Prifysgol Cymru
Y Drindod Dewi Sant
University of Wales
Trinity Saint David

ANTMANET: A Novel Routing Protocol for Mobile Ad-Hoc Networks Based on Ant Colony Optimisation

MABROUKA SALEM ABUHMIDA

STATEMENT:

This research was undertaken under the auspices of the University of Wales: Trinity Saint David and was submitted in partial fulfilment for the award of a Degree of PhD.

June 2017

Abstract

The core aim of this research is to present "ANTMANET" a novel routing protocol for Mobile Ad-Hoc networks. The proposed protocol aims to reduce the network overhead and delay introduced by node mobility in MANETs. There are two techniques embedded in this protocol, the "Local Zone" technique and the "North Neighbour" Table. They take an advantage of the fact that the nodes can obtain their location information by any means to reduce the network overhead during the route discovery phase and reduced the size of the routing table to guarantee faster convergence.

ANTMANET is a hybrid Ant Colony Optimisation-based (ACO) routing protocol. ACO is a Swarm Intelligence (SI) routing algorithm that is well known for its high-quality performance compared to other distributed routing algorithms such as Link State and Distance Vector.

The following Figure 1 highlights the contribution of this research with regards to the ACO algorithms based routing protocols history.

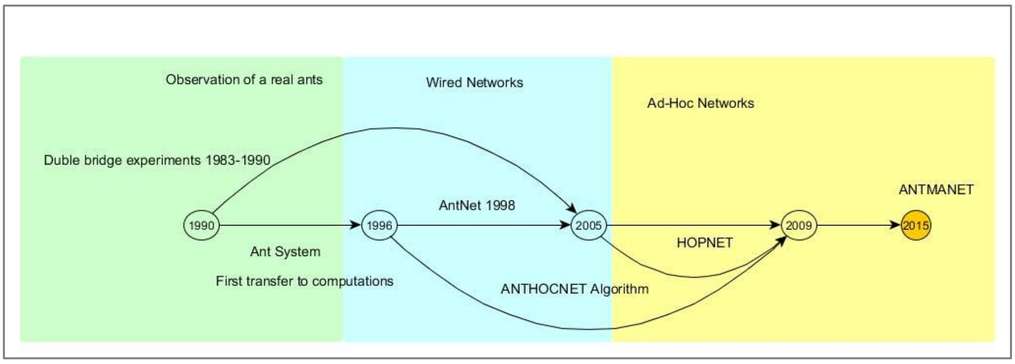


Figure 1: ANTMANET Timeline.

ANTMANET has been benchmarked in various scenarios against the ACO routing protocol ANTHOCNET and several standard routing protocols including the Ad-Hoc On-Demand Distance Vector (AODV), Landmark Ad-Hoc Routing (LANMAR), and Dynamic MANET on Demand (DYMO). Performance metrics such as overhead, end-to-end delay, throughputs and jitter were used to evaluate ANTMANET performance. Experiments were performed using the QualNet simulator.

A benchmark test was conducted to evaluate the performance of an ANTMANET network against an ANTHOCNET network, with both protocols benchmarked against AODV as an established MANET protocol. ANTMANET has demonstrated a notable performance edge when the core algorithm has been optimised using the novel adaptation method that is proposed in this thesis. Based on the simulation results, the proposed protocol has shown 5% less End-to-End delay than ANTHOCNET. In regard to network overhead, the proposed protocol has shown 20% less overhead than ANTHOCNET. In terms of comparative throughputs ANTMANET in its finest performance has delivered 25% more packets than ANTHOCNET.

The overall validation results indicate that the proposed protocol was successful in reducing the network overhead and delay in high and low mobility speeds when compared with the AODV, DMO and LANMAR protocols. ANTMANET achieved at least a 45% less delay than AODV, 60% less delay than DYMO and 55% less delay than LANMAR. In terms of throughputs; ANTMANET in its best performance has delivered 35% more packets than AODV, 40% more than DYMO and 45% more than LANMAR. With respect to the network overhead results, ANTMANET has illustrated 65% less overhead than AODV, 70% less than DYMO and 60 % less than LANMAR. Regarding the Jitter, ANTMANET at its best has shown 60% less jitter than AODV, 55% jitter less than DYMO and 50% less jitter than LANMAR.

Acknowledgement

All thanks to Almighty Allah, most Gracious, most Merciful for giving me the guidance, good health, and the strength to achieve my goal of obtaining my Ph.D.

I owe the greatest gratitude to my supervisors Professor Ian Wells and Dr. Kapilan Radhakrishnan, for their guidance and continuous support during my research. Their devotion to research and a serious attitude toward science has given me great encouragement and inspiration to accomplish this research. I think in the past four years, Professor Wells taught me how to do more than research. He taught me how to make gold out of stones. And Dr. Radhakrishnan has guided me to communicate my research in a better way. I will never forget their patience and their valuable comments during the period of my research. I feel very lucky that I finished my doctoral studies under their supervision.

To the most understanding and supportive person of my life who I dedicate this thesis to, my loving husband Esam Naas, he has been there with me from the start; he has been through all the struggles with me, I appreciate you so much for your limitless love, understanding and supporting, do know you are very special to me, and you have inspired me since day one and you still do. I also want to dedicate this research to my one and only loving son, Salem Naas your existence is my inspiration.

I am also grateful to Dr. T. E. Butt. For his ethical support. As an international student, I felt a certain difficulty in technical writing. Talib contributed much of his valuable time to help me improve my writing skills. He gave advice on how to communicate and co-operate with other researchers as a professional. Therefore, I do want to show my deep appreciation to him. I am also grateful to my colleague Arya

Sedigh for his moral support and valuable advice on this research via emails, talks, and group meetings.

I would like to take this opportunity to express my gratitude to all my friends and colleagues in UWTSD. I will always remember the time that we have spent together, and wish them all success in their lives. I would like to offer my regards to all of those who supported me in any respect during the completion of this project.

Finally, and most importantly, I wish to express my deepest gratitude to my late parents in Libya (Allah bless their souls). Who brought me up to see that failure is only a fact when we give up and whatever I do, I must see it through. I would like to say Thank you to all my family members and friends, who always supported in one way or another.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ACK	Acknowledgment
ACO	Ant Colony Optimisation
AODV	Ad-Hoc on Demand Distance Vector
AODV	Ad-Hoc on Demand Distance Vector Routing Protocol
BS	Base Station
CBR	Constant Bit Rate
CGSR	Cluster Head Gateway Switch Routing
CH	Cluster Head
DSDV	Destination Sequenced Distance Vector
DSR	Dynamic Source Routing
DYMO	dynamic MANET on demand
INI	T Initialization
LANMAR	Landmark Routing
LAR	Location Aided Routing
MANET	Mobile Ad-Hoc Networks
MN	Mobile Node
OLSR	Optimized Link State Routing
RREP	Route Reply
RREQ	Route Request
STAT	State
TC	Topology Control
ZRP	Zone Routing Protocol

Glossary

Term		Abbreviation	Description	Synonyms
Existed	New / Innovative			
Algorithm			A method or a process followed to solve a problem.	
Algorithm analysis			A less formal version of the term asymptotic algorithm analysis.	
Heuristic			A way to solve a problem that is not guaranteed to be optimal. While it might not be guaranteed to be optimal, it is generally expected (by the agent employing the heuristic) to provide a reasonably efficient solution.	
Parameters			The values making up an input to a function.	
Problem			A task to be performed. It is best thought of as a function or a mapping of inputs to outputs.	
Hard problem		HP	"Hard" is traditionally defined in relation to running time, and a "hard" problem is defined to be one whose best-known algorithm requires exponential running time.	
non-polynomial		*NP*	An acronym for non-deterministic polynomial.	
non-polynomial - hard		*NP-H*	A problem that is "as hard as" any other problem in NP. That is, Problem X is NP-hard if any algorithm in NP can be reduced to X in polynomial time.	
Probabilistic data structure			Any data structure that uses probabilistic algorithms to perform its operations. A good example is the skip list.	
Computability			A branch of computer science that deals with the theory of solving problems through computation.	
Cost		C	In graph representations, a synonym for weight.	Weight, Edge weight
Position		P	Defined as the physical space where the object is located at.	Location
Edge		Ed	The connection that links two nodes in a tree, linked list, or graph.	

Term		Abbreviation	Description	Synonyms
Existed	New / Innovative			
Graph		N-G	A graph G= (V, E) consists of a set of vertices V and a set of edges E, such that each edge in E is a connection between a pair of vertices in V.	Network, Network Graph
Path			In network or graph terminology, a sequence of vertices (v1,v2,...,vn) forms a path of length (n−1) if there exist edges from (vi to vi+1) for (1≤i<n).	Route
Routing protocol			A routing protocol is an intelligent set of processes and algorithms and messages that are used to select the best paths to reach network destinations.	
	ANTMANET		A novel MANET routing protocol based on ACO algorithm which has be developed in specifically this research study	
Router			A device responsible for making decisions about which of several paths network (or Internet) traffic will follow.	Router node,
Route Request		RREQ	If destination is not a source node's neighbour, then a broadcast RREQ message is generated.	
Route Reply		RREP	A node receiving RREQ will reply with a RREP containing the path to the destination or otherwise rebroadcast RREQ.	
Routing Table		RT	It is a hash table where all the packet forwarding information are stored in	
Forward Ant		FANT	A special type of RREQ	
Backward Ant		BANT	A special type of RREP	
Pheromone Table		PHT	Special type of RT	
	Local Zone	LZ	Is the zone in which the node is located at	
	North Neighbour Table	NNT	A special type of RT, where all the packet forwarding information of nodes located on the north side of the source node are stored in.	

Term		Abbreviation	Description	Synonyms
Existed	New / Innovative			
Node			Node has programmed or engineered capability to recognize and process or forward transmissions to other nodes.	Device, Mobile node
Physical Layer			The Physical Layer provides the procedures for transferring a single bit across a Physical Media.	
Physical Media			Any means in the physical world for transferring signals between OSI systems.	Message, Signal, Packet

List of Contributed Publications

[1] Abuhmida,M., Radhakrishnan, K., &Wells, I., 2016. Optimising the ACO Meta-Heuristic Probability Using Monte Carlo System, under process.

[2] Abuhmida,M., Radhakrishnan, K., &Wells, I., 2016. A novel routing protocol to address mobility issues on Wireless Sensor Networks, under process.

[3] Abuhmida, M., Radhakrishnan, K., &Wells, I., 2016. Modelling and Evaluating the Performance of A Highly Mobile Wireless Sensor Networks routing protocols, International Journal of Computer Technology and Applications (IJ CTA).

[4] Abuhmida,M., Radhakrishnan, K. & Wells, I., 2015. Evaluating the Performance of ANTMANET Protocol for MANET. In IEEE International Conference on Internet Technologies & Applications, (ITA15). Wrexham, North Wales, UK, United Kingdom.

[5] Abuhmida, M., Radhakrishnan, K., & Wells, I., 2015. Performance Evaluation of Mobile Adhoc Routing Protocols on Wireless Sensor Networks for Environmental Monitoring. In IEEE UKSim-AMSS 17th International Conference on Computer Modelling and Simulation, (UKSim2015). Cambridge, United Kingdom.

[6] Abuhmida, M., Radhakrishnan, K., & Wells, I., 2015. ANTMANET A Routing Protocol for Mobile Ad Hoc Networks. In IEEE 7th Int Conference on Computational Intelligence, Communication Systems and Network, (CICSyN2015). Riga, Latvia.

[7] Abuhmida, M., Wells, I. & X. C. A, 2011. Survey of Wireless Sensor Network Protocols Enabling High Mobility with a Focus on Swarm Intelligence. In IEEE Furth International Conference on Internet Technologies & Applications (ITA 11). Wrexham, North Wales, UK, United Kingdom.

Chapter 1. Introduction

1.1. Background

An Ad-Hoc network is a decentralized network, which requires no infrastructure; recent research has categorized Ad-Hoc networks into several network types such as Mobile Ad-Hoc Networks (MANET). MANET is a highly-devolved technology, which enables users to communicate without any physical infrastructure. Consequently, MANET devices can change locations and reconfigure themselves on the move.

Recently, the focus of research is being set on developing new technologies and routing protocols, which no longer require base stations, fixed routers, or any other infrastructure, this type of the decentralised communication largely widens the operational area of MANET (Li et al. 2012). Shifting the technology from the structured stereotype to be used in areas with little or no communication infrastructure to a more flexible decentralised mobile network.

To move these collected data from one node to another, a routing task has to be performed. The task of routing is defined as the action of forwarding data traffic between pairs of nodes known as “Source” and “Destination” following a set of rules namely “a routing protocol”. The nodes that have the responsibility of performing this action are known as “Routers”. Most existing routing protocols are designed to cater for MANETs specifications, although they are explicitly designed to cope with a low level of mobility.

One feature that routing protocols share with one another is their routing algorithm (Sarikaya 1993). Even though, a handful of protocols for MANET already exist, only a few of them are real can be considered as usable in mobile sensor networks. Those protocols, such as Ad-Hoc On-Demand Distance Vector (AODV) Routing for instance (Chakeres & Belding-Royer 2004), often rely on flooding a route request packet through the network, as soon as a node is willing to transmit data. The flooding is continued until the destination has been reached, an intermediate node knows a valid route to the destination, or until every node in the network has received the request. Upon reaching the destination the node is sending a route reply packet backward the same way the route request came from.

This type of routing is extremely challenging and can be ineffective due to the dynamically moving network, as the nodes constantly change their location, then the network topology changes frequently and so a good route will probably be unavailable after a short time. Consequently, this will cause that each node to update their routing table frequently, triggering the flow of many control packets through the network and so consuming precious network resources.

Over the last few years, self-configuring, self-healing algorithms have been considered as a solution to many large scale multihop MANETs (Elshakankiri & El-darieby 2016). There exist a number of swarm intelligence (SI) based protocols that try to meet these criteria (Giusti et al. 2012). They are based on the behavior of animals that form swarms. Recently, there has been increasing interest in the use of Swarm Intelligence or the naturally inspired algorithms for routing in especially in MANETs. Swarm intelligence is a computational intelligence technique that involves the collective behaviour of autonomous agents that locally interact with each other in distributed environment to find a global solution to a given problem. Ant Colonies, birds flocking, and fish schooling are examples in nature that use swarm intelligence. The similarities of the foraging behaviour of ants and MANTEs has inspired researchers to develop Ant based routing algorithms for MANETs (Mojana et al. 2011) (Karaboga & Akay 2009).

ACO is based on copying the behaviour of ants from the natures (finding the shortest path from food source to nest and vice versa) to computer networks, this is done by modelling artificial ants of the shortest path from the source to the destination. The ants deposit a chemical substance called a *pheromone* that other ants can sense on their journey to the destination. The ants interact with each other and their environment using the *pheromone* concentration. A MANET's environment is unstructured, dynamic and distributed, which is very like the ants' environment. The foraging behaviour of ants and the interaction behaviour of MANETs to deliver packets from source to destination are alike. The goal for both systems is to find the shortest path. ACO has been applied to many combinatorial optimization problems (López-Ibánez et al. 2015; Babaoglu et al. 2006; Babaoglu et al. 2005). In network optimisation problems, ant-based routing has been previously successfully applied to telecommunication networks (Sutariya & Kamboj 2014). Existing ant based routing protocols for Ad-Hoc (Di Caro, Ducatelle & Gambardella 2004a; Di Caro & Dorigo 1998b) are very promising in delivering packets when compared to conventional routing algorithms.

This research applies methods from the SI, specifically, the Ant Colony Optimization Algorithm (ACO) to reduce the network overhead and delay that is enforced by high node mobility and dynamic topologies in MANETs. The proposed protocol "ANTMANET", is a novel routing protocol for mobile MANET, inspired by techniques from previous work "ANTHOCNET" (Di Caro et al. 2005). ANTMANET is a hybrid routing protocol that combines the advantages of both proactive and reactive protocols. Hybrid routing protocols use reactive phase to guarantee more accurate metrics to determine the best paths to destinations and report routing information only when there is a change in the topology of the network. In addition, they use the proactive phase to allow rapid convergence and fresh routing information through the nodes.

ANTMANET has two phases. First is the reactive phase, this phase is divided into the initial stage and the pathfinding. The initial stage is the network initialization process, this process occurs in a very early stage of the network lifetime, where the nodes begin to build their own local topology and each node will create its own unique node structure. The second stage is proactive that is the maintenance phase where all nodes update portions of their routing tables as needed.

1.2. **Research Question**

This thesis proposes a novel ACO based routing protocol for a highly mobile MANET and the *research question is:* would the proposed protocol reduce the network overhead and delay in high mobile MANET network.

There are two main issues that are considered as the main motivation to conduct this research:

- *The first issue* is the rapid node mobility in specific scenarios, where the nodes are moving passively. Passive Node Mobility is uncontrollable node movement that affects many aspects of the network performance, for example, signal transmission rates and channel access, which affects the network overhead and delay. Rapidly moving nodes cause frequent link changes, which will invoke reactive or proactive events in both control packets will be used and the extensive use of them will lead to higher network overhead and end-to-end delay and more likely will result in lower throughputs.
- *The second issue* is the desire to investigate and discover the strengths of using a Hybrid ACO based routing protocol. To the best of our knowledge, most MANET routing protocols are either reactive, where the route is established only when a source node needs to send data to the intended receiver, or proactive, where the routes are established and maintained periodically. the approach this research is using is combining both techniques coupled with the naturally inspired algorithms to overcome the routing issues that is caused by the node mobility.

1.3. **Problem Statement**

In general, classification of a MANET routing methodology can be divided into two main categories: based on network structure or based on the protocol operation. Depending on the network structure, different routing schemes fall into this category. A MANET can be non-hierarchical or flat in the sense that every sensor has the same role and functionality (Alemdar & Ersoy 2010). Therefore, the connections between the nodes are at a short distance to establish radio communication. Alternatively, a MANET can be hierarchical or use a cluster-based hierarchical model, where the network is divided into clusters comprising a number of nodes. The cluster head, which is the master node within each respective cluster, is responsible for routing the information to another cluster head. Routing protocols are a key feature of any network. They enable each node to learn about the other nodes to find a link to their destination. Because some nodes can be moving in MANETS, routes between nodes change very often. Therefore, it is not possible to establish fixed paths and infrastructure between nodes (Farooq & Di Caro 2008). The traditional routing protocols have several shortcomings when applied in fast moving MANETS, such as scalability and control packet overhead problems since each node must keep in its routing table: the routing information of its neighbours to all other nodes in the network or to desired destinations.

Node mobility occurs when nodes can change locations and reconfigure themselves on the move, node mobility is considered as a serious routing challenge because it causes topology changes, which in turn results in triggering routing protocol interrupts that will increase the use of control packets as well as the network delay. Delay measurement relies on network synchronization. Traditional network synchronization introduces an additional overhead that makes the network unreliable due to packet losses caused by the high delay and network overhead.

The delay means that analysis of the collected information is challenging especially in real-time applications.

This research focuses on the routing issues that are caused by high node mobility and which affects the network performance. High mobility is described as a full mobility where both source and destination can freely and randomly move by increasing or decreasing the distance between them; whereas low mobility can be defined as semi-mobility where one of the nodes is stationary. Mobility levels are illustrated in Figure 2, where the shaded are the main mobility conditions that this research is focused on.

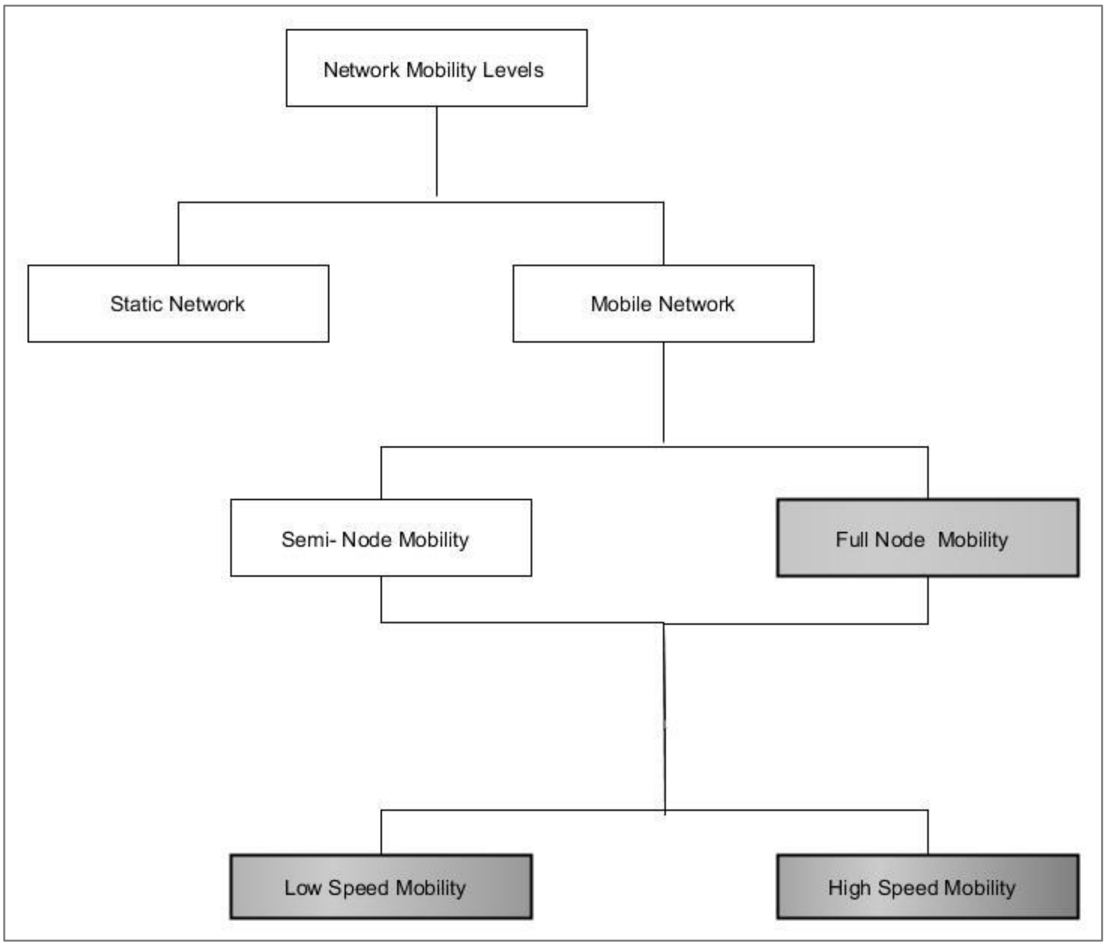


Figure 2: MANET Mobility Levels.

1.4. **Research Aims and Objectives**

The main aim of this research is to design and develop a novel MANET routing protocol based on an Ant Colony Optimisation (ACO) algorithm, which is referred to as "ANTMANET.

The aim of this research is addressed through the following objectives:

1. Review and study existing routing protocols to gain an understanding of issues associated with this field.
2. Survey the routing algorithms that have been implemented in Ad-Hoc routing protocols to identify the related advantages and disadvantages of each algorithm.
3. Review the area of Swarm Intelligence techniques to understand their principles and operations, as applied to the subject of this study.
4. Develop an architectural design for a communication network monitoring system that models the operational scenario for testing the proposed solution.
5. Design and develop the proposed protocol, which involves three tasks:
    a. Create the references (data) needed that represent MANET performance. This objective will be achieved through creating a set simulation based experiments with the support of the QualNet 7.3 software package.
    b. Create header functions and main functions of the proposed protocol.
    c. Implement the proposed protocol in the QualNet 7.3 software package.
6. Analyse, compare, and evaluate the performance of the proposed protocol using QualNet 7.3 software package.

The previously mentioned objectives can be summarised with the help of Figure 3 as follows: the steps of the research, starting by studying and analysing the MANET design space, routing issues and applications. This study has helped to elevate awareness of the routing complications introduced by the high mobility levels of the MANET nodes. This has led to in-depth study of the existing routing algorithms and classifications, which has resulted in forming a conclusion that most of the exciting protocols do not effectively handle high mobility and the performance of these protocols starts to degrade when mobility levels rise. This has expanded the research scope to cover, Swarm Intelligence (SI) routing algorithms as an alternative routing algorithm, where the core of this research is to design and develop a unique MANET routing protocol based on Ant Colony Optimisation (ACO) algorithm.

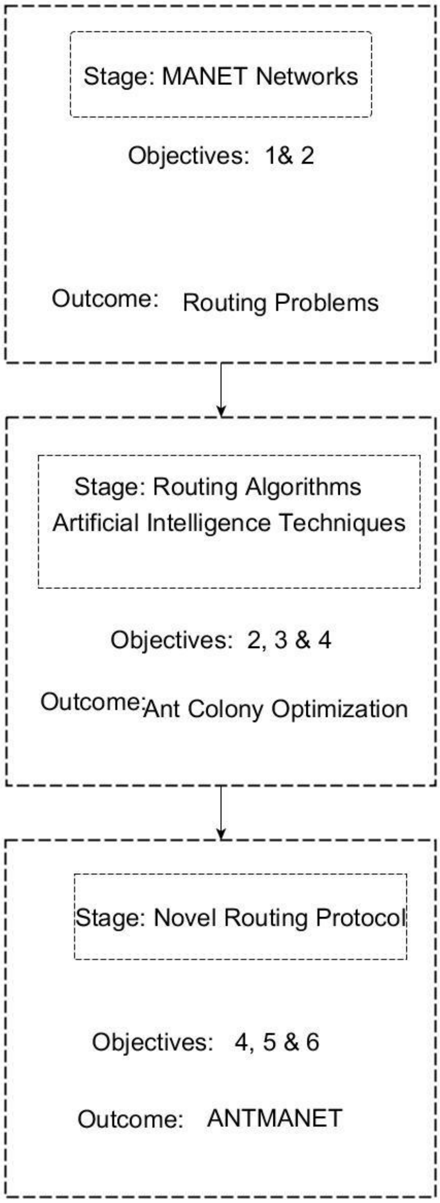


Figure 3: Research Scope.

