Decentralized Routing for Smartphone Networks – A Senior Thesis

A Thesis
submitted to the Faculty of the
Computer Science Department
of Georgetown University
in partial fulfillment of the requirements for the
degree of
Bachelor of Science
in Computer Science

By

Matthew Davis

Washington, DC
April 28, 2011

Decentralized Routing for Smartphone Networks – A Senior Thesis

Matthew Davis

Thesis Advisors: Micah Sherr and Lisa Singh

Abstract

This thesis proposes techniques for sending private and covert messages via smartphones. The ability to transfer data in this fashion has important applications. In places under rule of an oppressive dictatorship, often communication companies are run or controled by the government. This allows some observer working at that company to view all messages that are centrally routed, such as through cell towers. In order to maintain covertness, a peer-to-peer routing protocol will be used. Our routing techniques could be used to convey information to members of an opposition without fear or retribution. Our approach takes advantage of the unique properties and functionality of smartphones to route the data unknown to any agents of the state.

The major challenge in providing a solution is finding a decentralized method of routing that is able to quickly and efficently transport the message from sender to receiver. Addionally the routing must be performed with little overhead so that it is able to scale in environments where a phone may come into contact with many others at once. Using little overhead is also important, as one of the most pressing contraints when working with smartphones is battery life, and a complex protocol would drain this quickly.

We introduce two protocols for routing in networks of mobile nodes:he Centroid Routing Protocol and Grid Routing Protocol. These routing methods utilize location-based heuristics to identify places that are frequented by the phone's user, calculating them dynamically based on periodic updates from the phone. Our protocols use this

information to attempt to predict the future movements of nodes and route the message accordingly. This ensures that as the message has a strong chance to reach its destination as it is routed through the network.

Our results show that the Grid Routing Protocol performs better than both the Centroid Routing Protocol and other naive flooding and random walk protocols. We found that Our Grid Routing Protocol has a higher delivery rate and lower Median Time to delivery than the random walk protocol.

Index words:

Chapter 1

Introduction

In certain situations, secrecy is as essential a component to security as the best cryptographic algorithms. Consider the cases of citizens under a totalitarian regime who need to bypass censors and transfer covert information without fear of retribution, of intelligence agents deployed behind enemy lines who need safe methods of communicating back to their home country, and of Egypt in January 2011, where all communication infrastructure was disabled [7]. In these places, Internet and cellular data can be easily monitored by agents of the state. It must be assumed that this renders all centralized modes of communication compromised and thus useless, and as a result alternate methods must be available. While protecting the contents of a message is important in these cases, the ideal message passing protocol would transfer the data in a fashion that is undetectable to this central observer as to prevent any suspicion and ensure the safety of the message passers.

The increased use of smartphones and their additional hardware functionality over older model cellular tellphones, such as GPS receivers and 802.11 interfaces, allow for alternative message passing capabilities using non-traditional communication methods.

This thesis addresses the question of how to use the expanded capabilities of a smartphone to pass private messages independent of the centralized cellular network. Specifically, we will utilize data gathered by the GPS receiver and the phone's ability to create ad-hoc wireless networks to transfer messages to other phones. Currently,

when these phones send data, they use a form of centralized routing, making it possible for an observer at the cellular service provicer to monitor all messages. We conceive a message passing protocol that routes packets via an ad-hoc network using geographic information gathered from the GPS receivers and which takes advantage of the inherent portable nature of smartphones to route packets quickly and with a high rate of delivery. As users carry their phones with them, they enounter other potential carriers for the message, espcially when moving through heavily trafficked areas. **This thesis will propose two protocols for decentralized routing that can be used by smartphones to secretly pass messages based on simple location heuristics.**

Certain problems need to be considered when designing such a protocol. Many standard routing protocols, such as that used by the Internet, decide where to transfer messages based on a virtual address that has little or no correlation with physical location. Routers in the Internet tend to be stationary and fairly stable, and paths through the network are well-defined. Cell towers work in a similar fashion. These networks require extensive infrastructure to gain a great increase in speed and delivery rate. However, it also allows the channels to be easily monitored. Our protocols must be able to deliver a message using no hardware beyond the phones themselves.

The foremost challenge of our protocol is deciding how to handle routing in a network of highly mobile nodes. Such a network differs from one with non-moving nodes because standard forms of addressing would not be useful. It is possible that a node spends long amounts of time traveling before visiting the same place twice, and there is no easy way to determine which nodes will come into contact. Our protocol must have a quick way to calculate if any node seen will get the message closer to its destination. In essence, we would like to predict future movement.

To tackle this problem we first make the assumption that as humans go about their daily routines, they visit certain locations more often than others. In most cases these are homes and places of employment. We call these areas *home locations* and use two different location heuristics to determine where they are. Our protocol uses these home locations to predict where nodes are likely to travel in the future. A message is addressed with a physical location where the reciever can be found and the message is transfered based on which nodes are predicted to move close to this location.

Another significant challenge of mobility-based decentralized routing in smartphones is using as little computational power and storage space as possible. This is important because this protocol could possibly be used in busy cities, where a message carrier's range may contain hundreds of other smartphones at any given moment, and lengthy computations or storage access could cause a noticible drain of a battery which also must power the GPS receiver and wireless radio to function.

Keeping these points in mind, the goals of our protocol are:

- Secrecy - The protocol should be able to deliver the message anonymously and in a way that is unknown to the message carriers. We help achieve this by excluding any type of centralized routing in favor of a peer-to-peer network.

- Performance - The protocol should have a high delivery rate and low average time to delivery. We test this by comparing our results to those of two weaker routing methods.

- Accessability - The protocol should have to ability to be used on a wide variety of smartphone platforms. Therefore, our routing methods require only the phone's wireless adapter to transmit a signal and a GPS receiver for location information, both of which are standard on all models of smartphone.

This thesis proposes using a greedy algorithm that gets the message as close as possible to the receiver in as few number of transfers as possible. In order to best decide when to make a transfer, we introduce two protocols that use different location heuristics to determine home locations. Each node maintains only its own home location, and peridoically will poll the GPS receiver for its current position to update.

The first, the *Centroid Routing Protocol*, requires each node to maintain only a simple average of the coordinates of each location visited (ie, latitude and longitude), or its centroid. With each contact, the message is given to the phone with the centroid closest to the message's intended destination.

The *Grid Routing Protocol*, the alternative, divides the area on which the phone travels into a variable size grid and passes the message based on proximity to the intersection points of this grid. When gathering data, instead of keeping a record of the exact location, the phone keeps track of which point on the grid it is closest. The message is routed to the phone that is close to its destination point most often.

To gauge the performance advantages of our approach, we constructed a discrete event simulator that tested our methods against other, naive protocols such as flooding and random walks. The simulation utilizes human movement traces as input and outputs the results as the message moves through the network.

We found that one of the two routing methods which we introduce, the Grid Routing Protocol, performs better than both a random message passing protocol and another method which we introduce, the Centroid Routing Protocol. We found that grids with less space in between intersection points route a message takes longer but requires less intermediary steps, while a grid with more space in between intersection points sacrifices some accuracy for speed.

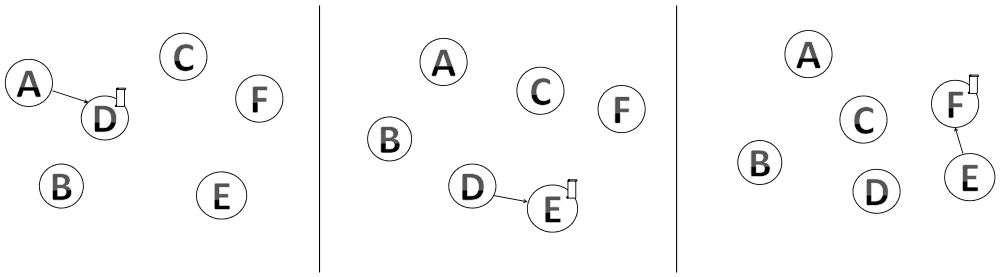

Figure 1.1: An example of mobile nodes routing a message from Node A to Node F. The nodes move as time progresses.

