Palestine Polytechnic University
College of Information Technology and Computer Engineering
Department of Computer Systems Engineering

جامعة بوليتكنك فلسطين

Graduation Project

Octopus: A Scalable Position-based Routing Protocol for MANETs

Prepared by :
Ahmad Amjad Jawabreh
Mariam Bader Hawamdeh

Supervised by: Dr. Liana Tamimi

A final research submitted in partial fulfillment of the requirements for the bachelor degree in Computer Systems Engineering

December 2018

Dedication

For our families.

Acknowledgment

First and foremost, we would like to thank our families for their continuous support. Every time we were ready to quit, you did not let us and we’re forever grateful. This work stands as a testament to your unconditional love and encouragement.

Also, we would like to thank our supervisor Dr. Liana Tamimi for her support and advice to be better each time.

At last, we would like to thank all the people who helped, supported and encouraged us to successfully finish our graduation project; teachers, friends or colleagues at Palestine Polytechnic University.

Mariam and Ahmad

List of Figures

List of Tables

Abstract

One of the major issues in current routing protocols for mobile ad-hoc networks (MANETs) is to develop a routing protocol that can satisfy different applications' needs and optimize routing paths to cope with the scalability of the networks.

In this document, we present the recent work in the field of position-based routing protocols and compare between them depending on network overhead, scalability, end-to-end delay, and routing strategy metrics.
After that, a new scalable position-based routing protocol called Octopus has been proposed to be used in MANETs. Octopus uses restricted directional flooding to reduce the number of packets sent over the network. The performance of Octopus has been studied using NS2 simulator. We show by detailed simulation that our proposed protocol reduces the packets routing load, bytes routing load, packet loss ratio, route acquisition latency, end-to-end delay and increasing the network scalability compared to LAR routing protocol.

Keywords: *Ad-Hoc networks, MANET, routing protocols in MANETs, position-based routing.*

Table of Contents

Chapter 1: Introduction

1.1 Overview

Mobile Ad hoc Networks (MANETs) are a decentralized type of wireless networks; they don't rely on a pre-existing infrastructure such as routers or access points that are used in wired networks. Hence, nodes share the routing information among themselves, then; routes between nodes are determined by routing protocols[1].

Routing protocols in MANETs are set of rules used to specify how routes in the network communicate with each other and which can be the optimal, efficient or robust route to share information between mobile nodes based on used strategy[1]. Although they are difficult in design, they have recognizable improvement in recent years because of widespread mobile networks and the urgent need of them.

There are different routing protocols developed by researchers since 1990s, that protocols depend on different strategies like the determining the destination position for the whole mobile network which called position-based routing protocols which use location determining services like GPS[1].

There are many position-based routing protocols used in MANETs, which have some drawbacks that influence the network performance, such as large overhead and large delay.

The aim of our research is to propose a new position-based routing protocol to be used in MANETs that can increase the network performance and scalability.

1.2 Motivation and objectives

What drives us to do this research project is our interest in this field because of its great importance in our life, like community networking, emergency deployment, and communication in disasters because MANETs are router free, fault tolerance and more economical[2].
Generally, we aim to focus on improving the scalability which is the capability of the network to handle a growing amount of work without affecting the performance of the network connection by reducing routing overhead to be more applicable.

The objectives of this research are as follow:

1. Survey the common position-based routing protocols used in MANETs.
2. Propose a new scalable position-based unicast routing protocol to be used in MANETs.
3. Implement and test the proposed routing protocol and measure its performance.
4. Compare the performance of the proposed routing protocol with LAR routing protocol and study the effect of number of network nodes variation, network area variation and network nodes speed.

1.3 Chapters Overview

The rest of this document is organized as follows:

Chapter 2 discusses the main concepts of MANETs. Chapter 3 is an overview of the recent work in position-based routing protocols used in MANETs. Chapter 4 discusses the methodology of the proposed routing protocol. Chapter 5 presents the simulation environment, results and performance evaluation of the proposed protocol and another MANETs routing protocol called LAR[3] and chapter 6 includes the conclusion and the future work.

Chapter 2: Background

2.1 Wireless Networks

Wireless networks are computer networks that use wireless connections to transmit data between network nodes, they emerged in the 1970's and became increasingly popular until today because they provide access to information regardless of the geographical location of the user.
There are two categories of wireless networks; infrastructure and infrastructure-less networks.
Infrastructure wireless networks, also known as cellular networks, have permanent base stations which are connected to other base stations through links. Mobile nodes communicate with another one through these base stations[4].
Infrastructureless wireless networks, also known as ad hoc wireless networks, decentralized type of wireless networks; they don't rely on a pre-existing infrastructure such as routers or access points that are used in wired networks[4].

2.2 Ad-hoc Networks

2.2.1 Definition

Ad-Hoc networks are a decentralized type of wireless networks, they are self-configuring networks formed when a set of mobile nodes joined together, and create a network by agreeing to route messages for each other.
Ad-Hoc networks don't have a fixed infrastructure, so there are no fixed routes between mobile nodes. Instead, each contract acts as a router and routes traffic for other nodes in the network.
MANETs are a type of Ad-hoc networks whose topology changes rapidly[1].

2.2.2 Applications of MANETs

With the widespread of mobile devices as well as progress in wireless communication, Ad-Hoc networking is gaining importance with the increasing number of widespread applications in commercial and military operations. MANETs allow users to access and exchange information regardless of their geographic position or proximity to infrastructure[5].
The distributed, wireless, decentralized, and self-configuring nature of MANETs make them suitable for a wide variety of applications, including the following[6]:

- Emergency Sector :

MANETs can be used in emergency/rescue operations for disaster relief efforts, e.g. in the fire, flood, or earthquake, while all infrastructure of communication equipment is destroyed, MANETs allow taking over the transmission of rescue request messages.

- Military Sector :

MANETs allow soldiers, vehicles, and military information headquarters. to communicate and transmit information regardless of their location.

- Education Sector :

MANETs allow exchanging data files in educations conferences and lectures.

- Weather Information

2.3 Importance of Routing Protocols in MANETs

A routing protocol supports the routing of packets over the network and it's the responsibility of routing discovery and packet delivery issues.

In wired networks, routes are discovered by distance vector or link state routing protocols, which are do not work well with MANETs, because MANETs have limited bandwidth and they do not have central control[2].

Routing Protocols in MANETs are difficult to implement because they have to face the challenge of infrastructure less and dynamic network.

In MANETs, nodes communicate with each other without the intervention of routers or access points. Each node in MANETs acts both as a router and a host. For all these reasons, routing in MANETs is an important and hard case to perform robustly and efficiency[1].

2.4 Classification of the routing protocols in MANETs

There are many routing protocols proposed for MANETs, these routing protocols are divided into two types: Topology-based and Position-based routing protocols (as shown in fig. 2.1).

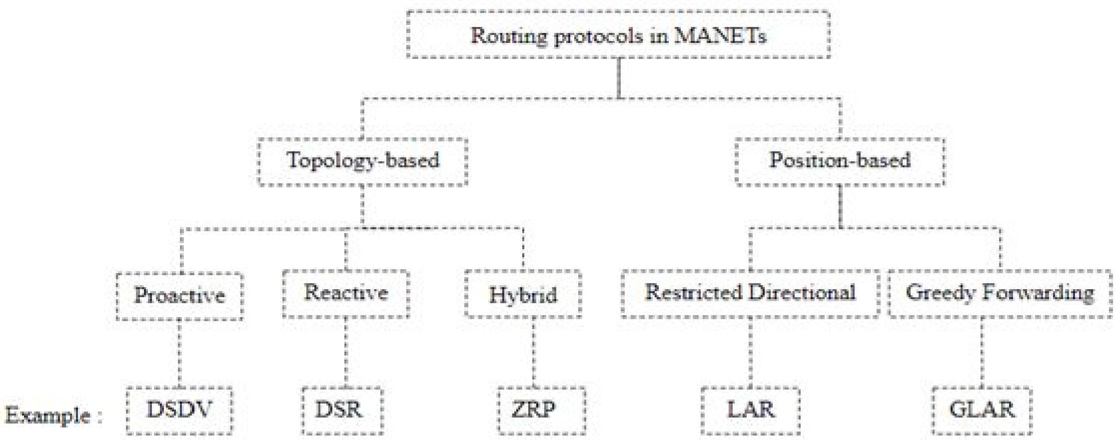

Fig. 2.1: Types of routing protocols for MANETs and their classifications

2.4.1 Topology-based routing protocols

In Topology-based routing protocols, the route between nodes is based on the information about the links existing in the network and it is divided into three types: Proactive (Table-driven) protocols, Reactive (on-demand) protocol and Hybrid (proactive/reactive) protocols[1].

- **Proactive routing protocols**

In this type, each node has a routing table which contains all possible paths towards the destination. However, this type has some drawbacks, such as high traffic and slow reaction to failure. Examples of this type are Destination Sequenced Distance Vector (DSDV), Optimized Link-state Routing (OLSR)[4].

- **Reactive routing protocols**

This type is the opposite of proactive protocols. Such the route only discovered and maintained when the source wants to send the data packet to the destination, therefore, reactive routing reduces the overhead in the network but in turn, it has some drawbacks, such as delay in the first transmission due to route discovery before exchanging packet. And it produces huge control packets due to route discovery during topology changes. Examples of this type are Dynamic Source Routing (DSR), Ad-Hoc On-Demand Routing (AODV)[1].

- **Hybrid routing protocols** use a combination of the strong points of reactive routing protocols and proactive routing protocols. For Example, Zone Routing Protocol (ZRP)[1].

2.4.2 Position-based routing protocols

Position-based Routing Protocols depends on the geographic position of the nodes to overcome the disadvantages in topology-based routing protocol like the overhead and flooding of request packet over the whole network. Each node aware of its own position, the position of its neighbors and position of destination node using some kind of Location services, like the Global Position System (GPS)[7]. Through this information, the efficiency and the performance of the protocol will be improved. Examples of this type, Greedy Perimeter Stateless Routing (GPSR)[1] and Location-Aided Routing (LAR)[3].

Chapter 3: Literature Review

In this chapter, we are presenting a survey of a set of recent work in position-based routing protocols used in MANETs.

3.1 Studied protocols

This section explains some of the recent and common position-based routing protocols.

3.1.1 Distance Routing Effect Algorithm For Mobility (DREAM)

DREAM routing protocol[8] has been proposed in 1998 and aims to reduce overhead, bandwidth loss, and transmission power consumption by implementing the following strategies: At the beginning, each node in the network knows its geographical coordinates using the GPS. Then the nodes exchange their coordinates with each other and record the location information in a location table stored in the memory of each node of them. If any node, source node attempts to send a message to another node, destination node, source node determines the destination node location using a location service. After that, the source node searches in its location table for all the nodes that are located between the source and the destination in one direction. The source node checks if the location data that came from the previous search is valid or not based on the difference between the current time and the time of the last update. If that difference is less than a predetermined value, the source will send the message to the valid next hop in the same direction as the coordinates of the destination. The node that receives this message checks if it's the destination or not. If it's the destination, it will send an acknowledgment to the source. If it's not the destination, it will pass the message after the checking procedure. But, if the difference between the current time and the time of the last update of the location table is not less than the predetermined value, this means that there's no information about the destination in the current location table. So, in that case, the source will flood the network with the message until it reaches the destination.

