The Pennsylvania State University

The Graduate School

Department of Computer Science and Engineering

MOBILITY PREDICTION BASED INTELLIGENT ALGORITHMS FOR MOBILE AD HOC NETWORKS

A Dissertation in

Computer Science and Engineering

by

Aravindhan Venkateswaran

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Philosophy

August 2008

The dissertation of Aravindhan Venkateswaran was reviewed and approved[*] by the following:

Raj Acharya
Professor of Computer Science and Engineering
Head of the Department of Computer Science and Engineering
Thesis Adviser
Chair of Committee

Thomas F. La Porta
Professor of Computer Science and Engineering

Wang-Chien Lee
Assistant Professor of Computer Science and Engineering

Carleen F. Maitland
Associate Professor of Information Sciences and Technology

Venkatesh Sarangan
Special Member
Assistant Professor of Computer Science
Oklahoma State University

[*] Signatures are on file in the Graduate School

Abstract

The advent of multi-hop mobile ad hoc networks (MANETs) has led to a number of application scenarios consisting of autonomous mobile agents performing long term sensing and communication tasks. Energy is a key concern in designing MANETs as the nodes usually have limited battery power. As communication will undoubtedly be one of the essential functionalities of such networks, optimizing the energy consumed for data transmissions is of utmost importance. In this dissertation, we have developed a mobility prediction based framework for deploying controllable mobile relay nodes for optimizing the energy consumption in MANETs. We present two instances of the relay deployment problem, together with the solutions, to achieve different goals. Instance 1, termed Min-Total, aims to minimize the total energy consumed by the traditional nodes during data transmission, while instance 2, termed Min-Max, aims to minimize the maximum energy consumed by a traditional node during data transmission. Our solutions also enable the prioritization of individual nodes in the network based on residual energy profiles and contextual significance. In addition, we also present a mobility prediction based clustering framework to dynamically organize the network into stable sub-structures to assist in an initial deployment of the relay nodes.

We perform an extensive simulation study to evaluate the performance of the relay deployment algorithms underying network conditions. We also investigate the performance of the proposed framework under different mobility prediction schemes. Results indicate that even when the relay nodes constitute a small fraction of the total nodes in the network, the proposed framework results in significant energy savings. Further, we observed that while both the schemes have their potential advantages, the differences between the two optimization schemes is clearly highlighted in a sparse network.

Table of Contents

List of Tables

List of Figures

Acknowledgments

I would like to thank my mentor and adviser Dr. Raj Achayra, whose guidance and encouragement have been of immense value to me. He has been a great source of inspiration. His patience and intellectual curiosity will always guide me to work sincerely, laboriously and prudently in my future endeavors.

I would also like to thank my co-adviser Dr. Venkatesh Sarangan, who has been a constant source of support throughout my thesis work. I have learnt a lot from him through the numerous intellectual conversations which really helped me in understanding the subject in detail.

I would also like to express my gratitude to the members of my doctoral committee (Drs. Thomas La Porta, Wang-Chien Lee, Carleen Maitland) for helping me in numerous ways despite their busy schedules. I would also like to thank Dr. Natarajan Gautam for his help during the early stages of this research.

This thesis would have been impossible without the motivation of my parents who have instilled in me the desire to learn. I would like to thank my fiance Subashini Krishnamurthy for being there and providing support when I needed it the most. Special thanks to my brothers for their constant support and encouragement throughout the course of my dissertation work.

While at Penn State, I certainly enjoyed the company of my numerous friends who have helped me throughout my stay. Special mention goes to Hari Thiruvengada and Ranjit for all their help and support.

This thesis is dedicated to my parents, Nagalakshmi and Subramanian Venkateswaran, for their love, endless support and encouragement.

Chapter 1

Introduction

Advances in wireless communication and the widespread use of mobile and handheld devices has resulted in an increasing popularity of mobile ad hoc networks (MANETs) - networks that consist of a collection of geographically distributed mobile nodes that communicate with each other over a wireless medium. The development of MANETs was originally motivated by military applications such as battlefield surveillance. However, in the recent years, MANETs have outgrown their military utility and are increasingly employed in a number of applications ranging from emergency search and rescue operations to environmental and infrastructure monitoring to network entertainment. MANETs are autonomous networks that do not have any fixed infrastructure in place. Communication takes place through wireless links among mobile nodes. Since the nodes have limited transmission range, communication between the nodes often takes place in a multi-hop fashion, i.e., two nodes that are not in the direct reach of each other converse by establishing routes through other nodes in the network. However, due to node movement, such multi-hop routes break frequently disrupting the traffic flow between the communicating nodes. Further, large scale mobility in a sparse network can also lead to an occasional loss of connectivity if nodes end up outside the range of each other. It is the mobility of the nodes coupled with transient nature of the wireless media that results in a highly dynamic network topology and varying network connectivity. As a result, node mobility is widely considered as an inherent characteristic of MANETs that more often than not, severely affects the operation of the network.

One way to address the effects of node mobility is devising proactive solutions that utilize mobility prediction algorithms to estimate the movement of the nodes. In recent years, mobility prediction has become increasingly popular as mobile devices are getting equipped with reasonably accurate positioning capability. In fact, in cellular networks, mobility prediction has been shown to be extremely effective at both the network and application level. For example, QoS degradation or forced termination may occur in cellular networks when there are insufficient resources to accommodate handoff requests [1]. One solution is to predict the trajectory of the mobile terminals so as to enable QoS provisioning by performing resource reservations in advance. Similarly, mobility prediction has also been applied to a number of tasks at the network level such as handoff management, flow control, resource allocation, congestion control and call admission control. There has also been a significant amount of work in the literature in the area of mobile location-based services that have also highlighted the effectiveness of mobility prediction techniques at the application layer.

Much of the research that utilize mobility prediction in MANETs has focused primarily on improving the performance of various routing protocols [2]. The basic idea behind such approaches is to predict the topology changes in the network by devising efficient link availability estimation algorithms. If such topology changes can be predicted fairly accurately then route reconstruction or route discovery can be completed prior to a topology change.

In addition to predicting the movement of nodes in the network, there has been a growing interest to study and build mobility-aided systems where the mobility of certain nodes, often referred to as relay nodes, can be controlled by the underlying network protocol. These relay nodes can be appropriately positioned and moved so as to optimize the performance of the network with respect to a specific objective such as network connectivity. As communication will undoubtedly be one of the essential functionalities of such networks, optimizing the energy consumed for data transmissions is of utmost importance. As a result, a number of recent studies aim to utilize controllable mobility for conserving energy in MANETs. While most of the existing work in the literature focus on deploying such relay nodes in a static network, it remains to be seen if such approaches can be effective in a truly *mobile* setting. This forms the primary focus of this dissertation wherein we investigate the use of mobility controllable relay nodes for optimizing the energy consumption in MANETs. We outline various strategies that utilize intelligent algorithms based on mobility prediction to deploy and position these relays nodes in a truly mobile network.

1.1 Contributions

In this dissertation, we have developed a mobility prediction based framework for deploying mobility controllable relay nodes for optimizing the energy consumption in MANETs. We have developed relay positioning algorithms that achieve two different optimization strategies based on an initial deployment of the relay nodes. Further, we also present a mobility prediction based clustering framework to dynamically organize the network into stable sub-structures to assist in an initial deployment of the relay nodes. In the following sections, we describe the goals, hypothesis, rationale and specific aims of this dissertation.

1.2 Overall Goal

The primary goals of the research presented in this dissertation are as follows:

1. Develop an intelligent mobility prediction based framework to deploy mobility controllable relay nodes for optimizing the energy consumption in MANETs.
2. Develop efficient algorithms for positioning the relay nodes so as to achieve two different optimization strategies:
   (a) Minimize the total energy consumed in the network.
   (b) Minimize the maximum energy consumed in the network.

1.3 Hypothesis

The hypotheses that we test in this dissertation are as follows:

1. Mobility control algorithms based on mobility prediction enable a feasible deployment of relay nodes in a truly mobile network where nodes move independently based on the requirements of the underlying application.
2. Different relay positioning strategies can be designed to minimize the total energy consumption across the entire network while simultaneously catering to the requirements of specific nodes in the network

3. Mobility prediction based clustering algorithms are an effective way to dynamically organize the network into stable sub-structures to assist in an initial deployment of the relay nodes.

1.4 Specific Aims

1. Define the relay positioning problem for conserving energy for a heterogeneous network consisting of two kinds of mobile nodes – the traditional nodes with limited energy and a few mobility controllable relay nodes with relatively abundant energy resources.

2. Facilitate the prioritization of individual nodes in the network based on residual energy profiles and contextual significance.

3. Develop intelligent algorithms to solve the relay positioning problem to achieve two optimization strategies – (a) Min-Max and (b) Min-Total.

4. Design and develop a mobility prediction based relay deployment framework that enables the interaction between the relay nodes and the traditional nodes in a truly mobile network. The proposed framework should be designed so as to work with any standardized underlying network layer MANET routing protocol.

5. Investigate the performance of the relay deployment framework under different mobility prediction schemes and study the tradeoffs involved in such systems using extensive simulation study.

6. Develop of a mobility prediction based distributed clustering algorithm that organizes the MANET nodes into long-lived cohesive groups by recognizing the similarity in mobility patterns.
7. Investigate the impact of existing mobility prediction schemes on the temporal stability of the obtained clusters with respect to different node mobility models under varying network and mobility conditions.

1.5 Significance

Wireless systems are continuing to evolve towards networks consisting of nodes that communicate without the need for infrastructure. As technology rapidly progresses, diverse sensing and mobility capabilities have become readily available to devices and consequently MANETs are being deployed to perform a number of important tasks. As communication is one of essential functionalities of such mobile networks, it is important to devise schemes to improve the communication energy efficiency at individual nodes. While it is important to minimize the overall communication energy consumption across the network, excessive or disproportionate energy consumption among nodes can also lead to premature failure of network. The work described in this thesis explores the use relay nodes with controllable mobility to enable the design of networks that are capable of optimizing their energy requirements.

We propose novel mobility prediction based relay deployment strategies and demonstrate that such schemes indeed result in significant energy savings. We also show that deploying and positioning such nodes is feasible in a truly mobile network where all the nodes move independently based on the requirement of the underlying applications. Although the full strength of controlled mobility can only be demonstrated by completely functional prototypes, the results from a thorough simulation study presented in this dissertation show that controllable node mobility is indeed an effective tool for improving the energy efficiency of mobile ad hoc networks.

1.6 Organization of the dissertation

The rest of the dissertation is organized into seven different chapters. We presented a brief outline of the dissertation in chapter 1. In chapter 2, we present the summary of the related work done in the literature. We define the relay deployment problem for a heterogeneous network in chapter 3. Chapter 4 outlines the solution strategies for the proposed relay deployment problem. The results from an extensive simulation study of the proposed algorithms are presented in chapter 5. In chapter 6, we describe a mobility prediction based clustering algorithm for MANETS. We conclude the dissertation by presenting a summary of the work done in chapter 7.

Chapter 2

Controllable Mobility in Mobile Ad Hoc Networks

2.1 Introduction

The mobility of the nodes in MANETs is traditionally presumed to be beyond the control of any network protocol. The nodes generally move about to fulfill the requirements of the underlying application. As a result, node mobility is often treated as a constraint and early research in the area was focused more on assessing the impact of varying node mobility on the performance of various protocols and algorithms. As the undermining effects of mobility in MANETs were realized, several researchers proposed a number of mobility-aware schemes in which the system changes behavior in response to, or by predicting, node movements [3]. For example, if the dynamics of the network topology could be predicted in advance, a route-discovery mechanism could select paths that were more stable or long-lived in order to avoid or reduce route failures.

More recently, there has been a growing interest to study and build mobility-aided systems where the mobility of certain nodes, often referred to as relay nodes, can be controlled by an underlying network protocol. These relay nodes can be appropriately positioned and moved by the protocol so as to optimize the performance of the network with respect to a specific objective. For example, the relay nodes can act as special infrastructure nodes [4] and help to maintain and improve network connectivity and communication in a sparse network. One can also imagine a network where the relay nodes can mimic the mobility patterns of a quorum of actively communicating nodes while serving as an intermediate hop for communication [5]. The presence of such relays would enable the traditional nodes to transmit data at reduced power, and thus result in energy savings. While the cost of mobility for certain nodes may be high depending on their physical dimensions and environmental conditions, an intelligent framework that utilizes such mobility controllable nodes can not only overcome some of the difficulties posed by node mobility but also result in significant performance gains.

In this chapter, an overview of several such approaches in the literature wherein mobility is viewed as a controllable network primitive is presented. A list of a number of objectives that can be achieved using mobility controllable nodes is discussed followed by a discussion on the various issues that such frameworks should address. Finally, we outline the idea behind the use of relay nodes as it applies to the research described in this dissertation.

2.2 Objectives of mobility control frameworks

Irrespective of being wired or wireless, a network is typically deployed in order to achieve a desired objective. Similarly, it is no surprise that the main task of a framework that utilizes nodes with controllable mobility is to decide where to position and move the nodes so that a desired objective is achieved. A brief overview of the objectives that researchers have pursued so far in the literature is given in the following paragraphs.

1. **Improve network connectivity:** A simple and straightforward way one can envision the use of nodes with controllable mobility would be to sustain or improve

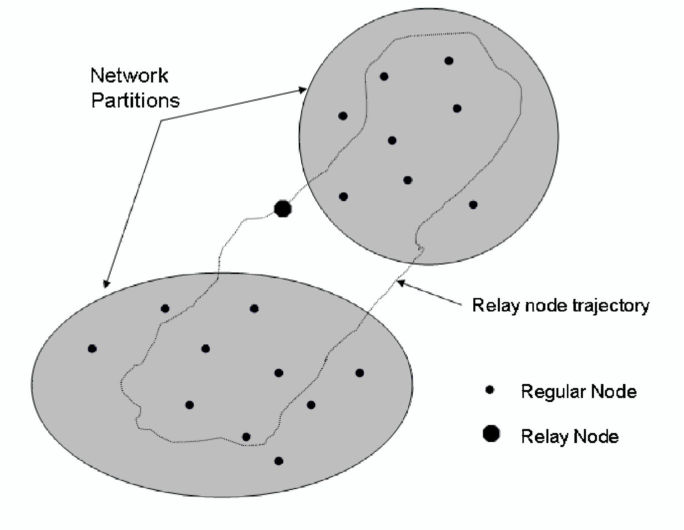


Fig. 2.1: Relay node as a bridge between network partitions

the connectivity of a MANET. From an application's perspective, the connectivity is an important property of the underlying network that enables a node to communicate with any other node in the network for a desired amount of time. In MANETs, the mobility of the nodes may lead to network partitions resulting in a temporary loss of connectivity. If the nodes are sparsely distributed, then such network partitions can last for a significant duration. Under these scenarios, routing algorithms fail to deliver packets as no routes are found to reach the destination. It is just ironical that the same node mobility, if controllable, can be harnessed to provide an easy solution. For example, in a sparse network, people have proposed the use of message ferries [6] that utilize the store-carry-forward paradigm. Analogous to the real-life ferries, a set of special mobile ferry nodes are introduced that move around the deployment area and take responsibility for carrying data between disconnected partitions as shown in figure 2.1. Such a scheme is often suited for delay tolerant applications such as sensor data collection, messaging and file transfer. Controlled mobility can also be used to improve connectivity by prolonging the lifetime of an existing route. As two neighboring nodes move further away from each other, the wireless link between them would fail disrupting the traffic flows along any route that utilizes this link. If one can introduce a relay node that could bridge the gap between the two nodes, it could sustain the existing route long enough to support any existing traffic flows. Such an approach would be crucial for providing QoS guarantees in MANETs.

2. **Improve Network Lifetime:** In MANETs, nodes usually communicate with each other in a multi-hop fashion by establishing routes through other nodes in the network. A significant improvement in network lifetime can be gained if the energy spent in data forwarding can be minimized. Such a scenario is of an even greater significance in wireless sensor networks. While they share many common characteristics, sensor networks are significantly different from traditional MANETs. Once deployed, most of the sensor nodes are generally static and collaboratively monitor the environment and process raw sensory data which is then relayed back to the sink over multiple hops. Further, sensor networks are often deployed in remote harsh environments and are designed for unattended operation. Consequently, one of the main goals of routing algorithms in sensor networks is to perform aggressive energy optimizations so as to prolong the lifetime of the network since it is often impractical to replace batteries post-deployment. A number of solutions have been proposed in the sensor networks literature that utilize mobility to achieve energy

efficiency in sensor networks. Most of these approaches either utilize mobile nodes as data sink ([7], [8]) or as data relays ([9], [10]).

3. **Improve mission utility:** Sensor networks are being widely used for remote environment monitoring and military surveillance. This task often involves operating in potentially hostile, toxic and unknown working environments making the manual and precise deployment of sensors impossible. As a result, the sensors are typically scattered by aircraft and randomly deployed which often fails to meet the adequate coverage level required for the successful completion of the task at hand. To overcome this problem, researchers have proposed mobility-assisted sensor deployment protocols where mobile sensor nodes move from densely populated areas to sparse areas to increase the coverage [11]. In addition, once deployed, the sensor nodes maybe required to relocate to different locations in response to the change in the mission or to overcome the loss of other sensors. Results indicate that having sensors nodes with controllable mobility provide high coverage under varying network conditions [11; 12].

While controlled mobility can be exploited to realize any of the aforementioned objectives, the precise goal of a given framework will be dependent on the underlying application. More so, the application might require optimizing the performance based on one objective while treating another objective as a constraint. For example, a network designer may be posed with a problem to randomly deploy a network over a predefined area such that the coverage and connectivity of the network is maximized while still providing a minimum guarantee on the lifetime of the network. Consequently, designing mobility control algorithms for communication in MANETs becomes non-trivial. Further, any such algorithm should also address a number of related issues, some of which have been highlighted in [13].