1.1 Terminology

Figure 1.1 shows an example of our network. A *network*, for our purposes, is a group of smartphones. *Nodes* are the members of this group. Figure 1.1 shows a sample network containing 7 nodes labeled A through F. The pictures represent the locations of these mobile nodes over time. In a simulation the *sender* is the node which generates the message and makes the initial transfer. In Figure 1.1, the sender is node A. The goal of the simulations is to deliver the message to the *receiver*. In Figure 1.1, the receiver is node F. Intermediate nodes in the network who help transport the message are *carriers*. In Figure 1.1, nodes A and D are carriers.

In our location-based routing protocols, each node needs to maintain a *home location* or *home area*, which serves as an approximation of the place or places where it spends most of its time. The specific method of calculating a home area depends upon the routing protocol being used. These will be explained in more detail in in Chapter 3

1.2 Outline

The remainder of the thesis is organized as follows. Chapter 2 discusses the related literature. Chapter 3 explains our methods of creating home locations and the types of routing we compare. Chapter 4 contains details of our simulation our results. Chapter 5 contains discussion about the findings. Conclusions and Future Work are presented in Chapter 6.

Chapter 2

Related Literature

Routing is a common computer science problem and significant research has been performed in that area. This section highlights work specifically pertaining to subjects that we considered while designing our protocols.

2.1 Greedy Routing Algorithms

Lee et al. use physical location to identify the best path through a network using Least Action Trip Planning (LATP) [6]. In their work, nodes need to visit multiple locations and use a greedy algorithm in order to minimize travel by visiting the closest locations first. Unlike LATP, our nodes only have one destination; however we similarly use a greedy algorithm to cover the most distance in the shortest amount of time. Glance et al. devise a system where messages are transfered between PDAs when they are docked to different locations [4]. These docks act as routers in the network, taking all of a node's messages and giving them to others who will bring them closer to the destination. Our method is a more advanced version of this as it considers a different mobile device architecture. The intermediate docking step is replaced by radio and GPS hardware built into and maintained by the smartphones themselves.

2.2 Human Networks

An important component to testing the proposed decentralized routing protocols is being able to simulate its results in a realistic setting. Lee et al. discuss the challenge of simulating human movement and some techniques for creating simulations [6].

Along with mobility, an important consideration with human networks is the possibilty of a node leaving the network for an extended period of time. Spyropoulos et al. identify the inherent problems of routing in an intermittently connected network, such as our network of smartphones where some users may leave their homes for vacation, allow their phone to power off, and battery or safety concerns require the disabling of their GPS. They propose two routing types that take these challenges into consideration. The first, Spray and Wait, initially floods the network with a set number of message copies and then waits for delivery, with the added use of network resourses ensuring a higher rate of success. Spray and Focus has each node maintaining its last contact with all other nodes which it uses to direct its routing [10]. Our system considers this, but achieves better scalability. Our centroid routing requires only two numbers be maintained to govern transfering, while our Grid Routing Algorithm maintains a fixed number of points on a grid. Kemp et al. also discuss *influence maximization* as a way of identifing the nodes which are likely to encounter each other in a network [5]. Though their approach acheives high accuracy, our emphasis on speed and secrecy discourages us from using such a complex system.

2.3 Anonymous Routing

Anonymous message passing has been explored in detail in the past. Anonymous network systems such as Tor [3], Crowds [8], and Hordes [9] are available for use on personal computers when browsing. These networks share a common trait that

messages sent through the network maintain only information about the receiver [3] [8] [9]. Every message carries this same format, so messages sent from the sender are indistinguishable from messages sent between carriers. Messages sent with our routing protocols share this anonymity feature. Each is addressed to aid the protocol's location heuristic in routing and maintain no information about previous nodes.

Another common trait of these anonimity networks is that they operate by layering messages with cryptography to prevent eavesdropping [3] [8] [9]. This is a strong defense against an attack with only a partial view of the network, but a central observer with knowledge of all of the network messages would negate all anyonoimity effects because they see the message before and after each transfer. Thus, using such methods with our smartphone network is infeasable because we wish to prevent an attacker who controls the entire network from seeing the message. However, certain aspects of these protocols can be useful for this thesis. Tor and Crowds networks maintain anyonimity by passing messages along paths of nodes within their network before reaching their destination [3] [8]. Our protocols would acheive this same result while routing a message closer to its destination through peer-to-peer networks, which will not only protecting the identity of the sender while also adding a layer of covertness.

Chapter 3

Decentralized Routing

In this chapter, we describe our routing protocols. We designed four methods: two strawman methods for comparison and two based on location heuristics. Certain assumptions can be made about our network:

1. Smartphones are able to create ad-hoc wireless networks for peer-to-peer data transfer. This is how our protocol would transfer the messages. We assume tha the adversary cannot monitor these p2p communications.

2. The phones update their location at regular intervals. Each phone maintains a record of its home location that is modified with each update.

Table 3.1 shows the steps common to all our routing protocols. In the first step, a message is inserted into the network by giving it to a random node. The message is always addressed with the name of the receiver (or a pseudonym, to help preserve anonymity), and may contain a physical address as well depending on the routing algorithm chosen. In the second step, the nodes become mobile and begin to encounter other nodes. The third step utilizes the radio on the smartphone to establish a connection between two nodes that come within range of each other. In the fourth step, the carrier may or may not transfer the message. The way each protocol handles step four is the only major difference between our protocols. The final step mirors step two in that the nodes continue on their paths.

Steps	Description
1	The sender has a message addressed to the receiver
2	The nodes in the network begin to move and come into contact with one another
3	When a carrier encounters another node, a wireless connection is made between them
4	The message is either transfered to the node or kept by the carrier, depending on the routing algorithm chosen
5	The nodes continue to move and transfer until the message reaches the receiver

Table 3.1: The major steps of our routing protocols

It is important to note, however, that our protocols are meant for networks in which none of the nodes ever disappear. Actual smarthpone networks would consist of nodes that could be powered down temporarily or that travel especialy far from their home area. We made this decision to prevent the possiblity of a node obtaining the message then leaving the network with it. We leave as future research the evaluation of message duplication strategies to mitigate loss due to nodes leaving the network.

3.1 Strawman Solutions

We used two naive routing protocols as a base case when examining our results, the first uses probabilistic flooding and the second is based on a random walk algorithm. Neither use a location heuristic as the criteria for transfering a message, but rather some random probability. These are not meant to be strong or even feasable methods of routing. Rather, each serves a specific purpose in helping us measure the strength of our proposed location-based protocols. The flooding method performs exceptionally well, and similar results would indicate fast, reliable routing. Conversly, the random walk method contains directionless routing, but provides for better scalability.

Steps	Description
1	The carrier encounters a node
2	The carrier checks to see if this node is the receiver, and if so transfers a copy of the message immediately
3	If this node is not the receiver, a copy of the message is transfered with probability p and not transfered with probability $p - 1$.
4	After the interaction, the carrier or carriers continue moving and begin looking for other nodes

Table 3.2: Message tranfer using the Probabilistic Flooding Protocol

3.1.1 Probabilistic Flooding

The *Probabilistic Flooding* protocol delivers the message from the sender to the receiver by generating multiple copies of the message that propogate quickly throughout the network. The sender begins with his single copy of the message addressed with the receiver's name. Each time any node with the message comes into contact with one that does not have it, a copy of the message will be transfered with some probability. If this new node is the receiver, the copy is transfered to it right away. The Probabalistic Flood Protocol delivers the message at a high rate because as more copies of the message propagate through the network, more nodes come into contact with infected nodes attempting to spread the message. Table 3.2 lists the steps performed by the protocol to route the message.