The implementation results show that DREAM reduces the routing overhead compared with other proactive routing protocols because the node passes the message only when it's in the direction between the source and the destination. In addition to consuming less bandwidth compared with other proactive routing protocols because it uses the distance-vector information instead of link-state as in other proactive routing protocols. Results also show that DREAM is a robust protocol, where the message will reach the destination by following the possible routes. But on the other hand, the node that will work on DREAM has to have a large memory for the location table, and DREAM cannot work without location services[8].

3.1.2 Location-Aided Routing Protocol (LAR)

LAR[3][9] has been proposed in 2000 to reduce the overhead of topology-based routing protocols. LAR describes how location information may be used to reduce routing overhead in MANET by limiting flooding of routing request packets in a smaller set of nodes related to the same zone which called request zone and discovered by GPS.
Two different schemes of LAR have been proposed to make the request zone. In LAR scheme 1;

request zone is set to be as rectangular, the rectangular zone includes the source (S) node and the expected zone, where the destination is expected to be there based on the recent location checked (as shown in fig. 3.1). The source node sends route request packet to each neighbor node discarding the nodes outer the request zone until reach the destination node, then the destination replies with its current location for future use.

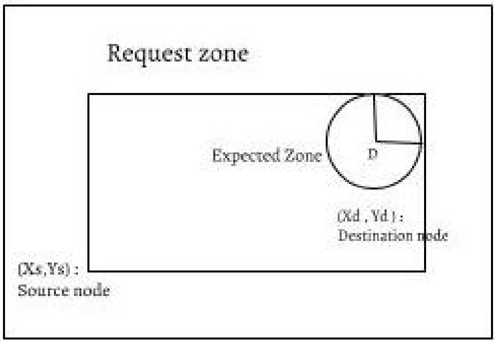

Fig. 3.1: LAR scheme 1

In LAR scheme 2; request zone represents the set of nodes closer to the destination (as shown in fig. 3.2), the source node (S) sends a route request packet to that zone, nearby nodes which are called intermediate nodes receive that packet, then the packet is forwarded to destination if the distance is less, otherwise it's dropped.

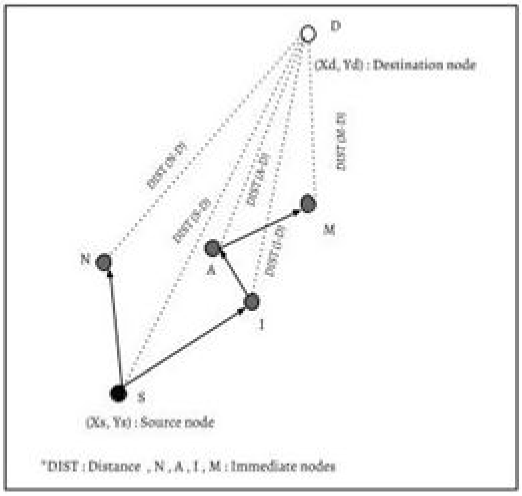

Fig. 3.2: LAR scheme 2

The performance of LAR was studied in comparison of pure flooding routing, based on several cases by varying the number of nodes, the transmission range of each node and moving speed. The implementation results show that LAR routing protocol precedes pure flooding in minimizing of network overhead by reducing the number of control packets sent because of the minimization of search zone depending on GPS. Moreover, LAR performs better than pure flooding at various speeds and transmission ranges and results show that it's a scalable routing protocol. But, due to the dependency on GPS, some errors in location estimation may be happen because of the mobility of nodes in MANET. LAR routing protocol became the basis of subsequent studies to enhance its performance[3].

3.1.3 Greedy Location-Aided Routing Protocol (GLAR)

GLAR[10] has been proposed in 2009 to improve the efficiency of LAR. The main difference between LAR and GLAR is that LAR uses the restricted directional flooding of request packet over the request zone for route discovery process, while GLAR uses greedy forwarding of request packet based on the shortest distance to the destination. GLAR first decides a baseline (as shown in Fig. 4) which is a virtual direct connection between the source node (S) and the destination node (D). The source sends the request packet to the neighboring nodes, each node computes the distance between it and the baseline, the node with the shortest distance will be chosen to be the next node in flooding process until reaching the destination node.

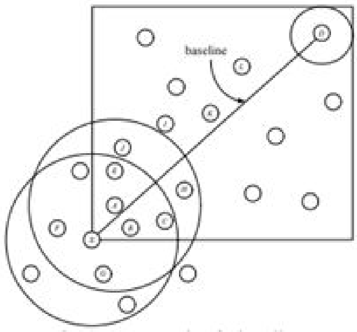

Fig. 3.3: An example of GLAR's baseline[10]

The implementation results show that the control overhead of GLAR is lower than that in LAR at different mobility speeds, so it surely improves the efficiency of LAR. But GLAR has low packet delivery ratio comparing with LAR, and GLAR cannot guarantee the route because of its greedy nature[10].

3.1.4 Distance-Based Location-Aided Routing (DBLAR)

DBLAR[11] has been proposed in 2009, in order to solve the problem of flooding, when the route discovery process in LAR was failed, by tracing the location information of destination node and depending on the change of the distance between nodes. In this protocol, when the source wants to send data to a target node. Firstly, it will use the LAR protocol for routing discovery process. But, if the destination is not inside the request zone the routing discovery will fail. So in this situation, DBLAR sends the location request outer the request zone. When a node receives a location request, the location list will be checked, if it has information about the destination node or not. If it has, the node returns the location reply. Otherwise, location error will be returned. When the source receives the location reply, it will send route request. When the nodes receive this request, each node will check if it has a route to the destination. If yes, the node will return route reply. Otherwise, the node sends the packet, when the distance between itself and the destination node is less than the distance between the preceding node and destination node and so on until to find the destination.
The implementation results of DBLAR show that DBLAR avoids flooding over the network. In addition of that, DBLAR has more packet delivery ratio and less end-to-end delay, less routing-load comparing with LAR. Also, DBLAR reduces the network overhead of LAR. But if all routes in DBLAR are broken off pure flooding will be used[11].

3.1.5 Improved Hybrid Location Aided-based Routing Protocol (IHLAR)

IHLAR[12] has been proposed in 2011 to overcome the major problems of reactive and reduce the end-to-end delay. IHLAR combines the advantages of the reactive routing protocol Ad-hoc On-demand Distance Vector (AODV)[13] and geographical routing protocol Angle Routing Protocol (ARP)[14]. IHLAR divided into two modules. The first module is topology-based routing (Intra-zone-Routing). In this algorithm, each node has a table that consists of information about nodes that are still alive within a number of P hops. Based on this table, each node can determine the zone with its neighbors. So when the destination present within P hop, the source or intermediate node will use AODV for routing the packet (as shown in Fig. 3.4).

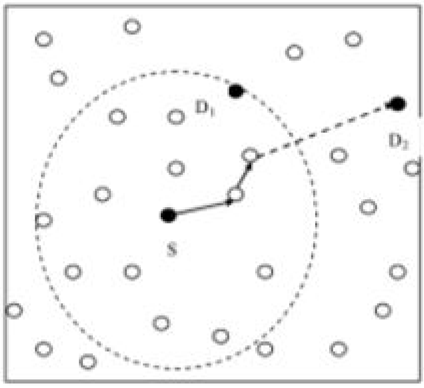

Fig. 3.4: IHLAR’s routing discovery using topology-based routing[12]

The second module is the geographical routing (Inter-Zone-Routing). This algorithm is used when the destination cannot be reached within P hops. In this situation, The source or intermediate node uses the greedy forwarding to sends the packet to the nearest node to the destination. But, when the source is the nearest one, the source will use ARP to sends the packet to the closest node counterclockwise (as shown in fig. 3.5).

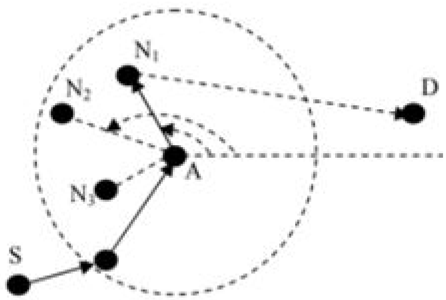

Fig. 3.5: IHLAR’s routing discovery using geographical routing[12]

The implementation results show that IHLAR achieves higher percentage delivery rate and less end to end delay than AODV. IHLAR overcomes the problems of reactive routing, reduces the end to end delay and improves the performance of path length of geographic routing. But in some situations, IHLAR can't guarantee the path from source to destination, because it’s depending on greedy forwarding strategy[12].

3.1.6 Location-based Routing Scheme with Adaptive Request Zone in Ad-hoc networks (LoRAReZ)

LoRAReZ[15] has been proposed in 2013 to reduce the routing overhead of LAR. LoRAReZ reduces routing overhead by applying two mechanisms. First one is to select the appropriate expected zone level depending on the distance between the source and destination nodes according to a selection mechanism called the Expected Zone Selection Mechanism. Then using the Request Zone Setting Mechanism to set the request zone depending on the expected zone level selected in the first step. So the size of the request and expected zones is set dynamically based on the distance between the source and the destination nodes.

The simulation results of LoRAReZ scheme has shown the better performance compared to LAR and Location-aware adaptation of request zone for mobile ad hoc networks (LARDAR) routing protocols. LoRAReZ reduces routing overhead and average end-to-end delay and increases packet delivery fraction and throughput comparing with LAR. But the calculation operation of the appropriate expected zone level is complex[15].

3.1.7 Greedy Zone Routing Protocol (GZR)

GZR[16] is a two-level routing technique proposed in 2015 and aims to divide the whole network into very small subnetworks. Each subnetwork contains a group of nodes that are geographically close to each other. That subnetwork is called zone, where each node in the network belongs to a unique zone.

Each zone contains bordering nodes, these nodes are located in the border of each zone and neighboring the other zones (as shown in fig. 3.6).

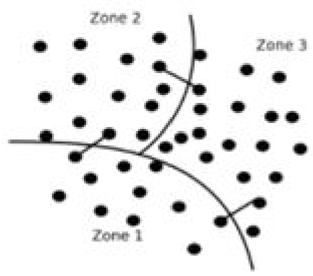

Fig. 3.6: Zones and bordering nodes[16]

Bordering nodes maintain a routing tree within their zone. This routing tree shows how each node can establish a connection with other nodes in the same zone (as shown in fig. 3.7).

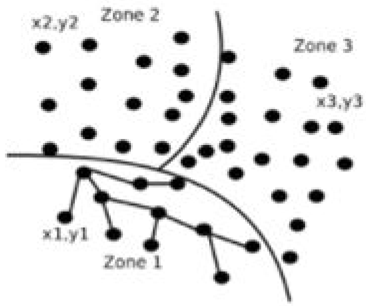

Fig. 3.7: Routing tree within a zone[16]

A connection between two nodes from different zones can be established by the connection between the source node and the bordering node nearest to the zone of the destination node. Then the next bordering node can direct this connection to the destination if the destination in the zone of that bordering node using routing tree.