1. Should the relay nodes be explicitly introduced in the network or should we instead control the mobility of existing nodes in the network? Having explicit relay nodes have their advantage in the fact that the loss of such nodes would not effect the network's mission. However, it might be difficult to determine an optimal number of such relay nodes that may be required for the task at hand during deployment.
2. A related issue is how to define the roles of the network nodes. A simple approach would be to statically designate a given node as an application node or a relay node based on the resources available at the node. On the other hand, dynamism in node behavior might be more useful in sensor networks where all the nodes have similar form and shape. Also, it might not be possible to control the mobility of certain specialized nodes in the network.
3. The mobility of the relay nodes may also be influenced by the network's traffic pattern. For instance, the mobility of relay nodes deployed in a network with a single source-destination pair is likely to be different from those deployed in a network with multiple source-destination pairs. It might be difficult to design an algorithm that can be effectively applied to a broad range of traffic patterns.
4. Designing mobility control algorithms often require the ability to predict the movement of nodes in the network. In some scenarios where nodes follow a predefined trajectory, predicting node mobility is often a trivial exercise. However, more often

than not, a more sophisticated scheme might be required to predict the movement of nodes in the network so that the relays can be optimally positioned to achieve the desired objective. Typically, the choice of the prediction scheme is governed by the underlying application.

5. There is also the issue of deciding whether there should be a centralized entity that controls the movement of all relay nodes or if each relay node should compute its own path in a distributed fashion. Ideally, mobility control algorithms should be totally distributed with each relay node making its movement decisions locally, while collectively achieving the desirable global properties.

6. It is also important to note that node movement may itself incur certain energy costs. Consequently, the mobility control framework may need to account for it so that controlled mobility is executed only when the benefits exceed the mobility cost. On the other hand, there may also exist situations where the power source for mobility is renewable but separate from a non-renewable power source for communication [13].

As simple as the idea of using mobility controllable nodes may sound, any realistic framework that aims to utilize controllable mobility needs to address the issues outlined above. A summary of several frameworks proposed in the literature that utilize controllable node mobility is provided in table 2.1.

Reference Number	**Objective**			**Relay Type**		**Relay Trajectory**	
	Connec-tivity	Energy Efficiency	Mission Control	Implicit	Explicit	Pre-defined	Dynamic
[8]		×		×		×	
[14]		×		×		(Random)	
[7]		×		×		×	
[10]	×	×			×		×
[9]		×			×	×	
[11]	×		×	×			×
[12]	×		×	×			×
[13]	×	×		×			×
[6]	×				×	×	
[4]	×				×		×

Table 2.1: Overview of frameworks that utilize controllable node mobility

2.3 Relay nodes for conserving energy in MANETs

Communication between nodes in MANETs typically takes place in a multi-hop fashion by establishing routes through other nodes in the network. As a result, the network topology and the source-to-destination path chosen can significantly affect the amount of energy spent in data forwarding. Several energy optimization approaches such as clustering [15], topology control [16; 17], etc., have been proposed to address this problem. Alternatively, one can exploit mobility controllable relay nodes in a number of ways to minimize the energy consumption in MANETs. For example, a mobile node moving through the network deployment region can collect data from the sensor nodes over single hop radio link as and when the mobile node is within radio range of the static nodes. This naturally avoids multihop relaying and reduces the energy overhead at nodes near the base station [8].

In this dissertation, we describe a different approach that is based on the idea that transmission to a distant device may consume a disproportionate amount of power in comparison to transmission to a node in closer proximity [18]. We introduce special mobility controllable relay nodes in the network that adopt a mobility pattern similar to that of the regular nodes, and move along with them while acting as intermediate hops. The presence of such intermediate hops enables the regular nodes to transmit data at reduced power, and thus save energy as shown in figure 2.2. The proposed approach is described in detail in the next chapter.

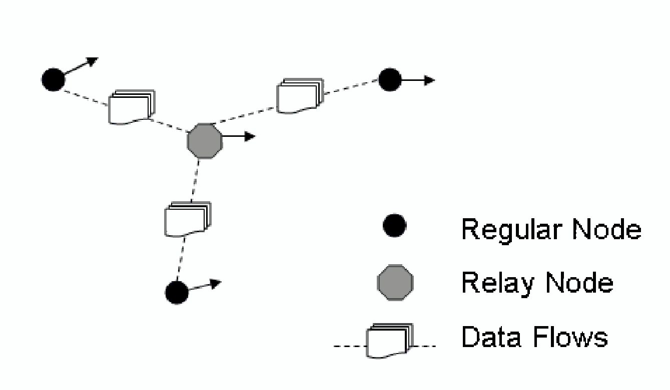


Fig. 2.2: Relay nodes as intermediate hops for conserving energy

Chapter 3

Relay Positioning Problem for Energy Optimization

In this chapter, we explore the possibility of using mobility controllable relay nodes to optimize the energy consumption in MANETs.

3.1 Introduction

It is well-known that the use of the wireless radio is one of the primary sources of energy consumption in MANETs. Typically, a node consumes much more power when it is actively transmitting data packets as compared to the packet reception and idle periods[19]. Further, transmission over shorter distances consumes significantly less power when compared to long-distance transmissions [18]. Therefore, one way to minimize the energy consumption in the network is to have the source nodes utilize intermediate nodes to transmit data over shorter hops even if the destination node is within its transmission range. This forms the basic principle behind the proposed approach.

Consider a heterogeneous network consisting of two kinds of mobile nodes–the *traditional* MANET nodes with limited energy and a few mobility controllable *relay* nodes with relatively abundant energy resources. One can envision a scenario where a certain number of specialized robots could act as relays while being outfitted with renewable power sources such as solar energy cells. Alternatively, the relay nodes could periodically return to a central base station to recharge their power sources while the traditional nodes continue to perform the tasks of the underlying application. The primary objective behind the proposed framework is to efficiently deploy these relay nodes in the network such that traditional nodes can utilize them as intermediate hops for communication. Once deployed, a relay node estimates the movement patterns of the traditional nodes in its neighborhood using a mobility prediction scheme. It utilizes the predicted network state to adopt a mobility pattern similar to that of the traditional nodes and move along with them serving as an intermediate hop for data flows. The presence of such intermediate hops would enable the traditional nodes to transmit data at reduced power and thus save energy.

As simple as the idea may sound, one has to address a number of issues in order to implement such a framework in a truly *mobile* network as discussed in chapter 2. While some aspects pertain to network planning issues such as determining the optimal number of relay nodes to deploy, there are also a plethora of protocol level challenges that need to be addressed. While transmission over shorter hops may result in considerable energy savings, it has been shown that increasing the number of nodes in an ad hoc network may decrease the wireless channel capacity [20]. Further, since the relay nodes are sophisticated devices, the cost of deploying these specialized relays is significantly higher in comparison to the traditional nodes. Any relay deployment framework such as the one proposed in this work needs to efficiently utilize the relay nodes in order to maximize the energy savings.

Trusting a relay node to forward data packets in an open network could also potentially lead to some security concerns. While it may be sufficient to ensure that the relay node cannot modify the forwarded packet without being detected at the destination, addressing this issue in a real-world setting is non-trivial and is outside the scope of this work. We however note that there are a number of different strategies that have been proposed in the literature to address the problem [21; 22; 23].

Although there are some studies in the wireless network literature that propose the use of relay nodes for minimizing the power consumption in ad hoc and sensor networks, they are either designed for environments devoid of mobility [10; 9] or consist of relay nodes that have a pre-defined trajectory (i.e., the mobility cannot be dynamically controlled by network protocols) [14]. A few other studies [13; 3] aim to minimize the energy consumption across all nodes (including the relay nodes) by often positioning the relay nodes near the geometric center of the communicating node pairs. In addition, they also often assume that the communicating nodes are relatively static.

Unlike existing approaches in the literature, in this work we consider a *truly* mobile network where both the source and destination nodes are assumed to be moving independently based on the requirements of the underlying application. We define the relay positioning problem for a heterogeneous mobile ad hoc network that compromises on the energy consumed at the relay nodes in order to optimize the energy consumed at the traditional nodes. We contend that under such a setting, the relay nodes can be positioned closer to the source nodes to minimize the energy consumed during packet transmissions.

Based on this core idea, we propose a novel relay deployment framework that utilizes mobility prediction and works in tandem with the underlying MANET routing protocol to optimally define the movement of the relay nodes. We present two instances of the relay deployment problem, together with the solutions, to achieve different goals.

Instance 1, termed *Min-Total*, aims to minimize the total energy consumed across all the traditional nodes during data transmission, while instance 2, termed *Min-Max*, aims to minimize the maximum energy consumed by a traditional node during data transmission. Our solutions also enable the prioritization of individual nodes in the network based on residual energy profiles and contextual significance.

We implement our solutions using the OPNET simulation software [24] and perform an extensive experimental analysis in order to better understand their behavior. We observe that the difference between the two instances is accentuated when the density of nodes in the network is lower. Further, the results indicate that even when the relay nodes constitute a small fraction of the total nodes in the network, the proposed framework results in significant energy savings.

3.2 Problem Formulation

In terms of energy consumption, a data flow occurring between two end nodes over a multi-hop path can be viewed as a sequence of one hop flows between adjacent nodes along the multi-hop path. Therefore, without the loss of generality, the term flow in this work refers to a single-hop flow unless specified otherwise.

Let $\mathcal{S}_\mathcal{T}$ be the set of traditional nodes and $\mathcal{S}_\mathcal{R}$ be the set of controllable mobile relay nodes in the network. Let $\mathbb{F}$ be the set of all one-hop flows in the network. Specifically, the set $\mathbb{F}$ consists of tuples $\mathbf{f_i} = (s_i, d_i, \lambda_i)$, $i = 1 \dots K$, where each $\mathbf{f_i}$ represents a flow between a communicating node pair (s_i, d_i), $s_i, d_i \in \mathcal{S}_\mathcal{T}$ with a data rate of λ_i. The energy consumed for data transmission by a flow $\mathbf{f_i}$ per unit time at node s_i is given by

$$E_i(\lambda, \delta) = \lambda_i \cdot P_T(\delta_{s_i d_i}) \tag{3.1}$$

where $\delta_{s_i d_i}$ denotes the distance between the two nodes s_i and d_i and $P_T(\delta_{s_i d_i})$ is the power needed for transmitting one unit of data between the nodes at rate λ_i. The power function is usually given by $P_T(\delta_{s_i d_i}) = a + b\delta^{\alpha}_{s_i d_i}$, where $\alpha \geq 2$, and a and b are parameters whose values depend on the characteristics of the communication channel [13].

Let $\mathbb{F}_{R_j}$ denote the *service set* of the relay node $R_j \in \mathcal{S}_{\mathcal{R}}$ that consists of all the flows that are being relayed through the node R_j. Note that $\mathbb{F}_{R_j} \subseteq \mathbb{F}$. The energy consumed per unit of time for data transmission at node s_i for a $\mathbf{f_i} \in \mathbb{F}_{R_j}$ is then given by

$$E_i^j(\lambda, \delta) = \lambda_i \cdot P_T(\delta_{s_i R_j}) \tag{3.2}$$

Based on these definitions, we now formulate the two instances of the relay positioning problem.

3.2.1 Minimize Total Energy: Min-Total

The objective of this strategy is to place a relay node in such a way as to minimize the overall energy consumption across the traditional nodes in its service set. The total energy consumed per unit time for data transmission at the traditional nodes in the absence of any relay nodes is given by:

$$E_{tot}(\mathbb{X}_{\mathbb{T}}, \mathbb{F}) = \sum_{\mathbf{f_i} \in \mathbb{F}} E_i(\lambda, \delta) \tag{3.3}$$

where $\mathbb{X}_T$ is the vector denoting the locations of the traditional nodes at any given time instant. Let $\mathbb{F}_R = \bigcup_{j \in \mathcal{S}_{\mathcal{R}}} \mathbb{F}_{R_j}$ denote the combined service set of all the relay nodes and $\mathbb{X}_R$ denote the location of the relay nodes at any given time instant. Given $\mathbb{F}_R$ and $\mathbb{X}_R$, the total energy consumed for data transmission by the traditional nodes in the presence of relay nodes is given by:

$$E_{tot}(\mathbb{X}_T, \mathbb{X}_R, \mathbb{F}, \mathbb{F}_R) = \sum_{j \in \mathcal{S}_{\mathcal{R}}} \sum_{\mathbf{f_i} \in \mathbb{F}_{R_j}} E_i^j(\lambda, \delta) + \sum_{\mathbf{f_k} \in \mathbb{F} - \mathbb{F}_R} E_k(\lambda, \delta) \tag{3.4}$$

It is clear from the above equation that the total transmission energy consumed in a relay-enabled network at any given time instant is dependent on $\mathbb{X}_R$ – the location of the relay nodes, $\mathbb{X}_T$ – the location of the traditional nodes, $\mathbb{F}$ – the set of flows in the network, and $\mathbb{F}_R$ – the combined service set of the relay nodes. Out of these, the controllable quantities are $\mathbb{X}_R$ and $\mathbb{F}_R$. Thus, with these preliminaries, we define the relay positioning problem for minimizing the total energy consumed as:

Min-Total: Given a network with $|\mathcal{S}_{\mathcal{T}}|$ traditional nodes and $|\mathcal{S}_{\mathcal{R}}|$ relay nodes, find the optimal relay node position $\mathbb{X}_R^*$ and the optimal service set $\mathbb{F}_R^*$ such that $E_{tot}(\mathbb{X}_T, \mathbb{X}_R^*, \mathbb{F}, \mathbb{F}_R^*) \leq E_{tot}(\mathbb{X}_T, \mathbb{X}_R, \mathbb{F}, \mathbb{F}_R)$ for any $\mathbb{X}_R$ and $\mathbb{F}_R$.

3.2.2 Minimize Maximum Energy: Min-Max

One of the drawbacks with the *Min-Total* formulation is that although the overall energy consumption is minimized, the energy consumption across individual nodes may be rather uneven. To address this issue, we formulate a different strategy where the objective is to minimize the maximum transmission energy consumed across the traditional nodes in the network. Given the service set $\mathbb{F}_{R_j}$ of a relay node $R_j \in \mathcal{S}_{\mathcal{R}}$, the maximum energy consumed by a traditional node in $\mathbb{F}_{R_j}$ is given by

$$E^j_{max}(\mathbb{X}_{R_j}, \mathbb{F}_{R_j}) = \max_{i \in \mathbb{F}_{R_j}} E^j_i(\lambda, \delta) \tag{3.5}$$

The objective of this formulation is to find the optimal position of every relay node $R_j \in \mathcal{S}_{\mathcal{R}}$ such that the resulting configuration minimizes E^j_{max} for the service set $\mathbb{F}_{R_j}$. Clearly, this also requires that one must define and then compute the best possible service set for every relay node. Based on these preliminaries, we define the relay positioning problem for minimizing the maximum energy consumed in a traditional node as:

Min-Max: Given a network with $|\mathcal{S}_{\mathcal{T}}|$ traditional nodes and $|\mathcal{S}_{\mathcal{R}}|$ relay nodes, find the optimal service set $\mathbb{F}^*_{R_j}$ for every relay node $R_j \in \mathcal{S}_{\mathcal{R}}$ and for this service set find the optimal position $\mathbb{X}^*_{R_j}$ such that $E^j_{max}(\mathbb{X}^*_{R_j}, \mathbb{F}^*_{R_j}) \leq E^j_{max}(\mathbb{X}_{R_j}, \mathbb{F}^*_{R_j})$ for any $\mathbb{X}_{R_j}$.

3.2.3 Incorporating Node Weights

The problem definitions described so far only take into account the energy consumption at individual nodes based on the existing data flows. In other words, solutions to both the *Min-Total* and *Min-Max* problems would favor the source nodes with higher data flow rates. However, it is also desirable that the relay nodes be deployed closer to source nodes with low residual energy even when the data rates of the existing flows through these nodes are small. Further, the network may contain certain critical nodes that are essential for the successful operation of the underlying application. In such scenarios, it is important that the relay nodes be deployed so that the resulting configuration prolongs the lifetime of such nodes while still achieving the required energy optimization.

We incorporate these factors in the problem formulation by assigning a weight w_i to each each flow $\mathbf{f_i} \in \mathbb{F}_{R_j}$. The weight w_i represents the priority assigned to the flow $\mathbf{f_i}$ and is derived from the corresponding source node. Let ϵ_i and p_i denote the residual energy and contextual priority of the source node s_i corresponding to the flow $\mathbf{f_i}$. The weight w_i is then given by

$$w_i = z_1 * \epsilon_i + z_2 * p_i \tag{3.6}$$

where z_1 and z_2 are the relative weights assigned to the residual energy and contextual priority, respectively, by the network operator. Given the node weights, the optimization function for the *Min-Total* scheme is given by

$$E_{tot}(\mathbb{X}_T, \mathbb{X}_R, \mathbb{F}, \mathbb{F}_R) = \sum_{j \in \mathcal{S}_{\mathcal{R}}} \sum_{\mathbf{f_i} \in \mathbb{F}_{R_j}} w_i \cdot E_i^j(\lambda, \delta) + \sum_{\mathbf{f_k} \in \mathbb{F} - \mathbb{F}_R} w_j \cdot E_k(\lambda, \delta) \tag{3.7}$$

Similarly, the optimization function for the *Min-Max* scheme is then given by

$$E^j_{max}(\mathbb{X}_{R_j}, \mathbb{F}^*_{R_j}) = \max_{i \in \mathbb{F}^*_{R_j}} w_i \cdot E_i^j(\lambda, \delta) \tag{3.8}$$

3.3 Towards a solution

It is evident from the definition of the relay deployment problem that the movement and positioning of the relay nodes is primarily contingent upon two network parameters - the position of the traditional nodes and the set of active flows between them. In the remainder of this dissertation, we refer to these two network parameters as the *network state information*. Given this network state information, the solution to the relay deployment problem is effectively a two step process – (1) Decide the *service set* for every relay node, and (2) Compute the optimal position of every relay node based on its service set. However, as we will see later, these two steps are rather closely intertwined which often leads to a an iterative solution strategy instead of a sequential approach.