1.5. **Main Contributions**

The specific contributions of this research presented can be considered from two different points of view. First, the thesis contains contributions that are aimed within the field of computer networking, and secondly, they are also contributing to the field of Swarm intelligence. The following highlights the main contribution of this research.

New routing protocol: From a networking point of view, this research proposes ANTMANET, which is a novel routing protocol for MANETs, based on ideas from Swarm Intelligence. The proposed protocol shows a novel way of combining ACO algorithms in a hybrid design by incorporating ideas from ANTHOCNET with the geo-flat network structure. This thesis presents its design and implementation using the simulation-based prototyping methodology. The protocol utilizes the nodes location information to minimise the search area and to enhance the routing process in MANET. The initial version of ANTMANET is detailed in [Chapter 4].

Evaluation study: The Performance measurements show that the control overhead and the network delay results of ANTMANET are better than the existing MANET protocols, all results are discussed in details in [Chapter 5].

1.6. **Structure of Thesis**

Chapter 2: Has explicitly introduces the definition of an Ad-Hoc network and highlight the growth in demand for Ad-Hoc networks, which has resulted in the creation of new applications and uses.

Chapter 3: Presents of different classifications of Ad-Hoc routing protocols per different criteria. The various classifications give a better overview of the MANET routing protocols

Chapter 4: Discusses in details the structure of the proposed protocol and illustrates all the different components and events.

Chapter 5: Lays the groundwork for meaningful evaluation of a protocol's performance by creating large number of experiments and comparing the ANTMANET to several the standard protocols.

Chapter 6: Compares the proposed protocol with another ACO based protocol by creating several scenarios that differs number of criteria

Chapter 7: Concludes this research with a summary of the experimental results and future work resulting from this study.

Chapter 2. Ad- Hoc Networks

Overview

This chapter discusses the definition of Ad-Hoc networks in general. The main aim is to raise awareness of how much Ad-Hoc networks have improved and developed from their initial use as a classified military tool, to becoming the highly used commercial tool it is today. In addition, it will highlight the growth in demand for Ad-Hoc networks, which has resulted in the creation of new applications and uses. This chapter will also point out how the theory has classified Ad-Hoc networks into different categories, such as Mobile Ad-Hoc networks (MANET).

The remainder of this chapter is organised as follows, 2.1 illustrates details on Mobile Ad-Hoc networks (MANET) followed by detailed examples of the application of MANETs in 2.2. Section 2.3 describes the mobility models of MANET. While, followed by section 2.4 which is summarising the chapter.

2.1. **Introduction**

Today, information systems are based on wireless technology; therefore, demand for unlimited capabilities and flexibility is rising. In addition, there is the increasing need to continuously collect, elaborate, and present data. Such activity requires significant standardization efforts, over different perspectives, to deal with dynamic, open, and not statistically predictable deployment conditions. These demands, merging with recent advances in wireless technology and communication in general, are opening up new services through specific integration opportunities, such as the Ad-Hoc network (Di Caro, Ducatelle, Heegarden, et al. 2004). The English dictionary definition of Ad-Hoc means, “improvised for a specific purpose”. It comes originally from the Latin, Ad-Hoc, which literally means "for this," or "for this special purpose" and, by extension, improvised or impromptu. Ad-Hoc networks are, typically, composed of nodes, which communicate over wireless links without any centralized control. Ad-Hoc nodes are equal in their capabilities so that each device can be, simultaneously, a router and an intermediate node (Di Caro, Ducatelle, Heegarden, et al. 2004). Ad-Hoc Networks can be traced back to the early 70s, specifically to the Defence Advanced Research Projects Agency (DARPA) (Hui 1992), and the Packet Radio Networking (PRNET) project in 1972 (Jubin & Tornow 1987), where technology belonged to the territory of military use only. This technology has played an important role specifically in battlefield monitoring. In the middle of the 90s new advances in commercial radio technology, wireless communication systems and mobile devices spanned several different application domains, ranging from environmental and habitability monitoring (noise, light pollution, animal monitoring, beach profiling), to security controlling (anti-theft protection, structural monitoring to prevent collapses of old buildings and bridges), and also to assist citizenship, urban living and roaming (elderly assistance services, emergency response teams) (Akyildiz & Kasimoglu 2004).

A few years after Ad-Hoc networks emerged for commercial use, researchers categorized the Ad-Hoc network into several types as part of a classification of the technology. Mobile Ad-Hoc networks, as the name implies, are decentralized self-configured networks with different mobility models and patterns. For example:

- Vehicular Ad-Hoc Networks (VANETs), are used for communication between vehicles and roadside equipment. Intelligent vehicular Ad-Hoc networks (InVANETs) are a kind of artificial intelligence that helps vehicles to behave in an intelligent manner during vehicle-to-vehicle collisions and other kinds of accidents (Yousefi et al. 2006).
- Smart Phone Ad-Hoc Networks (SPANs), leverage the existing hardware, primarily Bluetooth and Wi-Fi, in commercially available smartphones, to create peer-to-peer networks without relying on cellular carrier networks, wireless access points, or traditional network infrastructure. SPANs differ from the traditional hub and spoke networks, such as Wi-Fi Direct, in that they support multi-hop relays where there is no notion of a group leader so that peers can join and leave at will without destroying the network (Vandenberghe et al. 2011).
- Wireless sensor networks (WSN), possibly low-size and low-complex devices, are denoted as nodes that can sense the environment and communicate the information gathered from the monitored field through wireless links; the data is forwarded, most usually through multiple hops, via a sink that can use the

data locally, or is connected to other networks (e.g., the Internet) through a gateway (Akyildiz & Vuran 2010).

This research specifically focuses on routing in Mobile Ad-Hoc networks highlighting the significant features and main applications.

2.2. Mobile Ad-Hoc networks (MANET)

A Mobile Ad-Hoc Network (MANET) is an autonomous system of mobile hosts (MHs), which also serves as both a router and intermediate node connected by wireless links. Simply, a MANET is the network that comes together if there is a common medium (usually wireless) to the nodes, and the network formed does not require the support of an existing infrastructure or any other kind of fixed station. Nodes in MANETs have different mobility patterns, starting from the random to the controlled (de Morais Cordeiro & Agrawal 2011). MANETs have several key characteristics that affect the design of the network as well as the performance. According to (Muralishankar & Raj 2014; Romer & Mattern 2004; Yang et al. 2002; Bellavista et al. 2013) these characteristics can be summarized as follows:

- **Dynamic Topologies**: Although MANETs have brought in a lot of advantages to set up new applications, a number of issues still remain to be addressed. The most important issue is the routing of data packets in a MANET. As the nodes enter and leave the network and move randomly following unpredicted patterns at different speeds, each node has its own individual mobility pattern, the topology changes continuously and it becomes very tough to select a forwarding node to route the packets. Selection of an optimal path from the source to the destination also remains a challenge.

- **Energy-consumption Operation**: Ad-Hoc nodes in general and MANET specifically are highly likely to be battery operated. Therefore, the main concern is energy management technique to keep the nodes up and running for longer periods of time. Failing nodes will cause issues, such as path loss and topology change.

- **Limited Bandwidth**: Wireless links continue to have significantly lower capacity than infrastructure networks. In addition, many factors affects the wireless communications, such as fading, noise, and interference conditions, etc., often causes a reduction in the throughput compared to the radio's maximum transmission rate.

- **Security Threats**: MANETs are generally more exposed to physical security breaches than wired networks. The increased possibility of snooping and spoofing attacks makes the design of MANET networks more complex, especially if the application of the designed network is one of the classified profiles.

The absence of a centralized control in MANETs also adds to the above-mentioned issues. Apart from routing, there are some more issues in MANETs that need to be addressed. One of the major challenges is dealing with the limited wireless channel bandwidth. Wireless links are also prone to errors from interference. However, the issues have not limited the use of this technology

Besides the legacy applications that move from a traditional infrastructure environment into the MANET context, a lot of new services is generated for the new environment. This explains the diversity of MANET applications, ranging from small, static networks in large-scale, mobile, and highly dynamic ones. In the following sections, a range of MANET applications is discussed.

2.3. Applications of MANETs

MANETs have gained a great deal of attention over the past few years because of these significant advantages brought about by multi-hop, infrastructure-less transmission. MANETs provide an emerging technology for both civilian and military applications. They are found to be very suitable for military applications and also for emergency communication purposes. Due to this growing demand of MANETs, over these past years, a lot of research has been carried out to move different applications from a traditional infrastructure environment into the MANET context so that a lot of new services will be generated for the new environment. Some examples of the highly demanded applications are as follows:

- **Battlefield:** MANETs are frequently used in military applications in order to maintain an information network between soldiers, vehicles, and military information headquarters. The military can benefit from using this everyday network technology. The basic techniques of MANETs were developed originally for service in the Military field. Nowadays, MANETs are used universally by the world's militaries (Agrawal & Zeng 2015; Bansal et al. 1999). Ongoing developments in swarm drone technology may well require fast moving MANET approaches.
- **Personal Area Networking (PAN):** The wireless standards for this type of MANET are usually IEEE802.16, IEE802.15 or Bluetooth (Agrawal & Zeng 2015; Bansal et al. 1999). PAN is a short-range, localized network where nodes are usually linked with a specific operator. These nodes can be defined as

someone's cell phone, tablet, printer, laptop or smart watch, or they could be small devices used to monitor life signs, such as health monitoring devices. Mobility is a major issue, in terms of designing a network with this kind of application.

- **Crisis-management Applications:** The wireless standards used in this type of application are usually IEEE802.11 or IEEE802.15.4 (Agrawal & Zeng 2015; Bansal et al. 1999). This class of applications has arisen as a result of the occurrence of natural disasters such as tsunamis, hurricanes, where the entire communications infrastructure becomes unavailable, which can lead to a humanitarian disaster. Through using MANETS, an infrastructure is able to be quickly setup, within hours instead of days or weeks. The battery life, bandwidth and, in some areas, mobility pattern are the main concerns.
- **Environmental Monitoring applications:** this class of applications is used mostly for research studies to understand some environmental events. This requires the knowledge of the specific locations. In comparison with geographical positioning systems, MANETS can support the built-in geographical location by using an extremely accurate form of triangulation. This feature means that MANETs readings can be faster than the geographical positioning systems because to forward information there is no need to wait for multiple satellites to acquire a centralized security.

The main factor or characteristic of MANET networks is node mobility whereas all nodes can dynamically and arbitrarily be in such a manner that the interconnections between nodes are changing on a continual basis. This mobility causes fast variations of their availability. At one time the node is in range and while at other that node is out of the range. Consequently, more routing must be done to deliver data that will me higher network overhead resulting in higher delays and in turn lower throughputs. Therefore, focusing the possible mobility models of MANETs is very important to understand the specific requirements and issues that comes with each type. The following section discuss the MANET mobility models.

2.4. **Mobility models in MANET**

Mobility models are a way of describing the real motions of objects to help evaluate the network or the protocols performance in certain scenarios (Divecha et al. 2007). Mobility models are considered a major concern, as the demand for unrestricted mobility patterns evolves to fulfill modern requirements and the design space of MANETs.

Mobility can be classified considering the following aspects: the element that is mobile and the type of movement- both are a concern the physical aspects of mobility. The following sub-sections discuss in details each of them.

2.4.1 Mobile element

This element describes what in the network is moving, there are two types of elements or nodes in MANET networks:

1. Sink node- Special nodes where data collected (sometimes, already aggregated data) is sent.
2. End node- peripheral unit in a network that has a sensor integrated on board and collects desired data from surroundings.

Table 1 summarizes the mobility characterization. As it can be seen in the table, two cases can occur mobility of the sink node, and mobility of the end node.

Table 1: Mobile Element

Sink node	Mobile Base Stations (MSB)
	Mobile Data Collectors (MDC)
	Rendezvous (Hybrid)
End node	Weak
	Strong robotic
	Strong parasitic

Sink node mobility was introduced in (Guo 2012), among others, with the objective of making sink nodes closer to each sensor node or sensor node cluster, in order to save the nodes' energy. A second objective was to avoid the high cost of maintaining long multi-hop paths.

Two classes of sink node mobility exist:

- Mobile Base Stations (MBS),
- Mobile Data Collectors (MDC).

With Mobile Base Stations the sink node is capable of moving across the network, increasing the coverage and decreasing the number of hops to reach each node. (Silva et al. 2014) evaluates sink node mobility performance for various network topologies and types of movement.

Mobile Data Collectors (MDC), in turn, take advantage of the capability of more powerful nodes (either sink nodes or other dedicated nodes) to perform on-demand collection, avoiding the need for data to travel through several hops. (Shah et al. 2003) introduced the concept of data mules, where mobile sink nodes move randomly, collecting data across the network. (Shanmugam et al. 2015) proposed a solution where the trajectory of the Mobile Data Collector is not controlled but is known a priori, while (Ghassemian & Aghvami 2008) proposed a controlled MDC in real-time.

End node mobility can be classified into two basic modes (Silva et al. 2014): weak mobility and strong mobility.

- Weak mobility is the mobility forced by the death of some network nodes. Due to their intrinsic characteristics, namely hardware restrictions and battery operation, nodes have limited, often short lifetime. Consequently, new nodes must be added to replace dead nodes, thus leading to network topology changes.
- Strong mobility, in turn, is the type of mobility associated with the movement caused by either an external agent (wind or water) or by an intrinsic characteristic of the sensor node. Strong mobility can be further subdivided into robotic and parasitic. In the former case, the sensor node has the capacity to move on its own. In the latter case, it is attached to a moving entity.

An example of robotic node mobility is Robomote (Le et al. 2013), a wheel-equipped sensor node designed for easy deployment and low cost. Robomote was also equipped with two engines, one infrared sensor to detect obstacles and a sun-rechargeable battery. Despite the interest in and potential of Robomote, most existing applications are based on nodes attached to mobile bodies, i.e., on parasitic sensor node mobility. In (Silva et al. 2014) an issue is analyzed in depth, using various types of parasitism to classify the possible forms of association between motes and mobile bodies.

Most of the previous work in this area focused on the node speed or the pause time when using established mobility models such as the Random Waypoint (Santi 2012). The Random Waypoint is a random model for the movement of mobile devices, representing a change in their location, velocity and acceleration over time (Bettstetter et al. 2003). This model was introduced for the first time by Johnson and Maltz (Johnson & Maltz 1996). The mobile nodes move randomly and freely without restrictions. More specifically, the destination, speed, and direction are all chosen randomly and independently of other nodes. It is one of the most widely used mobility models for evaluating MANET routing protocols, because of its simplicity and wide availability.

2.4.2 Types of movement

Mobility in MANETs can also be classified according to the type of movement of the moving entity. There are two variants of the random waypoint: the random walk model and the random direction model. Both models are very similar. Some literature describes the Random Walk model as the specific Random Waypoint model with zero pause time. Both models are based on emulating the unpredictable movement of particles in physics. It is also referred to as *Brownian motion*. Node movements have strong unpredictability and randomness in both models.

It has been noted that if nodes move with different speeds or pause times but in the same direction this might not cause topology change (Akyildiz & Kasimoglu 2004). On the other hand, if nodes travel at the same speed, or at least have similar pause times, but move in different directions, this will most likely cause topology change, as is the case, for example, when operating a scenario with the intention of monitoring the surface of a body of water when a wave is disturbed by winds or an object is thrown into the water. In this case, it will not be suitable to describe the mobility model using the random waypoint model. Instead, a Fluid mobility model would better to demonstrate motion, as the nodes are moving passively, derived by the water kinematic waves that are generated in different directions and speeds. This motion could be described by an unsteady kinematic wave equation (Singh 1997).

2.5. **Summary**

This chapter presented a detailed discussion on Ad-Hoc networks, their design space, types, classifications, and applications. Also, numbers of comparative studies were presented, to provide a better understanding of the issues that come along with mobility, both in terms of their hardware and software.

In the following chapter, a discussion of the existing routing techniques and algorithms of Ad-Hoc networks are carried out, the aim of which is to acquaint the reader with some of the important core techniques behind the routing protocols, as well as their advantages and disadvantages, so that the choices made for the work presented in this thesis can be better understood.

Chapter 3. MANET Routing Protocols and Algorithms

Overview

The main contribution herein is the presentation of different classifications of Ad-Hoc routing protocols per different criteria. The various classifications give a better overview of the MANET routing protocols. This chapter highlights an overview of the existing routing protocols, as these classifications are more beneficial than a lengthy listing of previous routing protocols alongside the updated ones.

The remainder of this chapter is systematised as follows, 3.1 illustrates details on the classification of the Mobile Ad-Hoc networks (MANET) routing protocols, followed by detailed discussion of the Distance vector routing algorithm in 3.2. In section 3.3 detailed discussion of the Link state routing algorithm. Section 3.4 discusses alternative routing algorithms for MANET, followed by an in-depth discussion of the Ant colony optimisation routing algorithm in section 3.5. Followed by chapter summary in section 3.6

3.1. Introduction

In the computer networks field, routing refers to the process of moving data packets from Source node to Destination. Routing is a key feature of any network since it is not only about exchanging packets, rather it reaches beyond that to the exchange of important network information, such as battery level, link quality and nodes location. Each intermediary device collaborates to deliver packets to the next device across the optimal path. Part of this process involves advertising a routing table, which is a set of rules often viewed in the form of a table used to determine the length of the route, which can be the one-hop route, two-hop, three-hop, etc. Different strategies and methods are followed to construct and advertise routing information in a network by the means of a routing algorithm, which in turn, defines the metrics used for evaluating the routes' quality, in terms of its reliability and length (distance).

In the early stages of MANET network development, Distance Vector and Link State were the first two routing algorithms designed to route network traffic. However, the network dynamics, limited bandwidth, and power constraints, in addition to the emerging applications of MANETs, added more complexity to the routing task. This has resulted in the development of a new type of self-configured routing algorithm as a solution to the large-scale dynamic networks. For instance, the Swarm Intelligence algorithms (SI), which are inspired by the collective behaviour of a group of creatures such as schools of fish, bird flocks, honey bee colonies and ant colonies, are considered amongst these intelligent solutions (Babaoglu et al. 2006). They exhibit routing behaviour through using the complex interactions of autonomous swarm members. This thesis is inspired by the Ant Colony Optimization algorithm (ACO), which is a routing algorithm, based on the accumulation of knowledge from the experimental observation of the foraging behaviour of ants. Ant colonies can find the shortest path from the nest to the food source and vice versa. These randomly wandering ants communicate with each other using a chemical substance known as a pheromone. It is a hormone that can be sensed by ants as they travel along other ants' trails. Therefore, they tend to follow the strongest pheromone trails (Bonabeau et al. 2000; Ducatelle et al. 2010; Dorigo et al. 2000).

3.2. **MANET Routing Protocols Classification**

A MANET is a dynamic network, due to the nodes' mobility. Therefore, efficient packet routing is considered to be a challenging problem (Abdullah & Ehsan n.d.; Abuhmida et al. 2015). The objective of routing is to relay packets from source to a destination by means of a routing algorithm. Thus, there are many MANET protocols, which have been proposed in the literature, which are categorised and classified in order to analyse, compare, and evaluate their performance which will aid researchers in designing new protocols.

Classification of routing protocols in MANET's can be done in many ways, but most of these are done depending on routing strategy and network structure. Per the design space, the routing protocols can be categorized as, Reactive, proactive and hybrid routing, while depending on the network structure these are classified as Flat and Hierarchical Routing and Forwarding paradigms, which are Unicast, multicast and broadcast initiated protocols come under the Forwarding Paradigms taxonomy as shown

in Figure 4.

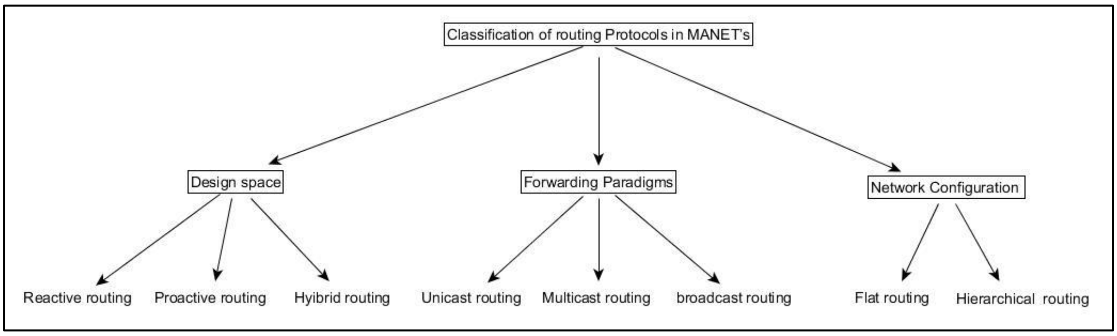


Figure 4: Classification of routing Protocols in MANET's

Furthermore, this classification can assist developers to have an appreciation of these protocols' characteristics and to find the relations between them. However, the routing protocols cannot be included under one category or one classification (Saeed

Abbod, & Al-Raweshidy, 2012). The known characteristics of the standard MANET routing protocols are classified per attributes related to their algorithm and forwarding Paradigms.

The focus of this next section is to present various routing protocol classifications that depend on either design space, network configuration, or on the routing algorithm characteristic, such as packet casting.

3.2.1 Design space

Every routing protocol has a routing taxonomy, which has its own properties are associated with an algorithm. The protocol philosophy will usually depend on the network capabilities and structure. For instance, proactive routing preserves fresh lists of destinations and their routes by periodically distributing routing tables throughout the network; this guarantees respective fresh routing information (Agrawal & Zeng 2015). However, it is known to have slower convergence time. In another example, the reactive routing finds routes on demand by flooding the network with Route Request packets (RREQ), which guarantees better and faster routing. However, it is known to have higher overhead. There is also hybrid routing, which combines the advantages of proactive and reactive philosophies. Hybrid routing is initially established with some proactively explored routes, and then serves the demand from additional routes through the reactive flooding method. The choice of one method or the other requires predetermination for case studies of the application of the network (Raju & Murthy 2015).

Protocols in this category differ in terms of the number of routing tables and the update methods. The following Figure 5 highlights the main three routing philosophies:

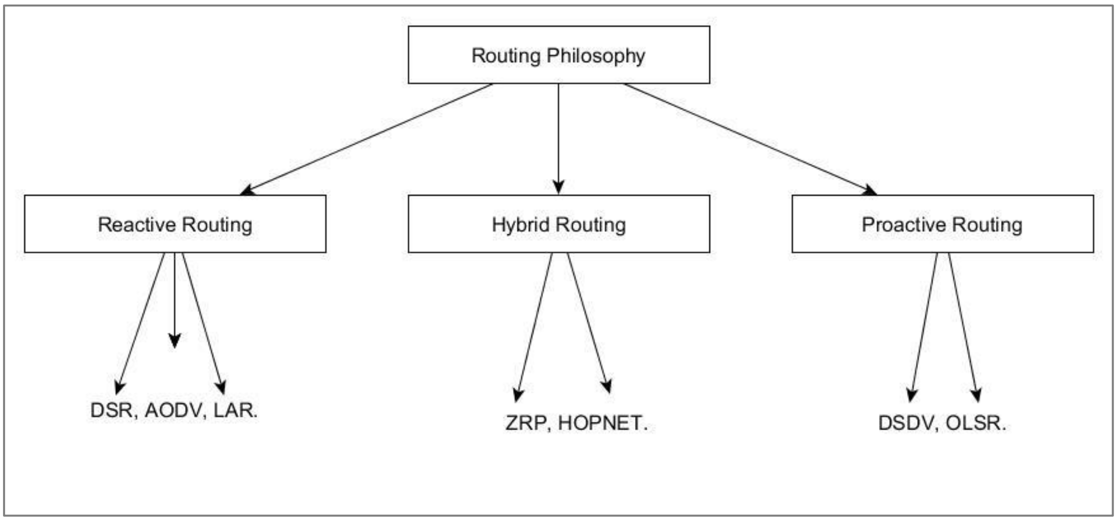


Figure 5: Routing Philosophy.

- Proactive routing: Proactive routing is also known as a table-driven protocol, in which the route to all the nodes is maintained in the form of a routing table. All packets are forwarded over the predefined routes specified in the routing table. With this approach, the packet forwarding is done faster, which results in lower network delay, as all routes are immediately available after the route setup phase is done (Raju & Murthy 2015). However, this approach is known to have higher routing overhead because of routing table updates due to node mobility. The expectation of the control overhead is proportional to the network size and level of mobility. Example protocols: optimised Link State Routing (OLSR) (Clausen et al. 2003), destination sequenced distance vector routing protocol (DSDV) (Rahman & Zukarnain 2009).