Figure 3.1 shows a network using the Probabalistic Flooding Protocol to send a message from sender A to receiver G. The first picture depicts A entering the network and making a connection with nodes B, C, and D. The message is transfered to Nodes B and D but not C. In the second picture, nodes B and D move and transfer the message to E and F, respectively. In the final picture, receiver G receives the

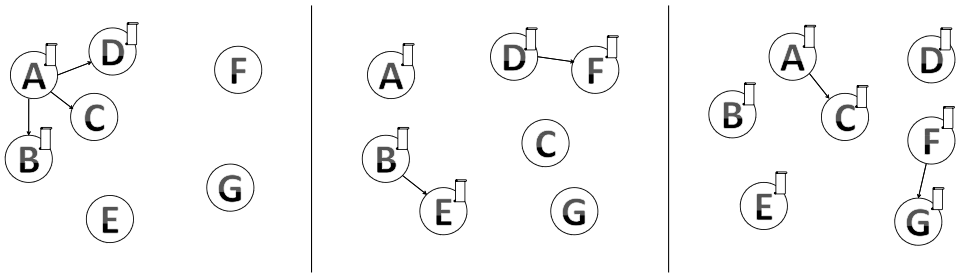

Figure 3.1: An example of a message routing using flooding. Node A is the sender and node G the receiver.

message from F. Concurrently, C receives the node from A, as all nodes continue to move through the network and propogate the message.

The Probabilistic Flooding Protocol finds the optimal path from sender to receiver. Recall from Chapter 1, that the optimal path is the fastest way to travel from one node to another. With a transfer rate of one hundred percent, the optimal path would be found every time, as every path is followed. We chose to consider this routing protocol with a one percent transfer rate. Even with a rate this low, maintaining multiple message copies make the Probablalistic Flood perform very well as far as success rate and time to delivery. However, certain problems would prevent the actual implemention of this method, for example letting the nodes know when a message copy has reached the receiver. In the network in Figure 3.1, A continues to transfer the message after the reciever has been reached, as would any other node with the message. The cost of performing so many transfers is very high. For these reasons, this protocol is meant to serve only as a guide, showing results close to what we think the optimal is for the protocols with single message copies.

Steps	Description
1	The carrier encounters a node or group of nodes
2	The carrier checks to see if any of these nodes are the receiver, and if so transfers the message immediatly
3	If none of these nodes are the receiver, the carrier transfers the message with probability $p = 1$.
4	If the carrier decides to transfer the message, it chooses a recipient at random.
5	After the interaction, the carrier continues moving and begins looking for other nodes again

Table 3.3: Message transfer using the Random Walk Protocol

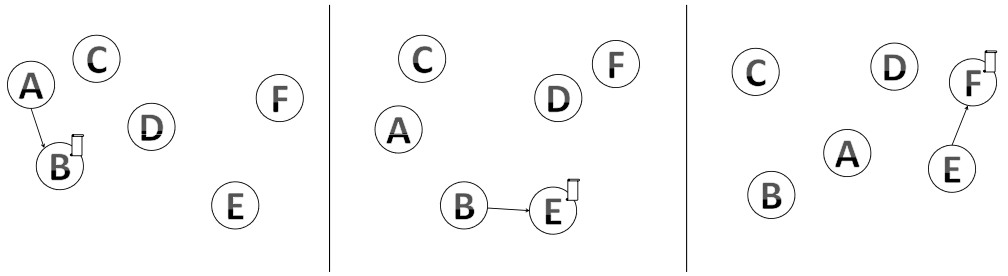

Figure 3.2: An example of a message routing using a random walk. Node A is the sender and node F the receiver.

3.1.2 Random Walk

The random walk protocol relies on pure chance to route the message from sender to receiver. The sender begins with a single copy of the message, addressed with just the name of the receiver. As a carrier moves through the network and encounters other nodes, it will transfer the message with some probability p. Only one copy of the message is maintained. Table 3.3 lists the steps performed by the protocol to route the message.

Figure 3.2 shows a message routed using the Random Walk protocol. Node A enters the network in picture one and transfers the message to Node B. Picture two shows the message being given to Node E and the final picture show it reacing the receiver F. Note that because the protocol transfers the message randomly, it will not necessary take the quickest path to the receiver, in this case, Node C.

This routing method relies on pure chance to govern how the message moves through the network. Each time the node currently holding the message comes into contact with another, it will transfer that message to it with some probability p. When it encounters multiple other nodes, it chooses one at random before making this decision. In this method, only one copy of the message is in the network at any given time and if a new carrier is selected the transfering node deletes its copy before moving on.

One potential weakness of this protocol is the possiblity that the message transfers to a node that will move it away from the receiver. In Figure 3.2, notice how B transfers the message to C in picture three despite C being farther from the receiver. For this reason, the Random Walk protocol is not meant to be a particularly strong routing protocol. Rather, the results given by this protocol act as a lower baseline with which to judge our other protocols by. It is our assumption that a good routing protocol must, at the very least, deliver a message faster and in less transfers than a message taking a Random Walk.

3.2 Location-Based Protocols

In the next two subsections, we introduce two protocols that route messages based on physical location: the *Centroid Routing Protocol* and the *Grid Routing Protocol*. To do this effectivly, the protocols utilize the smartphone's GPS receiver to build a location

Steps	Description
1	The carrier encounters a node or group of nodes
2	The carrier checks to see if any of these nodes are the receiver, and if so transfers the message immediatly
3	If none of these nodes are the receiver, all nodes present calculate their centroids
4	The carrier will transfer the message to the node with the centroid closest to the address on the message, or keep it if it's centroid is closest.
5	After the interaction, the carrier continues moving and begins looking for other nodes again

Table 3.4: Message transfer using the Centroid Routing Protocol

profile for each node. At regular intervals the nodes use their position to update their location profiles. It is important to note that the sender cannot perfectly predict the location of the intended receiver. Instead, we use past location data to attempt to predict where a node will be in the future and route the message accordingly.

3.2.1 Centroid Routing Protocol

We introduce the Centroid Routing Protocol, which bases its heuristic on an average of all places a node has been seen. It does this by calculating a centroid, or mean value of all visited locations, for each node, to use as a home location. A summation of the coordinates on each axis is maintained individually. Every time a node records its location, this new data is added to an existing tally of locations and new centroids are computed. To calculate the centroid, the coordinate totals are divided by the total number of records, resulting in the point on the map. As a result, the number of times the node has recorded data must be kept by each node as well. Table 3.4 lists the steps performed by the protocol to route the message.

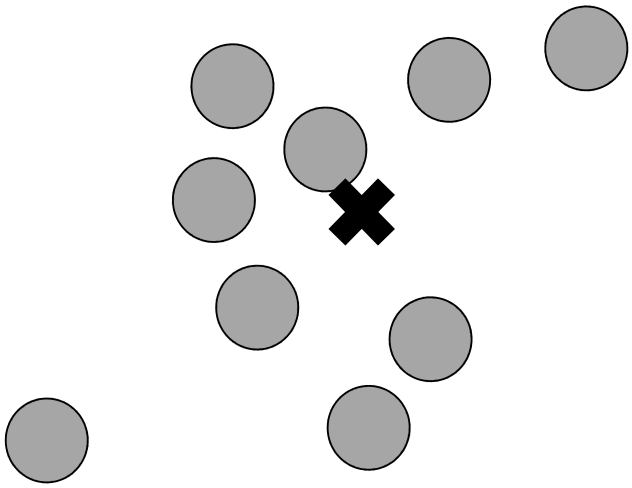

Figure 3.3: This figure shows all of the locations recorded by a single node, represented by filled in circles. The X represents the Centroid for this node given the data.