In a mobile network, the network state is highly dynamic due to the mobility of the traditional nodes. Extending the relay positioning solutions obtained for a static network to a mobile network is non-trivial due to the following reasons:

1. The location of the traditional nodes varies continuously with time. Consequently, the optimal position of a relay node should be computed for each time instant. In other words, this translates to defining the optimal *trajectory* for a relay node based on the movement of the traditional nodes in its service set.

2. The mobility of the nodes leads to frequent changes to the route traversed by a data flow which may further add to the dynamism in the set of one-hop flows in the network.

3. The set of end-to-end flows in the network could also be time-varying based on the requirement of the underlying application.

To address these issues, we first provide an overview of the proposed mobility prediction based relay deployment framework which will then be utilized by the various relay positioning algorithms proposed in chapter 4.

3.3.1 Mobility Epochs

In order to address the issues posed by node mobility, we propose a relay deployment framework that revolves around the idea of modeling the operation of the network over fixed-length time intervals called *mobility epochs*. Decisions pertaining to the positioning and movement of the relay nodes are based on the assumption that the changes in the network state over the duration of an epoch are either negligible or predictable. It is important to note that we do not assume that the nodes remain static over the duration of an epoch. Rather, we only assume that the mobility of the nodes during an epoch can be predicted with reasonable accuracy using a given mobility prediction scheme. However, the framework does assume that the set of active flows in the network remains constant over the duration of an epoch. As a result, the decisions pertaining to the positioning of the relay node could result in a sub-optimal solution if certain flows terminate or if new flows originate during the course of an epoch. Also, the service set computation for every relay node utilizes only the local network state information in the neighborhood of the relay node collected during every epoch. We do not use the global state information because obtaining such information in a mobile network is often difficult, if not impossible.

Finding an optimal value for the duration of a mobility epoch is often non-trivial. Much depends on the extent of dynamism in the network and the frequency of topology changes. It is quite possible that the calculated epoch duration could result in a solution which may not keep abreast with the changes in the network. In general, an ideal solution should have the flexibility to have variable length epoch durations with the number of relay nodes deployed in each epoch being decided by the network state that exist then. Nevertheless, for the sake of simplicity, we assume that, once they are chosen, the epoch duration as well as the number of relay nodes deployed during an epoch remain constant.

3.3.2 Significance of Mobility Prediction

In a mobile network, relay node deployment translates to defining the trajectory for the relay node such that it always remains at the optimal position relative to the positions of the nodes in its service set. A trivial way to achieve this could be to have the traditional nodes send location updates to the relay node at every time instant throughout the duration of an epoch. The relay node would then constantly compute its optimal location coordinates based on these location updates and position itself accordingly. Though this approach would result in an optimal trajectory for the relay node, it is clearly not feasible due to the communication overheads involved.

To address this issue, the proposed framework utilizes mobility prediction algorithms to estimate the movement patterns of the mobile nodes. Once the relay node finalizes its service set, it computes its optimal position coordinates at different time instants over the duration of an epoch using the predicted location of the source nodes corresponding to the flows in its service set. Based on these predicted position coordinates, the relay node then defines its trajectory for the duration of the epoch. The trajectory is nothing but the locus of the points defined the solutions at different instances in an epoch using the predicted position coordinates. Clearly, the accuracy of the prediction algorithm along with the number of predictions during an epoch will have a significant impact on the computed positions of the relay node.

Once the relay node defined its trajectory for an epoch, it needs to convey this information to all the nodes in its service set. To this effect, the relay node informs the traditional nodes of its calculated position coordinates at a fixed number of time instants during the course of an epoch. The traditional nodes would in turn utilize this information to predict their distance to the relay node in order to appropriately set the transmit power when communicating with the relay node.

3.3.3 Mobility prediction algorithms

Most of the prediction schemes proposed in the MANET literature are based on an underlying model that defines the mobility of the nodes in the network. For example, in [25], the authors predict the position of the nodes assuming the nodes follow a random walk mobility model. While such algorithms might result in strong characterizations, they might not work well if assumptions pertaining to the underlying mobility model do not hold good. As a result, in this dissertation, we restrict our analysis to two specific mobility prediction schemes - (1) Mobility Prediction using a linear model[2] and (2) Mobility Prediction using Linear Autoregressive Models [26].

3.3.3.1 Linear Prediction Model

The linear prediction model utilizes a simple algorithm that determines the position of a node at any future time instant by assuming that it's speed and direction of movement remain constant. The linear prediction model is based on the Link Expiration Time (LET) as described in [2]. Let the location of node i and node j at time t be given by (x_i, y_i) and (x_j, y_j). Also, let v_i and v_j be the speeds, and θ_i and θ_j be the directions of the nodes i and j respectively. If the transmission range of the nodes is r, then the Link Expiration Time, D_t, of the link between the two nodes, as defined in [2], is given by

$$D_t = \frac{-(ab+cd) + \sqrt{(a^2+c^2)r^2 - (ad-bc)^2}}{a^2+c^2} \tag{3.9}$$

where

$$\begin{aligned} a &= v_i \cos\theta_i - v_i \cos\theta_j \\ b &= x_i - x_j \\ c &= v_i \sin\theta_i - v_i \sin\theta_j \\ d &= y_i - y_j \end{aligned}$$

3.3.3.2 Linear First Order Autoregressive Model

The linear first order autoregressive (AR-1) model, as defined in [27] has been shown to effectively track the movement of a mobile node irrespective of the underlying mobility model. In the AR-1 model, the mobility state of a node at time step t is defined by the column vector $\mathbf{s_t} = [x_t, \dot{x}_t, \ddot{x}_t, y_t, \dot{y}_t, \ddot{y}_t]$ where (x_t, y_t) specify the position, $(\dot{x}_t, \dot{y}_t)$ specify the velocity, and $(\ddot{x}_t, \ddot{y}_t)$ specify the acceleration of the mobile node in the x and y directions in a two-dimensional grid. The AR-1 model for the mobility state $\mathbf{s_t}$ of a node given by:

$$\mathbf{s_{t+1}} = \mathbf{A}\mathbf{s_t} + \mathbf{w_t}$$

where $\mathbf{A}$ is a 6×6 transformation matrix and the vector $\mathbf{w_t}$ is a 6×1 discrete-time zero mean white Gaussian process, with a covariance matrix $\mathbf{Q}$. The matrices $\mathbf{A}$ and $\mathbf{Q}$ are called the parameters of the model and are estimated based on a training data which allows the model to accurately characterize a wide class of mobility patterns. The parameters of the AR-1 model are updated periodically using the actual observed values. If the state information $\mathbf{s_t}$ of a node at any time step t is given, then the mobility state $\mathbf{s_{t+\tau}}$ at any time step $t + \tau$ in the future can be predicted using the following equation:

$$\mathbf{s_{t+\tau}} = \mathbf{A}^{\tau}\mathbf{s_t}$$

3.3.4 Flow Information Aggregation Mechanism

The mobility of the nodes coupled with the characteristics of the underlying application often result in a varying number of data flows in the network where a certain number of flows may be shortlived whereas certain others may remain active over a longer duration. As a result, it is important for a relay node to periodically collect information pertaining to the active data flows between the traditional nodes in its neighborhood. In order to facilitate the exchange such flow-specific information between the relay nodes and the traditional nodes, we utilize three types of control messages – (a) HELLO, (b) Service Request (SREQ) and, (c) Service Response (SREP).

The HELLO message, which is broadcast by all nodes in the network once every epoch, serves two different purposes. First, the HELLO nodes are used to exchange location and movement information across neighboring nodes. In addition, the HELLO message broadcast by a relay node also acts as a service advertisement message that informs the traditional nodes in its vicinity of the availability of a relay. Any traditional node that wishes to solicit the services of the relay node, sends an SREQ message back to the relay node in response to the HELLO message. The SREQ message consists of information regarding all the data flows that the node is actively transmitting. If a traditional node receives HELLO messages from more than one relay node, it sends an SREQ message to the closest relay node in its neighborhood. The relay node collects all the service requests from the traditional nodes during an epoch and utilizes the aggregated flow information along with the location and movement information obtained from the HELLO messages, to compute its service set for the following epoch using the algorithms described in Sections 4.1.2 and 4.2.3. Once the relay node finalizes its service set for the duration of an epoch, it sends an SREP message to all the source nodes in its service set which completes the handshake between the nodes. The SREP message also contains information pertaining to the computed trajectory of the relay node for the duration of that epoch. The traditional nodes utilize this information to estimate their distance to the relay node in order to appropriately set the transmit power when communicating with the relay node.

Chapter 4

Solution Strategies for Relay Deployment

In this chapter, we provide a detailed overview of the proposed solutions to the relay positioning problem. It is evident from the definition of the relay positioning problem that the movement and positioning of the relay nodes is primarily contingent upon two network parameters - the position of the traditional nodes and the set of active flows between them. Given this network state information, any solution to the relay deployment problem is effectively a two step process – (1) Decide the *service set* for every relay node, and (2) Compute the optimal position of every relay node based on its service set. This holds true regardless of the optimization function under consideration. In describing the solution strategies for both the *Min-Total* and *Min-Max* problems, we first show that an optimal solution can be analytically obtained when the traditional nodes in the network are static. We then extend the solutions for positioning the relay nodes in a mobile scenario using the relay deployment framework proposed in chapter 3.

In this dissertation, the service set computation for a relay node utilizes only the local network state information in the neighborhood of the relay node. We do not use the global state information for the following reasons: (1) Obtaining global information about location and data flows in a mobile wireless network can be quite expensive in terms of energy which negates the very goal of using relay nodes; (2) Even if global information is available, it may not always be possible to put the information to good use. For example, the global state may indicate that a relay node could serve better if it moves to a different neighborhood. By the time the relay node moves to that neighborhood, the state there may change undermining the need for the relay node.

In devising a solution for the relay deployment problem, we assume that every node is aware of its geographical location and mobility information. This information can be obtained either through GPS or localization methods that utilize signal strength measurements. As done in similar works [18], we also assume that radios on the network nodes are capable of dynamically adjusting their transmission power on a per-packet basis. Whenever a traditional node transmits data packets through a relay node, it dynamically changes its transmission power while keeping the data rate constant. We assume that the transmission range of a radio is equal to the reception range and that the range of a traditional node is equal to that of a relay node.

4.1 Min-Total Optimization Strategy

Recall that the optimization function for the *Min-Total* problem as defined in eq. 3.7 is given by:

$$E_{tot}(\mathbb{X}_T, \mathbb{X}_R, \mathbb{F}, \mathbb{F}_R) = E_R + E_T$$

$$\begin{aligned} \text{where,}\quad E_R &= \sum_{j \in \mathcal{S}_\mathcal{R}} \sum_{\mathbf{f_i} \in \mathbb{F}_{R_j}} w_i \cdot E_i^j(\lambda, \delta) && (4.1) \\ E_T &= \sum_{\mathbf{f_k} \in \mathbb{F} - \mathbb{F}_R} E_k(\lambda, \delta) && (4.2) \end{aligned}$$

In order to minimize E_{tot}, it is evident that we need to compute the optimal values of $\mathbb{X}_R$ and $\mathbb{F}_R$ that minimizes E_R - the total energy consumed by all the flows that are served by the relay nodes in the network. In other words, for every relay node R_j, we need to find its service set $\mathbb{F}_{R_j}$ and position $\mathbb{X}_{R_j}$ such that it minimizes $\sum_{\mathbf{f_i} \in \mathbb{F}_{R_j}} w_i \cdot E_i^j(\lambda, \delta)$.

4.1.1 Solution for a static network

Let us consider a simple scenario of a network where the traditional nodes are *static* with a fixed set of flows between them. We use an example shown in figure 4.1a to illustrate the basic idea behind our solution. Consider a network of six static traditional nodes $\{N_1, \ldots, N_6\}$ and a single relay node R_1. Suppose that there are two active end-to-end multi-hop flows in the network with data rates λ_1 and λ_2 that traverse through multi-hop paths as shown. In the scope of our problem definition, this translates to five pairs of communicating nodes representing the single hop links along the flow path. The set of one-hop flows, $\mathbb{F}$, in the network can then be defined as:

$$\begin{aligned} \mathbb{F} &= \{(N_1, N_2, \lambda_1), (N_2, N_3, \lambda_1), (N_2, N_5, \lambda_2), \\ &\quad (N_3, N_4, \lambda_1), (N_5, N_6, \lambda_2)\} \end{aligned}$$

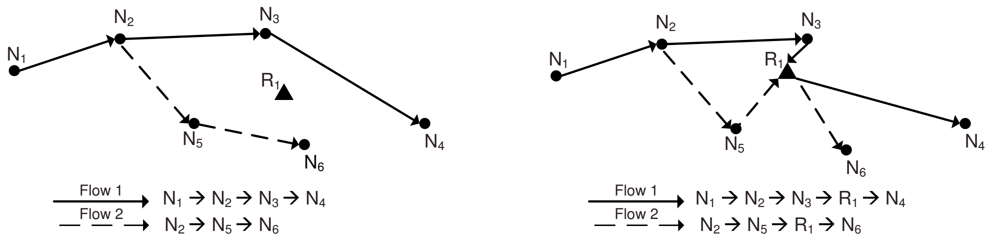
Suppose that the relay node R_1 is deployed in the neighborhood of nodes N_3, N_4, N_5, N_6. The only criteria for adding a given flow to the service set of R_1 is that the corresponding node pair should lie within the transmission range of R_1. If we assume that the flows (N_3, N_4, λ_1) and (N_5, N_6, λ_2) satisfy this criteria, then the service set of R_1 can be defined as:

$$\mathbb{F}_R^1 = \{(N_3, N_4, \lambda_1), (N_5, N_6, \lambda_2)\}$$

Further, let w_i denote the weights corresponding to each node N_i as calculated by eq. 3.6 based on their contextual priority and residual energy. In such a scenario, it can shown that the optimal position, (x_{R_1}, y_{R_1}), of the relay node R_1 is given by:

$$x_{R_1} = \frac{\omega_1 \cdot x_{N_3} + \omega_2 \cdot x_{N_5}}{\omega_1 + \omega_2} \quad y_{R_1} = \frac{\omega_1 \cdot y_{N_3} + \omega_2 \cdot y_{N_5}}{\omega_1 + \omega_2}$$

where (x_{N_3}, y_{N_3}) and (x_{N_4}, y_{N_4}) represent the position coordinates of the nodes N_3 and N_4 respectively, and $\omega_i = w_i * \lambda_i$. If $\omega_1 > \omega_2$, then R_1 should be deployed on a straight line joining the two source nodes N_3 and N_5, closer to N_3 to an extent determined by their node weights as shown in figure 4.1b.



(a) Data flows without using a relay node

(b) Data flows after deploying the relay node

Fig. 4.1: Deploying a relay node in a static network with fixed flows

If the relay node were to serve more than two nodes, it can be argued that the optimal position of the relay node should be the weighted geometric center of the source nodes in its service set for cases where the exponent α in the power function P_T equals two. For scenarios where $\alpha > 2$, it is challenging to obtain a closed form expression for the optimal relay position. In this work, for cases where $\alpha > 2$, we use the solution corresponding to $\alpha = 2$ as an approximation. In other words, if the service set of a relay node R_i is given by $\mathbb{F}_R^i = \{\mathbf{f_i} = (S_i, D_i, \lambda_i) | i = 1, \ldots, k\}$, then the position of R_i that minimizes equation 3.7 is defined in our work as:

$$x_{R_i} = \frac{\sum_i \omega_i \cdot x_{S_i}}{\sum_i \omega_i} \quad y_{R_i} = \frac{\sum_i \omega_i \cdot y_{S_i}}{\sum_i \omega_i} \tag{4.3}$$

where $\omega_i = w_i * \lambda_i$.

Now, let us consider the service sets $\mathbb{F}_{R_1}^2 = \{(N_3, N_4, \lambda_1)\}$ and $\mathbb{F}_{R_1}^3 = \{(N_5, N_6, \lambda_2)\}$. Since only a single source is being served in these cases, the relay can be placed as close as possible to the source node, in order to save maximum energy in the source node. To account for the physical constraints in such scenarios, we place the relay nodes at a distance of $10m$ from the source node.

Ideally speaking, the energy savings achieved when the relay is optimally placed in all the three service sets should be estimated and the set that gives the maximum power savings should actually be taken up for service. If there are n nodes in the neighborhood of a relay node, then a total of $O\left(2^n\right)$ service sets are possible. Evaluating all of them will result in an exponential time complexity. To avoid this, we present a fast heuristic algorithm in Section 4.1.2 that determines a good service set for a relay node together with the optimal relay node location for this service set. Of course, the trade-off achieved by the fast heuristic is that the service set selected may not be the optimum in terms of the potential energy savings.

4.1.2 Solution for a mobile network

The solution for *Min-Total* in a truly mobile network revolves around first determining the optimal service set of a relay and then computing the relay's optimal location based on this service set for different epoch instances. Recall that a flow can be added to a relay's service set as long as the communicating node pair corresponding to that flow is in the range of the relay node. In a mobile setting, this requirement translates to making sure that the communicating nodes continue to remain in the range of the relay node throughout the epoch duration. This constraint can be verified by examining the relative distance between the *predicted* position of the relay node and the communicating node pair at various time instants over an epoch duration. However, note that the movement of the relay node can be predicted only when its service set has been finalized. This creates a cyclic dependency wherein computing a relay node's service set depends on the movement of the relay node, which in turn depends on the service set of the relay node.