Reactive Routing

- Reactive routing: is also known as On-demand routing; these protocols find paths to a destination only when needed to transmit a packet. A source node will initiate a route discovery phase whenever a route is needed. This route discovery mechanism is based on the routing algorithm which employs different forwarding techniques, such as broadcasting control packets to all of the neighbours (Pandey 2015). This technique will be repeated until the route to the destination is found. Reactive routing is known to have smaller routing overheads because there is no need to update a route, due to the node mobility. However, it has a higher latency and it does not scale well. Example protocols: Dynamic Source Routing protocol (DSR) (Arora & Rizvi n.d.), Ad-Hoc On-Demand Distance Vector routing protocol (AODV) (Chakeres & Belding-Royer 2004), and Location-aided routing (LAR) (Ko & Vaidya 2000).

- Hybrid Routing: Hybrid routing algorithms combine the two previous techniques (the proactive and the reactive) in an attempt to bring together the advantages of the two approaches. As such, a hierarchical architecture is utilised in that these algorithms require an addressing system wherein the proactive and the reactive routing approaches are implemented at different hierarchical levels. Such algorithms are designed to increase scalability by allowing the nodes closest to each other to connect and form a number of groups, then, then assigning the group nodes different functionalities, both inside and outside the group, to reduce the route discovery overhead (Roberts & Das 2013). Example protocols: Zone Routing Protocol (ZRP) (Subramaniam 2003), HOPNET (Wang et al. 2009).

Table 2 compares the three routing philosophies in terms of the network configuration, route availability, mobility, etc.

Table 2: Routing Philosophies Comparison.

Parameters	Proactive	Reactive (on-demand)	Hybrid
Network Configuration	Flat and Hierarchical	Flat	Hierarchical
Route Availability	Always route is available	Determine on-demand	Depends on location of destination.
Network Mobility	Low	High	Very high.
Control Traffic	High	Low	Lower than other two types.
Periodic Message	Required	Not required	Sometimes used inside each zone.
Routing Information	Stored in routing tables.	Does not stored	If requirement is there then provided.
Delay	Low	High	Low (in Intra-zone) and High (in Inter-zone)
Benefit	Rapid establishment of routes and routing information is updated periodically.	Obtains required route when needed. Does not exchange routing table periodically and loop free.	Updated routing information, limited search cost & more Scalable.
Drawback	Convergence time is low, resource amount is used heavily, routing information flooded in whole network.	Routes are not up-to-date, large delay, more packets dropping.	Required more resources for larger size zones.

3.2.2 Network Configuration

This section will highlight the significant role the underlying network structure can play in the operation of routing in MANET. The network structure affects the routing algorithm choice, as it is considered as a function of the network level of mobility and scalability. There are several network structures based categories, but this thesis focuses on the Flat and Hierarchical Routing Structure as routing in MANET is mainly based on either, as shown in Figure 6.

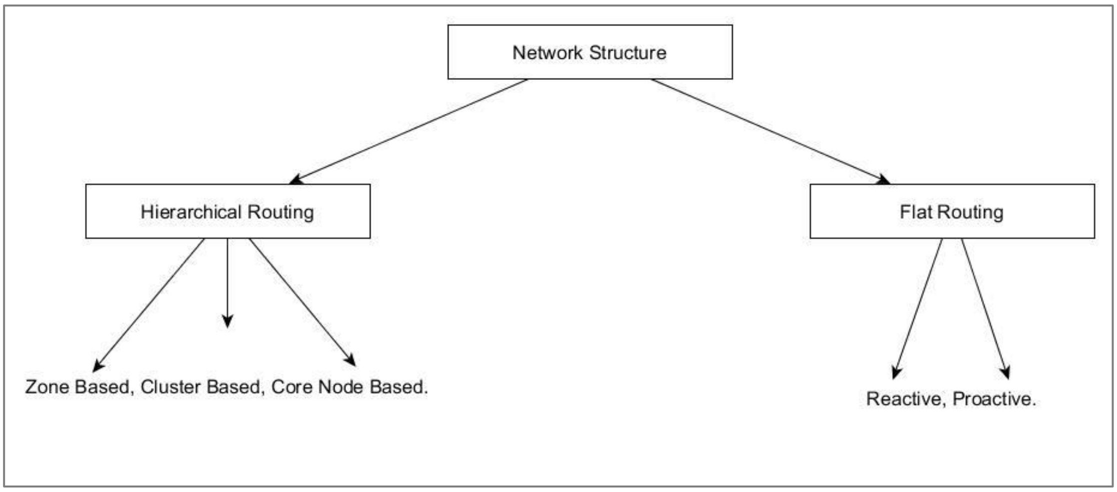


Figure 6: Network Structure.

- Flat Routing: The first category of routing protocol under consideration is the multi-hop flat routing configuration, also known as Fabric Routing. In flat networks, all nodes usually play an equal role; they collaborate together to perform different tasks (Jamatia et al. 2015). Flat routing mostly leads to a data-centric network that utilises a base station node (BS) to send requests to other BS in different regions so as to perform comprehensive routing. Early works on data-centric routing, for instance, the Sensor Protocols for Information via Negotiation (SPIN) routing protocols family and directed diffusion (DD) (Karl & Willig 2006) were shown to enhance the energy consumption. These two protocols motivated the design of many other protocols, which followed a similar design philosophy.
- Hierarchical Routing: Hierarchical routing is also known non-uniform routing. In contrast to flat routing, hierarchical routing usually assigns different roles to each network node. Non-uniform routing approaches are related to hierarchical network structures to facilitate node organisation and management, in other words, hierarchical network structures divide the network into a number of

regions, where each region works as sub-network (Heinzelman et al. 2000). Each region has a number of base station nodes (BS) normally used to organise large-sized networks. This scheme affects the routing in many ways; for example, reactive protocols are exploited to select the BS nodes, which carry out reactive routing functions.

- Non-uniform hierarchical routing protocols can be further sorted into three subcategories: cluster-based, zone-based and core-based. These protocols are categorised according to the organisation of the mobile nodes, their respective management, and their routing functions (Jamatia et al. 2015).
  - Cluster-based: In a clustering scheme, the nodes in a MANET are aggregated into different virtual groups, known as Clusters. A typical cluster will have at least one cluster head, as shown in Figure 7. The hierarchical cluster scheme will reduce the size of the routing table, which results in faster convergence. However, some clustering schemes may cause the cluster structure to be completely rebuilt over the whole network when some local events take place, e.g. the movement or node failure, resulting in re-clustering, which will increase the network overhead (Kuila & Jana 2012).

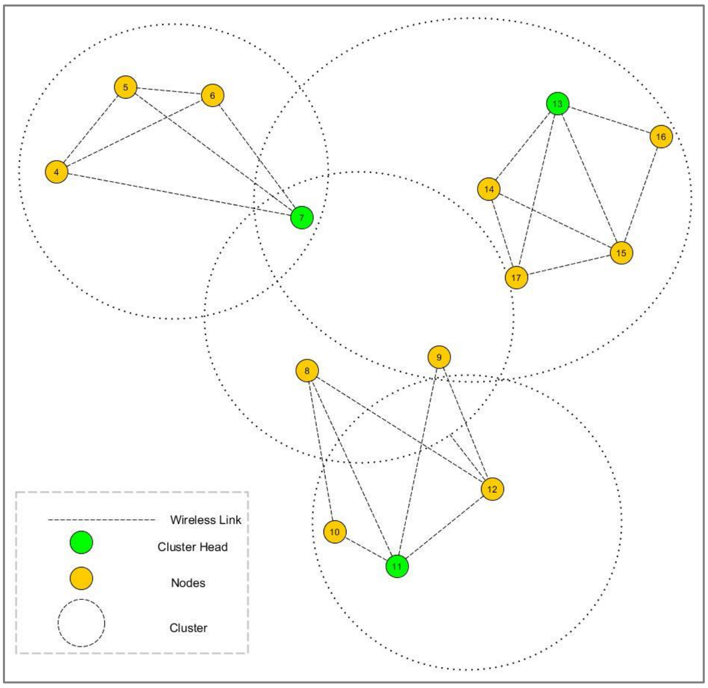


Figure 7: Cluster Structure.

- Zone-based: In a Hierarchical Zone-based protocol (HZB), the network is divided into non-overlapping zones. Unlike the Cluster scheme, there is no zone-head. ZHB expresses two levels of topology: node level and zone level. A node level topology defines how nodes of the same zone are connected to each other, so each node knows information about other nodes only in its zone. Zone level topology defines how zones are connected, so zone information is propagated globally to other zones. Consequently, each node will have full node connectivity knowledge about the nodes in its zone and only zone connectivity information about other zones in the network. So given the zone ID and the node ID of a destination, the packet is routed based on the zone ID until it reaches the correct zone(Husain & Sharma

2015). Then, once in that zone, it is routed, based on node ID to the destination, as shown in Figure 8. Unlike the Cluster scheme, the Zone based protocol is known to have a small routing table, which guarantees faster convergence time and it is changed adaptively.

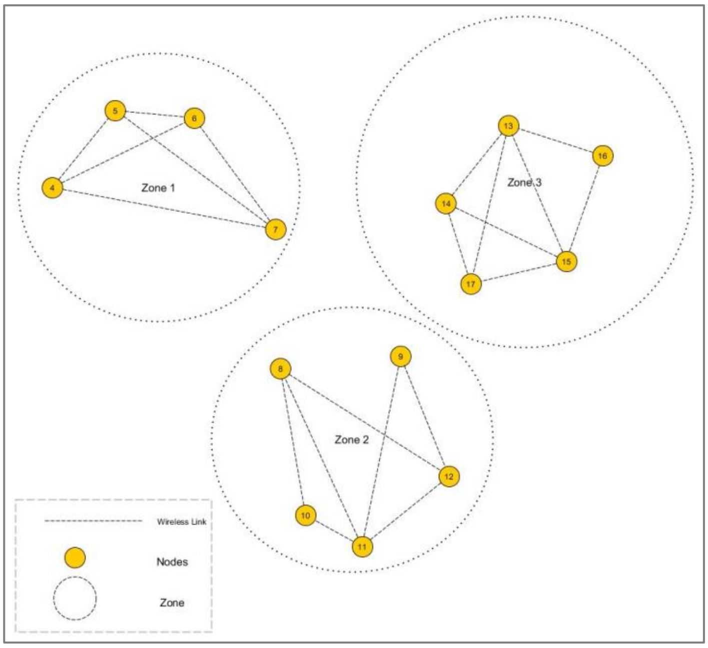


Figure 8: Zone-Based Structure.

- Core-node based: The Core-node based scheme is also referred to as the network "backbone". In this scheme, the network is divided into three layers, as in Figure 9: The Core layer consists of high-speed devices that represent the "backbone" nodes, which switch packets as fast as possible to aggregate routing information. The Core-node based method provides fast convergence time and lower network overhead (Vodnala et al. 2015). The Distribution layer aggregates the data received from the access layer switches before it is transmitted to the core layer for routing to its destination. Finally, the Access layer grants end devices access to the network (Vodnala et al. 2015).

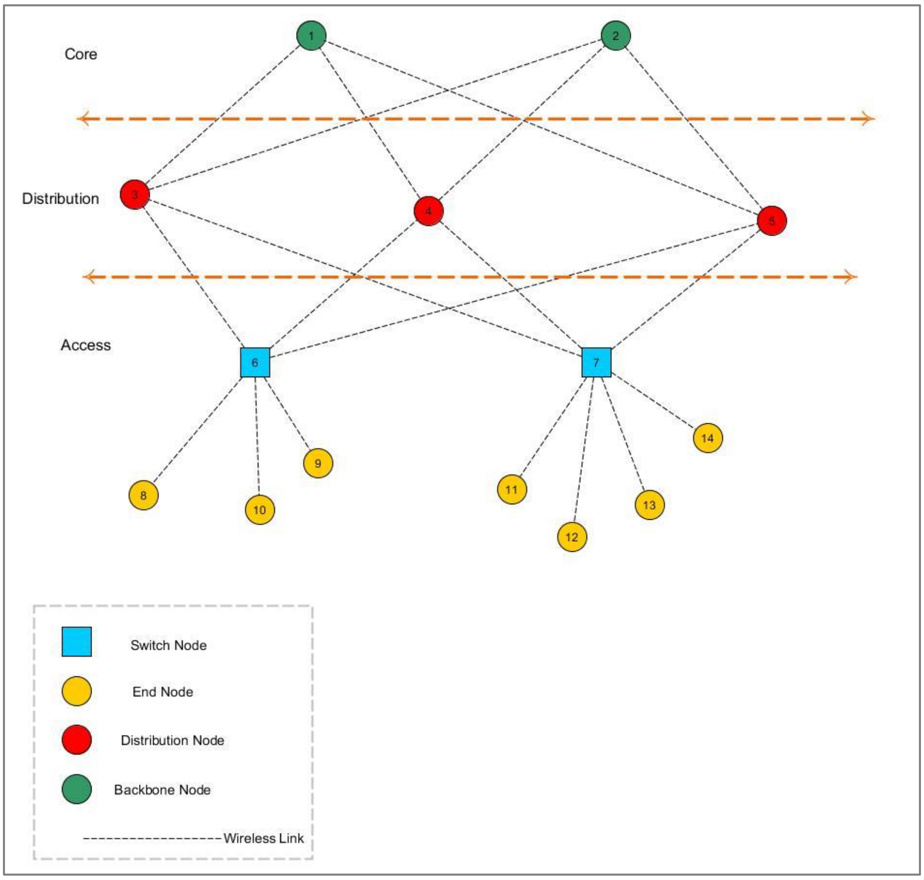


Figure 9: Core-Node based Structure.

3.2.3 Forwarding Paradigms

Casting control and/or data packets method is an important design factor that must be considered when designing routing protocols for MANET networks. This section describes the following main casting categories:

- Unicast: Source will send messages to a single destination. With the unicast method, the source node will forward packets to a specific destination, there are simply two devices involved in this communication at the time, and generally, it is something that is used to establish private sessions that will enable the exchange of private information, or else it is intended to go to one destination. This method does not scale very well for large-sized networks since each node can only communicate with a specific destination; these require a higher bandwidth to send information as efficiently as possible (Chun & Tang 2006).
- Multicast: Source will send same messages to several destinations. Multicasting in MANET is defined as the transmission of packets to a group of hosts identified by a single destination address. The main advantage of multicasting is to reduce the number of transmitting and forwarding packets. When multicast packets are generated by applications, each node handles them. In Figure 10, node 1 sends the packet to the multicast group; as node 2 is not a member of the group, it only relays the packet to the multicast group address (Dou et al. 2014).

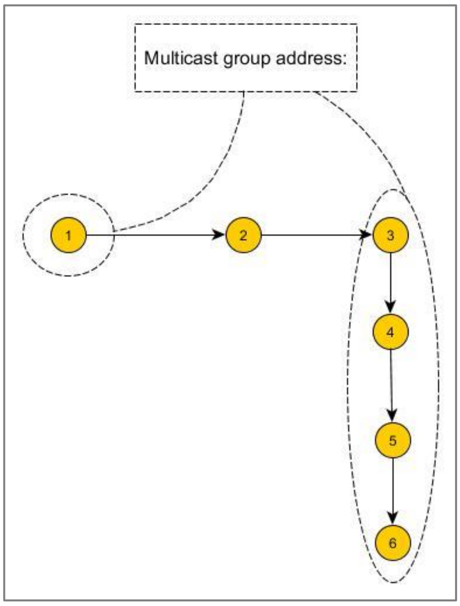


Figure 10: Multicasting.

- The existing MANET multicast routing approaches can be sub-classified into tree-based, mesh-based, core-based, and group forwarding-based multicast routing protocols (Chun & Tang 2006). This sub-classification is based on how the distribution paths among group members are constructed. Some of the multicast routing protocols could be included in more than one category, such as the Core-assisted Mesh Protocol (CAMP) (Farooq & Tapus 2014) which can be characterised as both a core and mesh multicast routing protocol.
- Broadcast: Source will send same messages to all possible destinations. Broadcasting in MANET is defined as the transmission of packets to all neighbouring nodes, each node examining whether it needs the received packet or not. Due to the limited signal range and bandwidth of MANETs, this mechanism alone is not effective enough to perform routing, but it can be used along with Multicast or unicast to aggregate routing information (Vecchio & López-Valcarce 2015). The broadcasting mechanism is most likely to be used in route maintenance by broadcasting a periodic control message. In the

literature, there are a number of proposed efficient broadcast protocols based on distributed and hierarchical methodologies, which can be subclassified according to their transmission methodology. For instance, with the probability-based method, the node decides whether to rebroadcast according to a specified probability or a simple conditional event which relates to the probability of reaching additional neighbours (Kim et al. 2014). Another example is the area based method; (Dou et al. 2014) which uses knowledge of sender node locations to estimate whether a transmission will reach a significant amount of additional coverage area, LAR includes an area based method to cast control packets (Ko & Vaidya 2000).

The following section highlights the main functionality of several the standard MANET protocols.

3.3. Standard MANET Protocols and Algorithms

Routing protocols have many properties to categorise them. Characteristics such as the speed with which they operate, the way they conduct updates, and the information they gather to perform their job make routing protocols unique. While many different routing protocols are available for use in MANETs, they all utilise one of only two different algorithms- link state and distance vector. The following sections discuss in details these two routing algorithms.

3.3.1 Distance Vector Algorithm

A Distance Vector routing algorithm (DVA) is the first routing algorithm class that uses the Bellman-Ford algorithm, named after the first author who contributed to this algorithm (Royer & Toh 1999). DVA works on finding the shortest path from the source node to the destination; which means distributed route computation using the neighbours' information. Each node knows its own address and the cost to reach each of its directly connected neighbours.

Before the first exchange, each node maintains a vector of distances containing exclusively its neighbours, rather than the entire topology. Each node advertises or exchanges its Distance Vector table (DV) to all the nearby destinations; updating the vectors of all destinations (Muralishankar & Raj 2014). For example, in the simple network shown in Figure 11, node (A) has no information about the rest of the network, therefore the cost to the all other nodes is set to infinity, and obviously, the cost to itself is set to zero as highlighted in Table (4).

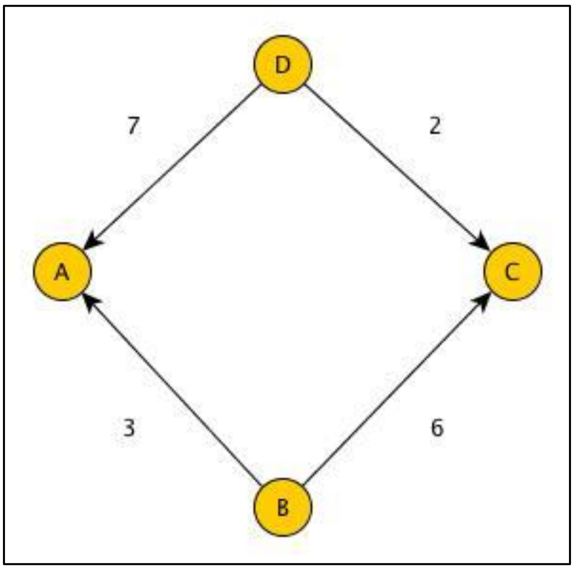


Figure 11: Distance Vector Network.

The algorithm works to define distances at each node and updates distances based on neighbours.

Table 3: Node A's Vector Table.

Destination node	Cost
A	0
B	∞
C	∞
D	∞

All nodes will send periodically updated copies of their DV table to all nearby neighbours; this will help them to add new links and discover new routes (Perkins & Royer 1999). Accordingly, the shortest route will be chosen to forward data and routing information. Referring to Figure 11, node (A) can only communicate directly with the nodes (B, D), and there is no direct link between (A, C).

During the first exchange, node (A) will learn one hop route through nodes (B+3, D+7) as (3, 7) are the cost of the links shown in the previous Figure. The DV table of the node (A) will be updated as shown in Table 4.

Table 4: First Exchange Node A's DV.

Next Hop	Cost
A	0
B	3
----------	∞
D	7

Now node (A) needs to learn the shortest route to the node (C); to accomplish this, node (A) needs to examine the DV tables advertised be its one-hop neighbours; in this case, nodes (B, D). The following table will show the routing tables of both nodes. Both intermediate nodes know how to get to (C), but node (A) will choose the shortest path, which will be calculated as follows:

- For the route through node (B): (B+3) + (C+6) = 9.
- For the route through node (D): (D+7) + (C+2) = 9.

In this case, both routes behave the same value; therefore, node (A) will take the most convenient one, which is the route through node (B).

At the end of the first exchange, all nodes will learn new routes to the rest of the network, as highlighted in Table 5.

Table 5: DV Table

Node (B) Distance Vector table.		Node (D) Distance Vector table.	
Destination node	Cost	Destination node	Cost
A	3	A	7
B	0	B	3
C	6	C	2
D	3	D	0

The routing tables advertised by all nodes by the end of the first exchange are as follows in Table 6.

Table 6: Routing Table for all nodes

Node (A) Distance Vector.		**Node (B) Distance Vector.**		**Node (C) Distance Vector.**		**Node (D) Distance Vector.**	
Destination node	Cost	Destination node	Cost	Destination node	Cost	Destination node	Cost
-----	0	A	3	A	∞	A	7
B	3	------	0	B	3	B	3
C	9	C	6	-----	0	C	2
D	7	D	3	D	2	-----	0

The quality of all routes is enhanced during the second exchange. This is because all nodes will discover more routes, which will be predominantly two-hop routes. The following Table 7 shows what all nodes will advertise by the end of the second exchange.

Table 7: Distance Vector Table.

Destination node	**Node (A) says**	**Node (B) says**	**Node (C) says**	**Node (D) says**
A	0	3	∞	7
B	3	0	6	3
C	9	6	0	2
D	7	3	2	0

By the end of the second exchange, node (C) will learn the route to node (A). All nodes will update their routing tables using the following formulas:

- A= min (B+3,D+7)
- B= min (A+3,C+6)

- C= min (B+6,D+2)
- D= min (A+7,C+2)

The results are as they appear in Table 8.

Table 8: Second Exchange Routing Table.

Node (A) Distance Vector.		**Node (B) Distance Vector.**		**Node (C) Distance Vector.**		**Node (D) Distance Vector.**	
Destination node	Cost	Destination node	Cost	Destination node	Cost	Destination node	Cost
-----	0	A	3	A	9	A	7
B	3	------	0	B	3	B	3
C	9	C	6	-----	0	C	2
D	7	D	3	D	2	-----	0

Given that the number of exchanges depends on the network diameter, the third exchange is the last one for the network in the example. During this stage, the nodes are attempting to find three-hop routes. Each node updates and advertises its own DV table to its neighbours only if a change occurred in the DV table. Neighbours then notify their neighbours if necessary.

Table 9: Routing Table.

Node (A) Distance Vector.		**Node (B) Distance Vector.**		**Node (C) Distance Vector.**		**Node (D) Distance Vector.**	
Destination node	Cost	Destination node	Cost	Destination node	Cost	Destination node	Cost
-----	0	A	3	A	9	A	7
B	3	------	0	B	3	B	3
C	8	C	6	-----	0	C	2
D	7	D	3	D	2	-----	0

3.3.1.1 Distance Vector Dynamics

The Distance Vector algorithm makes deprived routing decisions especially if directions or parts of it are not completely correct. Consequently, all routing tables will be incorrect until the routing algorithms have re-converged. For this reason, DV based protocols are not ideal for scenarios where the network is highly dynamic. There are number situations that can influence network dynamics, such as:

- Adding new routes**:** When new nodes are deployed in a network, they will advertise their DV table to their neighbours. In return, all nodes receiving the new DV will update their own routing tables and advertise their new DVs. This will increase the network traffic and overhead, thus producing a network collision (Chauhan & Dahiya 2012).
- Removing old and invalid routes: Although adding and removing routes work identically, adding routes is easier than removing them. An illustration for that is when new nodes are added they will immediately advertise their DV to notify other nodes. The same process will happen if a new link has been discovered. However, the complication of removing old routes occurs when the node that is responsible for this change is no longer operating. It cannot send failure notification messages, therefore network updates will take a longer time to be spread to all neighbours and adjacent nodes, causing less data delivery and more packets loss ratio(Chauhan & Dahiya 2012).
- Network partitions: Network partition refers to the failure of a network device that causes a network to be split into a number of subnets (Chauhan & Dahiya 2012). This could easily occur when a key link is no longer valid. Network

collision and routes duplications will occur if these partitions come back together.

3.3.1.2 Distance Vector Complications

As mentioned before, the Distance Vector algorithm has a single-path routing strategy; which means that the network will have one key route to each destination (Abuhmida et al. 2015a). This route will be enhanced in every DV exchange as shown in Figure 12.

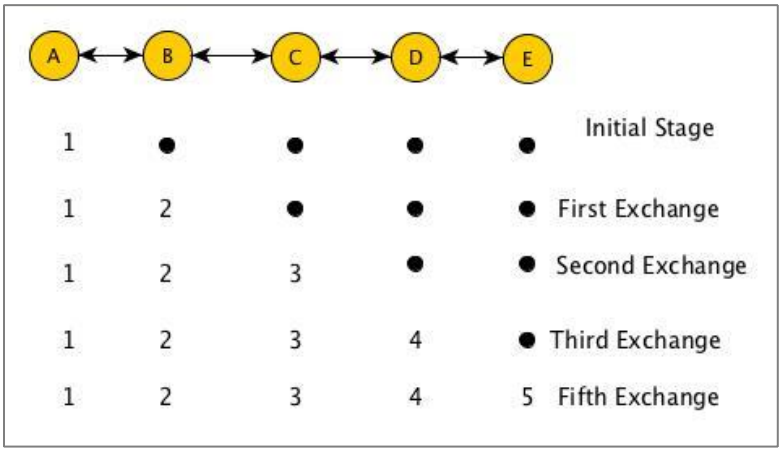


Figure 12: VD exchange Stages.

