadcast to all nodes, ability to carry specific traffic loads etc). The addition of mobility presents both challenges and new opportunities, and it is interesting to note that algorithms appear reasonable or at least competitive in fixed wireless are not capable of surviving the dynamics of mobility.

In the beginning of this thesis, we considered the problem of coming up with a single radius for all transmitters in a MANET. That radius would be the smallest possible to guarantee connectivity, and in this respect it would have been the most energy efficient. However, what we realized is that the velocity-dependent behavior of such "critical" radius is a function of the particular synthetic mobility model, the Random Waypoint Model (RWP). First we find out the range of critical radii, which is possible to have connectivity. As we cannot come up with a specific value that is suitable for all MANETs, we studied the impact of the non-uniform density exhibited by RWP. In fact, by studying the connected components of the underlying topology, we noticed a distinct bimodal distribution of component lifetimes. Hence, when comparing existing with future proposed routing (and not only routing but any MANET-related) protocol, the behavior of the underlying random graph may hold the key to explain within the proper context the observed behavior.

As a second step, we considered the Minimum Energy Broadcasting (MEB) problem and how it can be adapted to mobile environments. Because of the time dependent topology, the energy consumed with the original broadcast tree is not expected to remain near-optimal for long. Reconstructing the tree frequently requires more control messages sent than maintaining the original tree, and this results in decreasing the routing performance.

Although the cost of the rearranged MEB tree is not as low as the original, the aim is to exploit the wireless broadcast advantage and to minimize the computation complexity of the maintenance process (comparing to the tree construction). The objective is accomplished with the inclusion of simple on-demand "repair" schemes that re-associate nodes to their parents (or grandparents) with criteria of best energy consumption.

Finally, from the bandwidth constraint point of view, the reduction of transmission radius is expected to increase the wireless channel spatial reuse, which will minimize the consumption of energy and improve the throughput. Under this observation, we would expect that the throughput maximization and energy minimization both push path establishment algorithms towards the same direction. That is, it would be conceivable to expect good performance for one of them while optimizing the other. Unfortunately, the introduction of bandwidth constraints complicates matters. The freedom to establish paths is reduced not only because of the nodes actively relaying traffic, but it is also because of the "disadvantage" due to the broadcast nature of the medium. Nodes perceive as traffic load (and can be congested) by transmissions to which they are not interested but happen to overhear because of the transmission radius employed by the corresponding transmitter. While it is still early to drive any definite conclusion we speculate, based on simulations, that the idea of first making use of the spatial reuse to satisfy the higher source-destination load demands is key to accommodate the demands with reasonable energy efficiency.

We have only but briefly describe three relevant problems and their implications with respect to energy efficiency. The next step is to introduce a rigorous formulation in an optimization context (including both non-linear and integer optimization cases) in order to appreciate the order and the impact that the constraints have on the energy minimization problem, recognize standard techniques by which the objective can be, hopefully, decomposed and determined the underlying problem computation complexity.

Bibliography

[1] B.Liang, and Z.J. Haas, "Predictive Distance Base Mobility Management for PCS Networks," in *Proc. of INFOCOM* '99, pp.1377-1384, New York, NY, March 1999.

[2] J.Broch, D.A. Maltz, D. Johnson, Y.-C. Hu, and J. Jetcheva, "A Performance Comparison of Multihop Wireless Ad Hoc Network Routing Protocols," in *Proc of MOBICOM '98*, pp.85-97, Dallas, TX, October 1998.

[3] X. Hong, M.Gerla, G.Pei, and C.-C. Chiang, "A group Mobility Model for Ad Hoc Wireless Networks," in *Proc. of MSWiM '99*, pp.53-60, Seattle, WA, August 1999.

[4] Z.Hass, "A New Routing Protocol for the Reconfigurable Wireless Networks," in Proc. of *ICIPC*, pp.562-566, San Diego, CA, October 1997.

[5] C.Bettstetter, "Smooth is Better than Sharp: A Random Mobility Model for Simulation of Wireless Networks," in Proc of *MSWiM*, pp. 19-27, Rome, Italy, July 2001.

[6] M.Sanchez ,P. Manzoni, and Z.J. Haas "Determination of Critical transmission Range in Ad-Hoc Networks" , In Proc of *MMT'99,* Venice, Italy, October, 1999.

[7] L. Kleinrock, and J.A. Silvester, "Optimum Transmission Radii in Packet Radio Networks or Why Six is a Magic Number", National Telecommunications Conference, Birmingham, Alabama, pages 04.3.1-5, December 1978.

[8] D.B. Johnson, and D.A. Maltz, " Dynamic Source Routing in Ad Hoc Wireless Networks," in *Mobile Computing (ed. T. Imielinski and H. Korth)*, Kluwer Academic Publishers, Dordrecht, The Netherlands.

[9] C.E. Perkins, E.M. Belding-Royer, and S. Das, "Ad Hoc On-Demand Distance Vector (AODV) Routing," *MANET Working Group*, IETF, January 9, 2002.

[10]V. Park, and S. Corson, "Temporally-OrderedRouting Algorithm (TORA) Version 1 Functional Specification," *MANET Working Group*, IETF, July 20, 2001.

[11] R. Volkan, and T.H. Meng, " Minimum Energy Mobile Wireless Networks," in *Proc of IEEE International Conference on Communications*, ICC'98 Atlanta, GA, vol. 3, pp. 1633- 1639, June 1998.

[12] S. Singh, M. Woo, and C. S. Raghavendra. "Power-Aware Routing in Mobile Ad Hoc Networks," In The Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking, Dallas, TX, pages 181--190, October 1998.

[13] S. Singh, C. S. Raghavendra, and J. Stepanek, "Power-Aware Broadcasting in Mobile Ad Hoc Networks", *Proceedings of IEEE PIMRC '99,* Osaka, Japan, September, 1999.

[14] J.-H Chang, and L. Tassiulas. "Energy conserving routing in wireless ad-hoc networks," INFOCOM '2000, March 2000.

[15] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides "Algorithm for Energy-Efficient Multicasting in Static Ad Hoc Wireless Networks," MONET, Vol 6, no. 3, pp. 251-263, 2001

[16] J. Wieselthier, G. Nguygen, and A. Ephremides, "On the Construction of Energy Efficient Broadcast and Multicast Trees in Wireless Networks", in the proc.of *IEEE INFOCOM 2000.*

[17] F. Li, and I. Nikolaidis. "On Minimum-Energy Broadcasting in all Wireless Network", in proc. of *IEEE Annual Conference on Local Computer Networks*, November 2001.

[18] M. Cagalj, J-P hubaux, and C. Enz, " Minimum-energy broadcast in all-wireless networks: NP-completeness and distribution issues," to appear in the proc. of *MobilCom 2002*.

[19] B.Bollobas, Random Graphs: Second Edition, Cambridge University Press, 2001.

[20] Y.C. Hu, and D.B. Johnson, "Caching Strategies in On-Demand Routing Protocol for Wireless Ad Hoc Networks", Proc. of MOBICOM 2000, pp. 231-242, Boston , USA, August 2000.

[21] C.E. Perkins, and H. Harjono, " Resource Discovery Protocol for Mobile Computing," in the Proc. of *IFIP World Conference on Mobile Communications*, pp.219-238. 1996.