In order to address this problem, we propose a greedy heuristic algorithm shown in Figure 4.2 that iterates over different possible service sets of a relay node and jointly determines the relay node's service set and trajectory. This algorithm is executed distributedly by every relay node in the network at the beginning of each epoch. As the first step, the relay node discards all the service requests whose corresponding node pair is not presently in its transmission range[1]. It then adds the flow with the maximum weight in the request set to its service set and defines its trajectory assuming that this is the only flow that the relay would serve during the epoch. The trajectory of the relay node is defined using equation 4.3 based on the predicted positions of the nodes in its service set.

```
procedure $R_i$.ComputeServiceSet(SREQSet $\mathbf{Q_i}$)
    Input: $\mathbf{Q_i} = \{\mathbf{f_j} = (s_j, d_j, \lambda_j), j = 1 \ldots k\}$    ▷ The set of received SREQs
    Output: $\mathbf{F_i}$, where $\mathbf{F_i} \subseteq \mathbf{Q_i}$    ▷ The service set of $R_i$

    $\mathbf{Q_i} \leftarrow \mathbf{Q_i} - \mathbf{Q_i^*}$,   where $\mathbf{Q_i^*}$ is the set of SREQs whose communicating node pair is presently not in the transmission range of $R_i$
    for every $\mathbf{f_j} \in \mathbf{Q_i}$ do
        Compute $P_j \leftarrow$ the power consumed $\mathbf{f_j}$ .
        Predicted the position of the node pair $(s_j, d_j)$ at the end of the epoch
    end for
    Sort $\mathbf{Q_i}$ in the descending order of the power consumption

    $\mathbf{F_i} \leftarrow \{\mathbf{f_1}\}$    ▷ Add the flow with maximum power consumption to the service set
    Define the trajectory of $R_i$ based on the predicted positions of the node pair $(s_1, d_1)$
    $E_i \leftarrow$ the energy savings by serving a single flow
    for all $j > 1$ and $\mathbf{f_j} \in \mathbf{Q_i}$ do
        $\hat{\mathbf{F}}_\mathbf{i} \leftarrow \mathbf{F_i} \cup \{\mathbf{f_j}\}$    ▷ Add the next flow to the service set
        Define the trajectory of $R_i$ based on the predicted positions of the nodes in the service set $\hat{\mathbf{F}}_\mathbf{i}$
        if $(s_j, d_j)$ are not predicted to be within the transmission range of $R_i$ then
            Continue    ▷ Discard this flow and continue on to the next flow
        end if
        $\hat{E}_i \leftarrow$ the energy savings based on the service set $\hat{\mathbf{F}}_\mathbf{i}$
        if $\frac{(\hat{E}_i - E_i)}{E_i} < \Delta_{th}$ then    ▷ Check if the gain in energy savings is less than a minimum threshold $\Delta_{th}$
            Continue    ▷ Discard this flow and continue on to the next flow
        else
            $\mathbf{F_i} \leftarrow \hat{\mathbf{F}}_\mathbf{i}$
            $E_i \leftarrow \hat{E}_i$
        end if
    end for
end procedure
```

Fig. 4.2: Min-Total relay positioning algorithm for a mobile scenario

The relay node then iteratively attempts to add more flows to its service set based on the descending order of their weights. Every time a new flow is added, the relay node re-defines its trajectory and verifies that both the source and the destination nodes corresponding to the flows in its service set are predicted to be within its transmission

---

[1] Even though the relay node received the SREQ from the source, it is possible that the destination may not presently be in the relay node's transmission range

range. If this constraint is not satisfied, then the newly added flow is discarded and the next flow in the request set is considered. A flow is also discarded if the gain in the total energy savings is negligible when compared to the energy savings under the previous service set. The algorithm terminates after all the flows in the service request queue have been processed once. Note that this heuristic evaluates just n of the 2^n possible service sets and hence runs fast. The drawback is that the service set chosen by this heuristic may not always be optimal in terms of the total energy that can be saved across the traditional nodes.

4.2 Min-Max optimization strategy

The optimization function for the *Min-Max* problem as defined in chapter 3 is given by:

$$E^j_{max}(\mathbb{X}_{R_j}, \mathbb{F}_{R_j}) = \max_{i \in \mathbb{F}_{R_j}} w_i \cdot E_i(\lambda, \delta) \tag{4.4}$$

The objective under the *Min-Max* formulation is to minimize $\max_{i \in \mathbb{F}_{R_j}} w_i \cdot E^j_i(\lambda, \delta)$. As indicated earlier, $E^j_i(\lambda, \delta)$ is given by $E^j_i(\lambda, \delta) = \lambda_i \cdot P_T(\delta_{s_i R_j})$, where $\delta_{s_i R_j}$ represents the distance between the relay node R_j and the source node s_i. Substituting this in the objective function, we get

$$E^j_{max}(\mathbb{X}_{R_j}, \mathbb{F}_{R_j}) = \max_{i \in \mathbb{F}_{R_j}} \omega_i \cdot P_T(\delta_{s_i R_j}) \tag{4.5}$$

where $\omega_i = w_i * \lambda_i$. Notice that the power required by a traditional node s_i to transmit data packets to the relay node R_j is proportional to $\delta^{\alpha}_{s_i R_j}$. Minimizing the power consumed during transmission is same as minimizing the distance between the relay node and traditional node. The keen reader will recognize that if there are several nodes in the service set, then the min-max solution reduces to placing the relay node in the center of a circle that minimally encloses all the traditional nodes. In other words, for a given service set, the min-max formulation reduces to the *weighted minimum circle problem.*

1

Step 0 Pick any two points, P_a and P_b, and go to Step 1.

Step 1 If the AB circle covers all points, stop. Else let P_c be the weighted farthest point from the center of the AB circle. Rename the points if necessary so that the AC circle is not smaller than the BC circle. Go to Step 2.

Step 2 If P_b is covered by the AC circle, rename P_c as P_b and go to Step 1. Else go to Step 3 with P_a, P_b and P_c.

Step 3 If the ABC circle covers all points, stop. Else let P_d be the weighted-farthest point from the ABC circle center. Go to Step 4.

Step 4 Rename P_a, P_b and P_c if necessary so that the AD circle is larger than or equal to the BD and CD circles. If both P_b and P_c are covered by the AD circle, rename P_d as P_b and go to Step 1. Else, the solution of the four point problem is definitely a three point circle. The following cases can arise:

- **(i)** P_b, but not P_c is covered by the AD circle. The minimum circle is either the ACD or the BCD circle.
- **(ii)** P_c, but not P_b is covered by the AD circle. The minimum circle is either the ABD or the BCD circle.
- **(iii)** Neither P_b, nor P_c is covered by the AD circle. The minimum circle is the ABD or the ACD or the BCD circle.

Find the correct three point circle, rename the points as P_a, P_b and P_c, and go to Step 3.

Fig. 4.3: Algorithm for weighted minimum circle problem

4.2.1 Weighted minimum circle problem

The weighted minimum circle problem on a plane is defined as follows. Let $S = \{P_i = (x_i, y_i) | i = 1 \dots n\}$ be a set of distinct points in the XY plane. Let $\Omega = \{\omega_i, i = 1 \dots n\}$ be a positive-valued set where each ω_i represents a weight corresponding to every point $P_i \in S$. The weighted minimum circle problem is then defined as

$$\min_{X \in \mathcal{R}^2} \max_{P_i \in S} \omega_i \cdot d(X, P_i) \tag{4.6}$$

where X is a variable point in the plane $\mathcal{R}^2$, and $d(X, P_i)$ is the Euclidean distance between P_i and X. The special case where $\omega_i = 1, \forall i$, is equivalent to the geometrical problem of covering S with a circle of minimum radius. In order to understand the solution for the general case, let us first define a few terms.

AB equicircle: Given two points $P_a, P_b \in S$ such that $\omega_a > \omega_b$, let

$$L(P_a, P_b) = \{X : \omega_a d(X, P_a) = \omega_b d(X, P_b)\} \tag{4.7}$$

$L(P_a, P_b)$, known as the *AB equicircle*, is the locus of all points that are at the same weighted distance from both P_a and P_b. If the ratio, $r = \frac{\omega_b}{\omega_a} = 1$, then $L(P_a, P_b)$ is a straight line – the perpendicular bisector of the line joining P_a and P_b. If $r \neq 1$, then $L(P_a, P_b)$ is a circle with radius $\frac{r d(P_a, P_b)}{|1-r^2|}$, and center $\frac{P_a - r^2 P_b}{1-r^2}$.

AB circle: Given two points $P_a, P_b \in S$, the minimum circle covering the two points is called the *AB circle*. The center of the AB circle lies at the intersection of the AB equicircle and the line joining P_a and P_b. Specifically, the AB circle is centered at $\frac{\omega_a P_a + \omega_b P_b}{\omega_a + \omega_b}$ and has a radius $\frac{\omega_a \omega_b d(P_a, P_b)}{\omega_a + \omega_b}$.

Given, these two definitions, the following steps show how to determine the optimal solution for the 2 and 3 point weighted minimum circle problem:

Step1: For a weighted minimum circle problem with two points P_a and P_b, the optimal solution lies at the center of the AB circle.

Step2: For a weighted minimum circle problem with three points P_a, P_b and P_c, the optimal solution is determined either by one of the pair of points (P_a, P_b) or (P_b, P_c) or (P_a, P_c), or the optimal solution is given by the ABC circle whose center lies at the intersection of AB-equicircle, BC-equicircle and AC-equicircle.

Based on the above steps, an algorithm for the weighted minimum circle problem for a finite number of points on the plane is given in figure 4.3 [28]

4.2.2 Solution for a static network

Let us revisit the example of a simple scenario as shown in Figure 4.1a where the traditional nodes are *static* with a fixed set of flows between them. As before, there are three choices for the service set. Unlike *Min-Total*, choosing the service set in *Min-Max* may not be as straightforward for the following reason. Since the objective is to minimize the maximum power consumed in a service set, it is clear that a service set with only one node (say $\mathbb{F}^2_{R_1} = \{(N_3, N_4, \lambda_1)\}$) could be considered optimal since it will result in the least possible power consumption at the node being serviced (here N_3). However, such a service set may have only limited significance as only node is being served. Therefore, a scheme for ranking different service sets is required in order to pick the best service set for a relay node.

One possible ranking scheme could be to prioritize the service sets as per the total weight of the nodes contained in the set. The relay node could then pick and serve the service set with the maximum total weight. In case the nodes in this service set are outside the range of the relay node, the next best set can be chosen and so on. In our example, it is obvious that the service set with the maximum weight would be $\mathbb{F}^1_{R_1} = \{(N_3, N_4, \lambda_1), (N_5, N_6, \lambda_2)\}$. Once the service set of the relay node R_1 is finalized, computing the min-max position for R_1 is a straight-forward invocation of the weighted minimum circle problem where the input set S is derived from the position coordinates of the source nodes N_3 and N_4 as given below:

$$S = \{(x_{N_3}, y_{N_3}), (x_{N_4}, y_{N_4})\}\Omega = \{\omega_3, \omega_4\}$$

where $\omega_3 = \lambda_3 * w_3$ and $\omega_4 = \lambda_4 * w_4$.

Any strategy devised for solving the *weighted minimum circle* problem can be applied to solve for the optimal relay node coordinates *Min-Max* problem too. Several researchers have proposed solutions for the weighted minimum circle problem and in this work, we adopt the solution which is based on the algorithm proposed by Hearn et. al in [28]. The solution is based on a simple idea that considers every circle defined by two or three points, and finds the smallest of these circles that contains every point. In addition to the simplicity of the algorithm, it has also been shown to have a fast average case running time in comparison to several other approaches proposed in the literature [28].

While the ranking scheme discussed above seems reasonable, it can be shown that optimal service set selection with such a ranking translates the *maximum subset problem* which is NP-Complete [29]. In section 4.2.3, we present a fast heuristic algorithm that determines the optimal service set for a relay node in polynomial time.

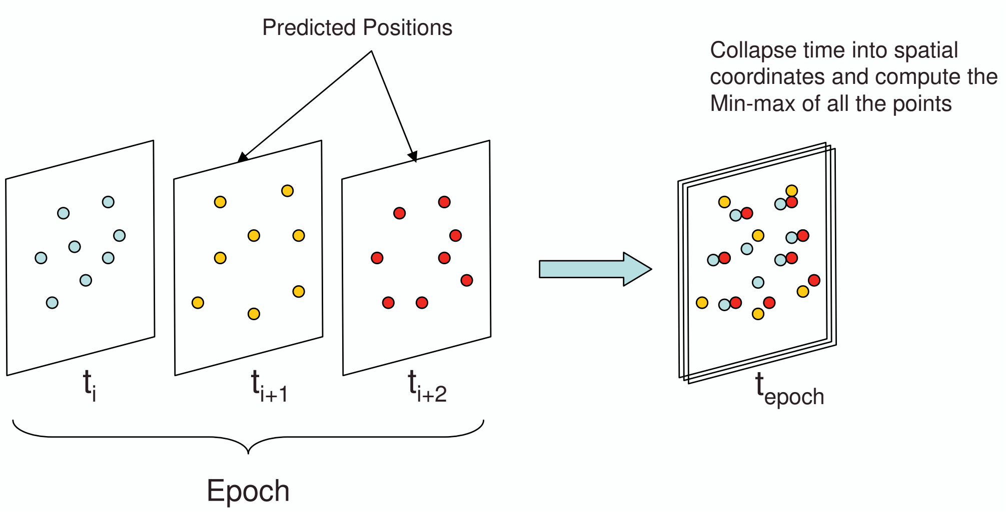


Fig. 4.4: Solution strategy for a mobile network

4.2.3 Solution for a mobile network

In order to find the optimal service sets and the optimal trajectory under the *Min-Max* formulation, one may be tempted to follow a similar algorithm as done in *Min-Total*, i.e., order the flows based on their weight and include each serviceable flow one by one as long as the energy savings increase. The trajectory of the relay node would then be the locus of the optimal locations for the resulting service set at various instances in an epoch. However, in practice such an approach may not be possible under the *Min-Max* formulation since it has been shown that the velocity of the center of the weighted minimum circle of the mobile nodes is unbounded in $\mathbf{R}^2$ [30; 31; 32]. Perhaps the most significant work related to the mobile weighted minimum circle problem is the work of Bereg et al. [30] who first raised some of the fundamental questions relating to bounding the velocity of a mobile facility function. They show the velocity of the mobile Euclidean 1-centre is unbounded in $\mathbf{R}^2$. They further show that no bounded-velocity approximation function can guarantee an approximation factor of λ to the weighted minimum circle problem for an arbitrary λ and a fixed maximum velocity v_{max} that is independent of λ. In other words, this implies that a relay node may not be able to follow the trajectory formed by the optimal locations at various instances in an epoch interval.

Therefore, unlike the *Min-Total* problem, we adopt a different approach to compute the min-max solution in a mobile setup. Figure 4.4 illustrates the idea behind the proposed min-max solution for a mobile network. To start with, the service set of the relay node consists of all the flows whose communicating node pairs are currently in the transmission range of the relay node. For every source node in its service set, the relay node then predicts their position at a fixed number of time instants over the course of an epoch. However, instead of computing a separate min-max solution for each of these time instants, we consider the node locations at all these instants as different points and compute a single min-max solution for the entire epoch using the algorithm described in [28]. Based on this computed position for the relay node, we check to see if all the communicating node pairs are predicted to be within the transmission range of the relay node. If not, the flow with the lowest weight is removed from the service set and the process iterates until either all the nodes in the service set are predicted to be within the transmission range of the relay node or the service set consists of a single flow. Figure 4.5 shows the min-max algorithm for a mobile scenario as described above. If the service set consists of a single flow, the relay is positioned as close as possible to the source node, in order to save maximum energy in the source node.

```
procedure $R_i$.MinMaxDynamic(SREQSet $\mathbf{Q_i}$)
    Input: $\mathbf{Q_i} = \{\mathbf{f_j} = (s_j, d_j, \omega_j), j = 1 \ldots N_i\}$        ▷ The set of received SREQs
    Output: $\mathbf{F_i}$, where $\mathbf{F_i} \subseteq \mathbf{Q_i}$        ▷ The service set of $R_i$
            $\mathbf{X}_{R_i} = (x_i, y_i)$        ▷ The optimal position of $R_i$

    $\mathbf{Q_i} \leftarrow \mathbf{Q_i} - \mathbf{Q_i^*}$,   where $\mathbf{Q_i^*}$ is the set of SREQs whose communicating node pair is presently not in the transmission
                    range of $R_i$
    $S_i \leftarrow \{\emptyset\}$,   where $S_i$ represents the input set of weighted points in the plane to the static min-max problem
    while $|\mathbf{Q_i}| > 1$ do
        for every $\mathbf{f_j} \in \mathbf{Q_i}$ do
            $(s_j^k, d_j^k) = \text{Predict}(s_j, d_j), \quad k = 1 \ldots m$        ▷ Predict the position of the node pair at $m$
                                                                                      different time instants within the epoch
            $S_i \leftarrow S_i \cup \{(s_j^k, d_j^k, \omega_j)\}$        ▷ Add all the predicted positions to $S_i$
        end for
        Sort $\mathbf{Q_i}$ in the descending order of the power consumption

        $\mathbf{X} \leftarrow \text{Compute-MinMax}(S_i)$        ▷ Compute the min-max position
        $\mathcal{D}_j \leftarrow \max(\delta_{R_i s_j}, \delta_{R_i d_j}), \quad j = 1 \ldots N_i$        ▷ Check for reachability conditions for all nodes
                                                                                      based on the computed min-max position $\mathbf{X}$
        $\mathcal{D} \leftarrow \max_j \mathcal{D}_j$
        if $\mathcal{D} <$ Transmission Range of $R_j$ then
            $\mathbf{F_i} \leftarrow \mathbf{Q_i}$
            $\mathbf{X}_{R_i} \leftarrow \mathbf{X}$
            Stop
        else
            $Q_i \leftarrow Q_i - \{f_1\}$        ▷ Remove the flow with the lowest weight
        end if
    end while
    $\mathbf{F_i} \leftarrow \mathbf{f_1}$        ▷ The service set consists of only one flow
    Define the trajectory of $R_i$ based on the predicted positions of the node pair $(s_1, d_1)$
end procedure
```

Fig. 4.5: Algorithm for Min-Max positioning of the relay node in a mobile network

Chapter 5

Experimental Evaluation

In this chapter, we present results from a detailed simulation evaluation of the relay positioning strategies proposed. First, we describe the details of our implementation followed by a brief description of the simulation scenario.