If the key route becomes invalid for whatever reason, it will result in more network traffic. Since the hop count change must be propagated to all routers, it must be processed on each node of these routes leading to a slower routing convergence. The slow converging will expose the network to the risk of the count-to-infinity problem. The count-to-infinity problem occurs when a node is unable to reach an adjacent node. For example, giving the network as in Figure 13 where the node (A) has become unavailable, the node (B) that is one hop away from node (A) assumes that the unreachable node is two-hops away. Meanwhile, node (C) updates its records to say it is three hops away from the unreachable node, since it is one hop away from node (B).

The nodes continue incrementing their hop count until it reaches infinity, which means the network is jammed.

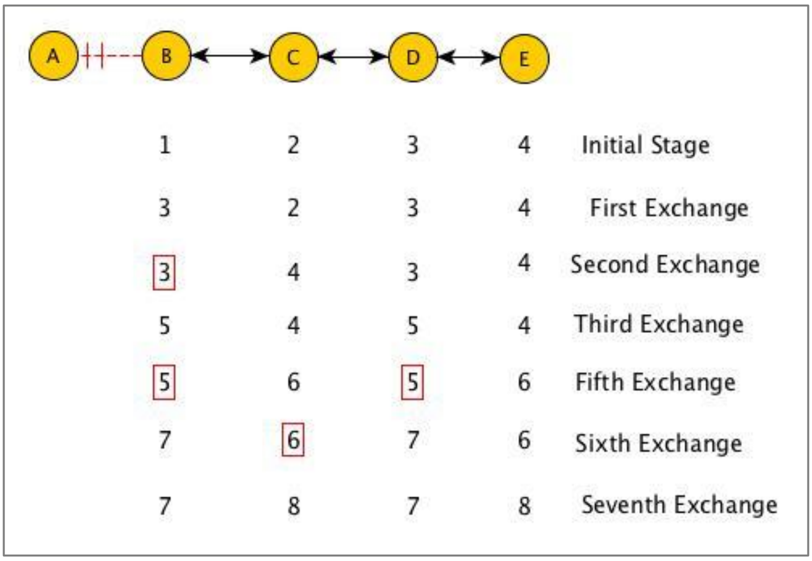


Figure 13: Node Failure.

There are two effective methods of preventing the count-to-infinity problem such as:

- Split Horizon: Split horizon is a method used by the DV algorithm to prevent the routing loops or count-to-infinity problem. Split Horizon follows one basic principle; that is, not to reply to control packets by the same route they came from (Arora & Rizvi n.d.).
- Triggered updates: Triggered updates allow nodes to announce changes in metric values almost immediately rather than waiting for the next periodic announcement. The trigger is a change to a metric entry in the routing table. Unreachable nodes are advertised with a hop count of 16 nodes by the triggered update. However, this is not the best method to be used, as all nodes will send triggered updates immediately. Each update could cause a cascade of broadcasted traffic across the network (Sharma 2010).

Distance vector based routing protocols are usually simple to configure and require little management. The best example of a DV based protocol is the Ad-Hoc distance vector (AODV). AODV is an on-demand protocols that routes data when needed. Once established routes are maintained, they stay until they become invalid.

3.3.1.3 Ad-Hoc On-Demand Distance Vector (AODV)

Ad-Hoc on Demand Distance Vector (AODV) (Royer & Toh 1999; Bansal et al. 1999) is a MANET reactive protocol. The term "Reactive" means that it searches for routes on demand (Royer & Toh 1999). AODV maintains a routing table with entries of routes to nodes that have been communicated with previously. The AODV nodes do not maintain information about the whole network; instead, they keep partial details of the previously used routing information.

To avoid loops, each node has predefined sequence numbers (SQ). The SQ along with the route information should be included by the nodes in their advertised (DV) table whilst finding the routes to a certain destination. When a source node anticipates an established route to a destination, it broadcasts a Route Request (RREQ), as highlighted in Figure 14 (Royer & Toh 1999; Bansal et al. 1999). When an intermediate node receives the RREQ, firstly, it checks the packet ID to guarantee that this packet has not been received before to avoid duplication. Secondly, it checks the destination SQ field of the RREQ message. Routes with the greater SQ are likely to be selected (Royer & Toh 1999; Bansal et al. 1999).

Next, a Route Reply (RREP) control packet is multicast by the source node's neighbours, to confirm that the route to the destination has been found. In some cases, where the route to the destination is not found, the intermediate node increases the number of hops and broadcasts a new RREQ.

Link failure messages will be broadcasted if there is no route found or if the link has become invalid. Link failure messages will be broadcasted if there is no route found or if the link has become invalid (Royer & Toh 1999; Bansal et al. 1999). A Route Error Packet (RERR) notifies defective nodes individually. AODV uses a loop-free to avoid the counting to infinity problem. The following is a summary of the main operations of the AODV protocol as stated above:

- Ad-Hoc On-demand Distance Vector Protocol Based on standard Distance Vector Algorithm.
- Nodes will do nothing even when connection between nodes is valid unless packets needed to be routed.
- Nodes maintain route cache and use destination sequence number for each route entry.
- Route Discovery Mechanism is initiated when a route to a new destination is required; this is achieved by broadcasting a RREQ and RREP.
- Route Error Packets (RERR) is used to erase broken links.

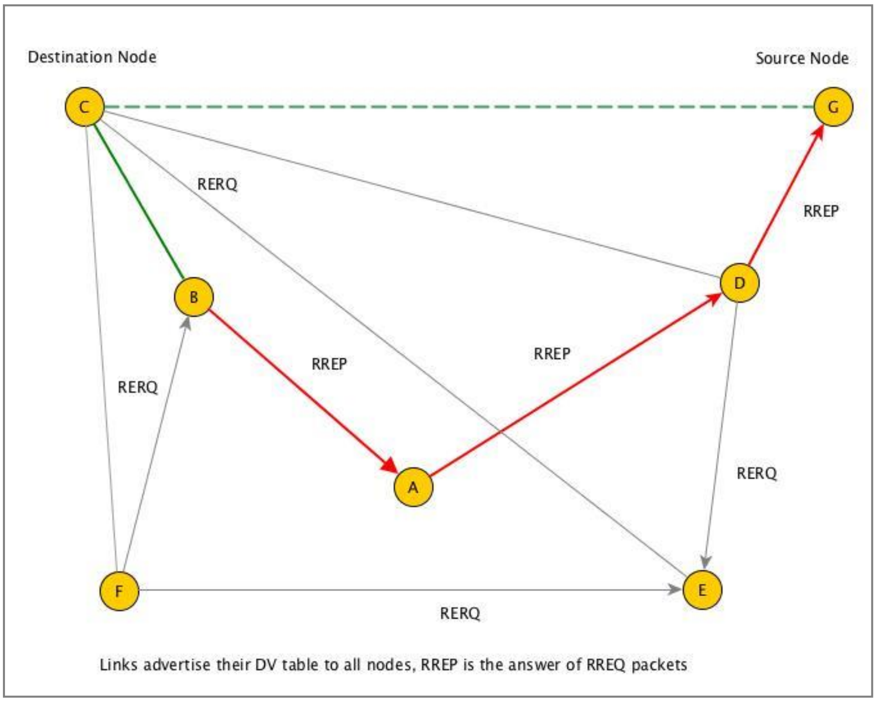


Figure 14: AODV Route Establishment.

3.3.1.4 Dynamic MANET On-demand

Dynamic MANET On-demand (DYMO) is a reactive, multi-hop unicast routing protocol (Yuan et al. 2006). DYMO is defined as an enhanced version of AODV. The routing operation within DYMO is divided into route discovery and route maintenance. Routes are discovered on demand when the originator initiates hop-by-hop distribution of a RREQ message throughout the network to find a route to the target, currently not in its routing table. This RREQ message is flooded to the network using broadcast and the packet reaches its destination. The target then sends a RREP to the source. Upon receiving the RREP message by the source, routes have been established between the two nodes. For maintenance of routes which are in use, routers can elongate route lifetimes upon the successful forwarding of a packet. In order to react to changes in the network topology, routers monitor links over which traffic is flowing (Yuan et al. 2006). When a data packet is received for forwarding and a route for the destination route is broken, missing or unknown, then the source of the packet is notified by sending a route error (RERR) message (Yuan et al. 2006).

3.3.2 Link State Algorithm

A Link State Algorithms (LSA) is a routing algorithm that is often better than DVA, in that it generates less traffic overhead. Although LSA is a completely different class that uses different routing methods, it still serves the same basic purpose as DVA, which is finding the best path for the source node to the destination. Unlike DVA, LSA does not broadcast the routing table. Instead, it broadcasts information about the entire network topology to guarantee that all active nodes have the same topology (Jacquet et al. 2001; Moussaoui et al. 2014). Nodes compute their Forwarding table in the same distributed way the DVA uses.

LSA proceeds in two phases; the first phase is the Reliable Flooding Phase (RFP); which is the first stage of the network lifetime. Nodes flood the topology in the form of Link State Packets (LSP) to describe their partition of the topology. By the end of this phase, each node will have learnt the full network topology (Moussaoui et al. 2014). For example, given a network such as that in Figure 15, node (E) can communicate with nodes (A, B, C, D, and F).

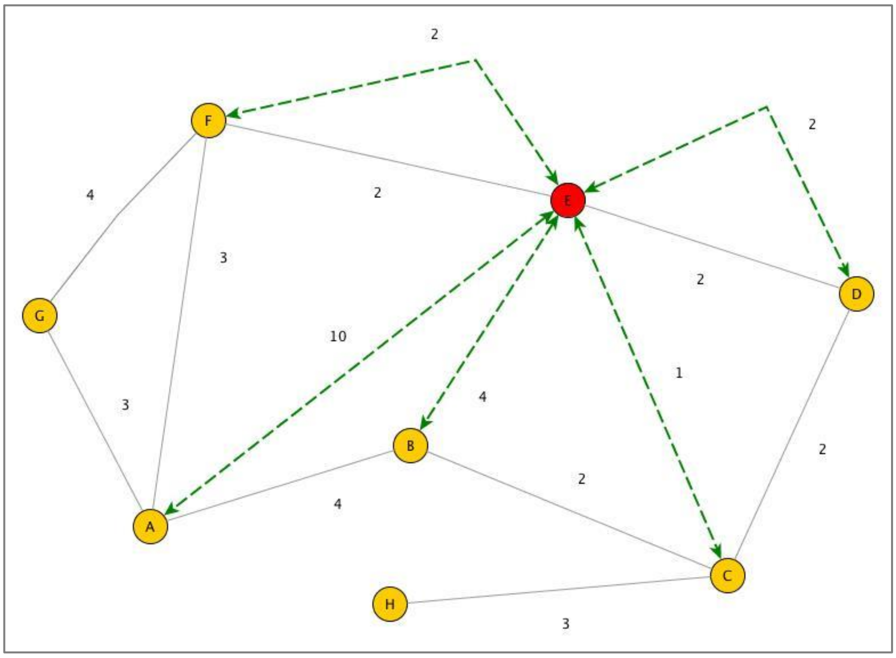


Figure 15: Link State Network.

The node E will construct its own LSP as in Table 10 and cast it to the network.

Table 10: Node (E) LSP.

Node Sequence Number: (E)	
A	10
B	4
C	1
D	2
F	2

When a node receives LSP from a neighbour, it will copy the new topology information into its own LSP. After a while, the node will have the full topology of the network. However, there will be redundant routes in the forwarding tables.

The second phase is the Path calculation phase, which starts only when data is being routed. To choose the best route, nodes run the Dijkstra algorithm (Skiena 1990) that can be defined as an algorithm used for finding the shortest paths between nodes in a network (Ravindranath & Rao 2015).

The Dijkstra algorithm, named after its first author E.W. Dijkstra, is an algorithm used to solve the problem of finding shortest paths from a source vertex to all other vertices in the graph. This algorithm is used by LSA to perform the shortest path routing, given that the network is a sort of directed weighted graph and the nodes are its vertices. Dijkstra's algorithm keeps four sets of vertices as shown in the following Table 11:

Table 11: Dijkstra sets of vertices

S	The set of vertices whose shortest paths from the source have already been determined
V-S	The remaining vertices.
d	Array of best estimates of shortest path to each vertex
Pi	An array of predecessors for each vertex

The basic Dijkstra mode of operation is as follows:

1. Initialise **d** and **pi**,
2. Set **S** to empty,
3. While there are still vertices in **V-S**,
    i. Sort the vertices in **V-S** according to the current best estimate of their distance from the source.
    ii. Add **u**, the closest vertex in **V-S**, to **S**.
4. Relax all the vertices still in V-S connected to u.

Any change in the values of (d) or (pi), topology change, will cause a change in the S and V-S, which in turn will cause degradation in the shortest path solution.

3.3.2.1 Handling Changes

LSPs are triggered if there is a topology change such as newly deployed nodes, a node failure or link failure. The process of restoring the changes caused by adding nodes is similar to removing them (Moussaoui et al. 2014). All nodes update their forwarding table by a new LSP coming from their neighbours. LSPs carry sequence numbers (SQ) to distinguish the new LSPs from the older ones. Nodes only accept and forward the "newest" LSPs.

In the case of link failure, the nodes will send a new LSP with infinity cost as an indicator of link down. For instance, given the network in Figure 16, the link to the node (G) has failed; the adjacent nodes (A, F) will learn that this link is down, using regular periodical checking.

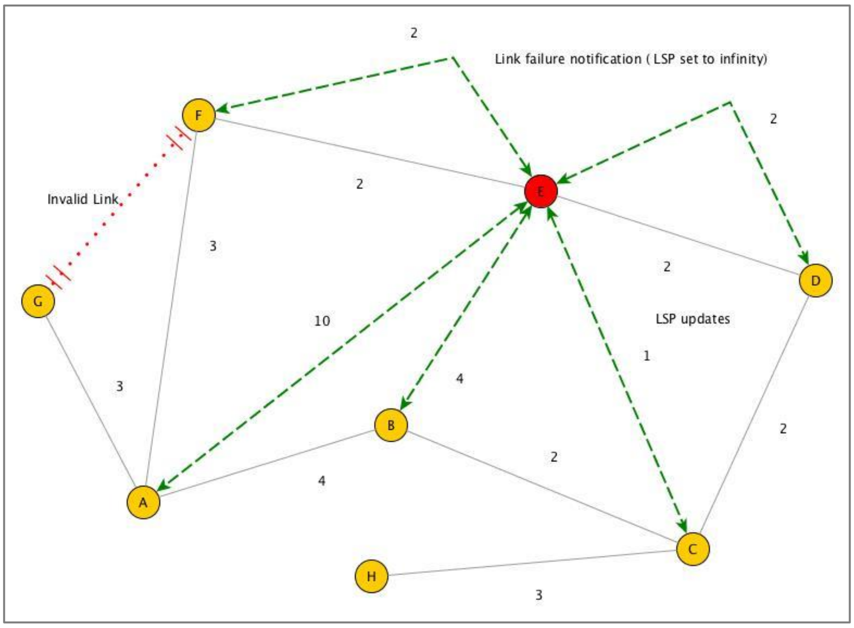


Figure 16: LS Link Failure.

Once (A, F) have learnt about the (G) link failure, they will cast a new LSP as shown in Table 12 to notify the rest of the network.

Table 12: Link Failure LSP.

Node A's LSP		**Node F's LSP**	
B	4	B	4
E	10	E	2
G	∞	G	∞

3.3.2.2 Link state Algorithms Complications

Despite all the advantages of LSA, it has several complications, which in some cases are hard to avoid. One of these complications is when the SQ number reaches its limit. LSA utilises the SQ to distinguish old LSPs from the new ones to avoid any redundancy (Moussaoui et al. 2014). LSPs are triggered in every topology change. A new SQ will be assigned to each new LSP, which is a crucial matter when the nodes are not stationary. The node's motion will cause a topology change, which will trigger LSPs. The extensive use of LSPs will result in increasing the SQ until it reaches the limit, which leads to termination of the search, consequently packet loss.

By the same token, in addition to the above, the network partitions are another vital LSA complication. When some nodes fail or links become invalid for whatever reason, some nodes do not detect a failed link or node immediately; therefore, they would forward data packets into a black hole. A black hole is a route created by a corrupted LSP with incorrect SQ. Some nodes discover the failure before others, which may cause inconsistency in finding the shortest path. The shortest path's inconsistency would result in transient forwarding loops. The existence of forwarding loops means that LSA based routing protocols are not ideal for VoIP networks, online gaming networks and MANETs.

LSA based protocols converge much faster than distance vector routing protocols, support classless routing, send updates using multicast addresses and use triggered routing updates. However, a disadvantage is that they require more router CPU and memory usage than DVA based protocols, which makes them harder to configure. There is a comment on the use of link state routing in MANET represented in the discussion of the OLSR protocol, which is discussed in the following section.

3.3.2.3 Optimised Link State Routing (OLSR)

Optimised Link State Routing (OLSR) is an LSA-based MANET routing protocol. OLSR exchanges topology information with other nodes; a few nodes are selected as Multipoint relays (MPRs), which are responsible for broadcasting messages and generating link state information during the flooding process. Using LSA reduces the message overhead by minimising the number of control messages flooded into the network (Black 2000)(Clausen et al. 2003). Nodes maintain the information of neighbours and MPRs, by exchanging periodic HELLO messages, which will help determine the link information, as illustrated in Figure 17.

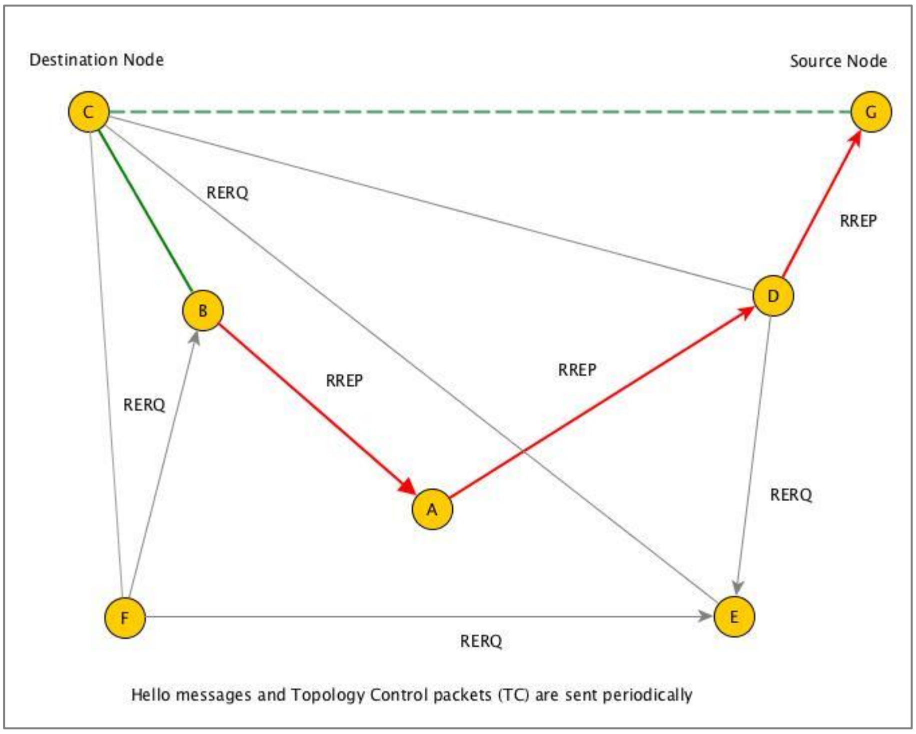


Figure 17: OLSR Route Establishment

For example, node (A) transmits a HELLO message to the node (B), a process known as an asymmetric link as in Figure 18. Asymmetric link formation will help the nodes to choose the most suitable MPRs. MPRs will send the topology control (TC) messages containing the information about link costs and MRP node information (Black 2000; Clausen et al. 2003).

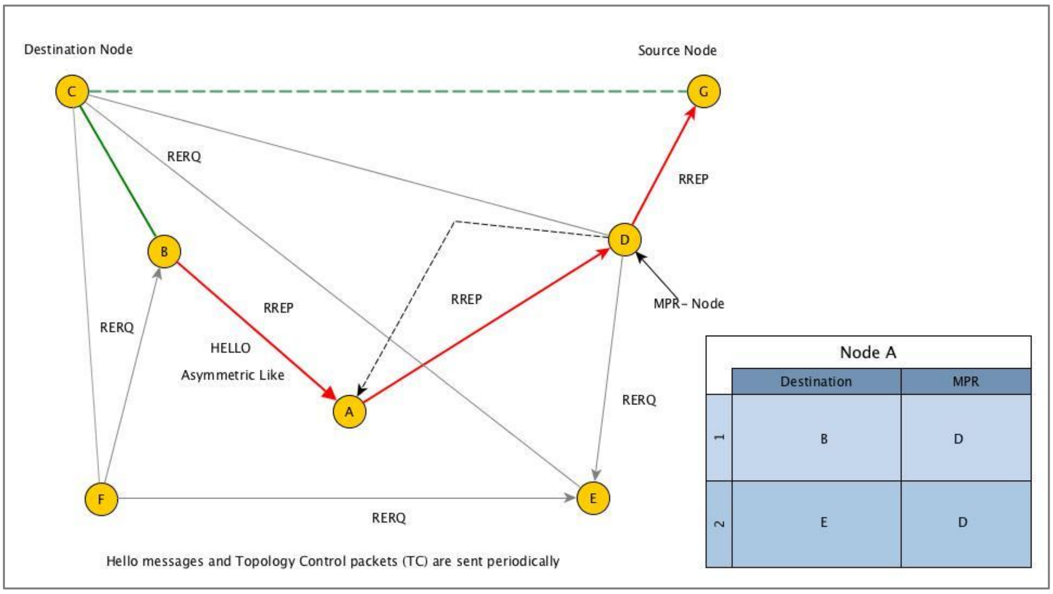


Figure 18: OLSR Asymmetric Like Information

The following is a summary of the main features of OLSR:

- OLSR stands for Optimized Link State Routing.
- OLSR maintains routes to all destinations in the network.
- OLSR uses two kinds of control messages: Hello and Topology Control (TC).
- OLSR uses Multipoint Relays (MPR) to reduce the possible overhead in the network.

A MANET routing algorithm should not only can find the shortest path between the source and destination, but it should also be adaptive, in terms of the changing state of the nodes, the changing load conditions of the network and the changing state of the environment. Traditional routing algorithms face many complications to achieve such features due to the dynamic behaviour and resource constraints in MANETs. To overcome these latent difficulties, new algorithms needed to be explored. And the literature has suggested a few them for instance, algorithms based on biologically inspired algorithms and the genetic algorithms**.** The next section will be discussing the verity of algorithms to justify the criteria for selecting the proper algorithm to be a baseline of the proposed protocol.

3.3.2.4 Landmark Ad-Hoc routing

Landmark Ad-Hoc routing (LANMAR) LANMAR is an effective proactive based routing protocol. Proactive means it periodically forwards control packets to search and maintain routes. LANMAR uses a few hops as a cost measurement to build its routing table. Although LANMAR does not need any established hierarchic to route data it still similar the Fish Eye algorithm (Furnas 1981). LANMAR uses some of the geographical promontories to keep track of its logical topology, to count the number of hops per route, LANMAR utilises a specific address for each node, which reflects its position within the hierarchy, this enables the protocol to discover and maintain a route faster (Pei et al. 2000).

3.4. **Alternative MANET Protocols and Algorithms**

Routing algorithms define the path taken by a packet between source and target destination. They must prevent deadlock, livelock (infinite loop), and starvation (bottleneck) situations (Ni & McKinley 2000). Deadlock is a cyclic dependency among nodes requiring access to a set of resources, so that no forward progress can be made, no matter what sequence of events happens (Moraes et al. 2003). Live lock refers to packets circulating the network without ever making any progress towards their destination. Starvation happens when a packet in a buffer requests an output channel, being blocked because the output channel is always allocated to another packet.

Routing algorithms can be classified according to three important standards as follows:

- The routing decisions
- The definition of the shortest path or a path
- The path length.

According to where routing decisions are taken, it is possible to classify the routing in source and distributed routing. In source routing, the whole path is decided at the source switch, while in distributed routing each switch receives a packet and defines the direction to send it. In source routing, the header of the packet has to carry all the routing information, increasing the packet size (Ni & McKinley 2000). In distributed routing, the path can be chosen as a function of the network instantaneous traffic conditions. Distributed routing can also consider faulty paths, resulting in fault tolerant algorithms.

The above-mentioned facts are the baseline for selecting the right algorithm to solve the specific routing problem. Routing is an optimisation problem, which cannot have a single optimal solution: more than one solution can be found and the best one is chosen. This type of problems needs an adaptive, decentralised, distributed and simple algorithm- fewer rules- to find the optimal solution such as the Swarm intelligence and Genetic algorithms, where the concept of these two algorithms is mainly based on the biological behaviours of natural objects. Evolution inspires both these algorithms.