Figure 3.3 displays the locations given by a node during its updates. Though the node does travel to the edges of the plane, its records are concentrated towards the center. The X shows the location of this node's centroid.

Figure 3.4 displays the execution of the Centroid Routing Protocol over time. The first picture displays the starting network of three nodes and their centroid locations. In the second picture, sender D is introduced to the network, and its centroid is added. In the third picutre, D routes the message to B because it has a closer centroid to C than either itself or A. In the final picture, B moves closer to its centroid and is able to successfully deliver the message when encountering C.

A carrier transfers a message to a nearby node if the latter node's centroid is closer than the current to the message's intended destination. The method for determining

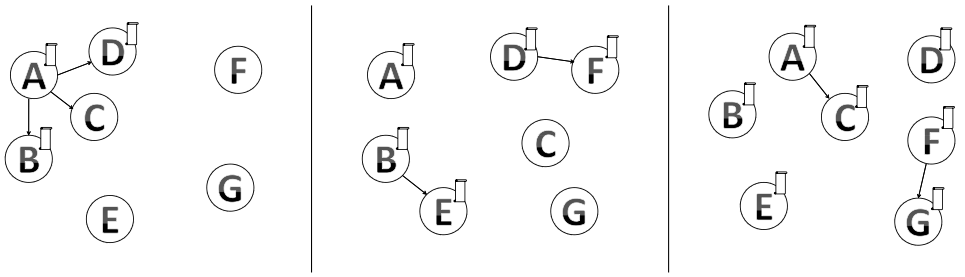

Figure 3.4: An example of a message routing using the Centroid Routing Protocol. Node A is the sender and node C the receiver. The stars indicate centroid locations for each node.

closeness can vary depending on the data. If node positions are expressed on a flat plane, Euclidian distance between centroid and address can be used to decide which node gets the message. If the records have GPS coordinates like points on Earth, the Haversine formula, a variation on Great Circle Distance, is used. This is an accurate way of determining short distances on a large sphere (i.e., the Earth).

3.3 Grid Routing Protocol

The Grid Routing Protocol was designed to compute home location with more precision than the Centroid Routing Protocol by dividing the network into a square grid. The intersection points on this grid serve as possible home locations and the protocol routes messages based on the frequency with which a node visits a point associated with the receiver.

Each time a node records its location, it calculates which grid point it is closest to. To do this, every node maintains knowledge of the size of the grid, which it uses to calculate where these intersection points are. Using this information, the frequency with which a node visits a point can be found. Frequency is defined as the ratio a node

Steps	Description
1	The carrier encounters a node or group of nodes
2	The carrier checks to see if any of these nodes are the receiver, and if so transfers the message immediatly
3	If none of these nodes are the receiver, all nodes determine the frequency with which they visit the message address
4	The carrier will transfer the message to the node that visits the address points with the highest frequency, or keep it if its frequency is highest
5	After the interaction, the carrier continues moving and begins looking for other nodes again

Table 3.5: Grid Routing Protocol

has visited a certain point to the total number of times it has visited any point. The message is addressed with an intersection point that sender believes the reciever is close to with a high frequency. Each time the carrier comes into contact with another, the message is passed to the node that is seen at this point more often. Table 3.5 lists the steps performed by the protocol to route the message.

Figure 3.5 shows ten locations where a node recorded its location. The intersection points on the grid labeled A through F are the places the node was closest to. The table lists the frequency with which the node visits each of these points. In this example, point A would be the most frequent point this node visits, and where its message would be addressed to.

The Grid Routing Method differs from the Centroid Routing Method in two major ways. First, it more precisely records where a node has been, which makes it better able to predict the future movement of nodes and more selective in which nodes carry the message. The second is that it allows a node to have multiple home areas. A node that splits its time between two locations could be a viable carrier for a message

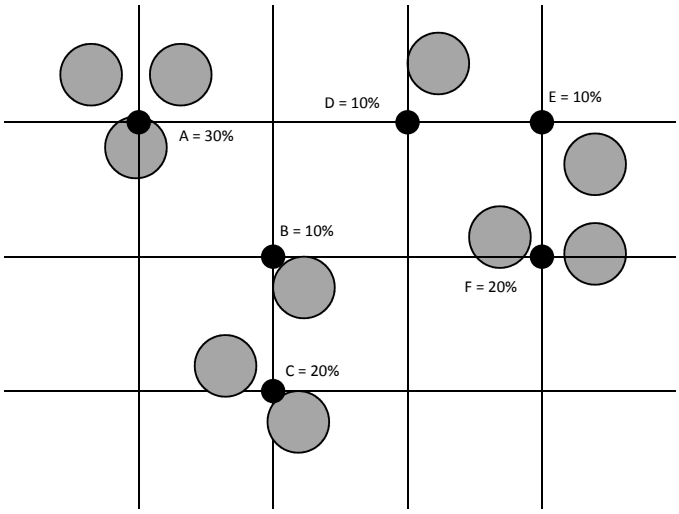

Figure 3.5: This figure shows all of the locations recorded by a single node, represented by filled in circles. Points A throuh F are the grid intersection points closest to these records. Each point is listed with the percentage of the time the node has been close.

addressed to either, because it will appear at both points with some high freqency. However, a node that primarly spends its time in a singular area will always be a more ideal carrier because we are able to better predict its future position.

Chapter 4

Simulation and Results

In this section, we discuss the specifics of our simulation and the results that it generated.

Recall from 3 our methods for decentralized routing. We designed this simulation to measure the performance algorithms. This is achieved through the use of the different routing protocols to simulate the passing of the message via trace data representing human movement. In our simulation, each time a carrier encounters another node, the chosen protocol is implemented to decide whether or not a transfer should be performed. This continues until it is delivered or the simulation runs out of trace data and exits unsuccessfully.

4.1 Datasets

In this subsection we give specific information about the datasets used for the routing simulation. Two sources were used. The first is a synthetic represenation of human motion generated though the use of the findings of the SLAW research [6]. The second is a collection of locations from over twenty years of research recorded by the Shark Bay Dolphin Project based off Monkey Mia beach in Austrailia [1].

4.1.1 SLAW Data

The SLAW Data uses various techniques to simulate human movement. It does so in two parts. First, fractal waypoints and power-law gaps are used to create areas on the

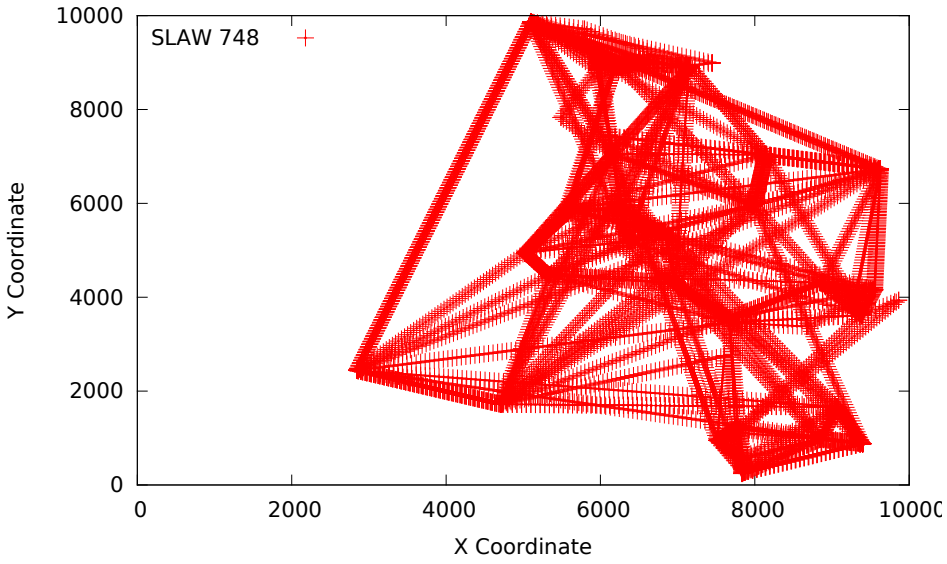

Figure 4.1: An example of the locations traveled to by a node during a simulation

map that represent areas that are commonly traveled. 200 waypoints are used with an alpha distance of 3 and a Hurst parameter, which governs how close they can be to one another, of 0.75. Second, Least Action Trip Planning and an Indiviual Walker Model are used to decide how the nodes move between the waypoints. Usually the nodes will visit the waypoints that are closest to them first before branching to those that are farther. In our dataset, waypoints within 50 meters are considered clustered and would be visited together.