5.1 Implementation Details

We implemented the the relay deployment framework in OPNET Modeler version 12.0 [24] as a mobility control layer that operates in tandem with the routing protocol at the network layer. In order to implement the flow aggregation mechanism employed by the relay nodes, each traditional node in the network maintains a *link cache* that stores information regarding all the active links between the node and its one hop neighbors. Specifically, each entry in the the link cache contains flow state information pertaining to all the end-to-end flows that traverse the link. Note that this state information maintained at each node along the route of an end-to-end flow is *soft state* and thus automatically expires when either the flow has terminated or the route taken by the flow is modified.

When a traditional node successfully establishes a service connection with a relay node for a given link, it updates the corresponding entry in its link cache and redirects the packets from all the flows traversing that link to the relay node. The implementation of such traffic redirections is acheived using the underlying MANET routing protocol employed by the nodes. The results presented in this dissertation utilize Dynamic Source Routing [33] as the underlying routing protocol. In this case, redirecting the packets of a data flow to a relay node is achieved by modifying the source route in the data packets itself. Note that the actual source of the end-to-end flow could be totally unaware of such route modification along the intermediate hops. Such a design enables the proposed framework to be inter-operable with most of the standard MANET routing protocols proposed in the literature as long as both the relay nodes and the traditional nodes employ the same routing protocol.

We compared the performance of the relay deployment framework using two mobility prediction schemes – AR-1 and linear prediction model. The linear prediction model, as described in section 3.3.3, is a very simple algorithm that determines the position of a node at any future time instant by assuming that it's speed and direction of movement remain constant. In Opnet, the movement pattern of nodes in a simulation scenario is usually defined using a trajectory data file. In the linear prediction model that is implemented in the Opnet modeler, a node's future position is simply looked up in the trajectory file and interpolated between the two closest points in time. The AR-1 model was implemented in MATLAB and the prediction algorithm was invoked from Opnet by establishing a callback interface between the MATLAB runtime engine and the Opnet discrete event simulator.

5.2 Simulation Setup

We simulated the relay positioning algorithm by randomly deploying a varying number of mobile nodes across a fixed network area. The mobility of the nodes was modeled using the Gauss-Markov mobility model [34] with the value of the tuning parameter α set to 0.7 with an average node speed of 5m/s. Each simulation run was carried out for a simulated duration of 1 hour and the results presented are averaged over ten simulation runs. When using the AR-1 based prediction scheme, the model was initially trained on a data set consisting of 300 data points. During the course of the simulation period, the parameters of the model were updated with the observed values at regular intervals of 10 seconds.

Source	Destination	Flow Rate
N1	N14	20 kbps
N3	N16	5 kbps
N5	N18	5 kbps
N7	N20	5 kbps
N9	N22	2 kbps
N11	N24	2 kbps
N13	N2	5 kbps
N15	N4	5 kbps
N17	N6	5 kbps
N19	N8	20 kbps
N21	N10	2 kbps
N23	N12	2 kbps

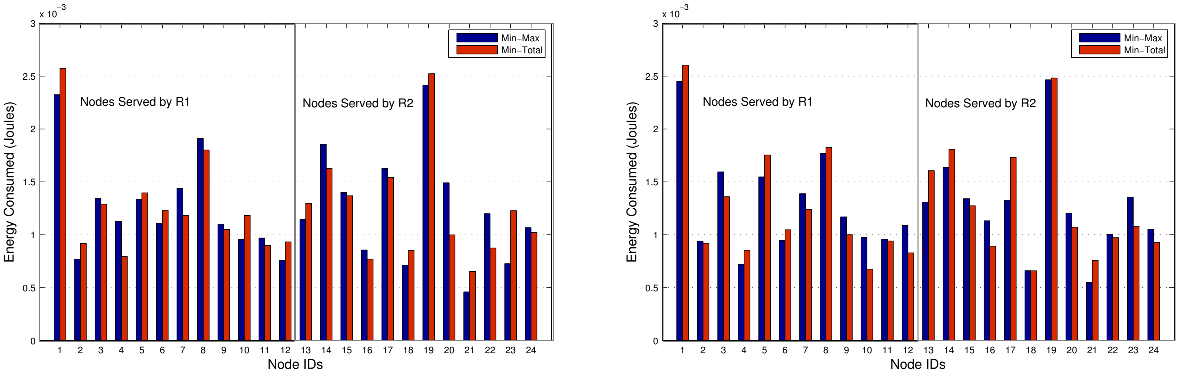
Table 5.1: List of end-to-end flows used in simulation study

The simulations utilize Opnet's implementation of the 802.11 MAC protocol for wireless transmissions which assumes a free space path loss model and a channel data rate of 11 Mbps. The maximum transmit power of the nodes was set to 0.2169 mW which translates to a transmission range of approximately 250 meters under a packet reception threshold of -95 dBm. The RTS/CTS threshold was set to 512 bytes and the transmission energy consumed for every packet transmission was computed based on the transmit power associated with the packet, actual packet size and the channel data rate. As done in similar works [18], we always transmit RTS/CTS packets at maximum transmission power to avoid the potential problem of nodes not being able to listen to the RTS/CTS packets with dynamic transmission power control.

5.3 Min-Max Vs. Min-Total

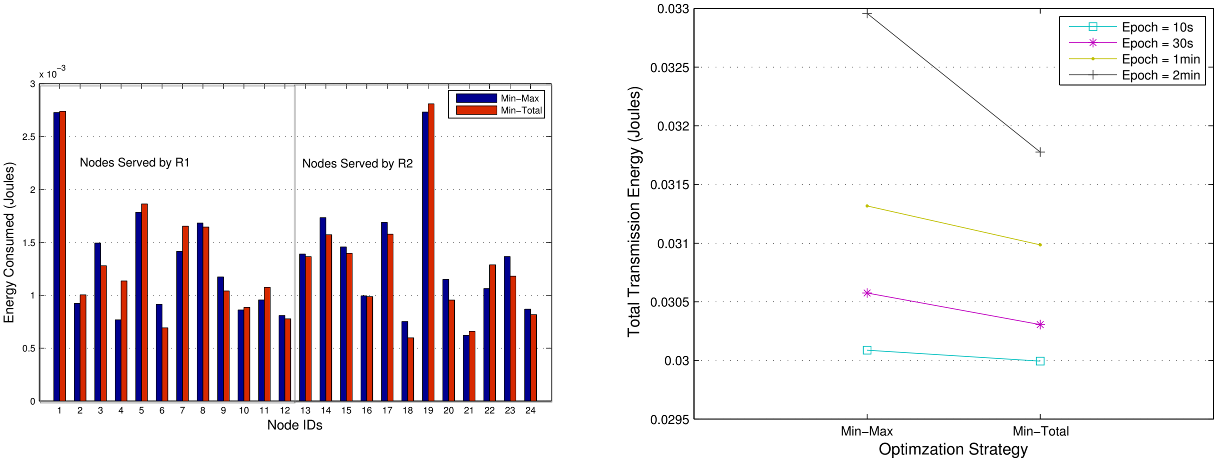
In the first set of experiments, we compare the performance of the *Min-Max* and the *Min-Total* algorithm by simulating a network consisting of 24 traditional nodes and two relay nodes in a 500mx250m network area. In order to ensure uniformity in the service sets across the two algorithms, the network was portioned into two groups containing 12 nodes each and one relay node was assigned to serve requests from each group of nodes. All the nodes in the network were assigned an equal contextual weight with each node having a similar initial battery power. A total of 12 end-to-end flows were setup between the groups as shown in table 5.1. For the sake of brevity, the results presented in this section are based on the Opnet prediction scheme. The simulation results for the AR-1 prediction schemes were also along the similar lines.

Figure 5.1a shows the transmission energy consumed across all the nodes in the network with an epoch duration of 10 seconds. The nodes $N_1, \ldots, N_{12}$ were served by the relay node R_1 and the nodes $N_1, \ldots, N_{12}$ were served by the relay node R_2. From the



(a) At individual nodes with epoch = 10s

(b) At individual nodes with epoch = 30s



(c) At individual nodes with epoch = 60s

(d) Total transmission energy consumed

Fig. 5.1: Transmission energy consumption in the network

graph, we observe that nodes N_1 and N_{19} account for the maximum energy consumption in their respective groups. Consequently, the *Min-Max* algorithm results in lower energy consumption across these two nodes in comparison to the *Min-Total* algorithm. On the other hand, the *Min-Total* algorithm does result in better optimization of the total energy consumed in the network as shown in figure 5.1d. A similar observation can drawn from the graphs in figures 5.1b and 5.1c where the maximum energy consumed within each group is the lowest under the *Min-Max* algorithm while the *Min-Total* optimization scheme results in a lower overall energy consumption in the network. However, as the epoch duration is increased, we observe that the performance of the *Min-Max* algorithm degrades. This can be attributed to the *temporal reduction* strategy employed by the *Min-Max* algorithm. Recall that in the *Min-Max* algorithm, we compute a single optimal position for the relay node for the entire epoch. As the duration of an epoch increases, the degree of variation in the network state of the traditional nodes in the service set of a relay node also increases. Consequently, the computed position of the relay node is no longer accurate resulting in sub-optimal performance.

5.4 Effects of temporal reduction

In order to better understand the nuances of temporal reduction, we implemented an alternate version of the *Min-Total* algorithm in which the relay nodes are deployed using a strategy similar to the *Min-Max* algorithm. In this approach, which we refer to as *Min-Total-Alt*, a single optimal position for the relay is computed for the entire epoch using the predicted positions of the nodes in its service set. Figure 5.3 shows the comparison between the two approaches. We observe that while the *Min-Total*

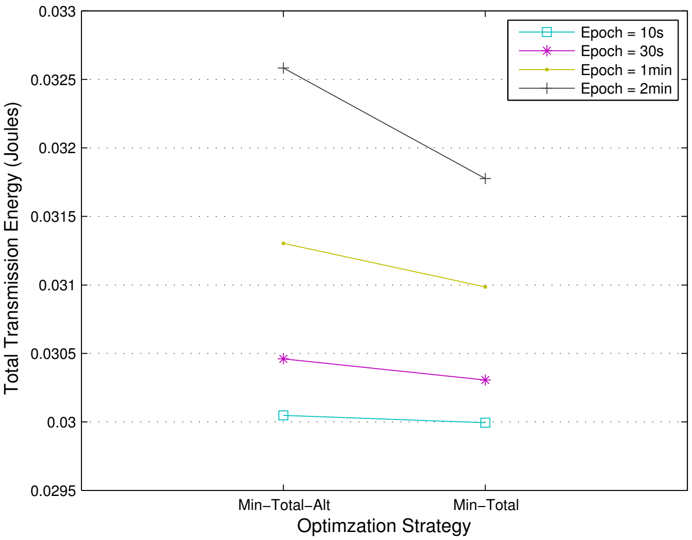
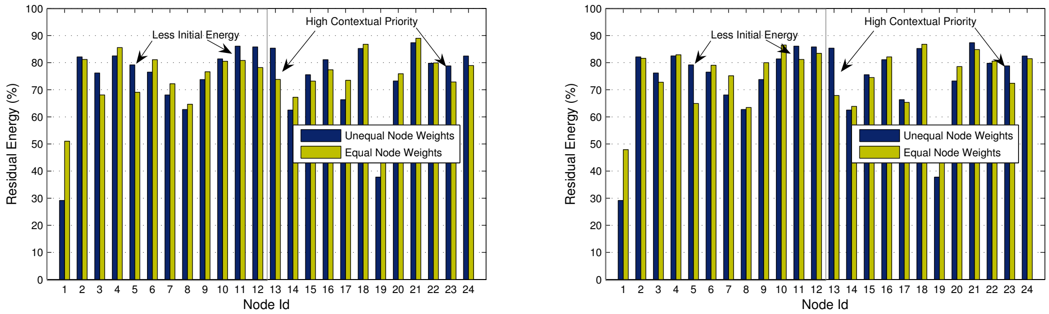


Fig. 5.2: Effect of temporal reduction

algorithm almost always results in a better performance in comparison to *Min-Total-Alt*, the difference is negligible when the epoch duration is small. As we increase the epoch duration, the performance of the *Min-Total-Alt* scheme degrades since the computed positions of the relay nodes are sub-optimal.

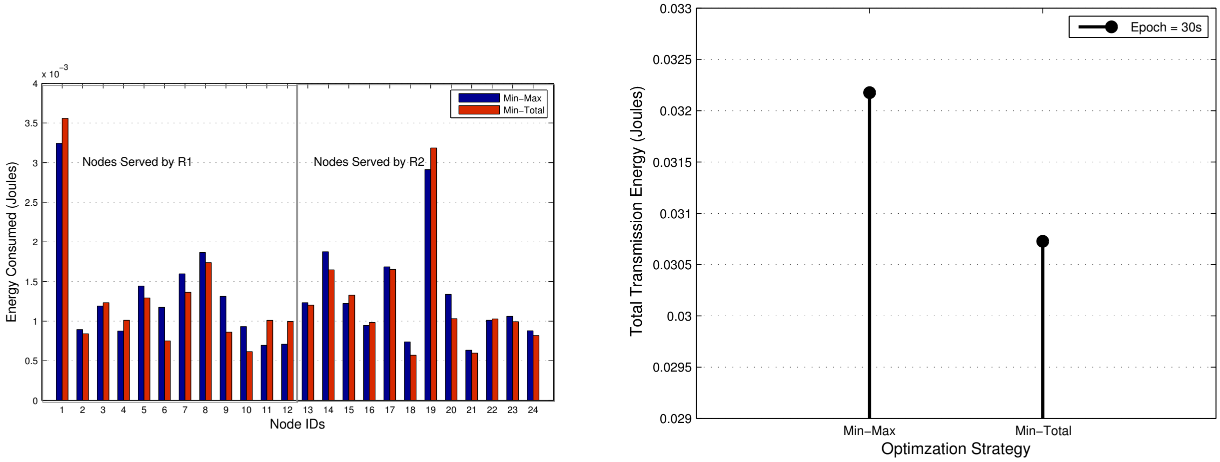
5.5 Effect of varying node weights

In the next set of experiments, we evaluate the performance of the different optimization algorithms when the nodes have unequal weights. The number of nodes and the set of flows in the network were similar to section 5.3. However, we assigned specific weights to four different nodes in order to prioritize them. Recall that the node weight is computed based on two factors - (a) residual energy and (b) contextual priority. Node N_5 and N_{11} were assigned an initial energy reserve of 0.003 Joules where are rest of the nodes were assigned a starting energy reserve of 0.005 Joules. On the other hand,



(a) Residual energy profile with Min-Max Algorithm

(b) Residual energy profile with Min-Total Algorithm



(c) Energy consumed at individual nodes with epoch = 30s

(d) Total transmission energy consumed

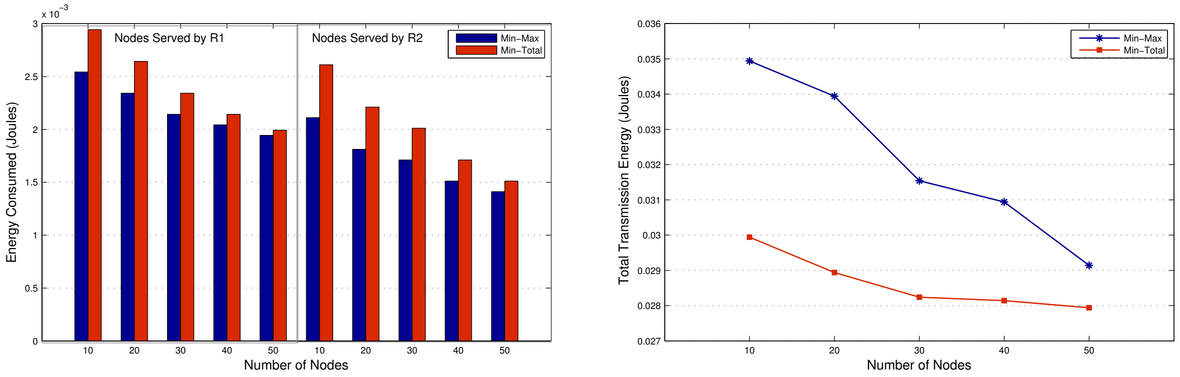
Fig. 5.3: Effect of node weights on energy consumption

nodes N_{13} and N_{23} were assigned a contextual priority of 5 and 8 respectively, compared to the other nodes which had a contextual priority of 1. Both the residual energy and contextual priority were given equal significance by setting the parameters z_1 and z_2 to 1. The epoch duration for this set of simulations was set to 30 seconds.

Figures 5.3a and 5.3b show the residual energy profile of the nodes in the network at the nodes after a simulated duration of one hour under the two different optimization algorithms. We observe that both algorithms indeed reduce the energy consumption at the prioritized nodes. For instance, both nodes N_5 and N_{11} have higher residual energy in the service area of the first relay node while the same is true for nodes N_{13} and N_{23} in the other group. Note that while both the algorithms do favor nodes with higher weights, they do so to a varying degree while still meeting the underlying optimization criteria. This can be verified in figures 5.3c and 5.3d which show that the *Min-Max* algorithm minimizes the maximum energy consumed in each group while the *Min-Total* algorithm leads to lower energy consumption across the entire network.