Nature has inspired researchers in many ways. Aeroplanes have been designed based on the structures of birds' wings. Robots have been designed to imitate the movements of insects. Resistant materials have been synthesised based on spider webs. The fascinating role that insects play in our lives is obvious. It is interesting how these tiny insects can find the shortest path for instance between two locations without any knowledge about distance, linearity, etc. Biologists studied the behaviour of social insects for a long time. For decades, entomologists have known that insect colonies are capable of complex collective action, even though individuals adhere to straightforward routines. When foraging, for example, workers appear to march to a drumbeat that dictates when to turn and when to lay down pheromone to guide other workers. As simple as these rules are, they create an effective dragnet to haul in food as efficiently as possible. In this manner, ants have been solving problems very skilfully every day of their lives for the last 100 million years (Panda & Padhy 2008).

Several modern algorithms have evolved in the last two decades that facilitate solving optimisation problems that were previously difficult or impossible to solve such as routing. These tools include evolutionary computation, simulated annealing, *Tabu* search, particle swarm, Ant colony optimisation, etc. Recently, genetic algorithm (GA), particle swarm optimisation (PSO) and the Ant Colony Optimisation (ACO) techniques appeared as promising algorithms for handling the optimisation problems. These techniques are finding popularity within the research community as design tools and problem solvers because of their versatility and ability to optimise in complex multimodal search spaces applied to non-differentiable cost functions.

GA can be viewed as a general-purpose search method, an optimisation method, or a learning mechanism, based loosely on Darwinian principles of biological evolution, reproduction and “the survival of the fittest” (Golberg 1989). GA maintains a set of candidate solutions called population and repeatedly modifies them. At each step, the GA selects individuals from the current population to be parents and uses them to produce the children for the next generation. In general, the fittest individuals of any population tend to reproduce and survive to the next generation, thus improving successive generations. However, inferior individuals can, by chance, survive and also reproduce. GA is well suited to and has been extensively applied to solve complex design optimization problems because it can handle both discrete and continuous variables, non-linear objective and constraint functions without requiring gradient information (Abido & Abdel-Magid 2003), (Varšek et al. 1993), (Ramírez & Castillo 2004) and (Abido 2005).

PSO is inspired by the ability of flocks of birds, schools of fish, and herds of animals to adapt to their environment, find rich sources of food, and avoid predators by implementing an information sharing approach. PSO technique was invented in the mid-1990s while attempting to simulate the choreographed, graceful motion of swarms of birds as part of a socio-cognitive study investigating the notion of collective intelligence in biological populations (Jolfaei et al. 2016). In PSO, a set of randomly generated solutions propagates in the design space towards the optimal solution over a number of iterations based on a large amount of information about the design space that is assimilated and shared by all members of the swarm (Zhang et al. 2015). Both GA and PSO are similar in the sense that these two techniques are population-based search methods and they search for the optimal solution by updating generations. Since the two approaches are supposed to find a solution to a given objective function but employ different strategies and computational effort, it is appropriate to compare their performance.

The following section presents a discussion of the swarm intelligence based routing algorithms and detailed discussion of an ant-based algorithm that is designed especially for distributed routing.

3.4.1 Swarm Intelligence Algorithms for the Routing Problem

Bio-inspired or Swarm Intelligence algorithms (SI) use the method of imitating the natural swarm behaviour of some social insects to solve optimisation problems. The synchronisation and collaboration of multiple intelligent agents have been studied extensively within the field of Distributed Artificial Intelligence (DAI) from the early 1970's (Kennedy et al. 2001). The DAI field itself was mainly focused on solving problems, including software agents. The robotics research community became an active field during the late 80's. Various studies were conducted in the area of cooperative robotics, such as CEBOT in 1987 (AbuKhalil et al. 2015; Ducatelle et al. 2009), SWARM in 1988 (Ducatelle et al. 2009), ACTRESS in 1989 (Ducatelle et al. 2009) and GOFER in 1990 (Arabshahi et al. 2001). These early projects were purely theoretical assignments that were conducted primarily on simulations.

In the early 1998, a new project was introduced by Dorigo (Dorigo 2006) that presented an original concept of utilising the DAI algorithms in order to enhance the routing in computer networks. Ant colonies and bird flocks were the main inspiration for this research, which has totally changed the definition of the Swarm Intelligence hypothesis in the following year by Bonabeau as follows:

> *"Swarm Intelligence (SI) is the property of a system whereby the collective behaviours of unsophisticated agents interacting locally with their environment, causing coherent functional global patterns to emerge."* (Bonabeau et al. 1999).

Since then, Swarm Intelligence has proven to have many powerful properties that are required by countless engineering systems, such as routing protocols, robotics and control systems (Bonabeau et al. 1999). The working mechanism of these biological systems have been reverse engineered and properly adapted to design unique distributed routing algorithms (Bonabeau et al. 1999). The new classes of routing algorithms, inspired by Swarm Intelligence, have been developed to solve numerous complex routing problems for MANETs (Farooq & Di Caro 2008). The SI based algorithms rely on the communication of a massive amount of simultaneously interacting artificial agents, such as Ant Colony Algorithms (ACO), Honeybee based and Slime based Algorithms. However, algorithms such as the Honeybee and Fish are mainly designed for solving optimisation computational problems.

The main SI algorithm that explicitly designed for routing is the ant-based algorithms is the Ant Colony Optimization (ACO) which is the main inspiration for this research. ACO is inspired by the collective behaviour of ant colonies and targets discrete optimisation problems. While comparing MANETs and ANTS in Table 13 it is concluded that though they have similarities like same physical structure, self-configuration and self-organization but still distinguished from each other in the route foundation, overhead, motive, routing table information. These differences are all in favour of the ant algorithms to provide better MANET routing.

Table 13: MANTEs vs ANTS

Parameters	MANETs	ANTS
Overhead	More	Less
Packet Delivery Ratio	Less	More
Route Discovery Procedure	Route-Request/Reply message are used	Pheromone value is used
Motive	Find shortest path for routing	To provide definite shortest path
Path Discovered	Single path, partially multipath	Multipath

The following introductory section will describe how real ants have inspired the definition of artificial ants that can solve optimisation computational problems such as dynamic routing.

3.4.1.1 ANT Colony Optimization

A simple individual ant can perform very complex tasks when it acts in the collaborative manner of a colony (Bonabeau & Théraulaz 2000). Ant behaviour has been observed by many researchers (Bonabeau et al. 2000), who have been able to document and summarise it as follows:

1. An ant colony can contain millions of ants.
2. The building and securing of a nest.
3. Ants communicate using their antennae and pheromones.
4. Ants use pheromones in a collaborative way to process information relating location of a food supply.
5. Ants can transport large items that could reach eight times its weight by cooperating with each other.
6. Ants sort food items into different groups based on type according to their own diet.

Several experiments were conducted between the 1980s and early 1990s by a group of biological researchers at the Libre de Bruxelles University (Dorigo 2006), which obtained original theoretical results reflecting the influence of pheromones on ant decision-making. A pheromone is a chemical substance that ants deposit on the ground to mark their path from their nest to food source. The following ants will choose the path with stronger pheromone, so they can retrieve the shortest path.

The results indicated that pheromone performance is a sort of “*dynamic collective memory"* of the colony, a depository of all the most recent “*foraging experiences”* of the ants belonging to the same colony (Dorigo 2006). By continually sensing and updating this depository chemical, the ants can indirectly communicate and influence each other throughout the environment. This basic method of indirect communication is enough to allow the colony as a whole to discover the shortest paths and alternatives, connecting a source of food to their nest (Dorigo 2006; Bonabeau et al. 2000).

The first generation of ants will wander randomly, laying down a pheromone trail in every path they take. When food is found, ants will return to the nest laying down the second portion of pheromone trails, which will increase the pheromone smell along the chosen path. The second generation will follow the strongest pheromone trails. Since the ants on the shortest path lay pheromone trails faster, it gets reinforced with more pheromone, making it more appealing to future ants. The ants become increasingly likely to follow the shortest path since it is constantly reinforced with a higher volume of pheromones as illustrated in Figure 19. The pheromone trails of the longer paths evaporate over time; therefore the probability of them being chosen is low (Dorigo 2006; Bonabeau et al. 2000; Dorigo et al. 2011; Dorigo et al. 2010).

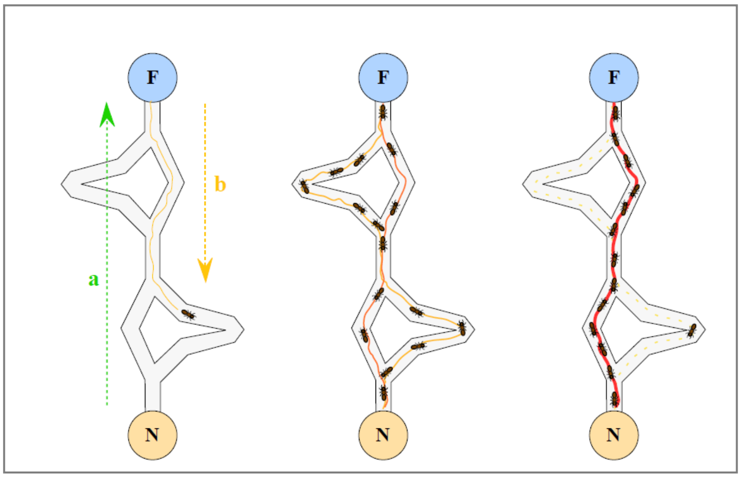


Figure 19: Natural Ant Path

The simple behaviour of an ant can be utilised to perform a complex task when deployed in a computer networks context as highlighted in the following Table 14. This foraging behaviour has been replicated in simulations that have inspired a class of ant algorithms that can be used to solve distributed routing problems.

Table 14: Ants in Networks

Nature	**Computer Networks**
Environment, Nature.	Graph, Network.
Nest to food source.	Source node to destination node.
Ants.	Agents, Artificial agents, Control packets.
Visibility.	Cost condition, e.g. Euclidian distance
Pheromone.	The link weight (τ)
Foraging behaviour.	Random walk through graph or a network.

An ACO algorithm can be described as the interaction of three main procedures as follows:

- Construction of Ants Solutions (CAS): This procedure manages a colony of ants that simultaneously visit adjacent nodes of the problem in question by moving through neighbour nodes of the problem's Construction Network (NC). They move by applying a probabilistic local decision behaviour that makes use of pheromone trails and heuristic information. In this way, ants incrementally build solutions to the optimisation problem. Once an ant has built a solution, or while the solution is being built, the ant evaluates the portion of the solution that will be used by the Update Pheromones procedure to decide how much pheromone to deposit.

- Update Pheromones (UP): This procedure is the process by which the pheromone trails are modified. The trails value can either be increased, as ants deposit pheromone on the paths they use, or reduced, due to pheromone evaporation.
- Solution Actions (SA): This procedure is used to implement centralised actions, which cannot be performed by single ants. For instance, the activation of a local optimisation procedure, or the collection of global information that can be used to decide whether it is useful or not to deposit additional pheromone to differ the search process from a non-local perception.

There have been several successful implementations of routing algorithms for the wired networks such as ANTNET as well as the wireless networks namely ANTHOCNET, this research is inspired by ANTHOCNET routing technique in MANET. More details on ANTHOCNET is discussed in the following section.

A General Description of ANTHOCNET

ANTHOCNET is a hybrid, adaptive routing algorithm that utilises both reactive and proactive routing. Specifically, it combines a reactive route setup process with proactive route maintenance and improvement process. The way ANTHOCNET gathers stores and uses routing information is inspired by the ACO and routing and distance vector routing; routing information is stored in two routing tables namely; Pheromone Table and Neighbor Table.

- Pheromone tables: entry τ_d of this pheromone table contains information about the route from node (*i)* to destination (d) over neighbor (j) this information includes the pheromone value (τ), which a value is indicating the relative goodness of going over node (j) when traveling from node (*i)* to destination (d) Virtual Pheromone value (for more information refer to Chapter 4).

- Neighbour table: This table keeps track of the wireless nodes to which it has a wireless link (for more information refer to Chapter 4).

The algorithm is composed of two main parts the Reactive part and the Proactive part. Next is a description of the two parts of the algorithm

Reactive Route Setup

This section describes the reactive component of the algorithm. It starts at the beginning of any communication session.

- The source node of the session controls its pheromone table, to see whether it has any routing information available for the requested destination.
- If it does not, it starts a reactive route setup process, in which it sends an ant packet out over the network to find a route to the destination. Such an ant packet is called a reactive forward ant.
- Each intermediate node that receives a copy of the reactive forward ant, forwards it. This is done via broadcasting in case the node does not have routing information about the ant's destination in its pheromone table. If routing information is available the packet is unicast to its neighbour.

Reactive forward ants: represents the route request packet and its function can be summarised as follows:

- As these ants broadcasted in the network towards their specific sink, they store the nodes that they have visited on their way in a list inside the packet.
- The first copy of the reactive forward ant to reach the destination is converted into a reactive backwards ant, while subsequent copies are destroyed.

Reactive backwards ant: represents the route reply packet and its function can be summarised as follows:

- Retraces the exact path that was followed by the corresponding forward ant back to the source.
- On its way, it collects quality information about each of the links of the path.
- At each intermediate node and at the source, it updates the routing tables based on this quality information.

Proactive route maintenance process

This process represents the proactive component of the algorithm. It works on updating, extending, and improving the available routing information. This process runs for if the communication session is going on. It consists of two different sub-processes: pheromone diffusion and proactive ant sampling.

Pheromone diffusion: It is the first sub-process of proactive route maintenance. It can be considered a deployable but unreliable way of spreading pheromone information. The function of the pheromone diffusion is summarised as follows:

- Spreads out pheromone information that was placed by the ants.
- Nodes periodically broadcast messages containing the best pheromone information they have available.

Proactive ant sampling: This sub-process turns the virtual pheromone into reliable regular pheromone. The virtual pheromone is the pheromone that is obtained via pheromone diffusion and is kept separate from the normal pheromone placed by the ants because of its potential unreliability, and the Regular pheromone is the reliable pheromone placed by the ants. This process is summarised as follows:

- All nodes that are the source of a communication session periodically send out proactive forward ants towards the destination of the session.
- These ants construct a path in a stochastic way, choosing a new next hop probabilistically at each intermediate node.
- Different from reactive forward ants, they are never broadcast.
- When calculating the probability of taking a next hop, proactive forward ants consider both regular and virtual pheromone.
- This way, they can leave the routes that were followed by previous ants, and follow the (potentially unreliable) routes that have emerged from pheromone diffusion.
- Once a proactive forward ant reaches the destination, it is converted into a proactive backwards ant that travels back to the source and leaves pheromone along the way (regular, not virtual pheromone), just like reactive backwards ants.
- Proactive ants can follow virtual pheromone and then, once they have experienced that it leads to the destination, convert it into regular pheromone.

3.5. **Summary**

In conclusion, the implementation of Distance Vector and Link State routing algorithms is insufficient to satisfy the dynamic features of MANET networks. Using Distance Vector or Link State routing algorithms in MANET means that frequent topology changes will greatly increase control overheads or suffer the slow route convergence that decays the algorithms' performance. If these issues are left unaddressed, the scarce MANET bandwidth is likely to be overused. Additionally, both algorithms are known to cause routing information inconsistencies and route loops when used for highly dynamic MANETs.

To overcome the dynamic behaviour and resource constraints in MANETs, an approach inspired by Ant Colony behaviour is utilised. The social organisation of these insects is based on the genetically evolved commitment of each individual to the survival of the group, which is a key factor behind their success (Di Caro et al. 2008b). Moreover, these insect societies exhibit the fascinating property that any explicit form of centralised control does not regulate the activities of the individuals, as well as of the society. The most successful and most popular research direction in ant algorithms is dedicated in their application to combinatorial optimisation problems, and it goes under the name of Ant Colony Optimization heuristic (ACO).

ACO finds its roots in the experimental observation that ants can select the shortest path among the few alternative paths connecting their nest to a food source. While searching for food, ants deposit a pheromone per probabilistic rule, and they travel the directions that are locally marked by higher pheromone intensity.

The vast majority of the existed effective routing protocols handling mobility are designed for MANETs and as the overwhelming current research mainstream on sensor networks consider it as a static network (Nikolidakis et al. 2013; Bandyopadhyay

& Coyle 2003). However, there exist many applications which require a higher mobility rate, for instance, habitat monitoring, battlefield surveillance, and object tracking. Which increases the unpredictable topology changes and frequent path failures (Wang & Yang 2007). This increased level of mobility causes higher rate of path breakage that leads to increase in the delay, overhead and packet loss. Therefore, this research is aiming to design an adaptive and effective routing protocol for highly mobile MANETs (Anastasi et al. 2009).

The following chapter represents the main contribution of this research. It describes in details the proposed routing protocol, its component, process, and detailed classification.

Chapter 4. ANTMANET Routing Protocol for MANET

Overview

MANETs are designed to provide service to their users at an acceptable level; performance modelling and evaluation should therefore play a crucial part in the designing and monitoring of those processes which ensure the successful deployment of a network. The higher node mobility of MANET adds a challenge to provide these acceptable levels of services. Therefore, this chapter presents an effective routing protocol based on ACO to handle mobility in MANET.

The remainder of this chapter is stretchered as follows, 4.1 illustrates a detailed overview of the proposed protocol (ANTMANET), followed by a detailed illustration of ANTMANET' structure in section 4.2. In section 4.3 a detailed discussions of Control packets structure and casting methods 4.4 discusses events, interruptions, and timers of the proposed protocol, complimented with a full protocol classification in section 4.5.

4.1. **Protocol Overview**

The function of a routing protocol in MANETs is to establish routes between several nodes. Generally, MANET routing protocols have a difficult design scope, the main reasons for this are: the highly dynamic nature of these networks due to the nodes mobility, and the need to operate efficiently with limited resources, such as limited network bandwidth and the limited processing capabilities and energy constraints. For these reasons, many Ad-Hoc routing protocols are not capable of scaling and handling high mobility well.

The main aim of this section is to introduce the ANTMANET protocol, which can work efficiently in MANET with a high level of mobility, regardless of the challenges. ANTMANET is essentially a hybrid routing protocol for MANETs, based on Swarm Intelligence to resolve the raising routing issues resulting from high mobility. Swarm Intelligence (SI) is the property of a system whereby the collective behaviours of simple agents interacting locally with their environment cause coherent functional universal patterns to emerge (Nagi et al. 2015). Swarm intelligence has many powerful properties required by countless engineering systems, such as routing protocols, robotic and control systems (Bonabeau & Théraulaz 2000).

Ant Colony Optimization (ACO) is a Swarm Intelligence algorithm based on modelling the collective behaviour of ants to solve computational hard problems such as travelling salesman problems (TSP) (Reinelt 1991). In the real world, each ant remembers only a small amount of information and it can only utilise a small number of simple rules; for instance, in their journey of searching for food, ants can transmit and receive pheromone inputs. A pheromone is a chemical substance that ants apply it on the ground to mark their path from their nest to their food source. The following ants choose the path with stronger pheromone, so they can easily retrieve the shortest path.

Despite the simple nature of an individual ant, the colony exhibits far more complex behaviour to adjust themselves per the changes in their environment; for instance, alternative paths will be retrieved if the shortest path becomes invalid for some reason. ANTMANET is based on demonstrating this simple aspect to perform tasks that are more complex. The following section discusses the design philosophy of ANTMANET.

4.2. **Protocol Taxonomy**

ANTMANET is a hybrid protocol; it combines the advantages of both proactive and reactive protocols. Hybrid routing protocols use reactive phase to guarantee more accurate metrics to determine the best paths to destinations and report routing information only when there is a change in the topology of the network. In addition, they use the proactive phase to allow rapid convergence and fresh routing information through the nodes.

The workflow diagram of ANTMANET is shown in Figure 20, firstly, the Reactive phase which is divided into two stages: The Initial stage and the Route Discovery stage. Secondly, the Proactive phase which is responsible for maintaining fresh routing information.

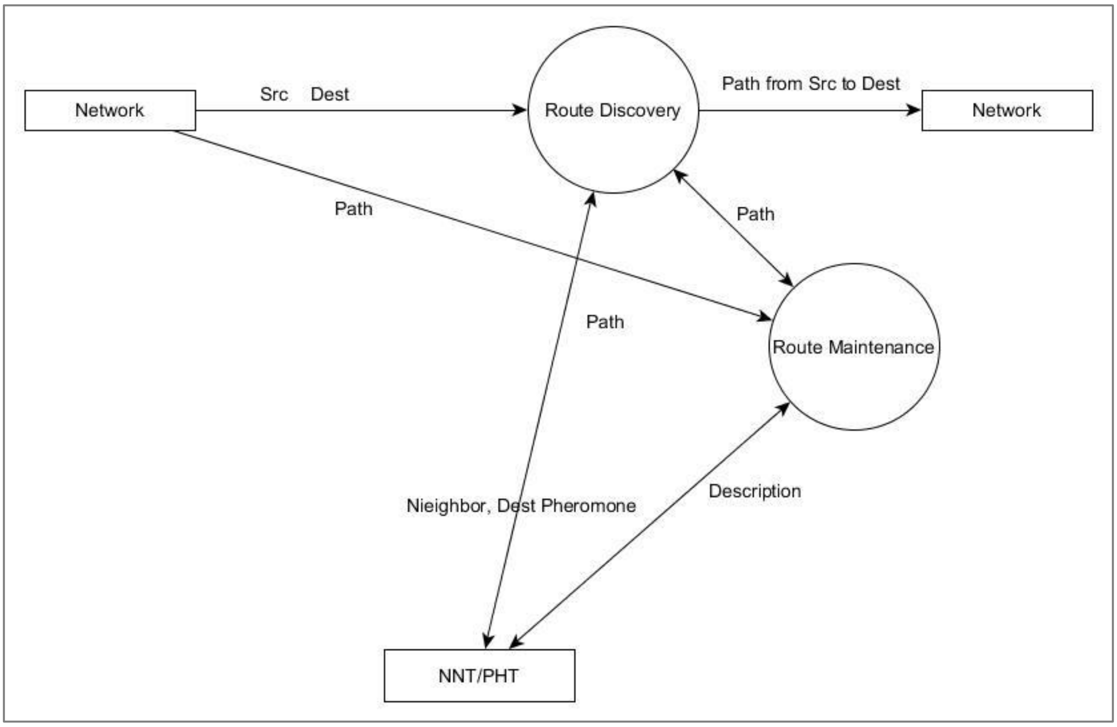


Figure 20: ANTMANET workflow diagram.

4.2.1 Initial stage

The initial stage occurs in a very early period of the network lifetime, where all or some nodes have just been deployed in the network. Apart from their own location and node-ID, they have no predefined knowledge about other devices in the same network. During this stage, the nodes begin to build their own local topology by constructing a unique node structure as shown in Figure 21. Each node maintains three tables and one vector, organised as follows:

- Statistical Vector: This table is a one-dimensional vector containing the fixed values of (α, β, ρ) which are an ACO probability parameters (Singh 2014), and $\varepsilon_1, \varepsilon_2, \varepsilon_3$ which are the initial Local zone pheromone values (for more details refer to section 4.3).
- Geo Table (GEO): This is a new table that is added to the original algorithm as part of this research. The entries of this table represent the nodes location

information such as coordinates and geo-lifetime of the node in the North Neighbour Table.

- North Neighbour Table (NNT): This is a new table that is added to the original algorithm as part of this research. The North Neighbour Table N_i) kept by node (i) is a one-dimensional vector with one entry for each of i's neighbours located at its north. (N_{ij}) is an entry in (i's) NNT table, which indicates the timestamp, related to geo-positioning of (i's) neighbour (j). This indicates when (i) last heard from (j).
- Pheromone Table (PHT): Each node (i) sustains a two-dimensional matrix. The entry of this matrix is $(\tau_{i,j}^{d})$ which represents the information about the route from node i to destination (d) over neighbour (j) (Singh et al. 2014; Dorigo 2006) (for more details refer to section 4.3).

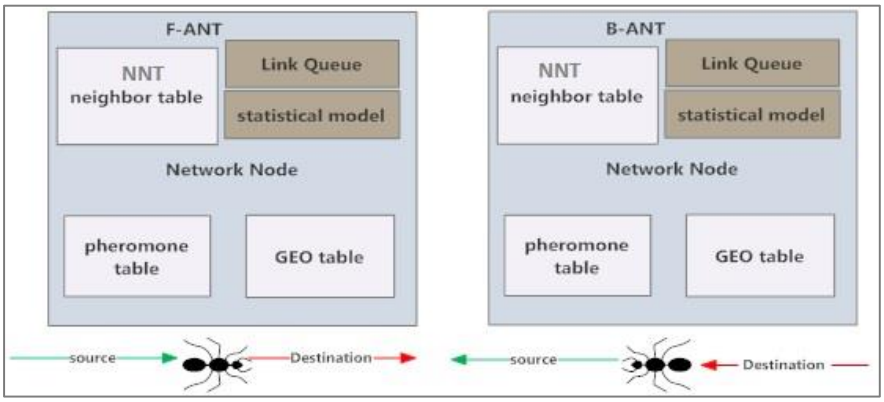


Figure 21: Node structure.

In this early stage of the network convergence, no data packets are exchanged between nodes. They only gather information about the network to build up their node structure. Nodes construct their routing tables by exchanging control packets, namely the Forward-Ant (FANT) and the Backward-Ants (BANT) (for more details refer to section 4.4). For instance, considering the network in Figure 22 Assuming that the Initial stage has just started, the source node (s) broadcasts FANTs to all nodes in its transmission range to construct a neighbourhood (local topology). The North Neighbour Table constructs the neighbourhood, which basically contains every node within the source transmission range and located in the north of the concerned node.