The trace takes place on a square $10x10$ kilometer plane and consists of 1000 nodes. The duration of the trace is seven days. Each node records its location once every 60 seconds. The maximum amount of time a node can remain idle is 4 hours.

4.1.2 Dolphin Data

The use of dolphin data as representative of human movement can be justified due to the social groupings and habits of their species. Connor builds off previous research in the area, writing about complex social relationships between dolphins. Dolphins live in *fission-fusion* like humans, meaning the animals form small, highly mobile social groups. His research extends this notion, claiming that dolphins show evidence of not just social groupings, but alliances within and outside of these groups that impact their composition [2]. There are similar complex social structure that are seen in human societies. Because of these similarities, we utilize this data set of Shark Bay dolphins as an approximation of possible human interactions. The use of this dataset is also advantagous because of the difficulty in gathering traces of actual human movement due to privacy concerns when collecting the data. We circumvent this issue when using the Dolphin dataset.

As an example of the dophin movements, figure 4.2 displays the places visted by a dolphin over the time used for our simulation.

The dataset contains records from 1988 through 2009. Or study focuses on data from 2000-2005 since the number of sightings is larger during that time peroid. The dolphins live in an area off the western coast of Austrailia between 25.4696 and 25.89158 degrees South Latitude and 113.46713 and 113.93434 degrees East Longitude. There are 734 dolphins that exist at that time.

4.2 Simulation Details

In this section, we describe the specifics of how our simulation executes.

All of the protocols share the same basic structure. First, there is an initial period of data gathering. This is used to initialize the home area of each node. During this

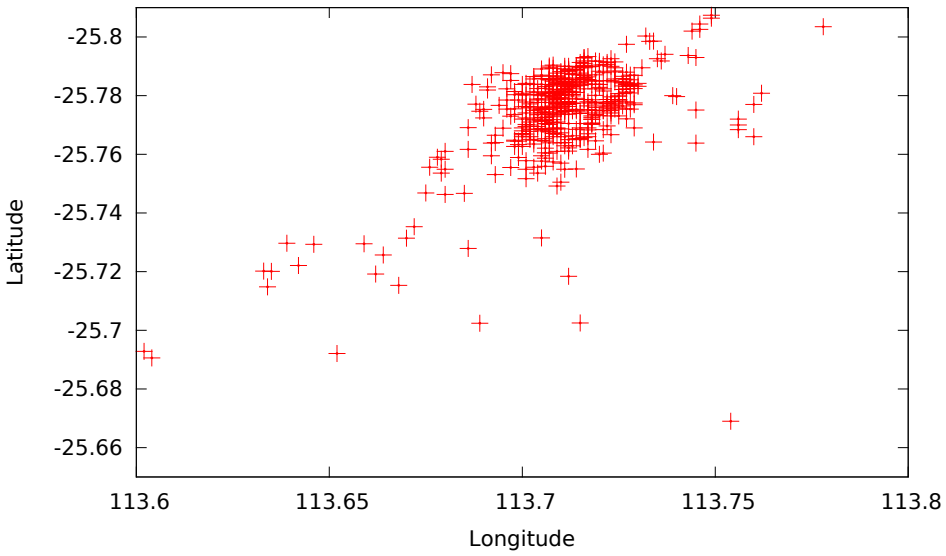

Figure 4.2: An example of the locations traveled to by a dolphin in the data

period, there are no messages in the network. Additionally, if a dataset is particularly large, certain periods of time may be filtered out and a subset used. If this is done, a list is made of all nodes that appear in the trace both before the start and after the end of this filtering. All other nodes are ignored. This ensures that a node cannot obtain the message and subsequently leave the network with it.

Once the initialization time has elapsed, a sender and receiver are randomly chosen and the message is introduced into the network. Each run of our simulation starts with a single copy of the message held by the sender. This message is addressed at this time. Although this address never changes after being initialized, the nodes continue to update their location as they are seen. In addition, each node maintains its location profile by keeping track of the last time and location that it was recorded.

Carriers can come into contact with other nodes in two ways. First, it is possible for other nodes to be present at the same location when the message is present. In this case, each of these nodes are elligible to receive the message. Second, a node may enter the area where the message is currently located, in which case it is considered eligible for transfer.

The simulation has two conditions for termination. It will exit successfully if the message was transfered to the receiver and output the time of the final transaction and the number of carries needed to deliver the message. It will exit unsuccessfully if the end of the trace is reached and the receiver still has not come into contact with the message.

The user inputs certain parameters to configure the simulation. Table 4.2 lists each parameter and its function. First, if filtering is required, a start and end time are input. Second, the user inputs the maximum distance over which the message can be transfered. For the SLAW data the unit of this input is meters and we used 30 meters as our default. For the dolphins, the distance is kilometers and we used one kilometer as our maximum. Third, the maximum time since a node was seen that a message can still be transfered. For the SLAW dataset. For the dolphins, one day was used as the amount of time a dolphin was considered in range to receive a message after visiting a location. Finally, for each run, the user inputs a seed for the random number generator and routing protocol to use, along with any relevant information for the protocol such as size of the grid or random probability.

4.3 Results

We ran our simulations one hundred times for each routing protocol and show the combined results. For the Probabilistic Flooding and Random Walk, a transfer prob-

Parameter	**Use**
Start Time	Used in conjunction with End Time if a subset of the trace data is to be used. This is the time of the start of the subset.
End Time	Used in conjunction with Start Time if a subset of the trace data is to be used. This is the time of the end of the subset.
Maximum Transfer Distance	The maximum distance over which a message can be transfered (i.e., the strength of the wifi signal).
Maximum Transfer Time	The maximum time before which a node can receive the message after being seen at a certain location.
Random Number Seed	A seed for the random number generator. This is used by all protocols to decide on the sender and receiver before the simulation begins, and by the flooding and random walk protocols to decide if a message is transfered.
Transfer Probability	For the Probablistic Flooding and Random Walk protocols. This is the probability that the message will be transfered after contact.
Grid Size	For the Grid Routing Protocol. This is the number of intersection points along each horizontal and vertical axis.

ability of one percent was used. For our Grid Routing Protocol, a grid of one hundred by one hundred intersections was used.

We decided to use one percent as the parameter for our flooding. This ensured the node had time with which to come into contact with the receiver itself before finding someone else to serve as a carirer. Unsurprisingly, the Probabilistic Flooding protocol delivered the message one hundred percent of the time. Maintaining multiple copies of the message ensure this while making the protocol infeasable in practice since the network would quickly become overwhelmed with message copies. However, we expect a strong routing protocol to be able to deliver the message in many less transfers and as close to as quickly as the flood.