The simulation results show that GZR is surly scalable and it guarantees the route from the source to the destination. But it increases the overhead especially over the bordering nodes because of the mobility of the nodes[16].

3.1.8 Hybrid On-demand Greedy Routing Protocol with Backtracking for MANETs (HGRB)

HGRB[17] is a routing protocol proposed in 2016, combines the features of position based and topology-based routing. In this protocol, each node that participates in route discovery process has seen-table that helps in the selection of the best neighbor and route table. So when the source or intermediate node wants to send a request packet, it selects the best neighbor based on the seen table. When the node receives the route request, it will search in its tables if it has the route to the destination. If yes, it will send the route reply to the source, otherwise, it will send the packet to best next hop. When the node can't be the best next hop, it will send a negative response to the previous node. After that, the source or intermediate node will see the remaining neighbors, but if the current node doesn't have neighbors to forward the packet or the neighbors of the current node can't forward the packet. The current node sends the route request back to the source which will see its remain neighbors. If no one of neighbors is valid to be next best-hop then the route request will be sent back to the node that forwards the packet.

The implementation results of HGRB show that HGRB minimizes the propagation of redundant route request. But HGRB is time-consuming and it can't guarantee the path between source and destination, since it is based on greedy forwarding[17].

3.1.9 Improved Greedy Forwarding Scheme (IGFS)

IGFS[18] has been proposed in 2017 to improve the efficiency of routing by combining two greedy forwarding methods. The first is distance-based, which selects the nearest node to the destination and the second is deviation-based, which minimizes the spatial distance between source and destination. This protocol defines an equation to combine the previous methods, where the equation is based on distance from a current node to the destination, transmission range and the angle of

deviation of the selected node from the destination. This equation assigns a specific value to each node where the node has the highest score is selected as the next forwarding node.
The implementation results show that proposed protocol increases the quality of the route in terms of both stability and reliability over conventional and direction-based algorithm if they are used separately. IGFS increases the stability and reliability of the route towards the destination. But in some situations, it can't guarantee a path from source to destination, because it’s based on greedy forwarding[18].

3.1.10 Adaptive Hybrid Geo-casting Routing Protocol for MANET (HGRP)

HGRP[19] has been proposed in 2018. HGRP uses the location services to enable each node in the network to be aware of its own location. Before any transmission process, the source will be aware of the location of the destination at specific time. If it does not exist in that location, the source builds a virtual triangle called Triangle region that connects the source(S) with the previous (Dt) and the current destination location (Dt+V) the area in the triangle region called restricted zone, the source forwards the request packet to one of the intermediate nodes located(e.g.: N1, N2) in the restricted zone. The same previous process will be conducted by the intermediate nodes (N1, N2) (as shown in fig. 3.8).

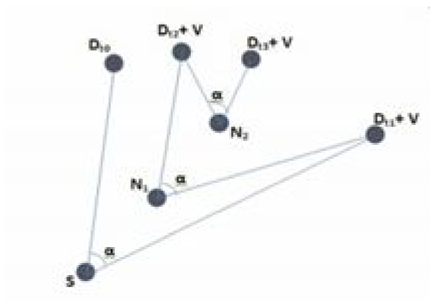

Fig. 3.8: Routing discovery process in HGRP[19]

Any node should fulfill these conditions before it forwards the request packet to next hop; the next hop should be closer to the destination, the received signal to noise plus the interference ratio should be higher than a predefined threshold and the next hop’s buffer has enough space to receive a new data packet.
The implementation results show that HGRP has less packet loss ratio and control overhead comparing with the classical proactive routing protocols like AODV, and it works efficiently with the networks with heavy traffic load and that contain nodes with high moving speed. But the selecting process of the next hop is somehow complex because it must fulfill three important conditions to be selected as a next hop. So HGRP is not always useful and it cannot guarantee the route to the destination[19].

3.2 Comparison and Summary

This section explains the metrics that considered in the comparison between the routing protocols studied in the previous section and the summary of this study.

3.2.1 Comparison Metrics and Survey Summary

The comparison of the studied location-based routing protocols is done based on the survey mentioned in the previous section. The studied methods are compared depending on five metrics provided in the studied routing protocols, which are network type, routing overhead, the scalability of the network, end-to-end delay and the strategy of routing. These metrics are defined as[20]:

- Network Scalability: is the capability of the network to handle a growing amount of work without affecting the performance of the network connection.
- Routing Overhead: is the additional bandwidth used by a communication protocol.
- End-to-end delay: is the time interval between the data packet generation time and the time when the last bit arrives at the destination.
- Routing Strategy: is the way of routing either restricted directional flooding, greedy forwarding, hierarchical routing or hybrid.
- Network Type: is the type of the network either Ad-hoc or MANET.

The following table 3.1 shows the results summary of the survey mentioned in the previous section based on the defined metrics.

Table 3.1: Comparison of location-based routing protocols

Protocol	Network type	Overhead	Scalability	End to end delay	Routing strategy
DREAM[8], 1998	Ad-Hoc	Low	Good	Low	Restricted
LAR[3], 2000	MANAT	Low	Good	Low	Restricted
GLAR[10], 2009	MANET	Low	Good	Low	Greedy
DBLAR[11], 2009	MANET	Low	Good	Low	Restricted
IHLAR[12], 2011	Ad-Hoc	High	Good	Low	Hybrid
LoRAREZ[15], 2013	Ad-Hoc	Low	Good	Low	Restricted
GZR[16], 2015	Ad-Hoc	Moderate	Good	Moderate	Hybrid
HGRB[17], 2016	MANET	High	good	High	Greedy
IGFS[18], 2017	MANET	Low	Good	High	Greedy
HGRP[19], 2018	MANET	Low	Good	Low	Hybrid

3.2.2 Summary of Comparison

After studying the previous routing protocols we found that restricted directional flooding has the preference in scalability, reduction of routing overhead and end-to-end delay over other protocols, so we chose this strategy to build our proposed protocol on.

Chapter 4: Proposed Protocol Model Design

4.1 Introduction

In this chapter, we propose a new scalable position-based routing protocol called Octopus.
Octopus is a routing protocol proposed to be used in MANETs and aims to reduce the routing overhead and end-to-end delay over the network by using restricted directional flooding strategy that by dividing the network for 8 sectors, then it floods the request packet in the sector that the destination node located in.
The source node can use some location services to be aware of the position of the destination node.

4.2 Assumptions

We assume the MANET is located in a two-dimensional area of A m * A m. N of mobile nodes are distributed randomly in this area and moving with different mobility speeds, as shown in fig. 4.1.

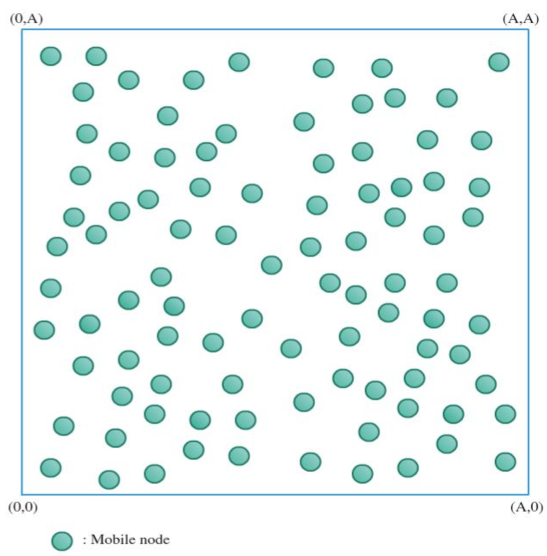

Fig. 4.1: Area A m * A m with N mobile nodes.

Each node is aware of its location using GPS service, Any node intends to send data called source node. Firstly, the source node divides the network -by default- into four sections based on its coordinates for the network, as shown in fig. 4.2.

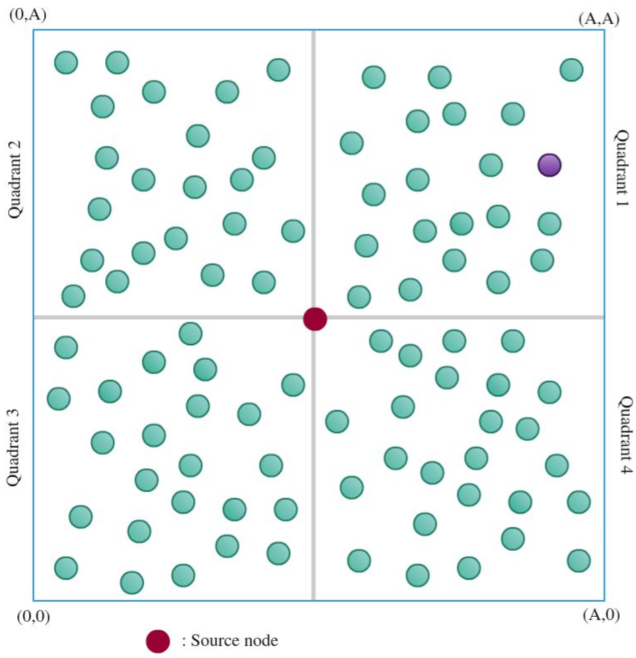

Fig. 4.2: Four sections based on source node location.

Any node can determine its location relative to the source node location, and decide in which section it's located.

4.3 Packets Format

This section explains the request and reply packets format that used in Octopus.

4.3.1 Request Packet Format

The request packet (REQ) is the first packet that sent from the source to reach the destination to perform route discovery process. REQ packet consists of 10 fields divided as shown in fig. 4.3.

Request ID
Packet Type
Source ID
Destination ID
Source Location Information
Destination Location Information
Destination Section
Destination Sector
Send time
Last Distance to the Destination

Fig. 4.3: REQ packet format.

Request ID : is an unique number for each request packet sent from the source to the same destination. Because of restricticted directional flooding routing, a node in the network may receive more than one REQ packet has the same request ID, so it will drop any REQ packet received after the first one to reduce packets routing load.
Packet Type distinguishes the request packet, reply packet and other packet types. It’s ‘R’ for REQ packet.
Source ID and **destination ID** explain the address of the source and the destination.
Source location information and **destination location information** determine the current location of source and destination. These fields contains the coordinates of the source and destination relative to the network area.
Destination Section: The section number that the destination located in relative to the source location.
Destination Sector: The sector that the destination located in relative to the section number and source location.
Send time: the time of sending this packet.
Last Distance to the Destination: which is the last computed distance between the last sent node and the destination. The receiving node computed the distance between itself and the destination using location information field. Then, it compares the result of that computation with the field of last distance to the destination, if the computed distance shorter than that stored in this field, the node floods the REQ packet after modifying of this field by replacing the previous value with the newly computed value.

4.3.2 The Reply Packet

The reply packet (REP) represents the response of the destination on the routing discovery process, it explains the final route discovered as a result of that routing discovery process. REP packet consists of 5 fields divided as shown in fig. 4.4.