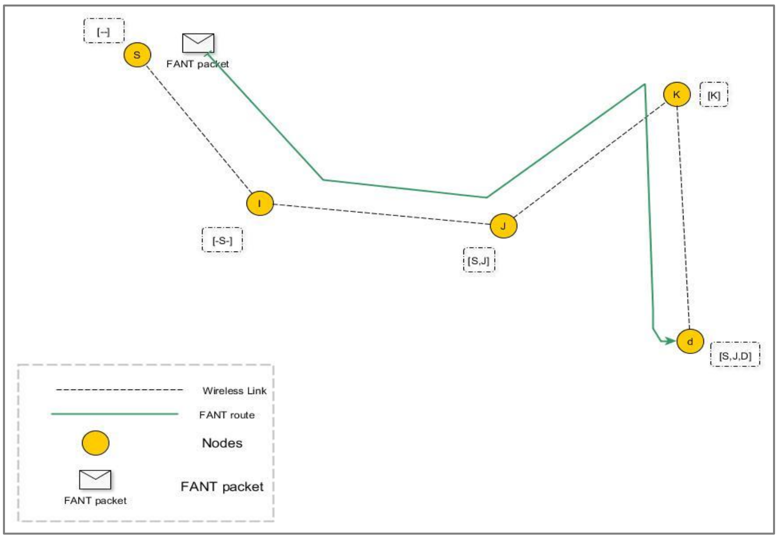


Figure 22: Initial Stage.

Each node determines its own geographical position using either a small inexpensive low power GPS receiver or any other techniques for finding relative coordinates. The coordinates (x, y) are exchanged while control packets are being aggregated.

Considering Figure 23, which shows a source node (s) and the destination node (d), if (s) randomly chosen neighbour node (j) per a geographical policy named Local Zone (LZ). The Local Zone technique is based on partitioning the neighbourhood into three Local Zones. Consider a segment between (d) and (s). This segment is diagonal line with slope angle. The slope angle is calculated using Equation 1 (Kamali & Opatrny 2007).

$$\theta = arcTan(\frac{(y_2 - y_1)}{(x_2 - x_1)} \bullet \frac{180}{PI}$$

Equation 1: Slope Angle.

Where:

X, Y are the coordinates.

PI is the (π) constant = 3.14

This angle denotes each zone, in other words, a neighbour node (j) will belong to (LZ1) if it is located within $\theta \leq 45$ from the source node (s). In the same way node (j) will belong to (LZ2) when it is located in $45 < \theta < 135$ and it will be belong in the (LZ3) if it located to $135 \leq \theta \leq 180$.

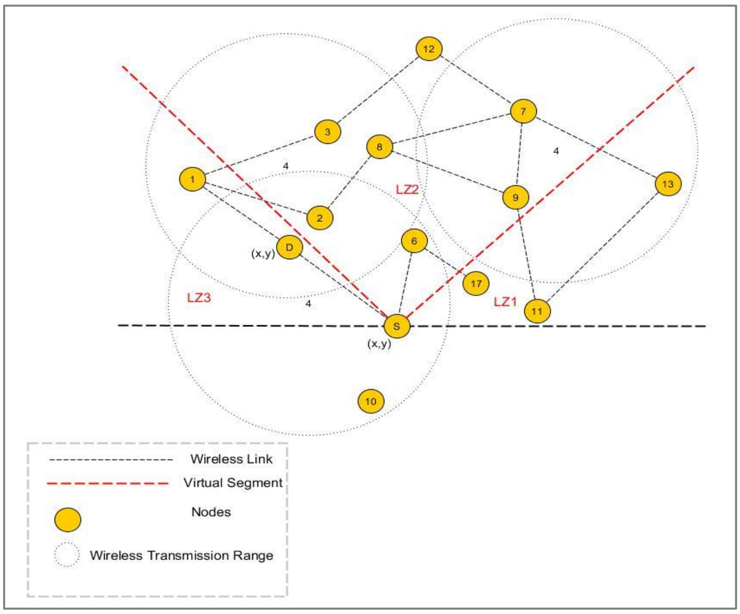


Figure 23: Local Zone Angle

Later, as the FANT populated throughout the neighbourhood, it collects a memory list P of all the nodes that it has been visited with their corresponding location at the time on its way from (s) to (d) (Dorigo & Di Caro 1999), for instance, P = (1, 2, 3,…, d-1).

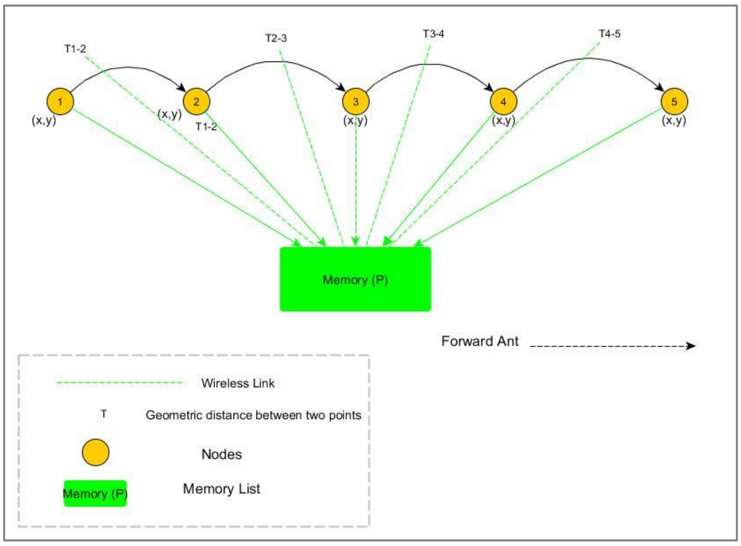


Figure 24: Forward Ant Memory List.

This is not the only information the FANT collects. It also keeps FANT lifetime, generation time and originates node ID, so each node will identify if the packet has been received before or not. When a node (s) needs to route data packets to a destination node (d), then the Route discovery stage is initiated to learn best ways to route the data packets.

When data packet needs to be routed, the node consults its pheromone table (PHT) if the route is within its neighbourhood it will choose the best pheromone value. The pheromone value (τ) denotes an artificial pheromone concentration value over that node which is modified whenever an ant transitions over it. (τ) represent the reversed distance between two nodes according to Equation 2 (Dorigo 2006).

$$\tau = \frac{1}{[(x_j - x_i)^2 + (y_j - y_i)^2]^{1/2}}$$

Equation 2: The Reversed Geometric distance between two points

Where:

X, Y are the coordinates of both nodes $i,\ j$

After the pheromone is calculated from the generic cost -distance- Three equations are used to manage the pheromone as follows: Pheromone update, Initial pheromone increase, and Path selection (Singh et al. 2014; Di Caro & Dorigo 1998a):

$$\tau_{ij}^{+} = \tau_{ij} + \varepsilon_x , x = 1,2,3$$

Equation 3: Initial Pheromone increase.

Where:

ε is the initial zone pheromone the value of this variable depends on the zone.

τ is the reversed distance between two nodes (Singh et al. 2014; Di Caro & Dorigo 1998a).

τ_{ij}^{+} is the pheromone increase (Singh et al. 2014; Di Caro & Dorigo 1998a).

As mentioned earlier, in the real world, the pheromone will start to evaporate if it was not enhanced by other ants using the same path, the same thing applies to the pheromone table; all the values will be updated using the evaporation formula in Equation 4 (Okdem & Karaboga 2009; Yoshikawa & Otani 2010).

$$\tau_{i,j}(t+1) = (1-\rho).\tau_{i,j}(t) + \rho.\tau_{i,j}$$

Equation 4: pheromone update.

Where: $0 < \rho < 1$ is the evaporation rate.

In the case of multiple routes, to simulate the exploratory behaviour of ants the control packet known as “artificial ant”, makes a stochastic decision based on probabilities of the next hop. The probability of an ant moving to node (j) from node

(i) towards node (d), where (N_i) represents a set of neighbours, is calculated by the Equation 5 (Dorigo 2006; Di Caro et al. 2008a).

$$P_{nd} = \frac{(\tau_{nd}^{i})^{\beta}}{\sum_{m \in N_i} (\tau_{md}^{i})^{\alpha}}$$

Equation 5: Path Selection Probability Equation.

Where:

$(\tau_{nd}^{i})^{\beta}$: is the amount of the deposited pheromone.

$\mathrm{N_{id}}$: is the set of neighbours of i over which a path to destination (d) is known.

$(\tau_{md}^{i})^{\alpha}$: is the pheromone value from neighbour m to i

Following this stage, if data packet need to be transmitted and a route to the destination node was not found, the second stage of the reactive phase will be established, which is the Route Discovery Phase. The next section will describe ANTMANET Route Discovery Stage.

4.2.2 Route Discovery Stage

Route discovery stage is invoked when a source (s) wants to forward a data packet to a destination (d), and there was no valid information for any path stored in the NNT and PHT. The source node then creates a FANT and propagates it in the network. The task of the FANT is to search for routes from the source (s) to the destination (d) and to update pheromone for the paths that the FANT has followed.

Once paths to the specific destination are discovered, the best path is maintained and used to send data. This approach reduces control overhead and maximises routing performance focusing on the routing metric estimator "pheromone" The pheromone is adjusted in such a way that the best possible utility link is chosen for delivering the packets to the destination.

The series of actions that will be taken in this stage are summarised as follows:

- If the pheromone value (τ_{nd}^{i}) is not defined for all neighbours (n) of the node (i), then node i multicasts FANTs to those nodes in which the destination might be located in their local zones.
- At the next node (j), *FANT* is multicasted again if there were no value of the pheromone (τ_{nd}^{j}) *for* all neighbours (n) of (j).
- FANT will be routed as a data packet, if there are pheromone values at (j) for destination (d).
- All FANTs keeps a $memory\ list\ p$ of nodes it has visited i.e., $(1{:}\ 2{:}\ 3{:}\ {::::}\ ;\ nd)$.

A flowchart summarising the route discovery stage is shown in Figure 25.

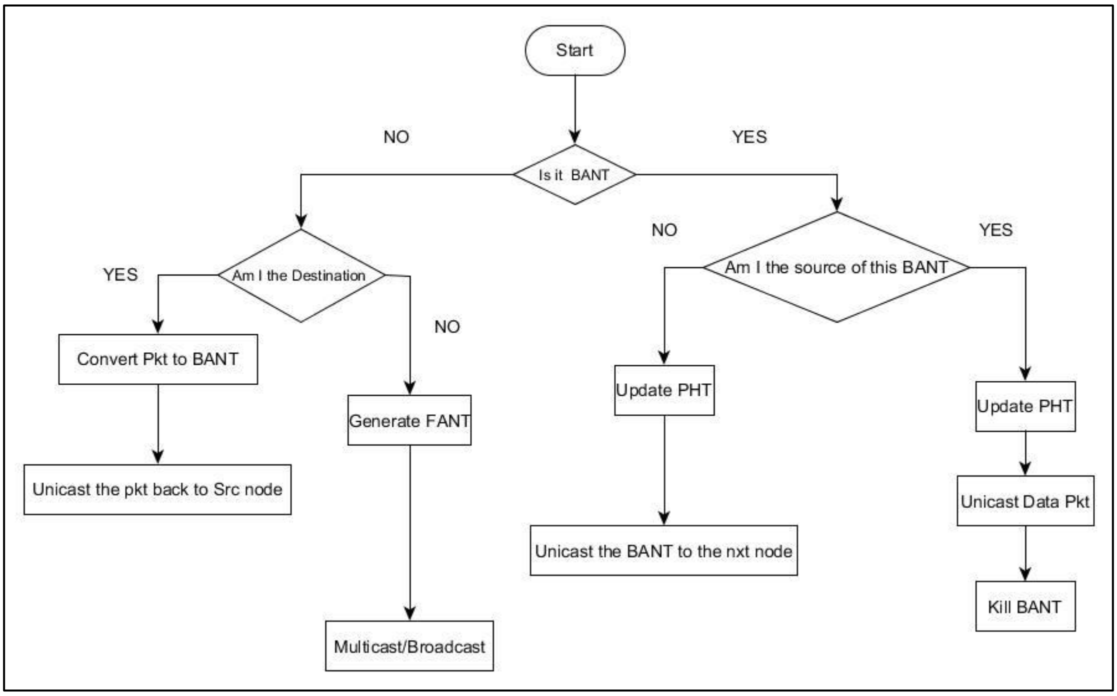


Figure 25: ANTMANET Route Discovery Stage – Flowchart

As previously mentioned, the scenario of the network application in which this protocol would be used for is a high mobility scenario where all nodes are non-stationary and change their location rapidly, this will cause some routes to be frequently invalid, which requires the use of a process to maintain the validity of the routing information in the routing table. The next section discusses the route maintenance stage that forms the proactive phase of ANTMANET.

4.2.3 Route Maintenance Stage

The Route Maintenance stage represents the proactive phase in ANTMANET. In this stage, all the routes kept in the PHT and NNT are updated and maintained fresh without a new demand for propagating FANT packets. Instead, it uses ANT-HELLO packets, which is the third type of the control packets ANTMANET uses. The reactive phase of the proposed protocol only uses this packet type. The significant difference between the ANT-HELLO and any regular HELLO packet is that the ANT-HELLO packet header contains the ($p\ memory\ list$) of the paths to the destination (d) (for more details refer to section 4.4).

A random source node S sends out an ANT-HELLO packet to randomly chosen destination node, e.g. every (t) minutes using (t_ hello) timer (for more details refer to section 4.5.2). To limit the network overhead, a maximum number (MAX) is used to limit the number of multicasts.

Generally, for a node to generate its ANT-HELLO, it needs to consult its PHT to randomly pick several destinations with the best-estimated pheromone values assigned to each one of them. Then an ANT-HELLO packet will be generated and forwarded over the same routes. Because of a successful ANT-HELLO reaching (d), a regular ACK packet will be sent back to the S. The ACK packet is a signal sent by the destination back to the source after the receipt of ANT-packet. If the source node did not receive an ACK for a certain amount of time by default it is 40 seconds, this will mean that a transmission failure has occurred (for more details refer to section 4.5.2). Transmission can fail because the PHT might contain values of paths that no longer exist. This can be due to a few possibilities, such as node's high mobility or link failure. In this case, the source node will remove the failed route from the PHT and NNT.

There is a special case, which could occur during the Maintenance Phase, when a node (i) receives an ANT-HELLO packet from a new neighbour (k), node (i) will update its PHT, GEO and NNT tables and create a new entry for (k). Figure 26 shows a flowchart summarising the mechanics of the route maintenance stage.

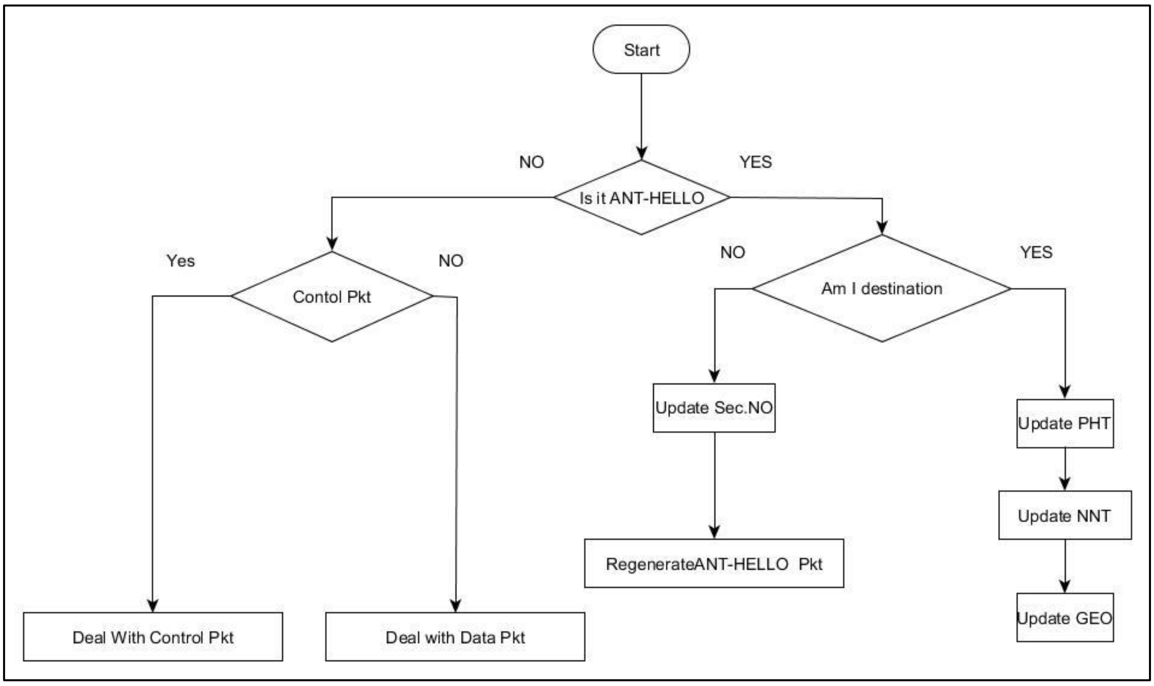


Figure 26: ANTMANET Proactive Phase - Flowchart.

The next section will discuss the node structure of ANTMANET in terms of routing tables used and the network structure during the different phases of ANTMANET.

4.3. **Routing Structure**

Routing structure is a method used to determine the protocol's forwarding techniques, for instance, flat structure uses a greedy forwarding technique.

ANTMANET routing structure can be defined as hybrid as it is neither flat nor geo- hierarchical, because, on one hand, it has no dominant nodes and all devices have the same capabilities and the same basic purpose, to sensing and routing. On the other hand, to minimise the use of control packet, which will reduce the network overhead, it utilises the node's coordinates to divide the forwarding regions into three local zones. It is true that in recent developments, position-based routing protocols exhibit better scalability, performance and robustness against frequent topological changes, but there are three main packet-forwarding strategies used for position-based protocols: *greedy* *forwarding, restricted directional flooding and hierarchical* approaches (Mahajan & Bang 2015). These forwarding methods are known frequently as broadcast control packets which cause higher network overhead. In preference to this ANTMAMET's local zone technique will allow the protocol to multicast control packets to the first zones, which reduces the route establishment time and reasonably maintains the routing table's size that guarantees faster convergence.

If a source node (s) demands to communicate with a destination node (d), forwarding data packets will not be feasible unless a route between (s) and (d) is established. This requires the exchange of fewer control packets.

ANTMANET uses the position information to keep track of coordinates of each node within its GEO table, which is coherent with PHT and NNT. These coordinates can be obtained using different methods such as Geodetic Datum (datum) (Santos et al. 2015)., which is a reference system by which measurements and coordinates are made. The earth is not flat, therefore many different methods for overlaying a grid and measuring system (datum) on its surface are devised for coordinates. These coordinates are usually obtained using the Global Positioning System (GPS) (Bruno et al. 2015). Generally, a GPS position is defined by a vector (X, Y, Z, t), where X, Y and Z are three-dimensional coordinates, and t represents the time when these coordinators were obtained (Misra, P. and Enge 2006). For simplicity, an assumption is made, that the z coordinate is always zero implying that the nodes are in the same plane, while the time coordinate (t) is maintained separately in the NNT.

4.3.1 Local Zone

ANTMANET assumes that the node is aware of its location and its neighbours' locations. Typically, a location service is responsible for this task. Each node will divide its neighbours into three local zones, the neighbour is any node, which can communicate with the source node located on its north direction as shown in Figure 27.

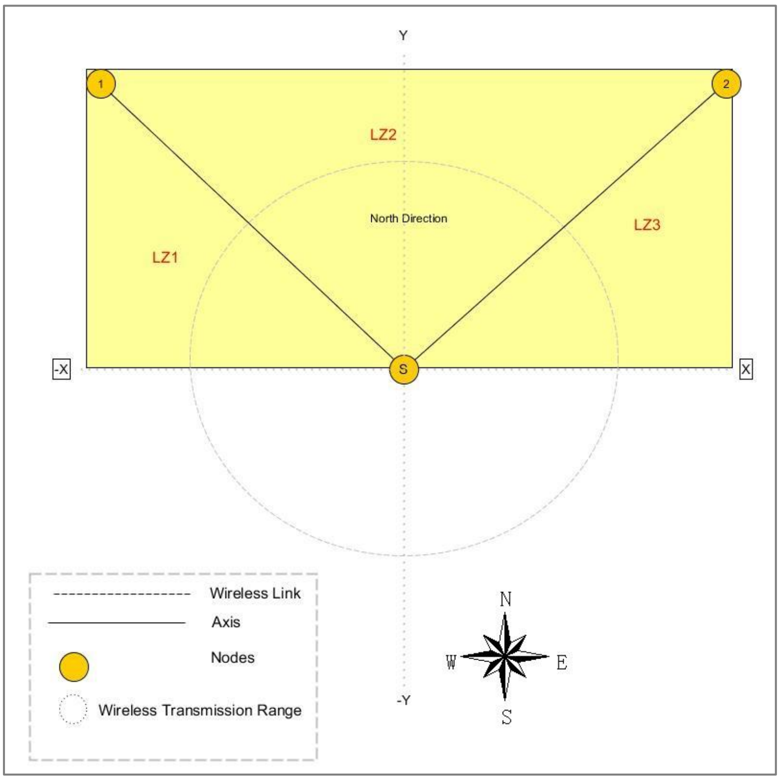


Figure 27: North Direction.

The zones are then calculated by the means of a slope angle as shown in Figure 28 (refer to Equation 1). A node will belong to the first zone (LZ1) if it is located within $\theta \leq 45$ from the source node (s), in the same way it will belong to the second zone (LZ2) if it is located between 45 $<\theta<135$, and at last it will belong to the third zone (LZ3) if it is between $135 \leq \theta \leq 180$.

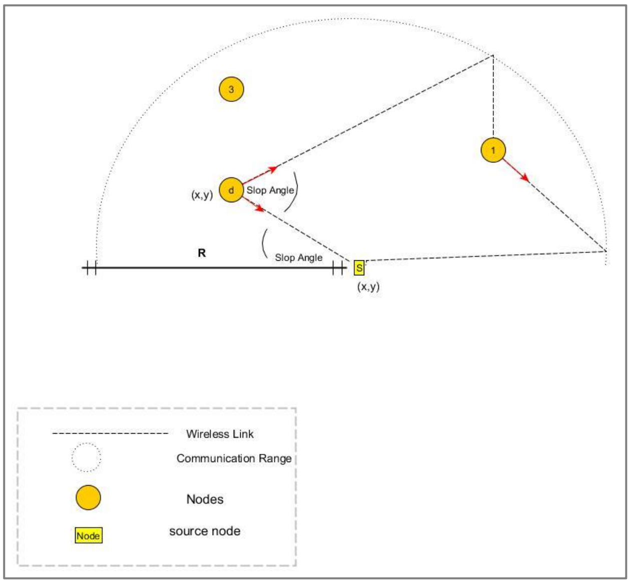


Figure 28: Slope Angle

The main reasons to only include the north side neighbours is, firstly, to keep the size of the routing tables as realistic as possible especially in a large network. Additionally, ACO algorithms such as ANTHOCNET have a collaborative behaviour (Di Caro, Ducatelle & Gambardella 2004b). Therefore, nodes do not need to know everything about the network, they only need to know a portion of the topology and collaborate with other nodes to make routing decisions, for example, the car's front light covering the area in front of the car, which is enough for the car to be aware of what is on the road. Secondly the approach guarantees faster convergence, as dividing the north side into smaller zones will speed up the reactive phase as well as the proactive one.

ANTMANET node structure as shown in Figure 21 has three routing tables: North Neighbour table (NNT), GEO table, Pheromone table (PHT) and ACO parameters (refer to section 4.2.1). These impact the routing decisions in ANTMANET. The next section describes the pheromone table.

4.3.2 Pheromone Table

Pheromone table (PHT) is organised similarly to the routing tables in distance-vector algorithms, but its entries $\tau_{nd}\ of\ n \in N_i$ are distances. The PHT entries abide by the same general concept attributed to pheromone variables in ACO. These are a measurement, of the goodness of forwarding packets towards destinations (d) through neighbouring nodes. Goodness is expressed as the inverse of a cost, which is expressed in Equation 2.

The τ_{ij} values are in the interval [0, 1] and sum up to 1 along each destination column as shown in Equation 6 (Singh et al. 2014; Di Caro & Dorigo 1998a):

$$\sum_{n \in N_k} \tau_{nd} = 1, d \in [1, N], \qquad N_k = \{\ neighbours(k)\}$$

Equation 6: Pheromone Matrix

The idea here, as in all ACO algorithms, is to learn an effective local decision policy by the continuance update of the pheromone values to obtain an effective routing policy. The pheromone table, in conjunction with information stored in the GEO and the NNT, forms the routing decision.

Along with the pheromone table, the nodes structure contains statistical parameters, which does not directly interfere with the routing decisions, instead it directly affects the ACO meta-heuristic (refer to chapter 6), the next section explains these parameters.