We decided to use one percent as the parameter for our Random Walk, as with our Probabalistic Flooding, so that each node would come into contact with a high number of nodes before passing the message on. This is even more important with Random Walk than with Flooding because only one copy of the message is maintained and we would like to ensure a node contacts as many other nodes as possible before transfering the message.

We used grids with measurements of between twenty five by twenty five and five hundred by five hundred depending on the dataset. We found that more precise grids tended to increase the delivery rate, while not impacting time to delivery.

4.3.1 SLAW Data Results

Table 4.1 displays the results for each protocol on the SLAW data. As expected, the Probabilistic Flooding Protocol performed best in deivery rate and time to delivery. We simulated the Grid Routing Protocol with grid sizes of 25 by 25, 50 by 50, 75 by 75, 100 by 100, and 200 by 200. We found that the 200 by 200 grid had the best delivery rate with this dataset. The Centroid Routing Protocol consistently performed the

	01% Probabilistic Flooding	**01% Random Walk**	**Centroid Routing**	**200x200 Grid Routing**
Delivery Rate	100	86	71	89
Mean Time to Delivery(TtD)	8 Hours, 26 Mins, 9 Seconds	2 Days, 4 Hours, 40 Mins, 25 Seconds	2 Days, 7 Hours, 54 Mins, 20 Seconds	2 Days, 5 Hours, 36 Mins, 29 Seconds
First Quartile TtD	4 Hours, 0 Mins	17 Hours, 15 Mins	18 Hours, 24 Mins	19 Hours, 18 Mins
Median TtD	7 Hours, 40 Mins	1 Day, 13 Hours, 13 Mins	1 Day, 23 Hours, 44 Mins	1 Day, 12 Hours, 56 Mins
Third Quartile TtD	11 Hours, 17 Mins	3 Days, 7 Hours, 9 Mins	3 Days, 14 Hours, 6 Minutes	3 Days, 15 Hours, 27 Mins
Mean Number of Transfers	43.25	18.55	20.97	2.43
First Quartile Transfers	8	7	10	1
Median Transfers	24	13	19	2
Third Quartile Transfers	59	30	31	3

Table 4.1: Statistics for the SLAW data after one hundred simulations with each type of routing

worst of all our non-flooding protocols in all metrics: delivery rate, time to delivery, and number of transfers.

Figure 4.3 displays the time to delivery for each of the routing methods displayed in the table. Flooding performs significantly better than all others, which are mostly similar.

Figure 4.4 shows the delivery rate after one hundred simulations of the Grid Routing Protocol with each different grid size. With grid sizes of 100 by 100 and

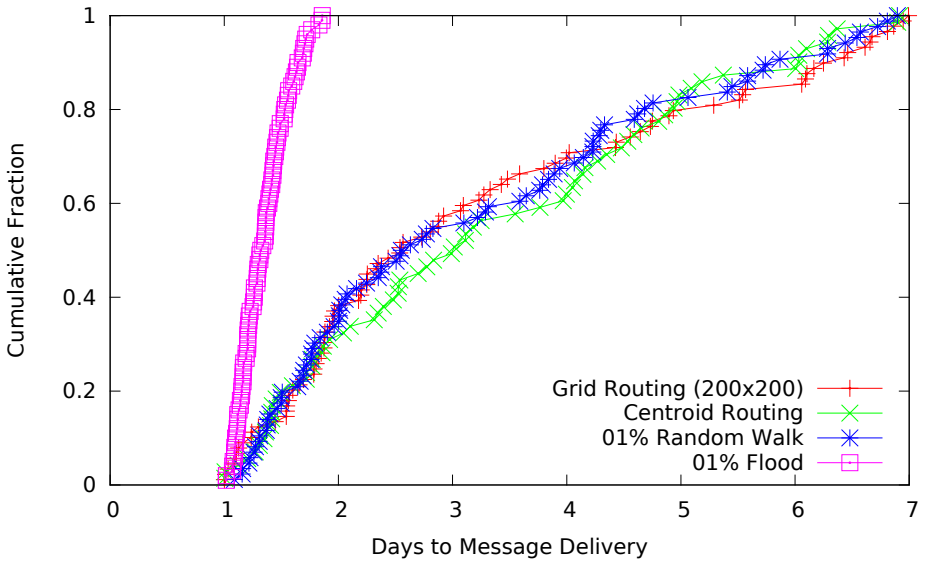

Figure 4.3: The amount of time until each successful transfer

smaller, the delivery rate is close to eighty percent. This increases to eighty nine percent when the grid size is increased to 200 by 200.

Figure 4.5 shows the time to delivery for each of the different grid sizes. The results show that size of the grid does not really impact the time to delivery.

We found that with this dataset, the Grid Routing Protocol performed best. With a grid size of $200x200$, we achieved a delivery rate of 89%, the highest for a non-flooding protocol, as well as delivery with the lowest number of transfers by far. This was achieved in a time consistent with those of the other protocols. Successful delivery in a small number of transfers is important because, since we are working with smartphones, we are concerned with the life of the device's battery. Using the

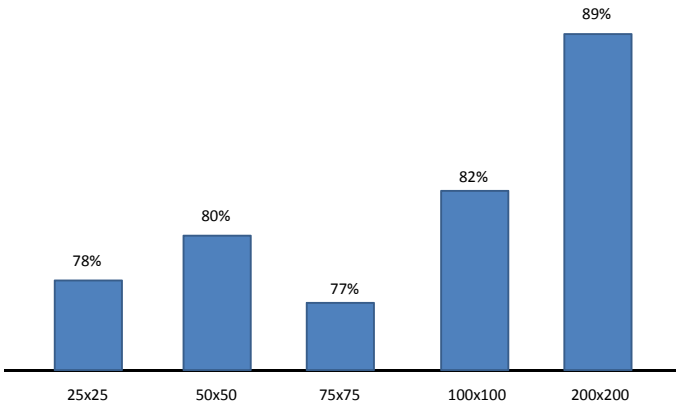

Figure 4.4: The delivery rate for the Grid Routing Protocol on the SLAW data with different grid sizes

phone's wireless capabilities to make a transfer will drain this so we wish to keep the amount of times which it is used to a minimum.

4.3.2 Dolphin Data Results

With the Dolphin data, all routing methods had very similar delivery rates and time to delivery statistics. As with the SLAW dataset, the Grid Routing Protocol did perform considerably better in terms of number of transfers necessary to reach the receiver.

We believe that this data is similar between the protocols because the areas traversed by the dolphins is well defined and they will often come into contact with

	01% Probabilistic Flooding	**01% Random Walk**	**Centroid Routing**	**100x100 Grid Routing**
Delivery Rate	99	96	98	97
First Quartile TtD	9 Months, 23 Days, 2 Hours, 4 Mins	9 Months, 23 Days, 2 Hours, 4 Mins	8 Months, 14 Days, 1 Hour, 6 Mins	9 Months, 12 Days, 10 Hours, 14 Mins
Median TtD	1 Year, 7 Months, 0 Days, 4 Hours, 18 Mins	1 Year, 7 Months, 7 Days, 9 Hours, 55 Mins	1 Year, 6 Months, 29 Days, 2 Hours, 47 Mins	1 year, 6 Months, 28 Days, 12 Hours, 20 Mins
Third Quartile TtD	1 Year, 8 Months, 23 Days, 4 Hours, 21 Mins	1 Year, 8 months, 23 Days, 2 hours, 15 Mins	1 Year, 8 Months, 22 Days, 10 Hours, 1 Min	1 Year, 9 Months, 5 Days, 10 Hours, 40 Mins
Average Number of Transfers	23.52	5.43	5.06	1.61
First Quartile Transfers	3	1	3	1
Median Transfers	13	3	4	1
Third Quartile Transfers	30	6	7	2

Table 4.2: Statistics for the Dolphin data after one hundred simulations with each type of routing