Request ID
Packet Type
Source ID
Destination ID
Send time

Fig. 4.4: REP packet format

Request ID is the request id that this REP packet follows.
Packet Type distinguishes the request packet, reply packet and other packet types. It is ‘A’ for REP packet.
Source ID and **destination ID** explain the address of the source and the destination.
Send time: the time of sending this packet.

4.4 Routing Discovery Process

In Octopus, any node intends to send data to other node in the network will follow these steps :

- Depending on the network dimensions, the source firstly divides the network into 4 sections, then divides each section into 2 sectors, as shown in fig. 4.5.

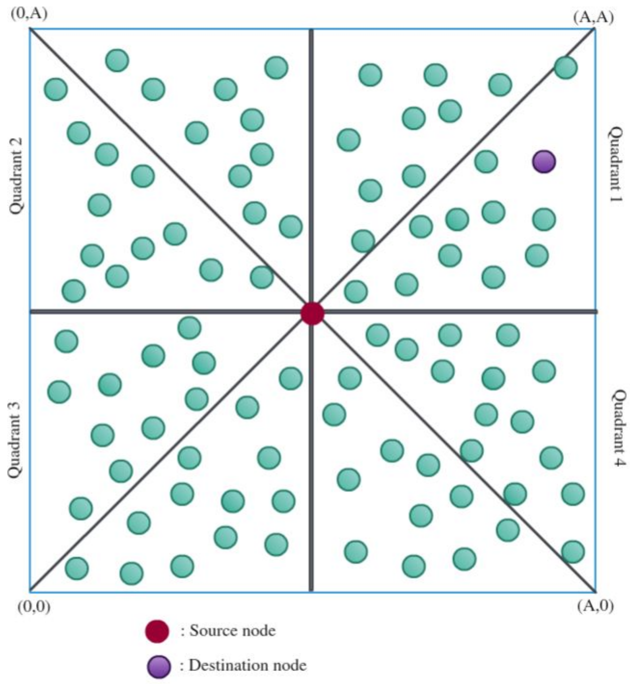

Fig. 4.5: How the source divides the network

- The source determines θ depending on the number of sectors, which equals $360^{\circ} / 8 = 45^{\circ}$.
- The source gets the position of the destination using location services. Then the source determines the request zone as shown in fig. 4.6.

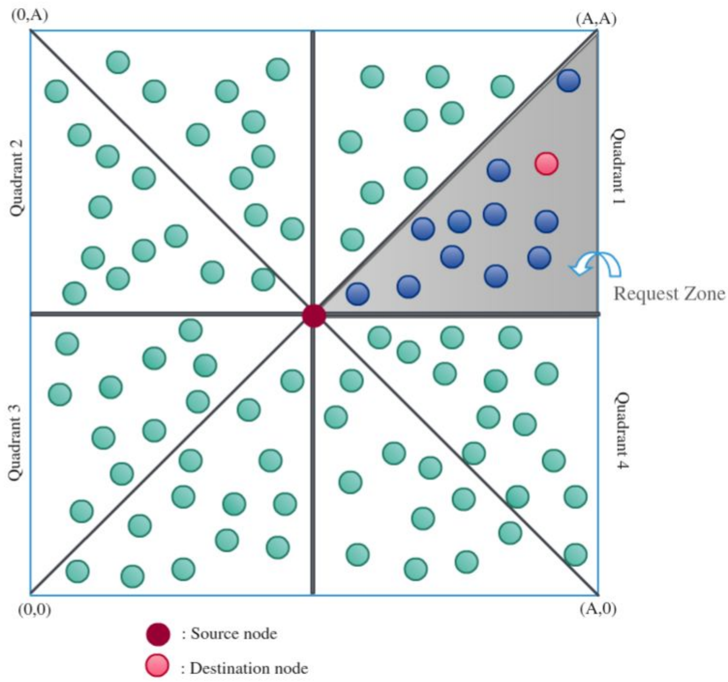

Fig. 4.6: Request Zone

- After that, the source sends a request packet (REQ packet) - to discover the specific route to the destination - using restricted directional flooding to the neighboring nodes located in the same request zone as shown in fig. 4.7.

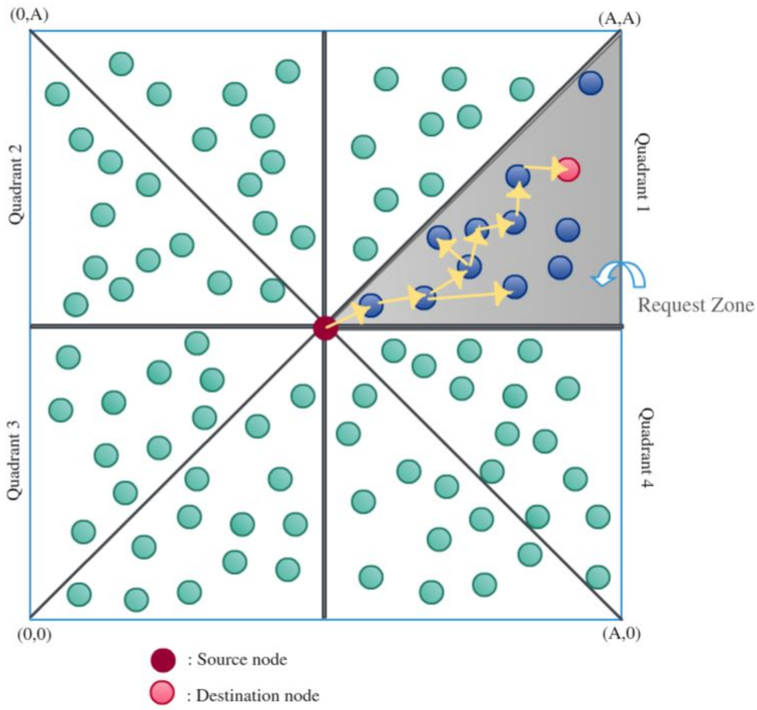

Fig. 4.7: REQ packet flooding over the request zone

- Each neighboring node that recieved the REQ packet subtracts the source coordinates from its coordinates, the result will be a pair of x and y coordinates. This result determines the sector that this node located in as explained by in the following algorithm and fig. 4.8.

Reaction of Intermediate node on receiving a REQ packet

INITIALIZATION :
QT = section that the destination located in.
ET = half of section that the that the destination located in.
I = Intermediate Node
D= Destination Node
$\theta = 45^{\circ}$
LAST DISTANCE = last less distance to the destination
$x_{Result} = x_{Intermediate} - x_{Source}$
$y_{Result} = y_{Intermediate} - y_{Source}$
IF x_{Result} IS GREATER THAN Zero And y_{Resul} IS GREATER THAN Zero
section = 1 // The Intermediate node located in first section
ELSE IF x_{Result} IS GREATER THAN Zero And y_{Result} IS LESS THAN Zero
section = 4 //The Intermediate node located in fourth section
ELSE IF x_{Result} IS LESS THAN Zero And y_{Result} IS GREATER THAN Zero
section= 2 //The Intermediate node located in second section
ELSE IF x_{Resul} IS LESS THAN Zero And y_{Result} IS LESS THAN Zero
section = 3 //The Intermediate node located in third section
IF section IS EQUAL TO QT:
Calculate $\beta = \tan^{-1}(y_{Intermediate} - y_{Source} / x_{Intermediate} - x_{Source})$

```
        IF  β IS LESS THAN θ Or IS EQUAL TO θ:
                 Eighth  = 1 of the section QT
             ELSE
                  Eighth  = 2 of the section QT
        ENDIF
       IF  Eighth IS EQUAL TO  ET  AND  Distance between I and  D IS LESS THAN  LAST
                  DISTANCE:
        LAST DISTANCE = Distance between I and  D
       LAST NODE = Intermediate  address
        SEND REQ packet  using restricted directional  flooding
       ENDIF
  ELSE
             Drop The packet
END
```

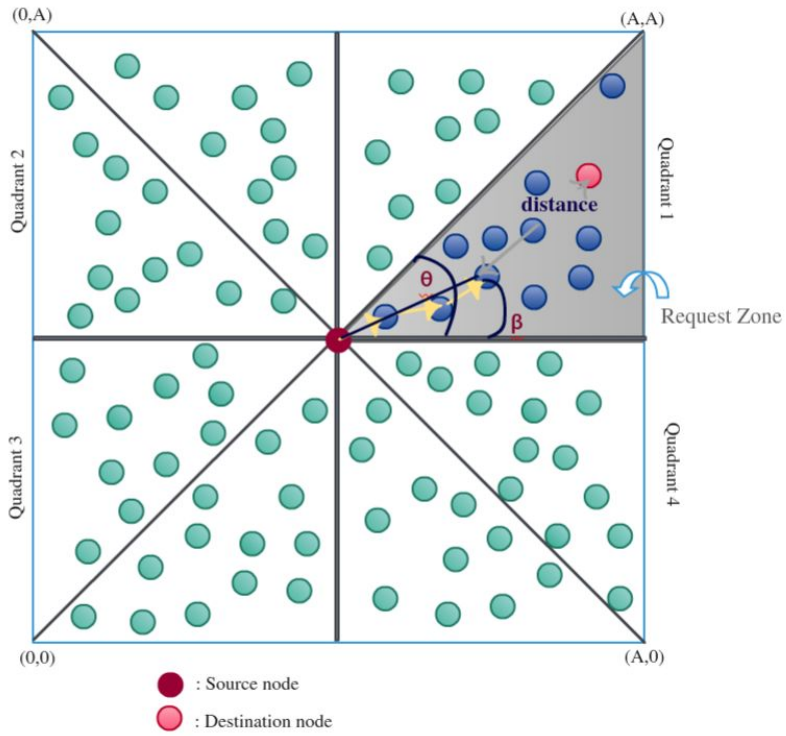

Fig. 4.8: Determining Theta θ and Beta β.

- Depending on the current position of the intermediate node relative to the source's current position. The intermediate node calculates Beta (β) which depends on the following equation.

β= $\tan^{-1}$ (Iy- Sy / Ix - Sx) Iy, Ix : intermediate node coordinates Sy, Sx : Source node coordinates

- β determines the sector that the intermediate node located in. if it is not in the request zone, it will drop the REQ packet.
- If this intermediate node located in the request zone, it will compute the distance between it and the destination using the destination location field in the REQ packet.
- If the distance is shorter than that to the previous node, the neighboring node with the least distances sent the REQ packet after modifying the Last Distance to the Destination field in

the REQ packet by the new distance the next neighboring nodes do the same process until reaching the destination node.

- When the destination receives the REQ packet, and accept the connection between it and the source, it returns REP packet to the source depending on reverse path strategy.

4.5 Reverse Path Strategy

- After receiving the first REQ packet and accepting the connection on the destination, it returns the REP packet to the last node that sent the REQ packet which that has the same request Id.
- The intermediate node returns the REP packet to the last node depending on the request Id of the REQ and REP packet.
- This process will be continued until reach the source node as shown in fig. 4.9.
- After receiving the REP packet by the source, the connection will be established depending on the discovered route, and the pending data will be sent to the destination using this route.