4.3.3 Statistical parametric

Statistical parametric model $\mathcal{M}$ is a vector that holds predefined constant values of the ACO parameters (α, β, ρ)[1], where β and α represent respectively the respective weights of pheromone trail and the distance between two nodes, which are network dependent, meaning they need altering in response to topology change or network status; which in turn improves ANTMANET routing decisions. Referring to Equation 5, if *(α = 0),* the selection will be based only on visibility, in this case, the heuristic function is turned into a stochastic search algorithm, meaning that only the nearest node are likely to be selected (Nallaperuma et al. 2015). On the other hand, if *(β = 0),* the selection will be based on the pheromone amplification, in this case the selection will be based only on the inactive situation of suboptimal routes. If *(α = 1),* no new solutions are generated, instead the same path using the old solution will be used (Stützle et al. 2012; Gholami & Mahjoob 2007). Therefore, these parameters can be optimised because they contribute weight to the function. ρ is a trail decay parameter that is problem independent but affects the quality of the solution, allowing the algorithm to ‘‘forget’’ bad decisions which have been previously taken. When a path is not being used for a certain amount of time, its associated pheromone value decreases exponentially by ρ (Hao et al. 2006).

These parameters define the balance between the exploration and exploitation nature of the ACO algorithm, which means the higher the exploration, the newer routes will be discovered that are different from the ones already known. On the other hand, emphasis on higher exploitation nature means improved solutions (Dorigo et al. 1999; Di Caro & Vasilakos 2000).

[1] ACO parameters are discussed in more details in chapter 6

The next section is discussing the different types of control packets and the casting methods used by ANTMANET to achieve better routing.

4.4. **Control Packets and Casting Methods**

In any routing protocol, the final quality of the routing policy critically depends on characteristics of the information maintained at the nodes. Different types of control packets obtain that information. A control packet is a formatted unit of data propagated in a network. A control packet consists of control information and user data, which is also known as the payload. Control information provides data for delivering the payload (Kim et al. 2000). ANTMANET has three main control packets: Forward-Ants, Backward-Ants, and ANT-HELLO packets.

Forward-Ant (FANT): FANT is ANTMANET control packet type 1, which is used in the reactive phase to obtain routing information. These represents the route request packet in traditional protocols, the difference is that FANTs build a solution by choosing probabilistically the next node to move to among those in the neighbourhood, this aid to avoid formation of routing loops. All the generated FANTs have the same characteristics; they differ only for the assigned source and destination and the casting method. Whenever a FANT reaches a node from one of its neighbours, the identifier of the neighbour, the sequence number of the packet and the identifier of the destination will be stored. Each node can see the same FANT from different adjacent nodes, therefore the sequence number is examined and the repeated ones are being terminated. When a FANT reaches the destination, it is dismissed and a reply will be generated in the form of a Backward-Ant (BANT) holding the same origin FANT's sequence number, which will be sent back to the source. Figure 29 graphically summarises the structures of the Reactive Forward Ant used by ANTMANET:

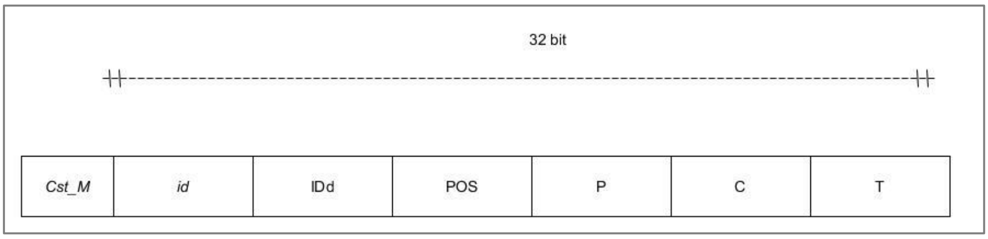


Figure 29: Forward Ant Packet.

Where:

Cst_M: is the casting method identifier.

id: The sequence number of the FANT.

ID_d: The identifier of the destination node.

Pos_j : The position of previous node$j$.

Ƥ: The memory list of visited nodes.

Ḉ: the list of the costs of traversed paths.

Ƭ: the packet generating time.

Backward-Ant (BANT): BANT is an ANTMANET control packet type 2, which represents the route reply in the traditional protocols. The use of the reversed memory list p allows the BANT to retrace the path that the FANT followed while searching for the destination node. Figure 30 graphically summarizes the structures of the Backward-Ant used by ANTMANET:

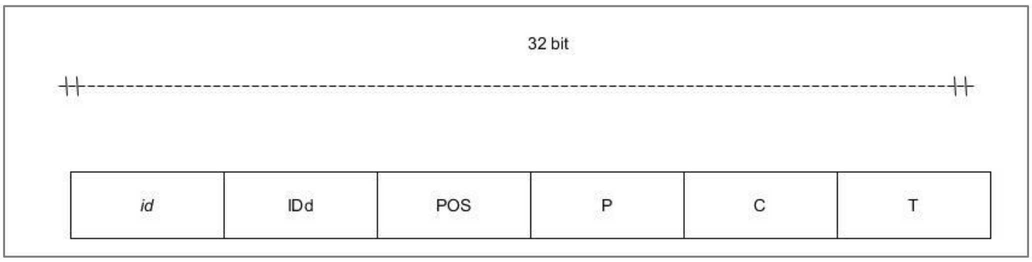


Figure 30: Backward Ant Packet

Where:

id: The sequence number of the ant.

ID_d: The identifier of the destination node.

Pos_j : The position of previous node$j$.

Ƥ: The memory list of visited nodes.

Ć: the list of the costs of traversed paths.

Ť: the packet generation time.

ANT-Hello packet: ANT-Hello packet is an ANTMANET control packet type 3. These packets are sent periodically by all nodes in order to establish and maintain neighbour relationships. In addition, ANT-Hello Packets are multicasted to enable dynamic discovery of neighbouring nodes. Along with the destination and the source addresses, the ANT-HELLO packets have unique parameters such as, Memory P list, Packet ID, Packet Generating time and Router-Dead-Interval. These parameters are included in ANT-Hello packet headers as illustrated in Figure 31.

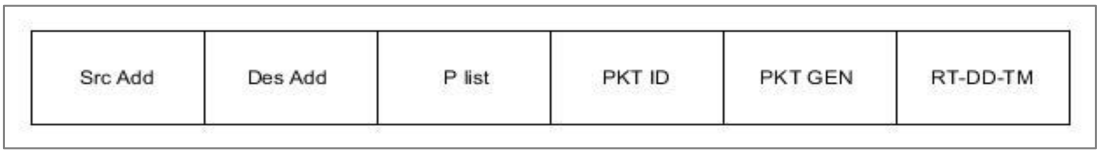


Figure 31: ANT-HELLO Packet.

Where:

$Src\ Add$: is the source node Address.

$Des\ Add$: is the destination Address.

$P\ List$: is the ANTMANET memory list.

$PKT\ ID$: is the sequence number of the packet.

$PKET\ GEN$: is the generation time of the packet.

$RT-$DD-TM: is the number of seconds before declaring a link failure.

4.4.1 Broadcasting in the Initial Stage

During the Initial Stage, nodes have no routing information about the network, they are only aware of their location and ID. A broadcasting method is used to build the first node structure to aggregate routing information. Broadcasting means the message will be sent to every node that is located within the transmission range of the source node. The routing tables will keep only the nodes located on the north side of the source node as shown in Figure 32. To avoid the extensive use of the FANTs the broadcast number is limited to a maximum value, which is 5 by default[2] (Di Caro, Ducatelle & Gambardella 2004a).

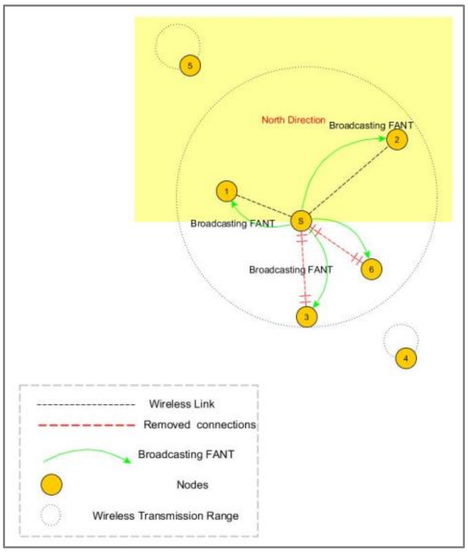


Figure 32: Initial Stage Broadcasting Method.

After the initial stage is finished, the nodes will stay in an "*Idle*" mode until an event occurs, events can be defined as an arrival of the type of packet or a timer interrupt (refer to section 4.5). This will trigger either Route Discovery stage or the Route Maintenance stage, which has different casting methods. The next section will explain the multicasting method used by ANTMANET during its reactive and proactive phases.

[2] This value can be changed; the default settings come from the original algorithm.

4.4.2 Multicasting in the Reactive/Proactive Phase

When an event of ANT-Hello packet or forwarding data, packet occurs with no valid information of a route to the destination, the node will aggregate Multicast FANT. Multicasting means that the packet will be sent to several nodes simultaneously to find a route to the specific destination. The number of multicasts was configured to a maximum value, which by default 4 packets per second (t_hello).

The selection of next hop multicasting node list is set with consultation to the PHT by the means of the best pheromone value, all nodes that have equal pheromone value are considered in this set. Call this set B. If B contains any single-hop neighbours, remove double-hop neighbours from B. then a node, X, is then chosen at random from B. If X is an adjacent neighbour, the packet is forwarded to X, otherwise -since X may be reachable from any number of nearby neighbours- the best neighbour is chosen and the packet is forwarded to that node. If the transmission fails, the chosen node is removed from consideration and the packet is reprocessed, starting with the original B.

In case the multicasted FANT found a route or routes to the destination, then the data packet will be forwarded on the best-found route. The following flowchart summarises the process explained above.

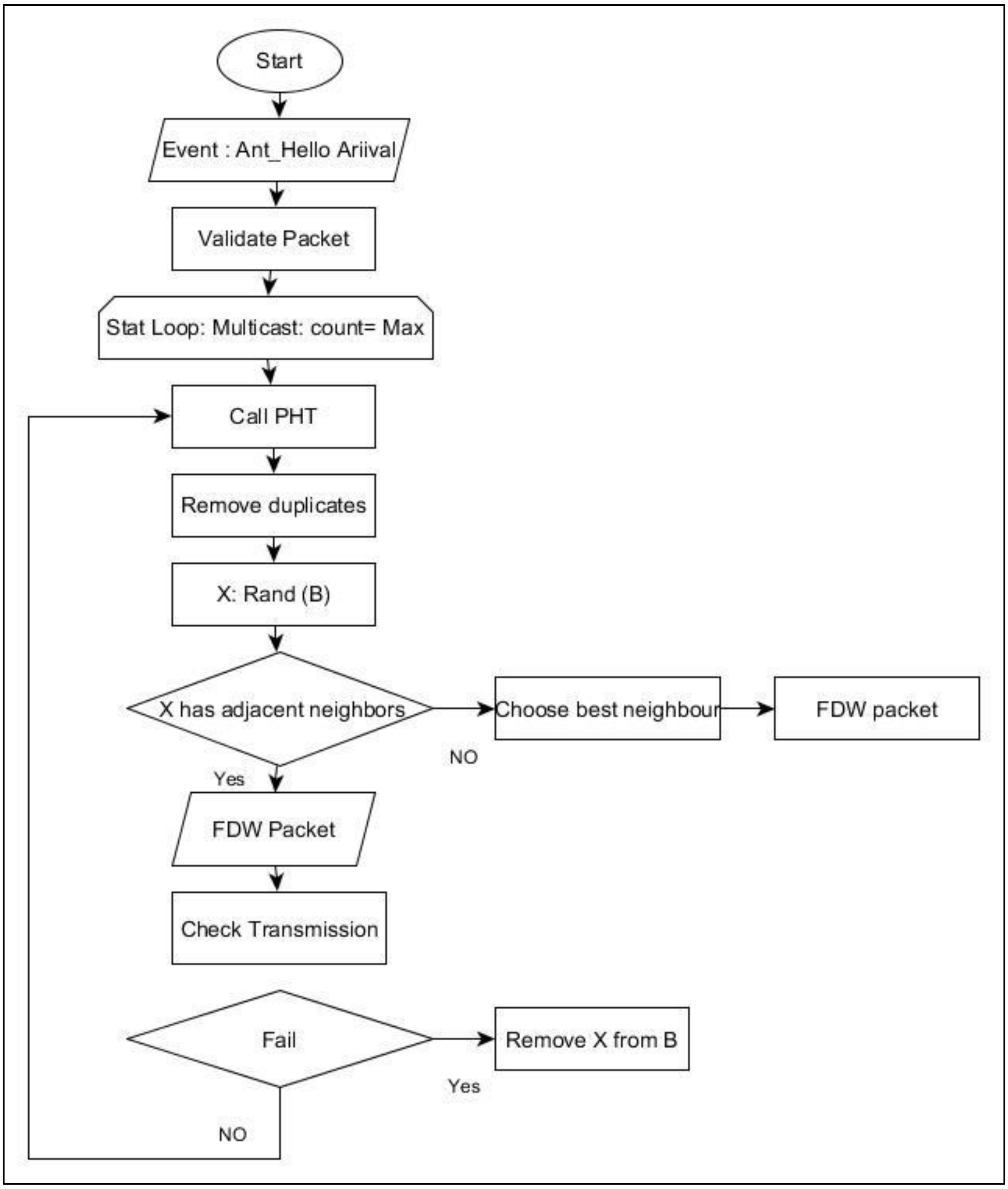


Figure 33: Multicasting in the Reactive/Proactive Phase.

The following section discusses the forwarding of data packet in ANTMANET.

4.4.3 Forwarding data packets:

Each node maintains its current neighbours' identities and geographic positions in its NNT. When a node needs to forward a data packet toward a destination (d), the node consults its NNT and PHT and chooses the neighbour closest to (d), then it unicasts the data packet to that neighbour. Consequently, the chosen neighbour itself applies the same forwarding method and this unicasting process will continue until the packet has reached the destination.

The next section describes the different events and timers ANTMANET utilises to perform the routing task.

4.5. Events and Timers

The following section contains detailed descriptions of the various events that can take place in the system. They can be divided into three types of event as follows:

- The arrival of data/control packets.
- The detection of link failure.
- Different timer for each event.

4.5.1 Packet Arrival Event

The following summarises the variety of packet arrival event that might occur at some point of the network lifetime.

- Data Packet Arrival*:* The protocol will unicast this packet to the destination in case there is valid route to the destination. If there was no valid routes to the destination a route discovery stage will be triggered where FANTs will be multicasted.
- Control Packet Arrival: In this case the algorithm will vary depending on which type of control packet it is.
    - *Forward-Ant (FANT):* at the arrival of a FANT, Assuming that the node was not the destination in this case the node needs to update its routing tables

and regenerate new FANT with the same identifier to distinguish whether it was broadcast or multicast. If the assumption was false and the node was the destination then this FANT will be changed into a BANT with reversed *P* list.

- *Backward-Ant (BANT):* when BANT arrives at a node, it unicast the same BANT to the first neighbour node exists in the reversed P memory list generated by the origin FANT.

The following Figure summarises ANTMANET packet handling process.

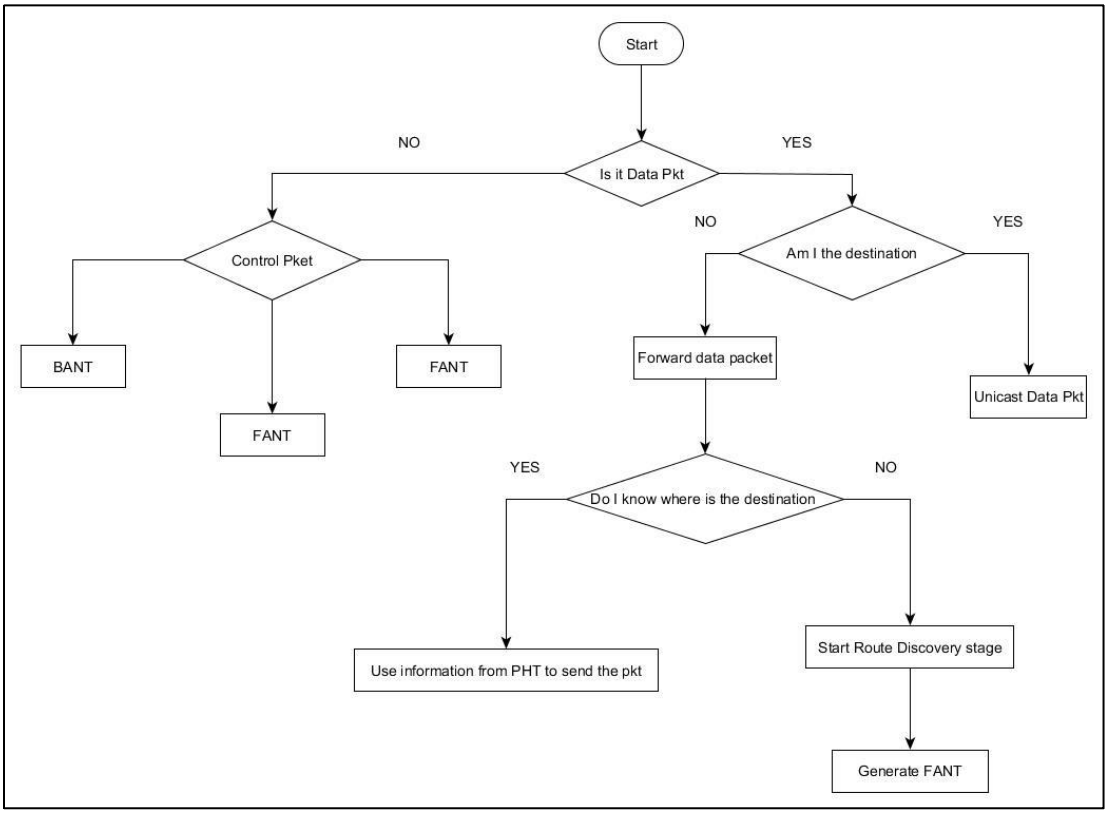


Figure 34: Packet Handling

Along with the packet handling methods NATMANET utilises several timers to execute different stages on different events, these timers are discussed in the following section.

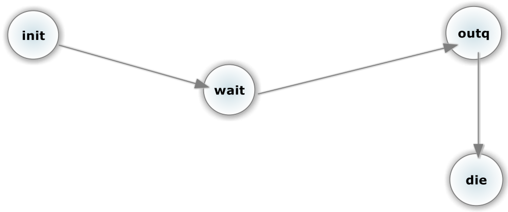
4.5.2 Timers Events

To plan delayed actions the node schedules the Event timers as follows:

- t_hello (200 second): This timer is scheduled from the actual hardware power up for as long as the node operates. The arriving of such timer event will provoke the node to send a Hello message in which its best PHT information is included.
- $RT-$DD-TM: This is the failed route timer, which defines the number of seconds before a node declares a link failure for a route and deletes it from its routing tables.
- Proactive-Ant timer (120 second): This time is scheduled at regular intervals from the moment the session starts until it ends; the arrival of this timer provokes the node to generate FANT .
- FANT timer (90 second): This timer is scheduled to be sent after a Multicast FANT has been sent. The arrival of this timer provokes the node to check if it received BANT; if NOT it will regenerate a reactive FANT in case the maximum number of transmissions has not yet been reached; in which case it drops the queued data.
- Evaporation timer (180 second): This is the evaporation process that causes the amount of pheromone deposited in each link to decrease with time.

For better description of the Events actions and timers, the Process Model Methodology (PMM) is highlighted in Figure 35.

state	Event Response Table Development		
	event	action	state
Ideal	packet arrival	select destination	destination selected
destination selected	launch forward ant	select link	put ant on link queue
wait on data link queue	cross link	launch backward ant	die



event name	Enumeration of Events	
	event description	interrupt
packet arrival	packet arrival from src	stream
launch forward ant	generate forward ant when current node = destination and it is not in the routing table	stream
cross link	cross link	stramt

state	process model methodology			
	event	condition	action	final state
Ideal	packet arrival	$t \bmod \Delta t = 0$	launch fant	wait in queue
wait on queue	queue cleaned	current node = next hop	cross link	out queue
out queue	arrival at destination	current node = destination	lanch backward ant	die

Figure 35: Process Model Methodology (PMM)

4.6. **Protocol Classification**

ANTMANET is an Ant Colony Optimisation based protocol and as it is known, all the ACO algorithms are stochastic algorithms, which essentially means that these algorithms can establish multiple paths between the source and destination. In addition, the distributed nature of routing is well matched by the multi-agent nature of ACO algorithms.

ANTMANET is a multipath routing protocol that utilises its positional information to perform faster routing. At the same time utilising the ACO helps ANTMANET to eliminate the disadvantages caused by using the greedy forwarding, methods used by most position based routing protocols. In addition, the design of ANTMANET minimises the draw of using the ACO algorithm such as higher control overhead and delay. An evaluation study is carried out in the following chapter to evaluate the performance of the proposed routing protocol.

4.7. **Summary**

This chapter has illustrated the proposed protocol and its functionality in details. ANTMANET is a novel ACO based routing protocol designed to handle high mobility in MANETs, it is a hybrid protocol that wisely combines both reactive and proactive phases. The protocol triggers a reactive phase only if needed- no known routes to destination- the protocol reduces the search area using a unique technique- Local Zone Technique- which reduces the use of control packets and provides scalability. It supports the proactive routing within the zone and reactive routing between routes.

The protocol has two new tables added to the original ACO algorithm to perform the Local Zone Technique: The North Neighbour table and Geo table; both impacting the routing decision and in the pheromone table only good routes are kept, which are chosen probabilistically based on the ACO algorithm.

The following chapter will present an extensive comparison study to evaluate the performance of the proposed protocol.

Chapter 5. Standard Protocols Comparison

Overview

Utilising simulation software packages is beneficial to the testing of any new design. Simulation can save time, energy, and money as there is no need to order equipment and connect it together to set a scenario. The main contribution in this chapter is the development of a strategy to represent the collected MANET performance metrics against the network context. Two comparison studies were conducted to compare the proposed protocol against standard MANET protocols and ACO based protocols.

The remainder of this chapter is organised as follows, 5.2 illustrates detailed experimental design, followed by section 5.3 that presents details on the mobility experiment. Followed by a chapter summary in section 5.4

5.1. Introduction

Simulation experiments are widely used to evaluate MANET routing protocols. These experiments must model the network topology, network traffic, routing methodology and other network attributes. In addition, the wireless and mobile nature of MANETs requires consideration of node mobility, the radio frequency channel, terrain, antenna properties, and battery characteristics.

There is no doubt about how important it is to establish a testbed for a system to measure its reliability in real the world, but this step would come after a successful software implementation. Utilising simulation software packages is valuable to the evaluation process of any new design. Simulation software packages save time and reduce the implementation cost compared to setting up a real network testbed. They are required to realistically model and emulate the network characteristics at the end of each simulation, statistics and network performance measurements are available for collection for evaluation and analysis. For example, QualNet is a simulation package that simulates any communication system (i.e. wired or wireless networks) in a short time with guaranteed accurate statistics to help with the evaluation of the performance of any proposed system (Jaikaeo & Shen 2005).

In this research, the experiment system is designed carefully to evaluate the proposed protocol performance through several network conditions. This is achieved by varying several factors to emulate realistic situations. The experimental system plan is shown in Figure 36.

1. The first condition is implemented to evaluate the effect of nodes velocity by varying the node's speed. To guarantee different levels of route convergence, several pause times is configured.
2. The second condition is implemented to evaluate the performance of the proposed protocol in stressed network conditions such as high network load by varying the number of packets in each case of the above factors.

The previously mentioned factors are organised in two experiments as follows:

1. Mobility experiments, which evaluates the effect of the different pause times to each node's speed that is along with varying the number of packets sent by the application per second. This experiment generates 180 single simulations per tested protocol.
2. ACO based protocol comparison experiment. The main aim of this experiment is to benchmark the proposed protocol against another ACO based routing protocol. This experiment will be discussed in details in Chapter 6.

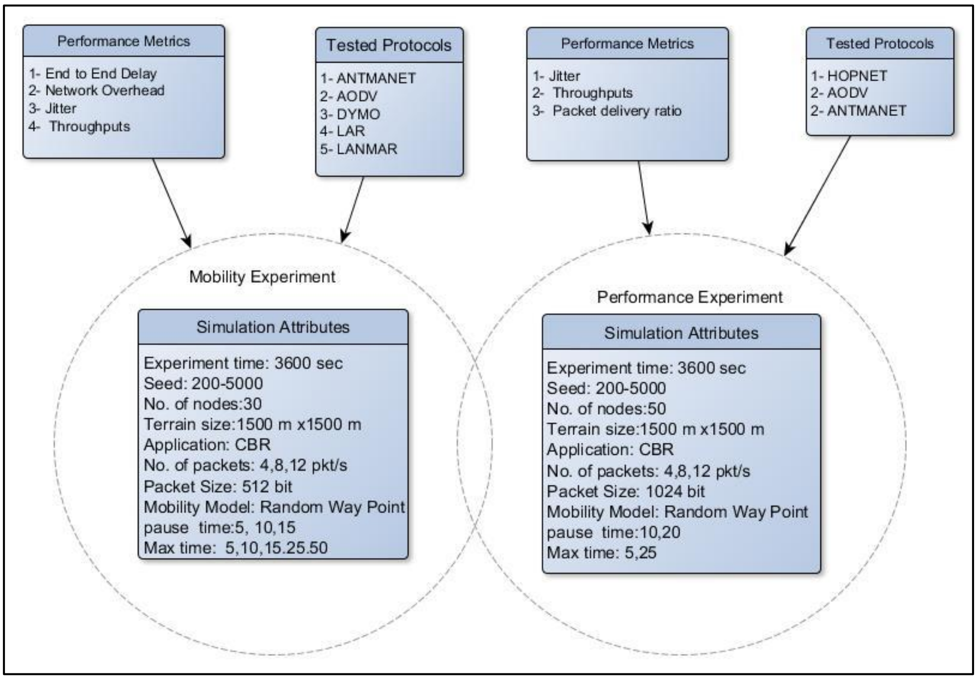


Figure 36: Experiment System plane

At the end of each simulation, several performance measurements are collected and analysed to evaluate the performance of the proposed protocol. The following section discusses these performance metrics that are used in all experiments.