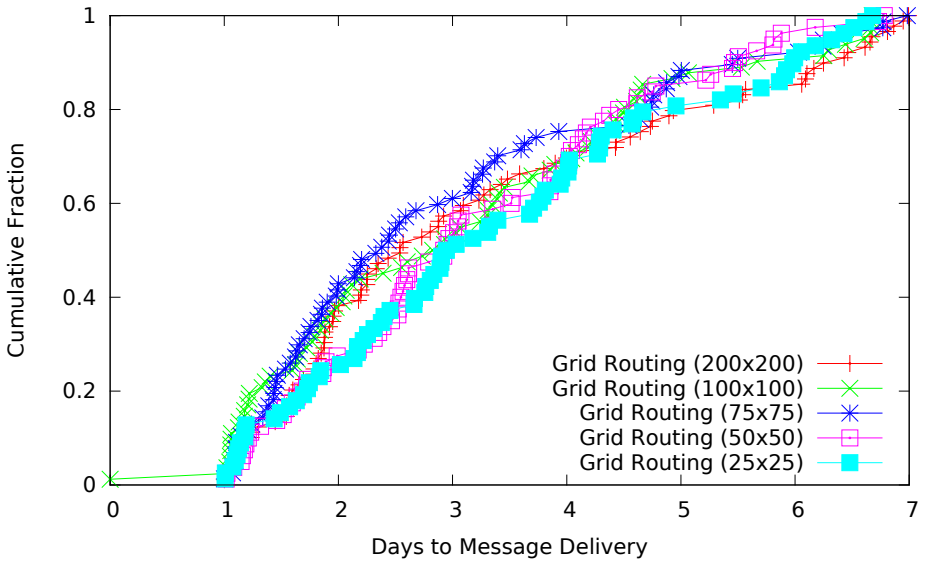

Figure 4.5: The amount of time until each successful transfer

many others when seen. However, there is enough of a distinction that the results are useful to us.

Figure 4.6 shows the amount of time needed for the successful transfer in our dolphin simulations. Though the median times are all very similar, in the quickest $35 - 40\%$ and slowest 20% of transfers, both grid and centroid routing protocols achieve success faster than the other protocols.

It is important to note that this function is a step function because it accounts for the researchers taking seasonal time off. This does not affect the results as we expect dolphin movement patterns to remain similar throughout the simulation, with message passing resuming in the same place it stopped at the end of the season.

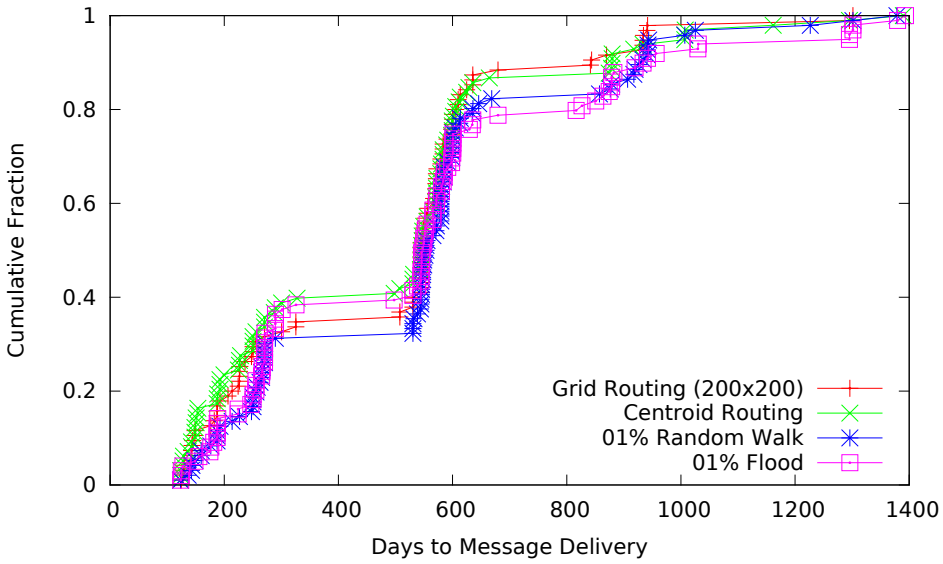

Figure 4.6: The amount of time until each successful transfer

Figure 4.7 shows the number of transfers needed to perform a message delivery in the successful simulations. As with the SLAW data, Grid Routing far outperformed the other protocols by this metric. We believe that the Grid Routing protocol is very successful and precise when finding viable carriers among the network nodes, which is reflected in the results.

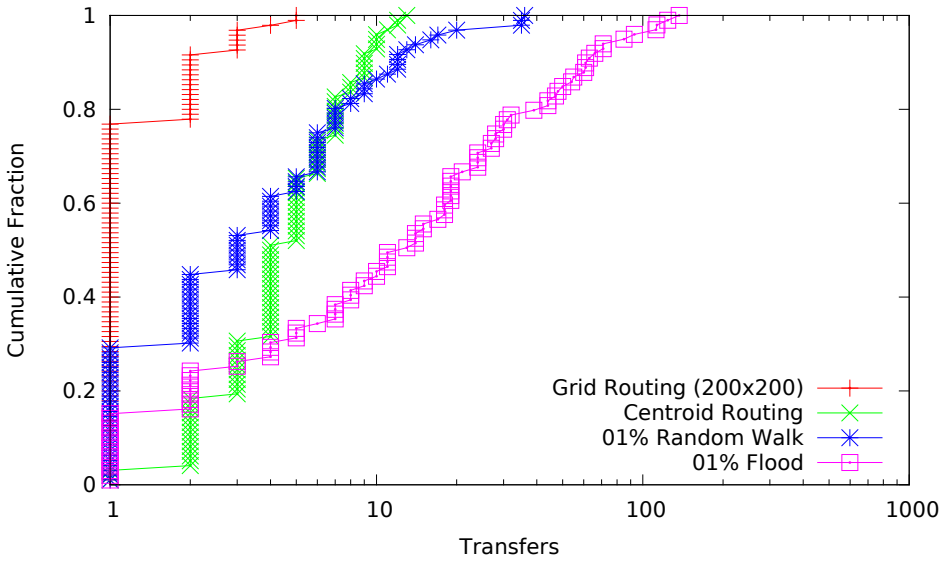

Figure 4.7: The number of transfers needed for successful delivery

Chapter 5

Discussion

In this thesis, we introduced the Centroid Routing Protocol and Grid Routing Protocol for decentralized routing of messages. Our Grid Routing Method had varying degress of success, but was on average the best of the non-flooding routing protocols, however the Cetroid Routing Method performed very poorly, with a lower delivery rate and longer time to delivery than a random walk. Our discussion considers the datasets while determining why the Centroid Routing Protocol underperformed. In addition, we discuss the limitations of our Grid Routing Protocol.

5.1 Centroid Routing Protocol

We believe that one possible reason for the Centroid Routing Method underperforming is the large range of locations traveled to by the nodes. Figure 5.1 shows a sampling of the paths traveled by four nodes in the network. Though there are areas visited much more frequently, the amount of time spent away from these areas is long and the distance traveled far enough that the centroids could become distorted.