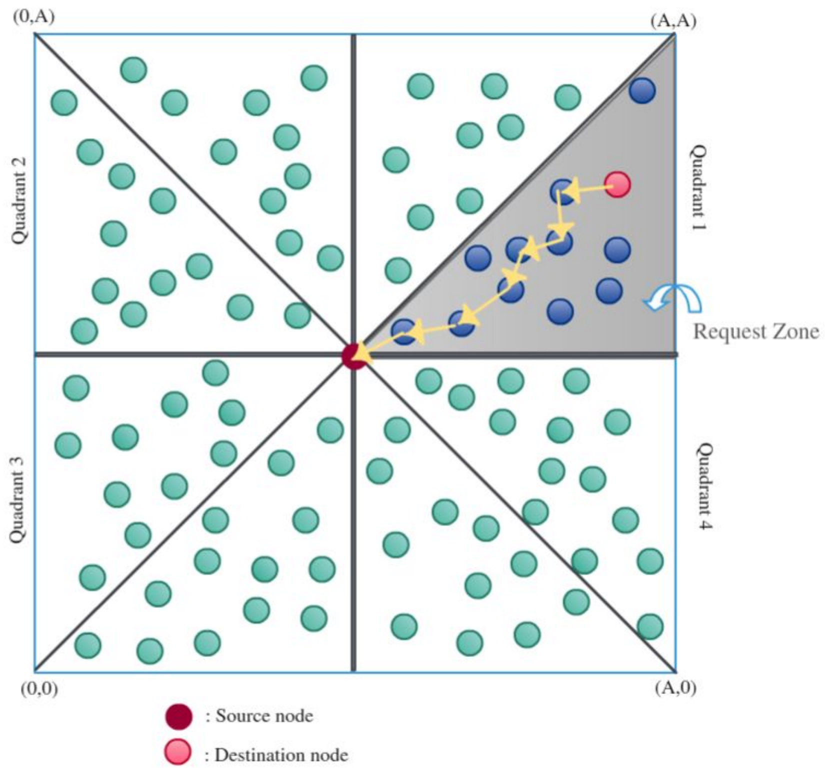

Fig. 4.9: Reverse path strategy.

4.6 Route Maintenance Phase

If the route between the source and the destination nodes is broken off, for example, the link i,j is no longer available as the node j is moved -where i and j are intermediate nodes in the route between the source and the destination-, then node i will search for another valid subroute to the destination.

If there is no another subroute, the node i will drop the data packet and returns an ERROR packet to the source which consists of the fields shown in fig. 4.10.

If a node receives an ERROR packet, it will check if it is the destination of this packet, which is actually the source and forwards the ERROR packet to the source if its an intermediate node.

If it is the source, it will create another route request packet with the previous information for that destination and send it to the neighboring nodes. Then a new route discovery process will be established.

Request ID
Packet Type
Source ID
Destination ID

Fig. 4.10: ERROR packet format.

Request ID : is an unique number for each route discovery request sent from the source to the same destination. all packets that belong to the same route request will have the same Request ID.
Packet Type distinguishes the request packet, reply packet and other packet types. It's 'E' for ERROR packet.
Source ID and **destination ID** explain the address of the source and the destination of this packet.

4.7 Expected Performance Measurement Results

This section explains the performance metrics that used to measure the performance of Octopus and LAR routing protocols and the expected results of this measurements.

4.7.1 Routing Protocols Performance Measurements

The performance of the routing protocols in MANETs can be measured through the average routing packets load, average routing bytes load, average packet loss ratio, average route acquisition latency and average end to end delay.
The average routing packets load is the average of total number of routing packets over the total number of packets received. The average routing bytes load is the average of total number of routing bytes over the total number of bytes received. The average packet loss ratio is the number of dropped packets over the transmitted packets. The average route acquisition latency refers to the delay in discovering the route to the destination. So, it is the average delay between the sending of REQ packet by the source for route discovery process and the receiving REP packet after reverse path strategy. the sending time for REQ and receiving time for REP are used to calculate route acquisition latency.
The average end-to-end delay is the difference between the time of transmitting a packet from the source and the time of receiving the packet at the destination node for all transmitted packets divided by the number of packets transmitted.
[1][21][22].

4.7.2 Expected Results

We expected that the performance of the proposed routing protocol will increase comparing to LAR routing protocol because Octopus specifies the sector that the destination located in relative to the source location and broadcasts a route request packet in this sector in specific terms until reaches the destination. While LAR floods the request zone with route request packets until reaches the destination.

More specifically, we think that the average routing packets load and average routing bytes load will be decreased because of the detection of the sector that the destination located in and the size of the request packet in Octopus is less than it in LAR.
Octopus uses a REQ packet of 34 bytes size, REP packet of 17 bytes size, Data packet of 19 bytes size and ERROR packet of 13 bytes size. While LAR doesn't use packets with fixed size because it stores the whole route in the route packet itself. While Octopus uses the nodes' storage to store the next and previous nodes in the route.
We expect also that the packet loss ratio and the average route acquisition latency and the average end-to-end delay will be decreased compared to LAR because of the packet size and the route maintenance phase. However, we think that the scalability will not be affected.

Chapter 5: Simulation And Results

The proposed protocol (Octopus) is simulated with different networking scenarios, ans compared with LAR routing protocol using ns-2.35 simulator.
The routing performance is measured by calculating the average routing packets load, average routing bytes load, average end-to-end delay and packet loss ratio.

5.1 Network Simulator 2

The network simulator 2 (NS2) is the second version of the Network Simulator (NS).
NS-2 is an open source, discrete event simulator targeted at networking research to validate researchers new theories. It is used for the simulation of routing protocols with different network topologies. It is capable of simulating wired as well as wireless networks. NS-2 is written in C++ and an Object-oriented version of Tcl which called OTcl[23].
In order to simulate via NS2, the user will use the OTCL to build the network topology, specify when from where and to where the data traffic will be sent. Moreover, the event scheduler will also be initiated by the OTCL. While the C++ in the simulator is used to program the NS2 internal objects for example the protocols.
The simulation is done by linking the OTCL and the C++ objects wia a linking language called TCLCL. The output of the simulation is found in a file called the trace file, which contains the all of the network communication information. and is animated via another file called NAM trace file.

5.2 Simulation Environment

The proposed protocol is simulated with different nodes speed ranges between 5 m/s to 10m/s and a pause time ranging between 5s to 25s, and by changing different networking factors for 300 second as a simulation time. The traffic pattern that is used is the constant bit rate (CBR), the MAC 802.11 protocol is the data link layer protocol is used, with a transmission range of 250 m and with Omni antenna as antenna type. Moreover, the nodes mobility behaviour follows the random waypoint model[24].

5.3 Simulation Results

Several experiments were performed, and the results were taken at different network areas and different numbers of nodes.

5.3.1 Nodes Number Variation Scenario

The packets routing load, bytes routing load, packet loss ratio, route acquisition latency and end to end delay are measured when varying the number of nodes in the network area. The number of nodes in the network increased between 20, 50 and 90 nodes at Area of 800m * 800m.

When increasing the number of nodes, the packets routing load of the proposed protocol is almost constant between 20 to 50 nodes then increasing when the nodes number is more than 50.
While in LAR, the packets routing load is much higher than Octopus, and sharply increasing when the number of nodes is increasing as shown in fig. 5.1.
The increase of packets routing load when the nodes density is increasing refers to the increasing of the number of nodes receiving and broadcasting the routing packets in the request zone.

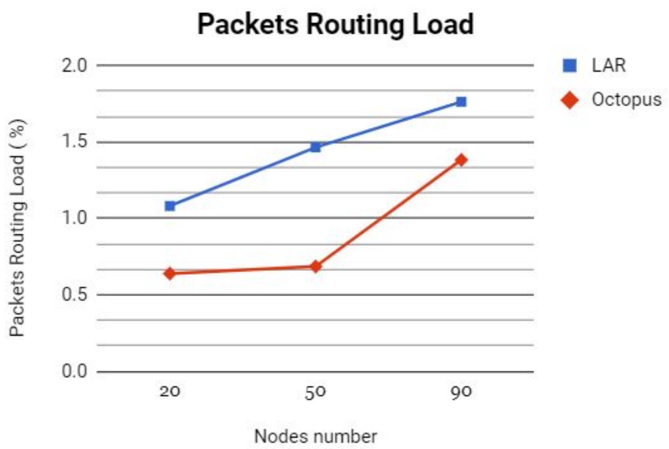

Fig. 5.1: Packets routing load vs. Network nodes number

Even more, fig. 5.1 shows that Octopus's packets routing load are clearly less than LAR. That's because the source in Octopus detects the sector that the destination located in and broadcasts REQ packet in that sector. The intermediate nodes have to check several conditions in order to rebroadcast the REQ packet. So, Octopus reduces as much as possible the useless routing packets.

The Bytes routing load in this scenario is very small in Octopus compared with LAR as shown in fig 5.2.

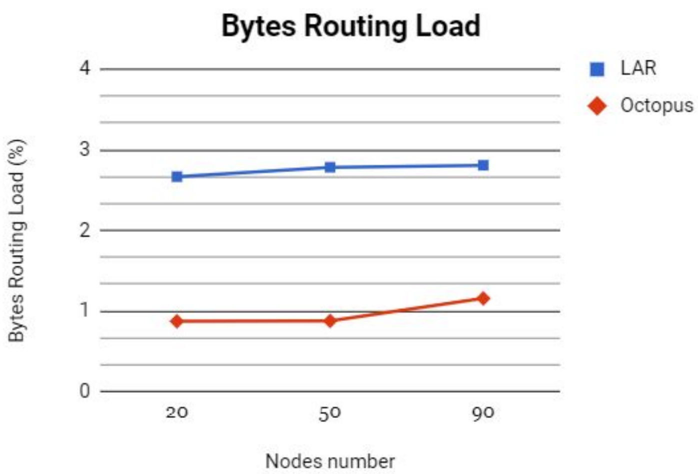

Fig 5.2: Bytes routing load vs. Network nodes number

The clear difference between Octopus and LAR in the case of bytes routing load is because the routing packets size in Octopus is fixed and less than LAR. However, packet size in LAR is variable refers to the route length, because LAR stores the whole route on the routing packet itself.
In fig. 5.3, as it can be seen, the packet loss ratio of Octopus is less than LAR, when the number of nodes is increased.