5.6 Effect of varying node density

Recall that both *Min-Max* and *Min-Total* optimization schemes are based around the idea of appropriately positioning the relay nodes to enable data transmissions over shorter hops. While the *Min-Max* scheme focuses on individual node fairness, the *Min-Total* scheme looks at the network as a whole. Even though we were able to observe such a behavior in the experimental results discussed so far, it is not clear what network scenarios would accentuate these differences. In order to understand that, we conduct a set of experiments in which we vary the density of traditional nodes in the network



(a) Maximum energy consumed in the network

(b) Total transmission energy consumed

Fig. 5.4: Effect of varying node density (Epoch = 30s)

while deploying a fixed number of relay nodes. We set up 10 flows in the network with the data rate of each flow set to 5 kbps. The epoch duration was set at 30 seconds.

Figure 5.4 shows the results from the simulations with two relay nodes. From the results it is clear that when the node density is high, both the schemes have similar performance. However, the difference in their behavior is clearly evident when the node density is less. In a sparse network, the data transmissions often utilize longer hops resulting in some nodes consuming significantly more energy. The *Min-Max* scheme effectively identifies such nodes and positions the relay node closer to them, significantly reducing their energy consumption in comparison to the *Min-Total* scheme as can be seen in Figure 5.4a. On the other hand, Figure 5.4b shows that the *Min-Total* scheme significantly reduces the total energy consumed in the network.

5.7 Effects of other parameters

There are a several parameters that could possibly affect the performance of the relay positioning algorithms such as the individual node speed, mobility models, and transmission range. While it is useful to study the impact of varying these attributes, there are other important questions that need to be answered. For instance, one would like to estimate the ideal number of relay nodes that should be deployed in the network for a given number of traditional nodes. Similarly, one may also want to understand how the choice of mobility prediction model impacts the performance of the relay positioning algorithms.

In this section, we conduct experiments to investigate such issues. We simulate a network consisting of 20 traditional nodes in a 1000m×550m network area. There are four categories of multi-hop data flows in the network based on the traffic scenario presented in [35]:

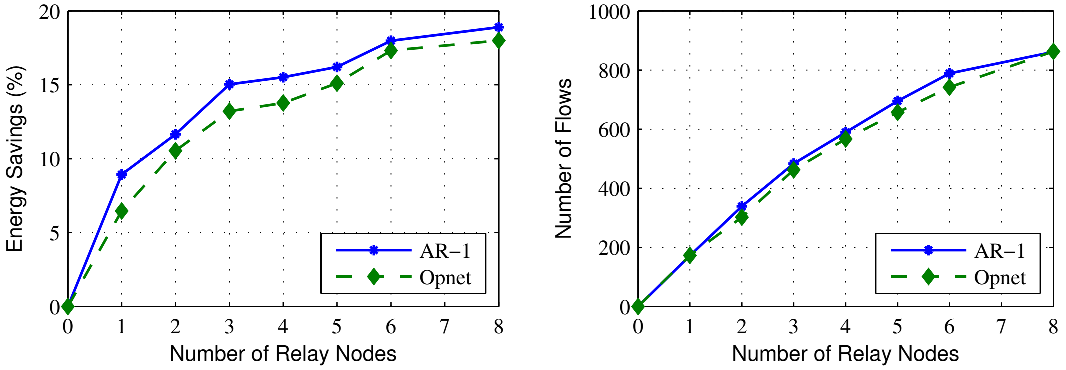
1. 4 High rate traffic with an average data rate of 120 kbps and 1 KB packet size that last throughout the simulation duration.
2. 4 High rate report traffic with an average data rate of 40 kbps and 1 KB packet size beginning at 900 seconds and continuing throughout the simulation duration.
3. 15 Bidirectional traffic between randomly selected node pairs with an average data rate of 40 kbps and 1 KB packet size. Each flow lasts for a duration of 300 seconds.
4. 15 Low rate report traffic with nodes sending 100 byte packets at an average rate of 0.8 kbps throughout the simulation duration.

For the sake of brevity, the results presented in this section are based on the *Min-Total* algorithm. The performance of the *Min-Max* algorithm also showed similar trends.

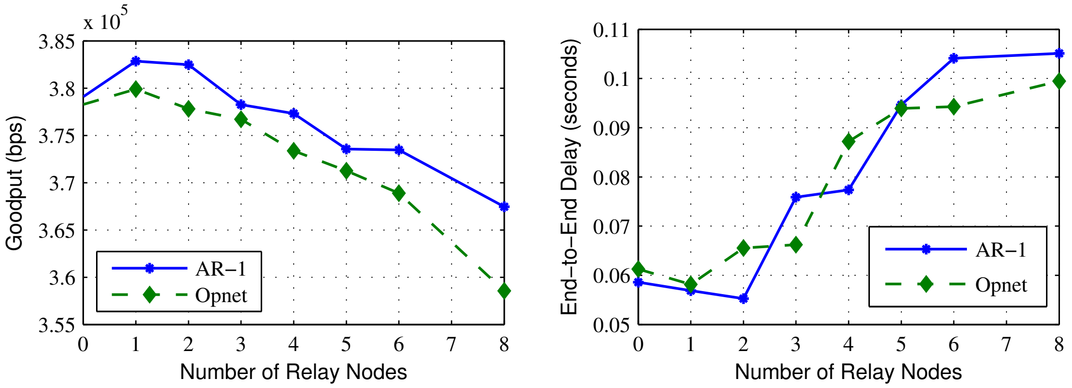
5.7.1 Results with deploying varying number of relay nodes

In the first set of experiments, we studied the effect of increasing the number of relay nodes deployed in the network. The epoch duration was fixed at 60 seconds and results presented in this section are averaged over 10 simulation runs. We measure the energy savings in the network as follows. Let E be the energy consumed at the traditional nodes for every successful data packet transmission in the absence of a relay node and E_R denote the the energy consumed when the relay nodes are deployed. The performance gain is then computed as $\frac{E-E_R}{E}$. It is intuitive that as we increase the number of relay nodes deployed in the network, the number of flows served by relay nodes also increases as observed by the results in figure 5.5b. Consequently, the transmission energy consumed by the traditional nodes also decreases resulting in increasing performance gains as shown in figure 5.5a.

However, introducing additional hops along the path of a data flow also has its tradeoffs. In [20], the authors show that in an 802.11 based ad hoc network, the end-to-end throughput available to each node is $O(1/\sqrt{n})$, where n is the total number of nodes in the network. This is due to the fact that as we increase the number of relay nodes, it not only reduces the number of transmission opportunities that the traditional nodes can use for their own transmissions, but also results in increasing channel contention every time a node attempts to transmit [18]. Consequently, the throughput in the network



(a) Performance Improvement (b) Number of flows served



(c) Throughput (d) End-to-end delay

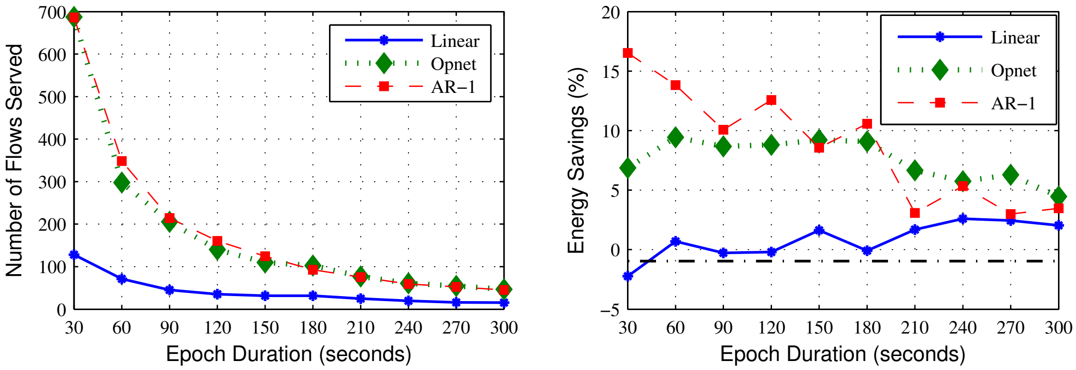
Fig. 5.5: Results with varying number of relay nodes

decreases while the end-to-end delay increases. While such a trend is clearly evident in figures 5.5c and 5.5d, we observe that the presence of small number of relay nodes actually improves the channel utilization and the transmission delays. This can be attributed to the fact that we utilize mobility prediction to define the movement of the relay nodes in such a way that it could potentially minimize the route disruptions due to the mobility of the nodes. Consequently, relaying the data packets through such additional nodes with a similar mobility pattern leads to improved performance and thus a higher throughput. Further, the AR-1 mobility prediction algorithm results in a slightly better performance when compared to the Opnet prediction scheme which can be attributed to the accuracy of the AR-1 algorithm.

5.7.2 Results with varying epoch duration

In the next set of experiments we deploy two relay nodes in the network and vary the duration of an epoch from 30 seconds up to 300 seconds. The goal of this study was to compare the performance of different mobility prediction schemes. In addition to Opnet's in built implementation of the linear prediction scheme, we also implemented a simpler linear prediction scheme that does not utilize the trajectory information of the nodes. We refer to this prediction scheme as a simple *Linear* scheme since the future position of a relay node is calculated only based on its current position without looking up the node's trajectory data.

Recall that the primary criteria for adding a flow to the service set of a relay node is that the corresponding communicating node pair should remain in the transmission range of the relay node for the entire duration of the epoch. The relay nodes employ



(a) Number of flows served (b) Performance Improvement

Fig. 5.6: Results with varying epoch duration

the mobility prediction algorithms to verify this constraint. Longer epoch duration mean that the mobility prediction schemes have to estimate the movement of the nodes over a larger interval of time. However it is quite possible that the movement pattern of the nodes could significantly change over this duration, often leading to inaccurate predictions. Consequently, as the epoch duration increases, the number of flows that would satisfy this criteria decreases. This can be verified in figure 5.6a, where we observe that, as the epoch duration increases, the number of flows served by a relay node sharply decreases.

The mobility prediction algorithm also plays a critical role in defining the trajectory of the relay node. As we increase the epoch duration, it is quite possible that the trajectory of the relay node could be incorrectly defined. The sub-optimal positioning of the relay node coupled with the decreasing service set size of the relay nodes result in decreasing performance gains as can be seen in figure 5.6b. Notice that while both AR-1 and the Opnet prediction schemes result in significant performance gains at smaller epoch durations, a simple linear prediction scheme performs poorly. This is due to the fact that predictions using a simple linear model are often error-prone leading to inaccurate positioning of the relay nodes. This result further emphasizes the fact that erroneous prediction schemes can significantly affect the performance of the proposed framework.

Chapter 6

A Mobility Prediction Based Clustering Framework

Scalability issues for routing in mobile ad hoc networks (MANETs) have been typically addressed using hybrid routing schemes operating in a hierarchical network architecture. Several clustering schemes have been proposed to dynamically identify and maintain hierarchy in MANETs [36]. To achieve significant performance gains, it is important that the underlying clustering scheme is able to identify stable clusters such that the cost associated with maintaining the clustered architecture is minimized. This chapter outlines the details of a simple framework for a mobility prediction-based clustering scheme that aims to provide temporal guarantees on link availability between nodes. The framework is then used to compare the performance of two generic mobility prediction algorithms: (1) Mobility Prediction using the Link Expiration Time [2] and (2) Mobility Prediction using Linear Autoregressive Models [27]. Simulations are performed to evaluate the temporal stability of the clusters defined in terms of the metrics - Cluster Survival Time, Cluster Residence Time and Number of Reaffiliations. In order to better understand the efficacy of mobility prediction, the results are compared with a clustering framework that is mobility sensitive, but does not utilize mobility prediction such as WCA (Weighted Clustering Algorithm [37]).

6.1 Introduction

The mobility of nodes coupled with the transient nature of wireless media often results in a highly dynamic network topology. This makes the task of routing in an ad hoc network more difficult when compared to a wired network. Routing protocols in ad hoc networks can be broadly classified into two types: reactive and proactive. However, a flat structure exclusively based on proactive or reactive routing does not perform well in large dynamic MANETs [38]. Consequently, a hierarchical architecture is essential for enhancing the routing performance in large-scale MANETs [36]. Unlike wired networks, it is essential to have a dynamic scheme to identify and maintain a hierarchy in an ad hoc network. A clustering scheme in MANET organizes the mobile nodes in the network into virtual groups known as clusters, based on certain criteria. A cluster typically consists of a cluster head and its member nodes. A clustered architecture provides an effective means for topology management, since topology changes local to a cluster need not be propagated across the whole network. Also, typically only the cluster heads are involved in route discovery which significantly reduces the control overhead associated with the routing process. There are many papers in the literature which focus on presenting an effective and efficient clustering scheme for MANETs. A survey of such clustering schemes is presented in [36].

An important argument against introducing a hierarchy in an ad hoc network is that, the overhead associated with maintaining the hierarchy may outweigh its potential benefits. For instance, the membership of a cluster can frequently change as the nodes move in and out of the range of the cluster heads. Hence, the clustering process may have to be run frequently creating additional computational overhead. Thus, it is important for a clustering scheme to identify stable clusters by minimizing the frequency of membership changes. Since it is not possible to partition the network into clusters which do not change at all, we need to design clustering schemes that exhibit temporal stability (i.e., identify clusters with a long life-time) in order to effectively apply hierarchical routing techniques. One way to achieve this is to use mobility prediction to identify clusters consisting of nodes that show some temporal similarity in their mobility patterns. Such an approach can also help in introducing a notion of quasi-stability in an otherwise unstable network topology.

6.2 Mobility prediction and clustering in MANETs

Several existing approaches utilize mobility prediction schemes to design efficient routing protocols for MANETs. In [2], William et al. compute the Link Expiration Time (LET) to predict the duration of a wireless link between two nodes in the network. Their approach assumes that the direction and speed of motion of the mobile nodes does not change during the prediction interval. This simple mechanism is then applied to enhance the reliability of existing unicast and multicast ad hoc routing protocols. In [39], an offline algorithm is proposed to predict link durations in the worst-case scenario for an urban MANET. The predicted link durations are then utilized to design a routing algorithm which finds minimum cost paths with required duration guarantees.

Dynamic clustering in ad hoc networks has also been extensively studied in the literature. Several distributed clustering algorithms for MANETs have been proposed. While some schemes try to balance the energy consumption for mobile nodes, others aim to minimize the clustering-related maintenance costs. Combined metrics based clustering schemes take a number of metrics into account for cluster configuration. The Weighted Clustering Algorithm (WCA) [37], is one such scheme, where four parameters are considered in the clusterhead election procedure which are representative of the degree, transmission power, mobility, and battery power of the mobile nodes. Such a scheme can flexibly tune the parameters to suit to different scenarios. Reference [36] presents a comprehensive survey of various MANET clustering schemes that exist in the literature.

One of the earliest approaches to utilize mobility prediction in clustering was the Distributed Dynamic Clustering Algorithm proposed by McDonald et al. in [25]. DDCA employs the (α, t)-clustering scheme, wherein generated clusters have the property that the path between any two nodes in the cluster will be available for time t seconds with a probability of at least α. Though this prediction scheme gives such a strong characterization, it is applicable only for those scenarios where the nodes follow a random walk mobility model. A (p,t,d)-clustering model is proposed in [40] which is based on mobility prediction derived from data compression techniques. The clustering is achieved by dividing the network into circular regions referred to as virtual clusters. A virtual cluster becomes an actual cluster whenever mobile nodes exist in it.

In [27], Zaidi et al. propose a two tier composite model of node mobility that captures the group behavior in a mobile ad hoc network. They use a first order autoregressive (AR-1) mobility model, originally proposed in [26] to track the mobility state evolution of an individual node. Their results indicate that with appropriate model parameters, AR-1 model is capable of representing a wide range of mobility patterns. A dynamic scheme to automatically recognize group mobility behavior in MANETs is also proposed in [27]. Though this could be considered as a clustering scheme, there is no explicit mobility prediction involved in the approach. Group mobility is identified by means of a correlation index test between the estimated mobility states of the individual nodes.

6.3 A Predictive Clustering Framework

In this section, we describe a simple mobility prediction-based clustering scheme that aims to provide temporal guarantees on the availability of links between mobile nodes. We assume that every node in the network has a unique id, which could be the node's IP address or a combination of one or more ids. Every node is also aware of its geographical location and mobility information either via GPS or using mechanisms such as [41] that use signal strength measurements.

6.3.1 Terminologies

We model a mobile ad hoc network as an undirected graph $G = (V, E)$, where V is the set of all mobile nodes, and E is the set of the undirected links between them. A link (u, v) is said to exist between nodes u and v, if and only if both are in the transmission ranges of each other. Let $\mathcal{N}_j$ denote the set of all nodes in the one hop neighborhood of node j. Cluster $\mathcal{C}_j$ is a set of nodes such that $\mathcal{C}_j = \{u | u \in \mathcal{N}_j\}$ for some $j \in V$. In addition, the members in the set $\mathcal{C}_j$ satisfy certain constraints, which will be discussed later on. Node j is called as the *seed* or the *clusterhead* of the cluster $\mathcal{C}_j$. Other nodes in the cluster are referred to as the *member nodes.* We define the residence time, τ_j^k of node k, as the amount of time k spends being a part of cluster j, before getting affiliated to another cluster. A node can get affiliated to another cluster if it moves outside the range of a clusterhead.