Changes to the way centroids are computed could help this method create a more accurate home area for the nodes. One possible solution would be to only consider a location relevant to the centroid calculation when the node changes direction. Currently, each node records its location once every 60 seconds. This is problematic when the node travels long distances from its centroid, as the same amount of records

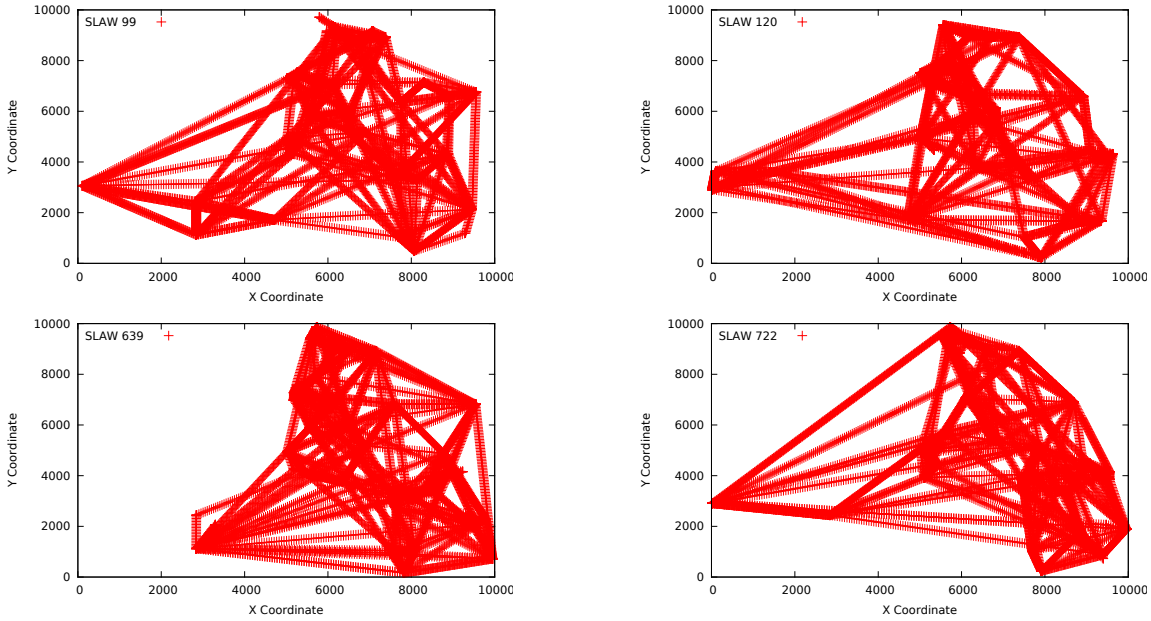

Figure 5.1: The paths traveled by four nodes in the SLAW network

would be taken on this trip than many short trips around a centroid. Though this node's home area should be close to the location of the short trips, it will instead fall somewhere inbetween. Figure 5.2 displays this senario. Another possible change to the protocol would be having the user of the smartphone inputing their own centroid, which is compared to others as they move around. This could be more accurate but is succeptable to user error due to the added layer of complexity.

5.2 Grid Routing Protocol

Despite our Grid Routing Protocol's positive results, we believe that it does contain certain limitations that could be improved upon. A message under this protocol will only be passed to a node that has been seen at the exact grid point of the address.

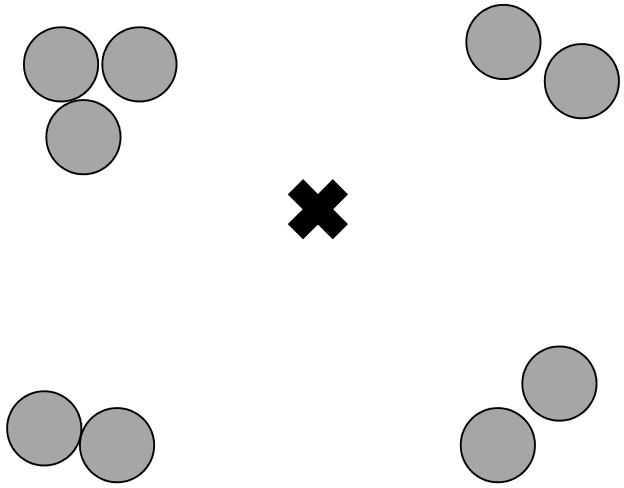

Figure 5.2: An an example of an unhelpful centroid

This makes the protcol very selective about which nodes are elligible to be carriers. It is possible to conceive of a situation where a sender never encounters a node that has been at an exact grid point, but has been seen close to some of those that are adjacent. A logical extension of our protocol would consider not just the single point of address, but also the surrounding points with some weighted probability. This would increase the complexity of the protocol as nodes come into contact while potentially finding a good carrier earlier.

Chapter 6

Conclusion and Future Direction

This thesis proposed two protocols for decentralized routing which could be used by smartphones to covertly pass messages. We introduced a Grid Routing Protocol which performed well compared to other naive solutions and our other protocol, the Centroid Routing Protocol.

Our Grid Routing Protocol performed best of the methods we introduced and our Strawman solutions. With the SLAW trace data, this protocol had up to an 18 percent increase in success rate with comperable time to delivery, and with the dolphin data a faster time to delivery with a comperable delivery rate. In all cases, the protocol was able to successfully deliver the message in far fewer transfers than the others, often an order of magnitude less.

We believe that the Centroid Routing Protocol performed poorly because a centroid is an inaccurate way of measuring a home location. However, we do believe this protocol could have uses. If messages needed to be transported over long distances, such as across a country, we believe the Grid Routing Protocol would fail because none of the nodes encountered by the sender will have records at the addressed grid point. For this reason, a combination fo the Centroid and Grid protocols may be ideal for such routing. The Centroid Protocol will be able to traverse large distances because it considers every place a node has been when deciding on a transfer, not just a single point, and the message will be routed closer through intermediary nodes that may not have centroids close to the address, but rather centroids that are *closer* to

the address than the receivers. We beleive that using the Centroid Routing Method until the message is within a certain distance from the address, than switching to the Grid Routhing Method would be an effective combination for routing messages over long distances.

One aspect of the challenge of routing in a smartphone network which we have not addressed is the possibility of nodes dropping out of the network. This could happen when a smartphone runs out of battery life or is powered down for some reason, as well as its owner moving out of range from any other smartphones for an extended period of time. A possible solution for this could include the sender transfering the initial message a fixed number of times, or having certain nodes in the network make copies. These solutions would not only help ensure delivery in a network with non-static nodes, but would also likely increase the delivery rate given our current network parameters.

Bibliography

[1] Shark bay dolphin project. http://www.monkeymiadolphins.org/.

[2] Robert C. Connor. Dolphin social intelligence: complex alliance relationships in bottlenose dolphins and a consideration of selective environments for extreme brain size evolution in mammals. *Philosophical Transactions of the Royal Society B*, 362:587–602, 2007.

[3] Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: The second-generation onion router. In *USENIX*, 2004.

[4] Natalie Glance, Dave Snowdon, and Jean-Luc Meunier. Pollen: using people as a communication medium. *Computer Networks*, 35:429–442, 2001.

[5] David Kempe, Jon Kleinberg, and Eva Tardos. Maximizing the spread of influence through a social network. In *SIGKDD Conference on Knowledge Discovery and Data Mining*, 2003.

[6] Kyunghan Lee, Seongik Hong, Seong Joon Kim, Injong Rhee, and Song Chong. Slaw: A mobility model for human walks. In *IEEE INFOCOM*, 2009.

[7] Elaine Quijano. Egypt cuts off communication amid crisis. http://www.cbsnews.com/stories/2011/01/29/eveningnews/main7297700.shtml.

[8] Michael K. Reiter and Aviel D. Rubin. Crowds: Anonymity for web transactions. *ACM Transactions on Information and System Security*, 1(1):66–92, 1998.

[9] Clay Shields and Brian Neil Levine. Hordes: a multicast based protocol for anonymity. *Journal of Computer Security*, 10(3):213–240, 2002.

[10] Thrasyvoulos Spyropoulos, Konstantinos Psounis, and Cauligi S. Raghavendra. Efficient routing in intermittently connected mobile networks: The multiple-copy case. *IEEE/ACM Transactions on Networking*, 16:77–90, 2008.