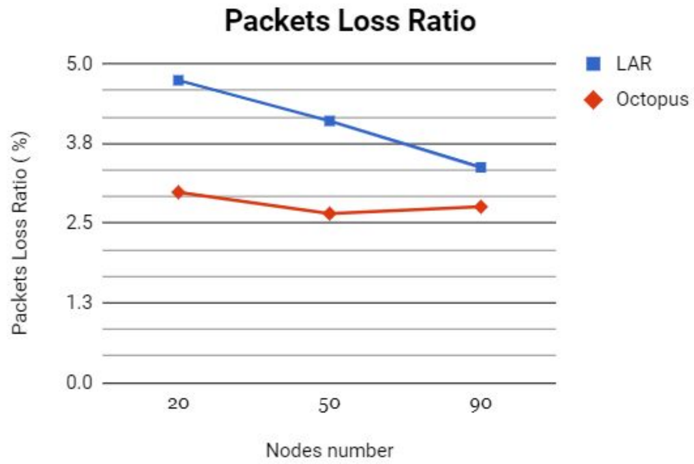

Fig. 5.3: Packet loss ratio vs. Network nodes number

The packet loss ratio of Octopus is less than LAR with the same number of nodes. However, Octopus's packet loss ratio is somewhat decreasing between 20 and 50 nodes. Then it's almost constant after 50 nodes. But the packet loss ratio in LAR is sharply decreasing when the number of nodes is increasing, but still much higher than Octopus.
The decreasing in packet loss ratio of both protocols when the number of nodes increased refers to increasing in the number of nodes in the request zone and high chance to pass the packets.

By increasing the number of nodes, the route acquisition latency figure of the two protocols is increased as shown in fig. 5.4.

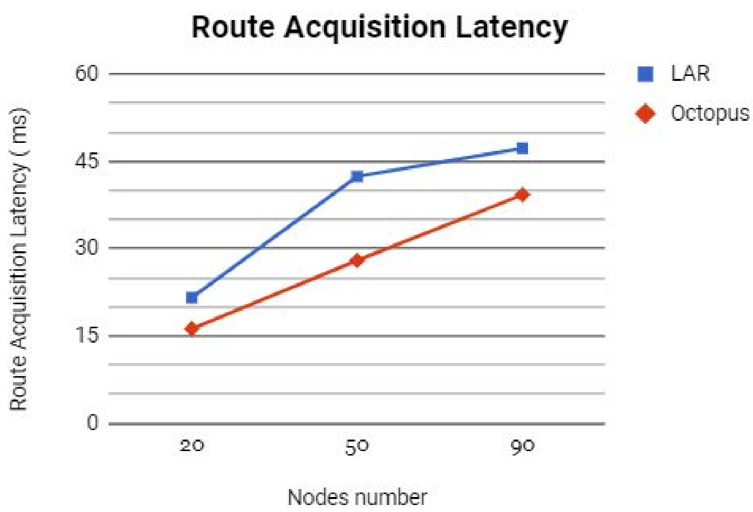

Fig. 5.4: Route acquisition latency vs. Network nodes number

Fig. 5.4 shows that the route acquisition latency of the proposed protocol Octopus is faster than LAR protocol. Moreover, Octopus's route acquisition latency increases linearly. While the increase in route acquisition latency of LAR protocol is much higher than Octopus.
The increasing in route acquisition latency refers to the increasing of delay in discovering the route to the destination when the number of network's nodes is increasing and increasing in the route length. However, it's an expected result because the distance between source and destination maybe longer when the number of nodes increased.

The average end to end delay in this scenario is very high and shows sharply decreasing in Octopus compared to LAR when the number of nodes increases from 20 to 50 nodes because the density of the nodes is increased. After 50 nodes, the average end to end delay of Octopus keeps less or equal values of LAR's end to end delay values as shown in fig. 5.5.

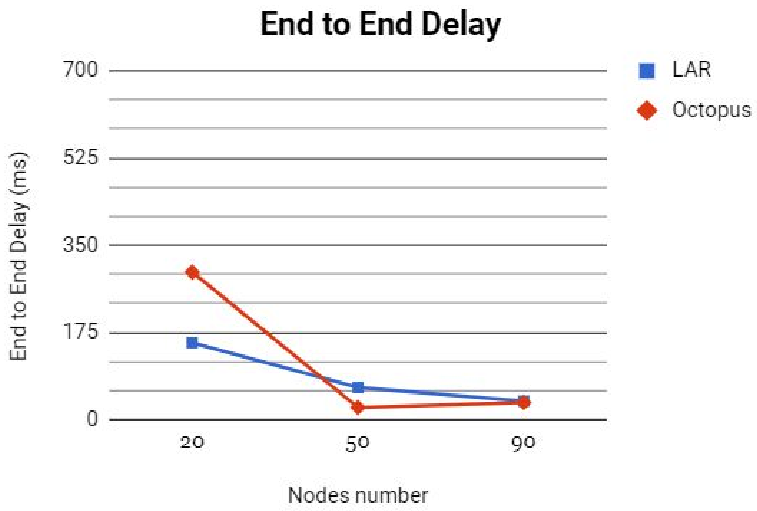

Fig. 5.5 End to end delay vs. Network nodes number

The decreasing of end to end delay in both protocols refers to the high chance to find the route and pass the data packets when the number of nodes increased.

5.3.2 Network Area Variation Scenario

The packets routing load, bytes routing load, packet loss ratio, route acquisition latency and end to end delay are measured when varying the area of the network between 200mx200m, 500mx500m and 800mx800m at the same number of network nodes.

When increasing the area of the network, the packets routing load of the proposed protocol is almost linearly increasing. While in LAR, the packets routing load is much higher than Octopus, and sharply increasing when the area of the network is increasing as shown in fig. 5.6.

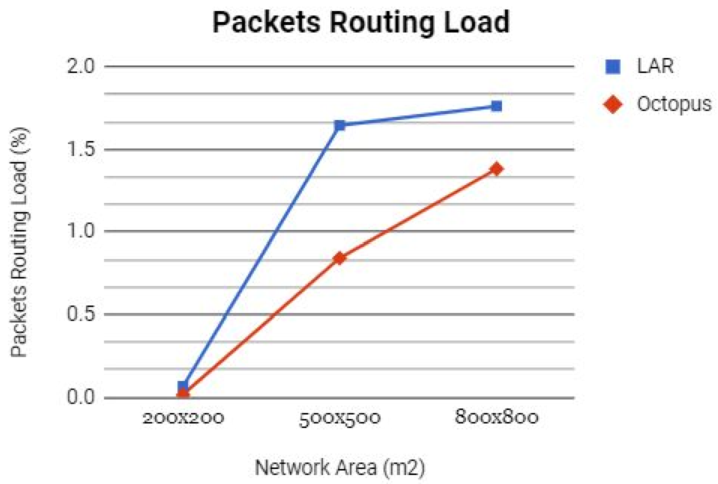

Fig. 5.6: Packet routing load vs. Network area

The increasing of packet routing load when the network area increased refers to the increasing in the number of routing packets needed to cover the connection in the request zone, because the size of request zone is increased.

The Bytes routing load in this scenario is very small in Octopus compared with LAR as shown in fig 5.7.

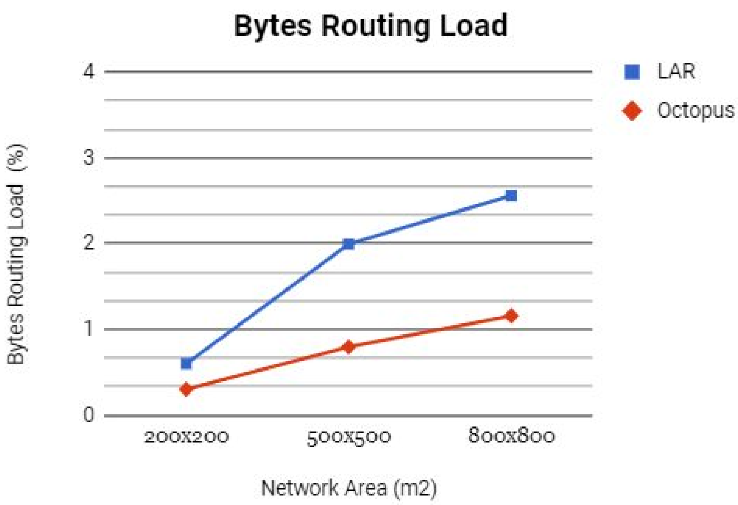

Fig. 5.7: Byte routing load vs. Network area

The increasing of bytes routing load for both protocols refers to the increasing in the number of routing packets needed to initiate the route in the request zone, because the size of request zone is increased. So, the byte routing load increases when the number packets routing increased.

In fig. 5.8, as it can be seen, the packet loss ratio of Octopus is less than LAR, when the area of the network is increased.

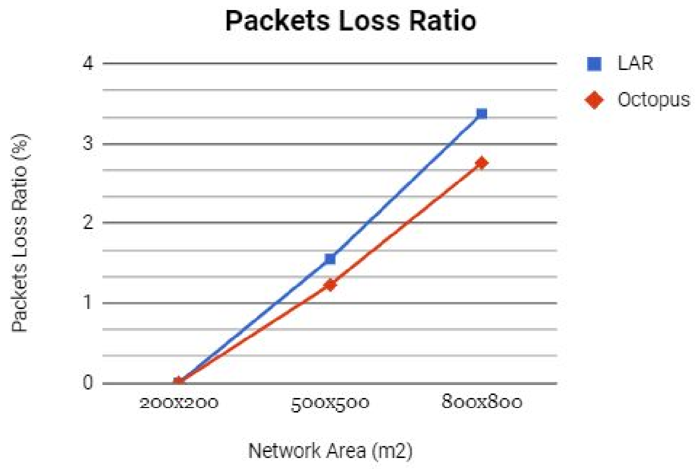

Fig. 5.8: Packet loss ratio vs. Network area

The packet loss ratio of Octopus is less than LAR within the same network area. However, Octopus's packet loss ratio is increasing. Moreover, the packet loss ratio in LAR is sharply increasing when the network area is increasing, but still somewhat higher than Octopus.
The increasing of packet loss ratio refers to the increasing of the area at the same number of nodes, so the chance of dropping data packets may be increased.

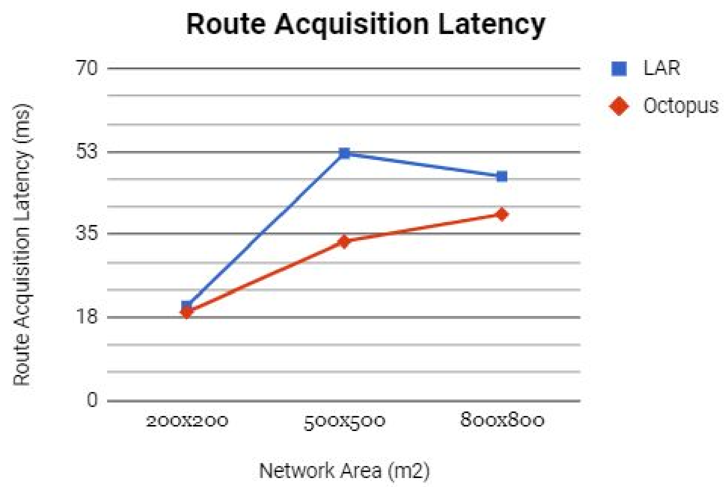

Fig. 5.9: Route acquisition latency vs. Network area

Fig. 5.9 shows that the route acquisition latency of the proposed protocol Octopus is faster than LAR protocol. Moreover, Octopus's route acquisition latency increases almost linearly. While the increase in route acquisition latency of LAR protocol is much higher than Octopus.
The increasing in route acquisition latency refers to the increasing of route length when the network area is increasing. However, it's an expected result because the distance between source and destination maybe longer when the area of the network increased at the same number of nodes.