6.3.2 Algorithm Specification

The proposed clustering framework aims to partition the network into clusters consisting of nodes that exhibit temporal similarity in their mobility pattern. The design of this framework is motivated by the (α, t)-clustering scheme originally proposed in [25]. Specifically, in order to join a cluster $\mathcal{C}_j$, a node i must satisfy the following conditions:

1. $i \in \mathcal{N}_j$
2. $\tau_j^k \geq T_j$, where T_j is the admission criteria associated with the cluster C_j.

A clusterhead uses the mobility prediction scheme to check if a given node can satisfy the admission criteria, before admitting the node in its cluster. The algorithm is designed to run continuously and asynchronously on each active node in the network, avoiding the need for a centralized control or periodic reclustering.

Every cluster head periodically broadcasts HELLO messages to the nodes in its neighborhood. The HELLO message contains the clusterhead's admission criteria, location, and mobility profile. Upon activation, a node rapidly seeks to join a feasible cluster based on the advertisements from the neighboring clusterheads. If there are multiple feasible clusters, the node joins the cluster with maximum number of member nodes. If no clusters are detected, the node itself becomes a clusterhead and starts broadcasting periodic HELLO messages. Adjacent un-clustered nodes are prevented from each forming a new cluster by forcing nodes with higher identifiers to back off and try again as described in [25].

Cluster maintenance is performed based on a soft-state approach. Each member node maintains timers that are reset on receiving the periodic HELLO messages from their cluster heads. If a member node does not receive the HELLO message from its clusterhead within a stipulated time, the associated timer goes off to indicate one of two possibilities: (1) the member node has moved out of the clusterhead's transmission range, or (2) the clusterhead has died. In both these cases, the member node tries to find out if there are any other feasible clusters in its neighborhood that it can join. If none is available, it becomes a clusterhead on its own and starts broadcasting periodic HELLO messages.

Similar to the HELLO message, the member nodes in a cluster send periodic MEMBER_UPDATE messages to the clusterhead. Every clusterhead proactively maintains the location and mobility information of all the nodes in its cluster. If a MEMBER_UPDATE message is not received within the stipulated time, it is assumed that the node has moved out of the transmission range of the clusterhead and is no longer considered a part of the cluster.

6.4 Experimentation Results

In this section, we present the results from detailed simulation experiments carried out using the OPNET simulation software [24]. Before we discuss the results, we first describe the mobility models and the performance metrics used to evaluate the prediction schemes.

6.4.1 Mobility Models

We model the movement of nodes in the network using three mobility models: (1) Gauss-Markov, (2) Random Waypoint and (3) RPGM mobility models. Although random node mobility has been widely used, there are a number of applications of ad hoc networks in tactical communications such as emergency response teams, battlefields, etc., where nodes do not exhibit complete random motion. Therefore, in order to effectively study the performance of any clustering algorithm for an ad hoc network, we need to have mobility models that simulate realistic movement of mobile nodes. Hence, we selected the Gauss-Markov mobility model which allows us to control the randomness in the movement pattern.

We consider the random waypoint mobility model as a worst case scenario for any mobility prediction scheme. While good mobility prediction schemes should be successful in identifying explicit group mobility in the network, accurate mobility prediction in the presence of absolute random mobility is tough, if not impossible. The RPGM model introduces explicit group mobility in the network. An effective mobility prediction scheme should be able to identify the groups accurately. Therefore, in order to evaluate the strengths of the prediction schemes, we also conduct simulations consisting of groups of nodes, each moving independent of each other in an overlapping fashion. Reference [34] presents a comprehensive description of the above mentioned mobility models.

6.4.2 Performance Metrics

The primary goal of using a mobility prediction scheme is to enable the underlying clustering framework to provide temporal guarantees on the availability of routes to all the nodes within a cluster. In order to analyze the performance of the prediction schemes, we consider the following factors:

1. The clusters identified through mobility prediction should exhibit temporal stability, i.e., there should be minimal changes in the membership of a cluster over a specified duration of time.
2.]The overhead associated with cluster maintenance should be minimized.
3. The number of clusters in the network should be minimized to achieve scalability.

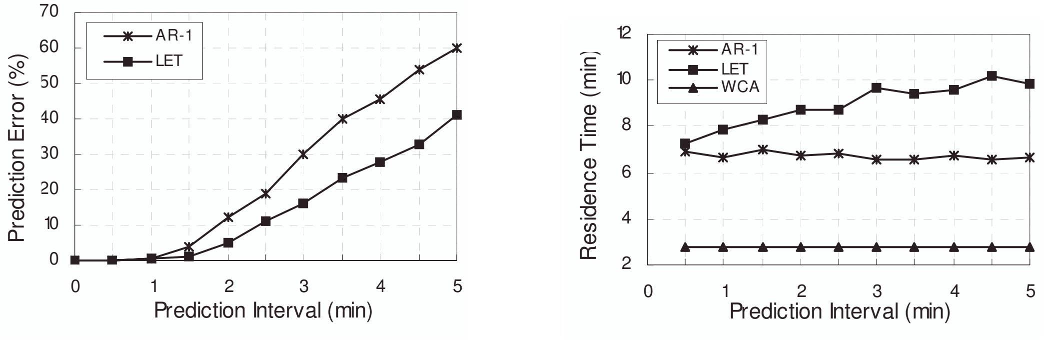
Metric for temporal stability: One way to evaluate cluster stability would be to observe the duration of time for which the membership a cluster remains unchanged. However, in the absence of explicit group mobility, it is very unlikely that nodes will remain with the same cluster for a long duration of time. Nevertheless, the stability of both inter- and intra-cluster routes critically depends on the frequency of the nodes leaving a cluster. Hence, we define the cluster survival time as the amount of time between two consecutive events of nodes leaving the cluster. We also record the cluster residence time which is the average amount of time spent by a node in a cluster. The cluster residence time is also a good measure of the stability of a cluster [25]. It is similar to cell residence time in cellular systems, which is a determinant of the distribution and rate of handoffs.

Metric for maintenance overhead: The maintenance overhead of the clustering algorithm can be evaluated using the reaffiliation count which represents the number of times mobile nodes change their cluster affiliations. A higher reaffiliation count means higher control traffic overhead since all active routes to the node need to be updated.

Metric for scalability: Finally, it is important to minimize the number of clusters in the network in order to improve scalability. Nevertheless, a clustering algorithm need not result in an minimal number of clusters, as long as the resulting clusters are relatively stable.

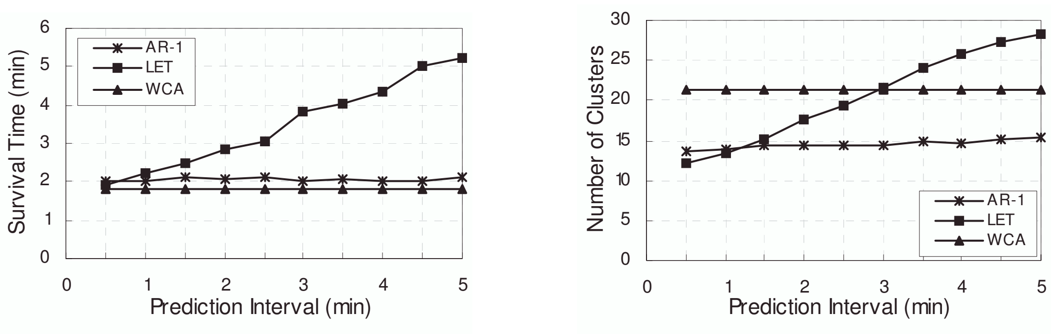
Measure for prediction accuracy: The performance of the clustering scheme is heavily dependent on the accuracy of the prediction algorithm. The decision to allow nodes to join a cluster is based on the future position of the nodes in the network, as estimated by the mobility prediction algorithm. We define the prediction error as the fraction of times a prediction turns out to be incorrect, i.e., the fraction of times a node leaves a cluster without satisfying its admission criteria.

It is important to note that the above mentioned metrics are not independent of each other. However, they all indicate the performance of the prediction scheme from different aspects. Clearly, there is a tradeoff between the size of the clusters and the stability of the clusters. A small cluster implies higher cluster survival times since membership changes will be less frequent. However, it is desirable to have clusters with multitude of nodes to localize the effects of topological changes. An optimal clustering scheme would be one which maximizes the stability of the clusters while still resulting in highly populated clusters.



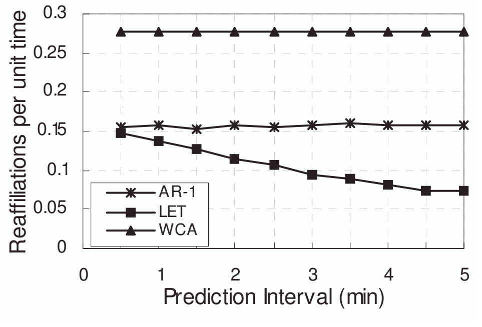
(a) Percentage Error in Prediction

(b) Average cluster residence time



(c) Average cluster survival time

(d) Average number of clusters



(e) Reaffiliations per unit time

Fig. 6.1: Effect of prediction interval under Gauss-Markov mobility model

6.4.3 Experimental Setup

We simulate an ad hoc network consisting of 70 mobile nodes on a 1000m×1000m grid. We used the OPNET simulator and each of the simulation runs were carried out for a 5 hour time period. We compared three clustering schemes, namely, LET-based predictive scheme, AR-1 model-based predictive scheme and WCA. For each of the simulation runs, the AR-1 model was initially trained on a data set consisting of 600 data points. During the course of the simulation period, the parameters of the model were updated with the observed values at intervals of 30 seconds.

6.4.4 Results with Gauss-Markov mobility model

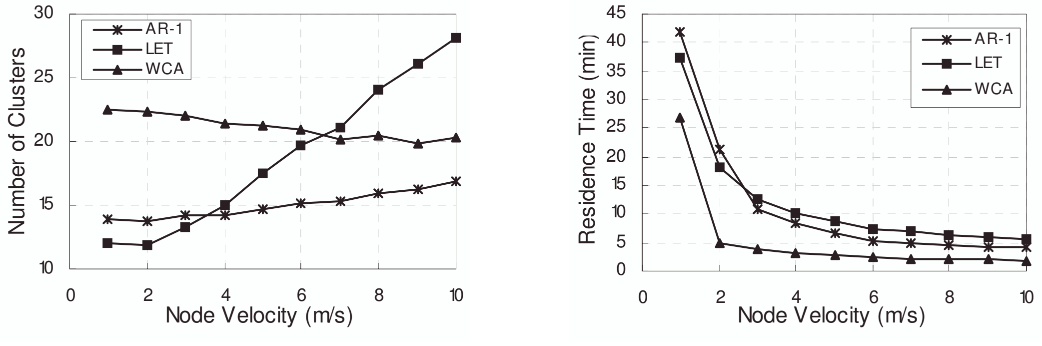
Gauss-Markov mobility model gives us the ability to control the randomness in the movement patterns of the mobile nodes through a tuning parameter α [34]. If $\alpha = 1$, the movement of the nodes is completely linear whereas a value of $\alpha = 0$, results in random node movements. For our simulations, α was set at 0.8.

6.4.4.1 Sensitivity to prediction interval

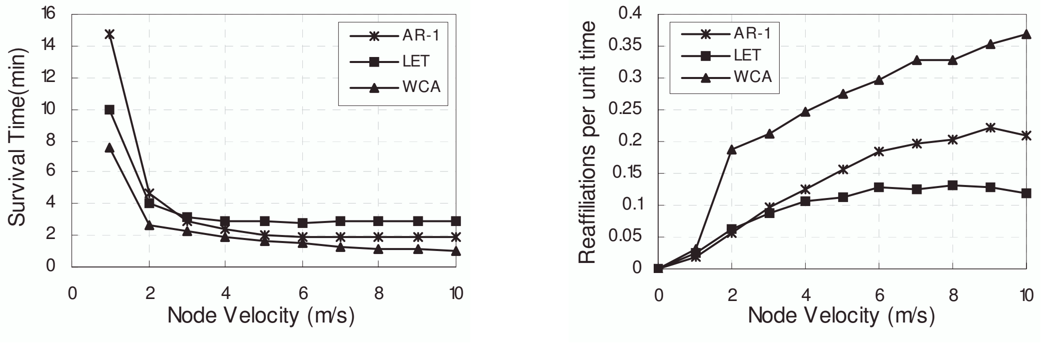
In our clustering framework, a node is allowed to join an existing cluster in the network only if it satisfies the admission criteria associated with the cluster. A clusterhead uses the mobility prediction scheme to check if a node satisfies the admission criteria. A stricter admission criterion would require the mobility prediction scheme to predict the movement of the nodes over a larger interval of time. If the predictions were to be accurate, as the admission criteria is increased, the resulting cluster will exhibit greater temporal stability with high cluster survival and residence times.

The graphs in figure 6.1a indicate that both the schemes have a high prediction error over intervals greater than 2 minutes. We also observe that while the first order linear autoregressive model accurately tracks node mobility, it has significantly higher prediction error, making it unsuitable for *multi-step predictions* (predictions over large intervals). In our simulations, we used the prediction scheme referred to as the *plug-in* predictor which is obtained by repeatedly using the fitted model with unknown future values replaced by their own forecasts. This often results in high prediction errors over large intervals especially if the model order has not been fitted well [42]. Erroneous predictions often lead to the inclusion of some nodes in a cluster that in reality, do not satisfy the admission criteria. As a result, high prediction errors in the AR-1 based scheme severely degrade its performance as indicated by cluster survival and residence times in figure 6.1b and 6.1c.

The LET-based scheme, on the other hand, does result in clusters with increasing survival times as the admission criteria of the clusters is increased. This is due to the fact that, in the absence of total random movement, a linear approximation of the movement of the nodes over a short interval of time holds good. *Nevertheless, both the mobility prediction-based schemes result in better temporal stability when compared to WCA which is insensitive to the admission criteria.* A prediction-based scheme also significantly increases the cluster residence time and hence the stability of routes in the network. Stable and long-lived clusters also result in significantly less maintenance overhead which can be verified in figure 6.1e.



(a) Average number of clusters (b) Average cluster residence time



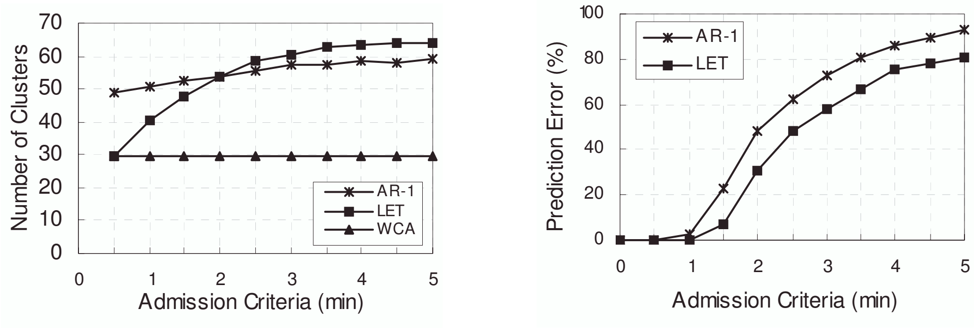
(c) Average cluster survival time (d) Average number of clusters

Fig. 6.2: Effect of node speed under Gauss-Markov Mobility Model

6.4.4.2 Sensitivity to node speed

In the next set of experiments, we vary the average speed of the mobile nodes from 1 m/s to 10 m/s. The admission criteria of the clusters is fixed at 2 minutes with a uniform transmission range of 250 meters for all nodes. As the average speed of the mobile node increases, they tend to move in and out of the clusters more frequently resulting in a highly dynamic network topology. Consequently, maintaining the temporal stability of the clusters becomes increasingly difficult. Nevertheless, a good mobility prediction scheme should be able to accurately identify nodes that meet the admission criteria of the clusters even at higher speeds. As a result, a clustering scheme that uses mobility prediction should adapt the cluster size to node mobility while maintaining the temporal stability of the clusters. Specifically, at low speeds, it results in less number of clusters with larger cluster size, while the average number of clusters in the network gradually increase in response to higher speeds.

From our results, we observe that both the prediction schemes exhibit this trend as can be seen in figure 6.2a. For typical walking speeds (less than 2 m/s), the AR-1 model-based scheme results in highly stable clusters in comparison to the LET-based scheme as seen in figures 6.2b and 6.2c. However, the LET-based scheme adapts well to increasing node speeds making it more suitable at higher speeds. While the performance degradation of the AR-1 model-based scheme could be offset partially by updating the model parameters more frequently, it will significantly increase the computational overhead. Nevertheless, we observe that both the prediction schemes do result in clusters with better temporal stability when compared to WCA which also has significantly high number of reaffiliations as shown in figure 6.2d.