The average end to end delay in this scenario is somewhat high and shows some linear increasing in Octopus. Compared to LAR, when the network area increases from 40000m^2 to 250000m^2, the average end to end delay increases clearly. But after that, the average end to end delay of LAR somewhat has a constant value until it has a value close to that in Octopus, but average end to end delay of LAR still higher than Octopus as shown in fig. 5.10.

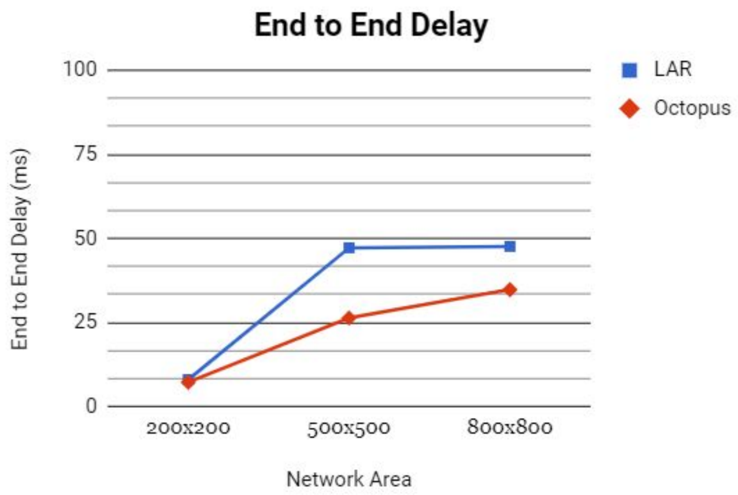

Fig. 5.10: End to end delay vs. Network area

The increasing in average end to end delay when the area is increased for both protocols refers to the increasing in route length.

5.3.3 Network Nodes Mobility Speed Variation Scenario

In this section, we compared the performance of the proposed protocol Octopus with that of LAR using the results from our simulation experiments when varying the network nodes mobility speeds between 5 and 15 m/s. This scenario is modeled a network area of 800 m x 800 m and 90 number of nodes.

When increasing the nodes mobility speeds, the packets routing load of the proposed protocol is almost constant while increasing from 5 to 10 m/s. After that, it's almost linearly increasing. While in LAR, the packets routing load is higher than Octopus, and linearly increasing when the nodes speed are increasing as shown in fig. 5.11. The increasing in packet routing load refers to the high mobility of the nodes, so the nodes has a high chance to change it position. That means the route has high chance to be broken and reinitiated, so the number of routing packets will increase.

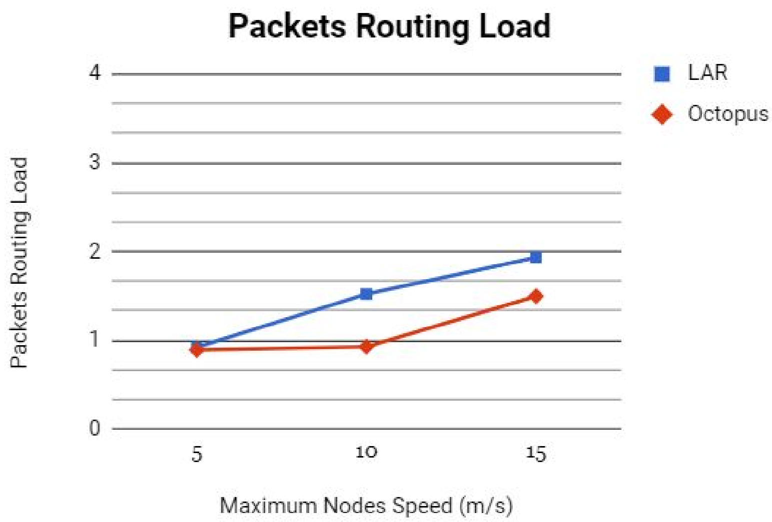

Fig. 5.11: Packets routing load vs. Maximum nodes speed

The bytes routing load is almost increasing in Octopus when the nodes speeds are increasing. But is shows some growing in LAR when the nodes speeds increasing from 5 to 10 m/s. After that, it shows some decreasing . as shown in fig. 5.12.

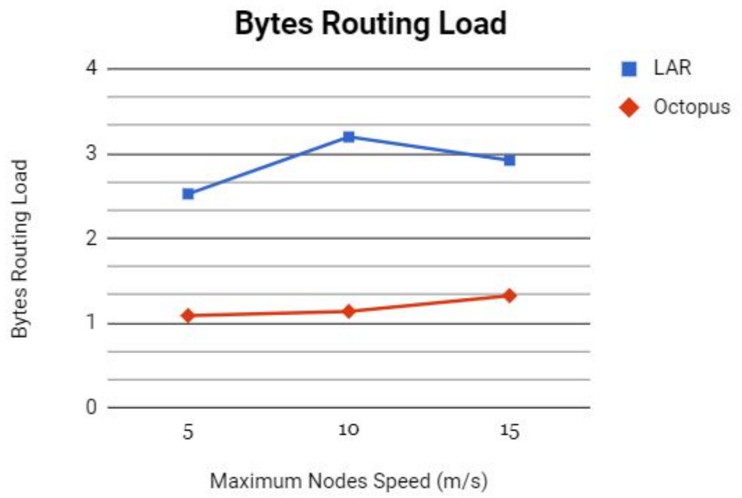

Fig. 5.12: Bytes routing load vs. Maximum nodes speed

The increasing in bytes routing load for both protocols refers to the increasing in nodes mobility speed and the need of more routing packets.

Fig. 5.13 shows that Octopus is more efficient than LAR when considering the packet loss ratio.
The increasing of packet loss ratio for both protocols refers to the high chance to drop the packets when the nodes move with higher mobility speed because of the probability of route broken.

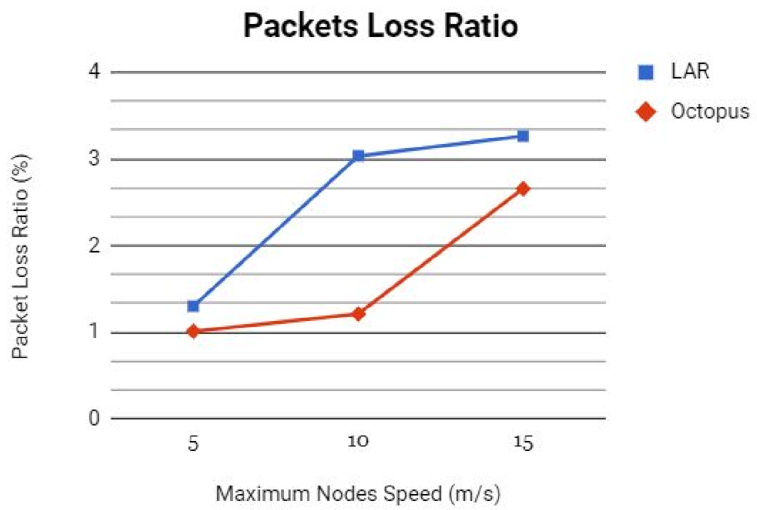

Fig. 5.13: Packet loss ratio vs. Maximum nodes speed

Fig. 5.14 shows that Octopus has less values compared to LAR. The route acquisition latency is increasing because of the increasing in node mobility speed and changing in the positions of nodes so the distance between source and destination maybe longer.

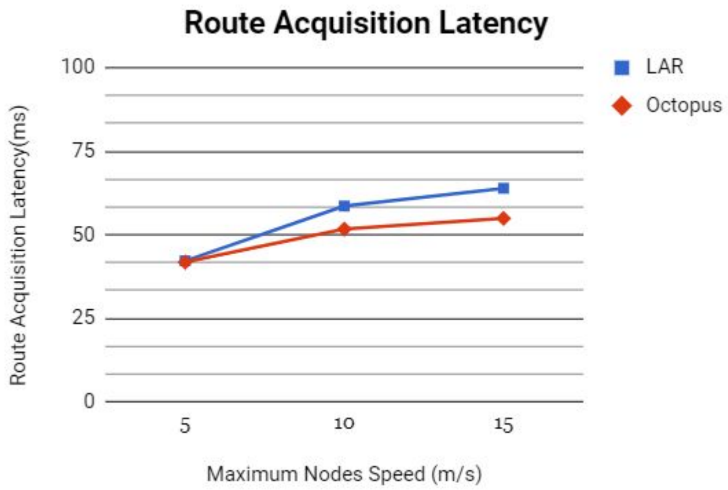

Fig. 5.14: Route acquisition latency vs. Maximum nodes speed

Fig. 5.15 shows that average end to end delay is clearly increasing when the nodes mobility speeds are increasing. That refers to the changing of the positions of nodes so the distance between node may be longer.

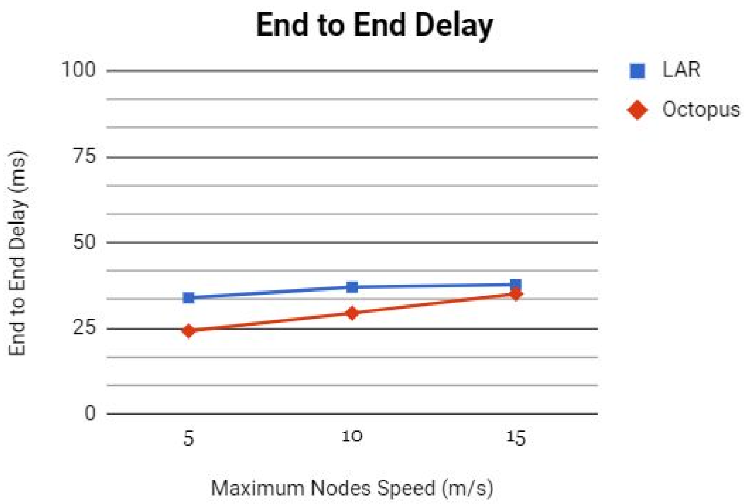

Fig. 5.15: End to end vs. Maximum nodes speed

5.4 Simulation Conclusion

The proposed protocol Octopus and LAR routing protocols were simulated using NS2 simulator with different network scenarios. Their performance was measured when varying the number of network nodes at the same network area,varying the network area and varying the maximum network nodes speed.

In all of the simulated networking scenarios, the proposed protocol outperforms LAR routing protocol when considering the packet routing load, byte routing load, packet loss ratio, and route acquisition latency.

In most of the scenarios, the average end to end delay values of the proposed protocol is less or close to that of the LAR routing protocol.

Chapter 6: Conclusion and Future Work

6.1 Conclusion

In this document, we presented a comparison of a set of location-based routing protocols for MANETs. After that, we proposed Octopus, a scalable position-based routing protocol for MANETs. Octopus uses the restricted directional flooding routing type by dividing the network into 8 sectors and sent the request packets only over the sector that the destination located in, to reduce the number of request packets sent over the network. Octopus reduces the routing packets load, routing bytes load, packet loss ratio, route acquisition latency and end-to-end delay over the network compared to LAR protocol without affecting the scalability of the network.

The simulation of the proposed protocol is done using the NS2 simulator. According to the simulation results, the performance of Octopus is increased compared to LAR routing protocol.

6.2 Future Work

As a future work, a modification can be made on the methodology of the proposed protocol to decrease the packet loss ratio and routing load by improving the route maintenance phase.

After receiving an ERROR packet by the source, it will apply the route maintenance phase mentioned in chapter 4. But if it receives more that one ERROR it will expand the route request zone for route discovery process by adding of the adjoining sectors as shown in fig. 6.1. After that, a new route discovery process will start on the expanded request zone.

Then, the proposed protocol will be simulated after the modifications, and its performance will be compared with the performance of its original version and with other existing protocols in different networking scenarios.

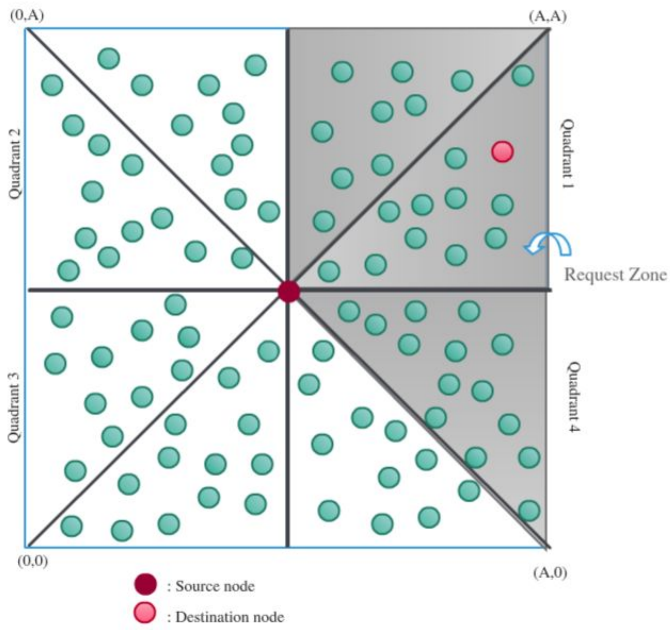

Fig. 6.1: Request zone expansion

Also, as a future work, the number of sectors can be modified and studied. For example, the source can divide the network into 4 or 16 sectors depending on the network density information. The performance of the proposed protocol will be studied on a different number of sectors and compared with each other.

A Simulation Results Tables

Table A.1: Network nodes number variation with area of 800m * 800m for Octopus routing protocol

Packet Routing Load (%)	Bytes Routing Load (%)	Packet Loss Ratio (%)	Route Acquisition Latency (ms)	Average End to End Delay (ms)	Nodes number
0.6393126	0.8741218	2.98616	16.22936	296.535	20
0.6857036	0.8789882	2.65048	27.96884	24.78018	50
1.382556	1.158076	2.757402	39.26638	34.81558	90

Table A.2: Network nodes number variation with area of 800m * 800m for LAR routing protocol

Packet Routing Load (%)	Bytes Routing Load (%)	Packet Loss Ratio (%)	Route Acquisition Latency (ms)	Average End to End Delay (ms)	Nodes number
1.081264	2.66865	4.741476	21.56724	154.7256	20
1.46468	2.785372	4.10432	42.41182	65.05278	50
1.762974	2.809816	3.376134	47.25864	37.66964	90

Table A.3: Network area variation at 90 network node for Octopus routing protocol

Packet Routing Load (%)	Bytes Routing Load (%)	Packet Loss Ratio (%)	Route Acquisition Latency (ms)	Average End to End Delay (ms)	Network Area (m^2)
0.0155	0.3017994	0	18.64056	7.27405	200x200
0.8425884	0.7943556	1.228102	33.49214	26.37626	500x500
1.382556	1.158076	2.757402	39.26638	34.81558	800x800

Table A.4: Network area variation at 90 network node for LAR routing protocol

Packet Routing Load (%)	Bytes Routing Load (%)	Packet Loss Ratio (%)	Route Acquisition Latency (ms)	Average End to End Delay (ms)	Network Area (m^2)
0.06491666	0.5995442	0	19.92924	8.07836	200x200
1.645622	1.99464	1.552522	52.10216	47.26562	500x500
1.762974	2.809816	3.376134	47.25864	47.66964	800x800

Table A.5: Network nodes maximum speed variation at 90 network node and 800m x 800m network area for Octopus routing protocol

Packet Routing Load (%)	Bytes Routing Load (%)	Packet Loss Ratio (%)	Route Acquisition Latency (ms)	Average End to End Delay (ms)	Maximum nodes speed (m/s)
0.8917312	1.089746	1.0128908	24.22776	41.83428	5
0.9299324	1.140098	1.2115402	29.44855	51.7291	10
1.498102	1.325592	2.65834	35.0936	54.95604	15

Table A.6: Network nodes maximum speed variation at 90 network node and 800m x 800m network area for LAR routing protocol

Packet Routing Load (%)	Bytes Routing Load (%)	Packet Loss Ratio (%)	Route Acquisition Latency (ms)	Average End to End Delay (ms)	Maximum nodes speed (m/s)
0.9244768	2.526376	1.301432	42.24978	33.91004	5
1.522078	3.198642	3.036764	58.64866	37.01892	10
1.93572	2.922754	3.264004	63.9649	37.74874	15

References

[1] I. Khalil, S. Bataineh, L. Qabajah, A. Khreishah, “Distributed secure routing protocol for Mobile Ad-Hoc Networks“, *5th International Conference on Computer Science and Information Technology*, Amman, Jordan, 2013.

[2] P. Kaur, Sukhman, “An Overview on MANET- Advantages, Characteristics and Security Attacks“, *International Journal of Computer Applications, 4th International Conference on Advancements in Engineering & Technology ICAET,* 2016.

[3] Y. Ko, N. Vaidya, “Location-Aided Routing (LAR) in Mobile Ad hoc Networks“, *Wireless Networks*, Vol.6, No.04, pp.307-321, 2000.

[4] M. Mamoun, “Location Aided Hybrid Routing Algorithm for MANET”, *International Journal of Engineering & Technology IJET-IJENS*, Vol.11, No.01, pp 30-35, 2011.

[5] L. Raja,S. Baboo, “An Overview of MANET: Applications, Attacks and Challenges”, *International Journal of Computer Science and Mobile Computing*, Vol.3, No.1,pp. 408-417, 2014.

[6] B. Yassein, O. Khaoua, “Applications of Probabilistic Flooding in MANETs”, *Ubiquitous Computing and Communication Journal*, Vol.1 No.1, pp.1–5 ,2007.

[7] A. Gupta, S. Sharma, “A Survey on Location Based Routing Protocols in Mobile Ad-hoc Networks*“,(IJCSIT) International Journal of Computer Science and Information Technologies*, Vol.5, No02, pp. 994-997, 2014.

[8] S. Basagni, I. Chamtac, V. Syrotiuk, B. Woodward, “A Distance Routing Effect Algorithm for Mobility”, *MobiCom '98 Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking*, Dallas, Texas, USA, 1998.

[9] V. Nikam, G. Chavan, “Survey on Location Based Routing Protocols in MANET “, *International Journal of Computer Science and information Technologies (IJCSIT)*, Vol.5, No.02, pp. 1659 -1663, 2014.

[10] N. Wang, J. Chen, Y. Huang, S. Wang, “A Greedy Location-Aided Routing Protocol for Mobile Ad hoc Networks“, *Proceedings of the 8th WSEAS international Conference on Applied Computer and Applied Computational Science*, Hangzhou, China, 2009.

[11] W. Kun, W. Meng,“A Distance-based Location-aided Routing For MANET”, *Journal Of Electronics* ,Vol.26, No.2, pp. 152–160, 2009.

[12] D. Luo, J. Zhou, “An Improved Hybrid Location-Based Routing Protocol for Ad Hoc Networks”, Wireless Communications, *Networking and Mobile Computing (WiCOM) 7th International Conference*, Wuhan, China, 2011.

[13] C. Perkins, E. Roye, “Ad-hoc On-Demand Distance Vector Routing”, *WMCSA '99 Proceedings of the Second IEEE Workshop on Mobile Computer Systems and Applications*, pp. 90, 1999.

[14] V. Giruka, M. Singhal, “Angular Routing Protocol for Mobile Ad Hoc Networks”, *Proceedings IEEE International Conference Distributed Computing Systems Workshops*, Columbus, OH, USA, 2005.

[15] P. Kirdpipat, S. Thipchaksurat,"Location-Based Routing Scheme with Adaptive Request Zone in Mobile Ad Hoc Networks", *IEICE Transactions on Information and Systems*, Vol.E96-D, No12, pp.2565-2574, 2013.

[16] K. Samarasinghe, P. Leone, "Greedy Zone Routing : Scalable Routing in Large Scale Wireless Ad-hoc Networks", *Twelfth Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*,Seattle, USA, 2015.

[17] B. Mahmood, A. Ibrahim, D. Manivannan ,"Hybrid On-demand Greedy Routing Protocol with Backtracking for Mobile Ad-Hoc Networks", *9th IFIP Wireless and Mobile Networking Conference (WMNC),* USA, 2016.

[18] P. Mishra, C. Gandhi, B. Singh, "An Improved Greedy Forwarding Scheme in MANETs", *Journal Of Telecommunications And Information Technology*, No. 01, pp. 50-55, 2017.

[19] A. Hashim, K. Bakar, A. Zainal, "Adaptive Hybrid Geo-casting Routing Protocol for Mobile Ad hoc Networks", *Lecture Notes on Data Engineering and Communications Technologies*, Vol.5, 2017.

[20] A. Kaur and M. Bansal, "A Review on Location Aided Routing Protocol in Mobile Adhoc", *International Journal Of Engineering And Computer Science* ,Vol.4, No.06, pp. 12936-12940, 2015.

[21] P. Rohal, R. Dahiya, P. Dahiya, "Study and Analysis of Throughput, Delay and Packet Delivery Ratio in MANET for Topology Based Routing Protocols (AODV, DSR and DSDV)", *International Journal For Advance Research In Engineering And Technology*, Vol.1, No.02, pp. 54-58, 2013.

[22] C. Bettstetter, G. Resta, P, Santi, "The Node Distribution of the Random Waypoint Mobility Model for Wireless Ad Hoc Networks", *IEEE Transactions on Mobile Computing*, Vol.2, No.03, pp. 257-269,2003.

[23] A. Khana,S. Bilal, M.Othman, "A Performance Comparison of Network Simulators for Wireless Networks ", 2013 , retrieved from https://arxiv.org/abs/1307.4129 on 10/4/2018, 2:07 a.m.

[24] S. Dabibbi, S.Guirguis, "Implementation and Performance Evaluation of Three Routing Protocols for Mobile Ad Hoc Network Using Network Simulator", *Lecture Notes on Software Engineering*, Vol. 3, No.1,pp. 61-66, 2015.