(a) Average number of clusters

(b) Percentage error in prediction

Fig. 6.3: Performance under random waypoint mobility model

6.4.5 Results with Random Waypoint mobility model

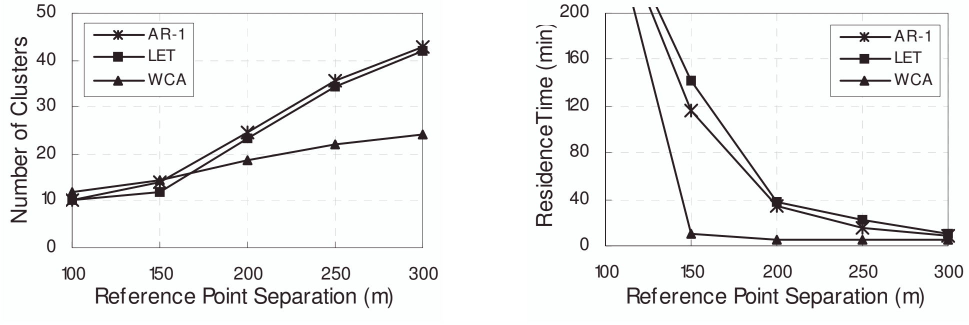
The accuracy of a mobility prediction algorithm is directly related to the movement patterns of the nodes in the network. In the presence of total random movement patterns, it is almost impossible for any prediction algorithm to perform well. We consider this case as a worst case scenario for clustering schemes that rely on mobility prediction. In figure 6.3a, we observe that the both the prediction schemes result in an almost un-clustered network for high cluster admission criterion. This is so because, every node forms its own cluster since the prediction scheme is unable to identify any feasible cluster. WCA always yields a well-clustered architecture since it does not try to meet any admission constraint. Though the AR-1 model has been shown to be successful in accurately tracking random node movement [27], multi-step prediction is worse than

LET since it tries to model the mobility of the nodes using a linear model. *Clearly, in the presence of random node mobility, it is advisable to use an algorithm that does not rely on mobility prediction.*

6.4.6 Results with RPGM Model

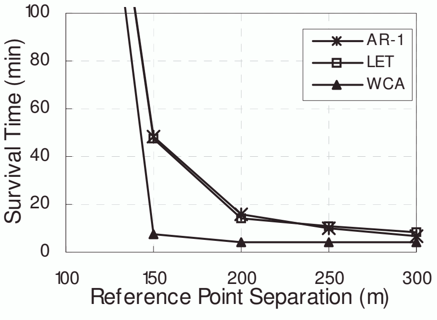
The Reference Point Group Mobility (RPGM) model represents the random motion of a group of nodes as well as the individual nodes with the group. Each group in the network is represented by its logical center. Individual mobile nodes randomly move about their own pre-defined reference points, whose movements depend on the group movement. Therefore, an accurate clustering scheme should be able to identify such explicit group mobility in the network. However, in order to evaluate the strengths of the prediction schemes, we conducted simulations consisting of 10 groups of 7 nodes each moving independent of each other in an overlapping fashion in 2000m×2000m grid. The reference point separation is increased across subsequent runs of the simulation to simulate groups which are loosely coupled. The transmission range of the nodes is fixed at 250 meters and the admission criterion for the prediction schemes is set at 3 minutes.

In figure 6.4a, we plot the number of clusters identified by the clustering schemes with respect to increasing reference point separation. We observe that when the nodes within a cluster are tightly coupled together, all the three schemes are able to accurately identify the groups in the network. But as the separation between the reference points is increased, even the nodes within a group are far apart from each other making it impossible to identify the groups using a single hop clustering scheme. As a result, the number of clusters in the network increases steadily. However, both the prediction



(a) Average number of clusters

(b) Average cluster residence time



(c) Average cluster survival time

Fig. 6.4: Performance under RPGM model

schemes have similar performance in terms of all the performance metrics. For node separations less than 150 meters, there is no change in the clustered topology once the actual groups are identified. Thus, the cluster residence times and the cluster survival times equal the duration of the simulation. The performance of WCA rapidly degrades in comparison to the predictive schemes as illustrated in figures 6.4b and 6.4c. However, as the reference point separation is increased, individual node mobility (which is similar to random waypoint mobility model) starts to significantly influence the results.

6.5 Summary

In this chapter, we studied the effect of mobility prediction on the temporal stability of clusters in MANETs. We proposed a simple mobility-aware clustering framework to compare the performance of two generic mobility prediction algorithms: (1) Mobility Prediction using the Link Expiration Time and (2) Mobility Prediction using Linear Autoregressive Models. Based on our simulation results, we make the following conclusions:

1. When the nodes do not exhibit total random motion, a predictive clustering scheme significantly improves the temporal stability of the clusters when compared to a mobility-aware non-predictive scheme. However, there is a tradeoff between the stability and the size of the clusters.

2. In the presence of total random node mobility, it is advisable to use an algorithm that does not rely on mobility prediction.

3. In the presence of explicit group mobility, both predictive and non-predictive clustering schemes are successful in accurately identifying the groups. However, when

the separation with the nodes in a group increases, mobility prediction helps in improving the temporal stability of the identified clusters. However, the performance gains are restricted due the one-hop clustering scheme used.

4. A predictive clustering scheme is able to adapt to varying network conditions by dynamically adjusting the cluster size in order to guarantee temporal stability.
5. While the AR-1 model has been shown to accurately track node mobility [26], it does not always result in accurate mobility predictions.

We make the following conclusions about the comparative study of the two prediction schemes.

1. The AR-1 model with the recursive plug-in predictor results in higher multi-step prediction errors in comparison to the LET-based scheme. However, for predictions over a small interval of time, the AR-1 based scheme results in an optimal number of clusters with comparable survival times.
2. The AR-1 model-based scheme performs well under low mobility whereas the LET-based scheme is a more suitable choice at higher speeds since the performance of the AR-1 model degrades faster than the LET-based scheme with increasing node speeds.
3. In the presence of explicit group mobility, both the prediction schemes perform equally well.

Chapter 7

Conclusion

In this dissertation, we explored the use of relay nodes with controllable mobility to enable the design of networks that are capable of optimizing their energy consumption. We studied the deployment of relay nodes in a heterogeneous network consisting of two kinds of mobile nodes - the traditional nodes with limited energy and a few mobility controllable relay nodes with relatively abundant energy resources. We defined the relay deployment problem that aims to minimize the energy consumption at the traditional nodes while compromising on the energy consumed by the relay nodes in a truly mobile network where the communicating nodes are assumed to be moving independently based on the requirements of the underlying application. We proposed a novel relay deployment framework that utilizes mobility prediction and works in tandem with the underlying MANET routing protocol to optimally define the movement of the relay nodes. We presented two instances of the relay deployment problem, together with the solutions, to achieve different goals. Instance 1, termed *Min-Total*, minimizes the total energy consumed across all the traditional nodes during data transmission, while instance 2, termed *Min-Max*, minimizes the maximum energy consumed by a traditional node during data transmission. Our solutions also enable the prioritization of individual nodes in the network based on residual energy profiles and contextual significance.

In addition, we developed a mobility-prediction based distributed clustering algorithm that organizes the MANET nodes into long-lived cohesive groups by recognizing the similarity in mobility patterns. We contend that such a clustering algorithm can be utilized to define an effective initial deployment of the relay nodes in a given network.

We conducted a comprehensive simulation analysis to evaluate the performance of the proposed optimization schemes. Results indicate that when the mobility prediction algorithms are fairly accurate, deploying a small fraction of such relay nodes using the proposed framework can not only result in significant energy savings but also improve the overall performance of the network. While both the schemes have their potential advantages, we observed that the differences between the two optimization schemes is clearly highlighted in a sparse network.

Bibliography

[1] W.-S. Soh and H. Kim, "Qos provisioning in cellular networks based on mobility prediction techniques," *IEEE Communications Magazine*, vol. 41, pp. 86–92, Jan. 2003.

[2] William Su, Sung-Ju Lee and Mario Gerla, "Mobility prediction and routing in ad hoc wireless networks," *Intl. Journal on Network Management*, vol. 11, no. 1, pp. 3–30, 2001.

[3] C. Tang and P. K. McKinley, "Energy optimization under informed mobility," *IEEE Trans. Parallel Distrib. Syst.*, vol. 17, no. 9, pp. 947–962, 2006.

[4] H. Hellbr and S. Fischer, "Mine and mile: improving connectivity in mobile ad-hoc networks," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 8, no. 4, pp. 19–36, 2004.

[5] E. R. Chittimalla, A. Venkateswaran, V. Sarangan, and R. Acharya, "On the use of nodes with controllable mobility for conserving power in manets," in *ICDCSW '06: Proceedings of the 26th IEEE International Conference Workshops on Distributed Computing Systems.* Washington, DC, USA: IEEE Computer Society, 2006, p. 88.

[6] W. Zhao, M. Ammar and E. Zegura, "A message ferrying approach for data delivery in sparse mobile ad hoc networks," in *Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing (Mobihoc '04).* New York, NY, USA: ACM Press, 2004, pp. 187–198.

[7] A. Chakrabarti, A. Sabharwal, and B. Aazhang, "Using predictable observer mobility for power efficient design of sensor networks," 2003.

[8] A. A. Somasundara, A. Kansal, D. D. Jea, D. Estrin, and M. B. Srivastava, "Controllably mobile infrastructure for low energy embedded networks," *Mobile Computing, IEEE Transactions on*, vol. 5, no. 8, pp. 958–973, 2006.

[9] W. Wang, V. Srinivasan, and K.-C. Chua, "Using mobile relays to prolong the lifetime of wireless sensor networks," in *Proceedings of the 11th annual international conference on Mobile computing and networking (MobiCom '05).* New York, NY, USA: ACM Press, 2005, pp. 270–283.

[10] Rajesh Rao and George Kesidis, "Purposeful Mobility for Relaying and Surveillance in Mobile Ad Hoc Sensor Networks," *IEEE Transactions on Mobile Computing*, vol. 3, no. 3, pp. 225–232, 2004.

[11] G. Wang, G. Cao, T. F. L. Porta, and W. Zhang, "Sensor relocation in mobile sensor networks," in *IEEE INFOCOM*, vol. 4, March 2005.

[12] G. Wang, G. Cao, and T. F. L. Porta, "Movement-assisted sensor deployment," *IEEE Transactions on Mobile Computing*, vol. 5, no. 6, pp. 640–652, 2006.

[13] D. K. Goldenberg, J. Lin, A. S. Morse, B. E. Rosen and Y. Richard Yang, "Towards mobility as a network control primitive," in *Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing (Mobihoc '04).* New York, NY, USA: ACM Press, 2004, pp. 163–174.

[14] S. Jain, R. C. Shah, W. Brunette, G. Borriello, and S. Roy, "Exploiting mobility for energy efficient data collection in wireless sensor networks," *Mobile Networks & Applications*, vol. 11, no. 3, pp. 327–339, 2006.

[15] S. Bandyopadhyay and E. Coyle, "An energy efficient hierarchical clustering algorithm for wireless sensor networks," *IEEE INFOCOM*, vol. 3, pp. 1713–1723, April 2003.

[16] Y. Tseng, Y. Chang, and B. Tzeng, "Energy-efficient topology control for wireless ad hoc sensor networks," in *International Conference on Parallel and Distributed Systems (ICPADS)*, 2002.

[17] J. Pan, Y. T. Hou, L. Cai, Y. Shi, and S. X. Shen, "Topology control for wireless sensor networks," in *9th annual international conference on Mobile computing and networking (MobiCom).* New York, NY, USA: ACM Press, 2003, pp. 286–299.

[18] J. Gomez, A. T. Campbell, M. Naghshineh and C. Bisdikian, "PARO: supporting dynamic power controlled routing in wireless ad hoc networks," *Wireless Networks*, vol. 9, no. 5, pp. 443–460, 2003.

[19] L. M. Feeney and M. Nilsson, "Investigating the energy consumption of a wireless network interface in an ad hoc networking environment," in *IEEE INFOCOM*, 2001, pp. 1548–1557.

[20] J. Li, C. Blake, D. S. D. Couto, H. I. Lee, and R. Morris, "Capacity of ad hoc wireless networks," in *MobiCom '01: Proceedings of the 7th annual international conference on Mobile computing and networking.* New York, NY, USA: ACM Press, 2001, pp. 61–69.

[21] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," in *MobiCom '00: Proceedings of the 6th annual international conference on Mobile computing and networking.* New York, NY, USA: ACM, 2000, pp. 255–265.

[22] P. Papadimitratos and Z. J. Haas, "Secure data transmission in mobile ad hoc networks," in *WiSe '03: Proceedings of the 2nd ACM workshop on Wireless security.* New York, NY, USA: ACM, 2003, pp. 41–50.

[23] A. Ghosh, R. Talpade, M. Elaoud, and M. Bereschinsky, "Securing ad-hoc networks using ipsec," *Military Communications Conference, 2005. MILCOM 2005. IEEE*, pp. 2948–2953 Vol. 5, 17-20 Oct. 2005.

[24] "Opnet v12.0," http://www.opnet.com.

[25] A. B. McDonald and T. F. Znati, "Design and Simulation of a Distributed Dynamic Clustering Algorithm for Multimode Routing in Wireless Ad Hoc Networks," *SIMULATION*, vol. 78, no. 7, pp. 408–422, 2002.

[26] Z. R. Zaidi and B. L. Mark, "Mobility estimation for wireless networks based on an autoregressive model," *IEEE Global Telecommunications Conference (GLOBECOM)*, vol. 6, pp. 3405–3409, Dec. 2004.

[27] Z. Zaidi, B. Mark, and R. Thomas, "A two-tier representation of node mobility in ad hoc networks," in *IEEE Comm. Soc. Conf. on Sensor and Ad Hoc Communications and Networks*, Oct 2004, pp. 153– 161.

[28] Hearn, Donald W. and Vijay, James, "Efficient algorithms for the (weighted) minimum circle problem," *Operations Research*, vol. 30, no. 4, pp. 777–795, jul 1982.

[29] Garey and Johnson, *Computers and Intractability, A Guide to the Theory of NP-Completeness.* Freeman, New York, 1979.

[30] S. Bespamyatnikh, B. Bhattacharya, D. Kirkpatrick, and M. Segal, "Mobile facility location (extended abstract)," in *DIALM '00: Proceedings of the 4th international workshop on Discrete algorithms and methods for mobile computing and communications.* New York, NY, USA: ACM, 2000, pp. 46–53.

[31] ——, "Lower and upper bounds for tracking mobile servers," in *Foundations of information technology in the era of network and mobile computing*, N. S. R. Baeza-Yates, U. Montanari, Ed. Kluwer Academic Publishers, 2002, pp. 47–58.

[32] S. Bereg, B. Bhattacharya, D. Kirkpatrick, and M. Segal, "Competitive algorithms for mobile centers," *Mobile Networks and Applications*, vol. 11, no. 2, pp. 177–186, 2006. [Online]. Available: http://dx.doi.org/10.1007/s11036-006-4470-z

[33] D. J. et al., "The dynamic source routing protocol for mobile ad hoc networks (dsr)," *IETF Internet draft, draft-ietf-manet-dsr-10.txt*, July 2004.

[34] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *Wireless Communications & Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications*, vol. 2, no. 5, pp. 483–502, 2002.

[35] J. Hsu, S. Bhatia, M. Takai, R. Bagrodia, and M. J. Acriche, "Performance of mobile ad hoc networking routing protocols in realistic scenarios," in *IEEE Military Communications Conference, 2003*, vol. 2, 2003, pp. 1268–1273.

[36] J. Y. Yu and P. H. J. Chong, "A survey of clustering schemes for mobile ad hoc networks," *IEEE Communications Surveys & Tutorials*, vol. 7, no. 1, pp. 32–48, 2005.

[37] M. Chatterjee, S. Das, and D. Turgut, "Wca: A weighted clustering algorithm for mobile ad hoc networks," pp. 193–204, April 2002.

[38] P. Gupta and P. Kumar, "The Capacity of Wireless Networks," *IEEE Transactions on Information Theory*, vol. 46, no. 2, pp. 388–404, March 2000.

[39] J. Tang, G. Xue, and W. Zhang, "Reliable routing in mobile ad hoc networks based on mobility prediction," *IEEE International Conference on Mobile Ad-hoc and Sensor Systems*, pp. 466–474, Oct. 2004.

[40] S. Sivavakeesar, G. Pavlou, and A. Liotta, "Stable clustering through mobility prediction for large-scale multihop intelligent ad hoc networks," *IEEE Wireless Communications and Networking Conference (WCNC)*, vol. 3, pp. 1488–1493, Mar. 2004.

[41] Srdjan Capkun, Maher Hamdi and Jean-Pierre Hubaux, "GPS-free Positioning in Mobile Ad Hoc Networks," *Cluster Computing*, vol. 5, no. 2, pp. 157–167, 2002.

[42] C.-K. Ing, "Selecting optimal multistep predictors for autoregressive processes of unknown order," *ArXiv Mathematics e-prints*, Jun. 2004.

Vita

Aravindhan Venkateswaran

Education

The Pennsylvania State University, State College, Pennsylvania, 2008
Ph.D. in Computer Science & Engineering, expected in August, 2008
Area of Specialization: Mobile Ad Hoc Networks

Birla Institute of Technology & Science, Pilani, Rajasthan, India, 2002
M.Sc. (Tech.) in Information Systems
M.Sc. (Hons.) in Mathematics

Research Experience

Doctoral Research The Pennsylvania State University, 2002–2008
This research involved the design of a unified framework for deploying mobility controllable relay nodes to optimize the energy and connectivity requirements in MANETs. In addition, a mobility prediction based clustering framework was designed that dynamically partitions the network into clusters consisting of nodes that exhibit temporal stability in their mobility pattern

Undergraduate Research Birla Institute of Technology & Science, Pilani, 2001–2002
This research work invoved the analysis of existing soft-state multicast algorithms. As part of the work, the announce-listen paradigm and its scaling properties in particular, were studied as it applies to a distributed directory service.

Teaching Experience

Teaching Assistant Operating Systems Design, Penn State University, Fall 2005

Teaching Assistant Discrete Mathematics for Computer Science, Penn State University, Spring 2006

Selected Publications

– A. Venkateswaran, V. Sarangan, N. Gautam and R. Acharya, "Impact of Mobility Prediction on the Temporal Stability of MANET Clustering Algorithms," in Proc. 2nd ACM Workshop on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks (PE-WASUN 2005), held in conjunction with MSWiM 2005, Montreal, October 2005.
– A. Venkateswaran, V. Sarangan, T. F. La Porta and R. Acharya, "A Mobility Prediction Based Relay Deployment Framework for Conserving Power in MANETs," to appear in Proc. of ACM/IEEE International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM) 2007.
– A. Venkateswaran, V. Sarangan, S. Radhakrishnan, and R. Acharya, "MANET: Network Mobility as a Control Primitive," Encyclopedia of Wireless and Mobile Communications, B. Furht (ed.), CRC Press, Taylor & Francis Group, Forthcoming 2007.