The following section presents mobility level experimental results for three different network load scenarios in a MANET environment.

5.2. **Mobility Experiment**

Node mobility, coupled with physical layer characteristics, determines the status of link connections. Link connectivity is an important factor that is impacting the comparative performance of any routing protocols. From the standpoint of the network layer, changes in link connectivity invoke routing events such as routing maintenance and routing discovery phase, which hugely impact the performance metrics, for example, the throughputs and the network control overhead.

Traditionally, simulation studies of MANET routing protocols have explicitly modelled mobility. Mobility models can be classified as independent models or group models. Independent mobility models assign movement vectors independently to nodes without considering the movement of other nodes in the system (Navidi & Camp 2004). Group mobility models consider correlated movements of nodes; therefore the movement vector is not independent of the group of members (Kaveh & Khayatazad 2013; Bettstetter et al. 2003)

In this research, experiments are carried out using the Random Waypoint Trajectory model (RWM) (Hua & Haas 2015; Navidi & Camp 2004), which is an independent mobility model. RWM is chosen as the trajectory in this experiment for being the most realistic mobility model that can capture the physical movement of floating sensor nodes with water waves and currents.

This experiment considers a network of 30 nodes placed randomly within the area of $1500\,(m^2)$.[3] The data application used is the Constant Bit Rate (CBR) to establish data sessions among a chosen source-destination pairs (SDPs). Three different network loads were utilised to examine the proposed protocol performance in a normal, medium, and high network loads, this is done by varying the number of packets sent per second to 4, 8 and 12 CBR packets. For example, 2 SDPs amongst 30 nodes are engaged in generating the traffic. However, during the data forwarding process, all the 30 nodes including the SDPs will be involved in generating background traffic to provide the necessary support for routing and data forwarding over the on-going communication session.

To emulate the mobility model to cause route convergence, fifteen levels of node mobility, those performed by varying two key factors; node speed (5, 10, 15, 25,

[3] These choices were based on the testbed in reference (Anon 2012)

50 m/s) and pause times (5, 10, 15 s). The following Figure 37 demonstrates a flowchart of the mobility levels.

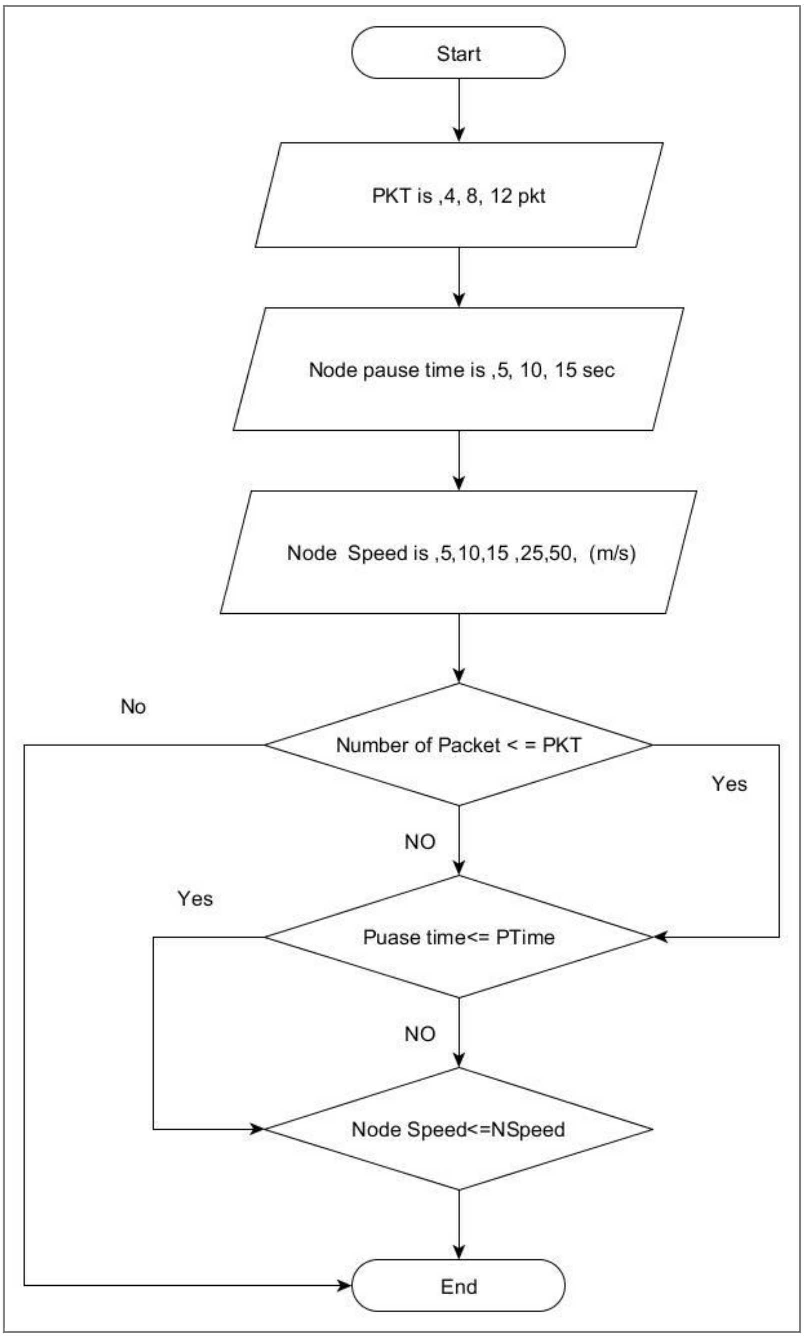


Figure 37: Mobility experiment Plane

The following Table 15 summarises the mobility experiment simulation Attributes.

Table 15: Mobility Scenario Attributes.

Parameters	Value
Experiment time	3 H
Number of nodes	30
Terrain size	1500 m x1500 m
Application	CBR
Packet Size	512 bit
Number of packets (packet/s)	4,10,15
Mobility Model	Random Way Point
Pause time (sec)	10, 20,30
Speed (m/sec)	10,50,100,150,200
Propagation model	Free Space
Channel frequency	2.4 GHz
Radio type	Accumulated noise model
Network protocol	IPv4

The following Figure 38, highlights a run-time scenario sample of the undertaken simulations. In order to perform the mobility experiment 2400 simulation has been conducted to achieve the simulation conditions variation represented in Figure 37.

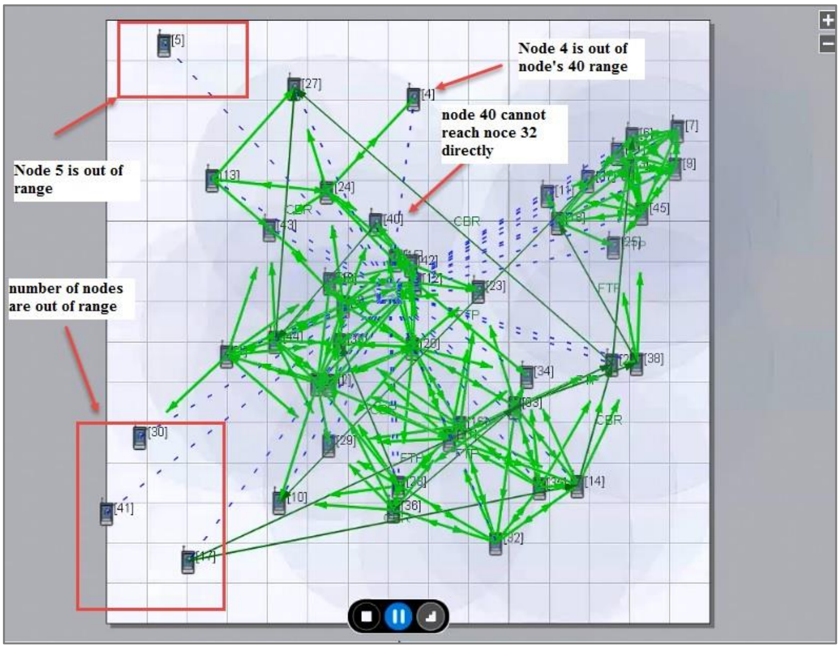


Figure 38: Capture of the QualNet Simulation.

The following section presents an analytical discussion of the collected statistics.

5.3. Results and analysis of the Mobility Experiments

This section presents the main effects and the interaction of each factor in the experiment as shown in Figure 36. For brevity and convenience, each factor is denoted in Table 16.

As mentioned earlier the node mobility is modelled using the (RWM), which is widely used in MANET simulations. In this type of mobility models, nodes move at some speed uniformly and distributed in [MIN SPEED, MAX SPEED]. Each node begins the simulation by moving towards a randomly chosen destination. Whenever a node reaches a destination, it rests for a pause time. It then chooses a new destination and moves towards it again. This process is repeated until the end of simulation time.

Table 16: Mobility Experiment Levels

Mobility Experiment			
Factors	Level 1	Level 2	Level 3
Pause-time (sec)	5	10	15
Speed (m/s)	5,10,15,25,50		
Network size	30		
Routing protocols	ANTMANET, AODV, DYMO, LANMAR		

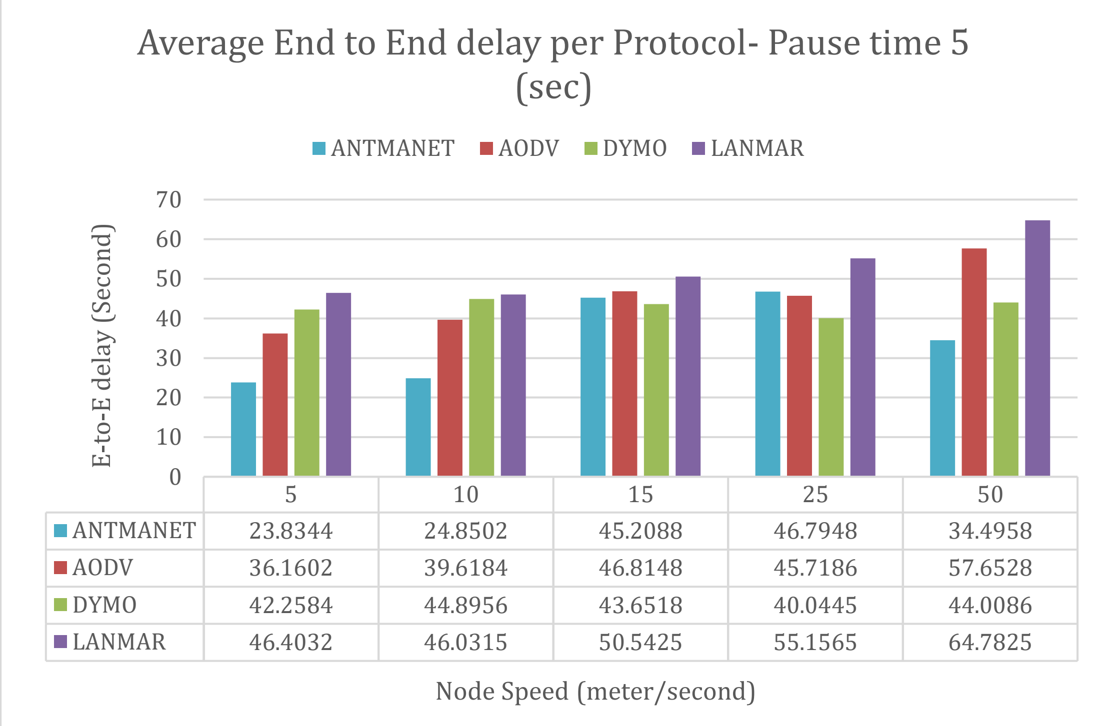
To explain the data collected of the simulation several bar graphs are charts used to visualise the performance of all tested protocols. Bar graphs are one type of data representation that is different from the histograms. These graphs have x-axis that represents a different category of data- in this case, the different five node speeds- (5, 10, 15, 25, 50 (m/sec)) and y-axis that is the numerical values which, represents the collected data- in this case the average performance metrics of ten different seed simulations. Each category displays the performance of the tested protocols within its conditions the results are represented in four different coloured bars that - denoted in the blue bar in all graphs is the proposed protocol ANTMANET- the legend of the graph is located on the top under the graph title. All results are presented in the following sections.

5.3.1 Scenario A.

The first condition examined is when the CBR application is generating traffic at the rate of 4 packets/sec, which is considered the most realistic packet rate of a MANET network because of the low bandwidth and the energy restraints of such network (Coiro et al. 2013; Reichenbach et al. 2005). This is tested in three different pause times, pause time is the period that the node will stay stationary after reaching the destination. The shorter pause time will cause more convergence in the routing matrix than the longer ones, meaning the network will stress more as the nodes will move rapidly in shorter pause time conditions. That will affect the network measurement metrics in question, which are the average E-to-E delay, throughputs, Jitter, and network overhead.

5.3.1.1 End to End delay

Overall the charts in Figure 39, Figure 40 and Figure 41 show that compared to other considered protocols, the average End to End delay of ANTMANET (denoted in blue) was lower in each category, which is an indication that the proposed protocol outperformed the standard protocols. It also shows that ANTMANET has a level of stability in its performance while varying node speeds and pause time, the bottleneck of ANTMANET performance along with all four tested protocols is at a very high speed represented in the last two speed categories.



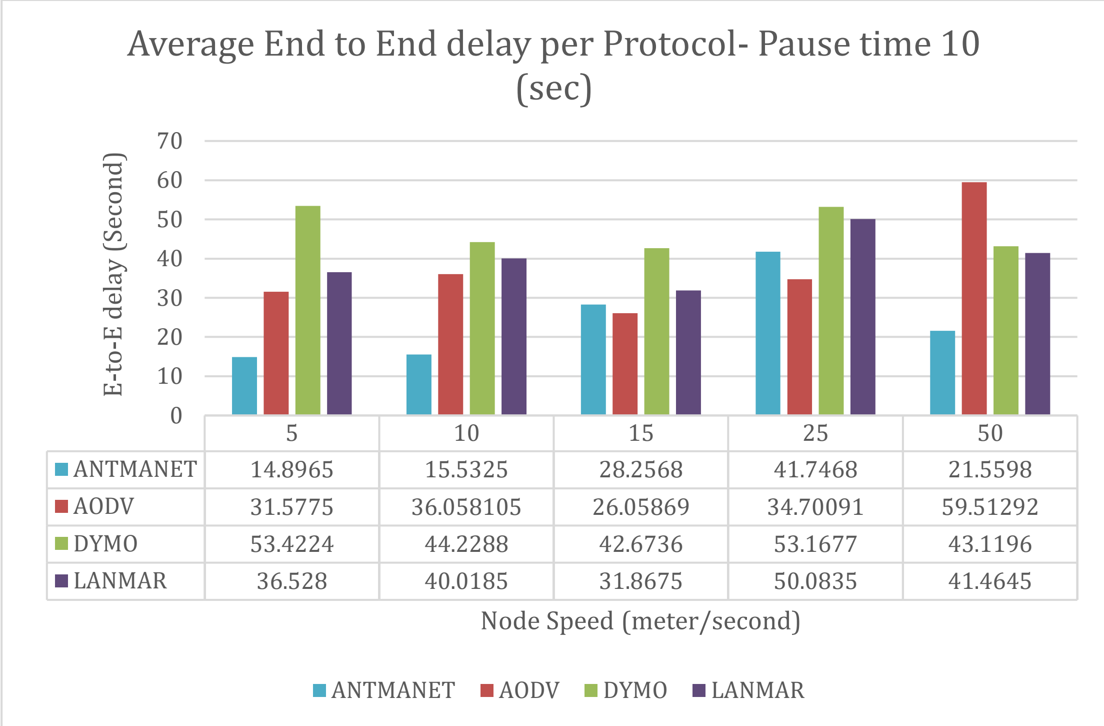
	5	10	15	25	50
ANTMANET	23.8344	24.8502	45.2088	46.7948	34.4958
AODV	36.1602	39.6184	46.8148	45.7186	57.6528
DYMO	42.2584	44.8956	43.6518	40.0445	44.0086
LANMAR	46.4032	46.0315	50.5425	55.1565	64.7825

Figure 39: Average End to End delay vs Node Speed per Protocol- Pause time 5 (sec)

In Figure 39, by looking at the bottom left-hand side of the graph- the vertical axes- the average E-to-E delay (in seconds) is showing that when the speed of the node – the horizontal axes- is 5 (m/sec) ANTMANET network has a delay that is less than the AODV network by 15%, and 10% less than LANMAR network as it has the worst performance of the tested protocol. The obvious reason for this poor performance of LANMAR is because it is using the Landscape details to calculate the routing cost. The nodes are moving with the lowest speed of the running simulation but they are still rapidly moving, giving a very short pause. The rapid movement has caused LANMAR's poor performance as the algorithm did not have enough time to converge. On the other hand, ANTMANET has performed better as the ACO algorithm is a distributed routing algorithm, therefore all nodes share their view of the network and each node needs to know at least one neighbour towards its destination, which helps in speeding up the convergence.

While the speed of the nodes increases, the E-to-E delay increases as well. As the nodes move faster they trigger more topology changes causing more packets to queue waiting for updated routing information to be routed to their destination. Moving to the second speed category, the delay in the ANTMANET network has increased by 20% as the node's speed increased to 15 (m/sec), and 20% when the speed increased again in the following category. Although 20% increase might seem high, but in fact, it is an acceptable result when comparing it with the 40% delay increase in the AODV network and a 65% delay increase in the LANMAR and DYMO networks. For the last speed category, the speed is high which scores 3 (km) per minute, therefore the increase in the delay shows a big jump in LANMAR and again ANTMANET kept its consistency with the lower E-to-E delay.

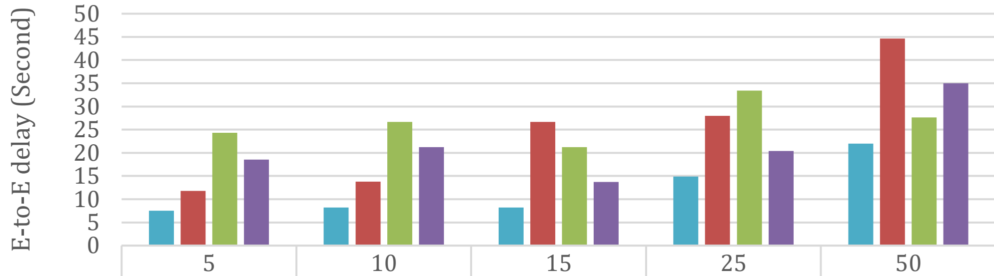
In terms of ANTMANET performance, it has the lowest delay in all categories as opposed to exactly 7.9 seconds' delay of the AODV network and a 5.6-second delay of DYMO, which is (40% and 55% delay increase respectively) in the first category. In addition, in the second category, the delay increased as expected in all protocols, where the delay increase of ANTMANET was 20% of the previous category opposed to 57% increase in the delay of the AODV network and 50% increase in the LANMAR network delay.



	5	10	15	25	50
ANTMANET	14.8965	15.5325	28.2568	41.7468	21.5598
AODV	31.5775	36.058105	26.05869	34.70091	59.51292
DYMO	53.4224	44.2288	42.6736	53.1677	43.1196
LANMAR	36.528	40.0185	31.8675	50.0835	41.4645

Figure 40: Average End to End delay vs Node Speed per Protocol- Pause time 10 (sec)

Figure 40 illustrates the amount of the average E-to-E delay while the pause time is increased to 10 Seconds, this increase has clearly affected the performance of all protocols and decreased the load on the network infrastructure. ANTMANET has the lowest average E-to-E delay that is illustrated in all node speed categories. Although DYMO operates like its predecessor i.e. AODV and does not add any extra modifications to the existing functionality, the operation is further simpler, that is purely based on sequence numbers assigned to all the packets. However, DYMO came second after LANMAR in the previous chart showing highest delay. Adversely LANMAER performance has slightly improved, this is because of the increase of the pause time, which allowed it to have more time to update its routing tables.



	5	10	15	25	50
ANTMANET	7.539	8.175	8.175	14.872	21.972
AODV	11.792	13.765	26.671	27.983	44.676
DYMO	24.352	26.679	21.245	33.389	27.643
LANMAR	18.575	21.21065	13.7151	20.4123	35.0076

Figure 41: Average End to End delay vs Node Speed per Protocol- Pause time 15 (sec)

Figure 41 highlights the expenditure of the average E-to-E delay while the pause time has increased again to 15 seconds, again the effect of this increase is quite clear on the performance of the proposed protocols. LANMAR showed a significant improvement in its delay performance as the pause time increased to 15 (sec), but this was not enough to outperform ANTMANET that is showing significantly low delay.

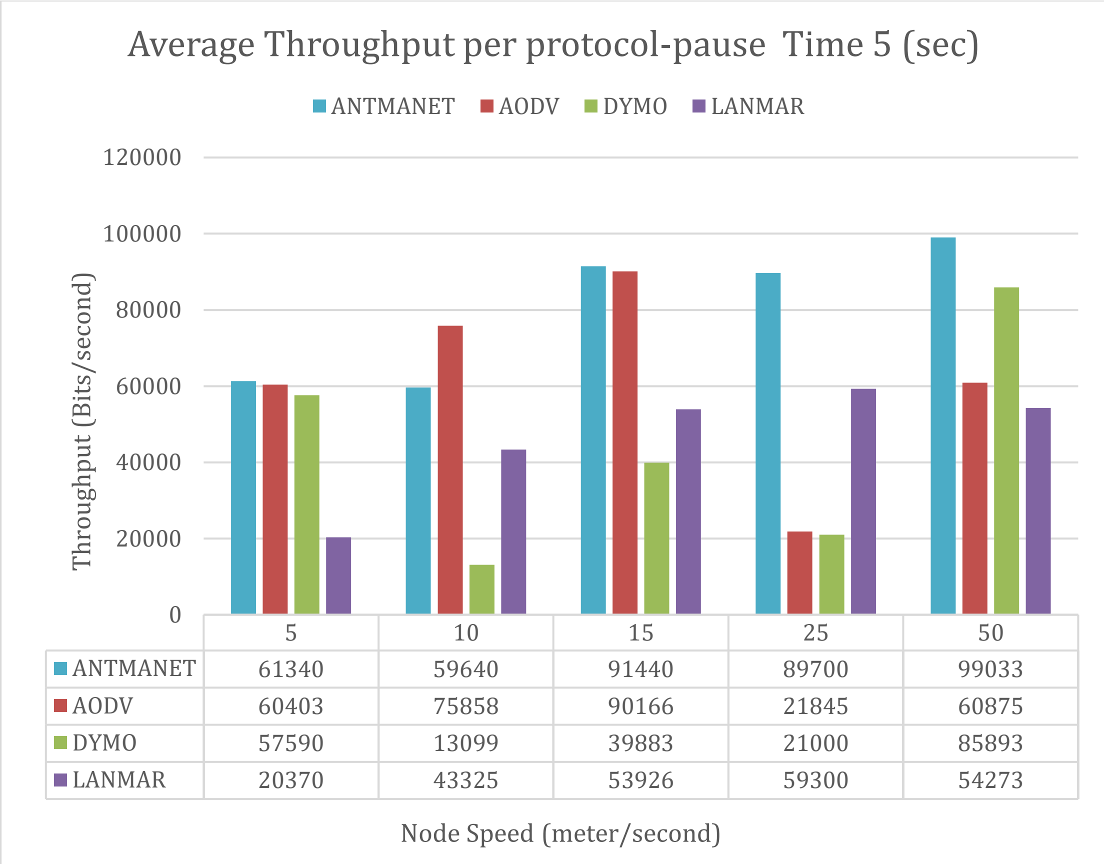
The following section presents a discussion of the throughput results, as well as performance models.

5.3.1.2 Throughput

The two main functions operated by any routing protocol are the selection of routes for various origin-destination pairs and the delivery of messages to their correct destination once the routes are selected. The second function is conceptually straightforward using a variety of protocols and data structures (known as routing tables). The delivery of packets is measured by one of the main performance metrics, known as throughput.

Overall, ANTMANET has offered an effective routing. The effect of good routing is to increase throughput while keeping the same value of average delay per packet under high offered load conditions and to decrease average delay per packet under low and moderate offered load conditions. Furthermore, it is evident that the proposed routing protocol can operate to keep average delay per packet as low as possible for any given level of mobility. While this is easier said than done, it provides a clear-cut objective, which is expressed by its structure and design (refer to chapter 4). This section illustrates the proposed protocol's performance in terms of the throughputs.

Figure 42, Figure 43 and Figure 44 consist of Five sub- figures showing the tested protocols following the same colour code as in the previous figures –legend is in the top of the graph- illustrating the number of packets that been received by the destination. ANTMANET shows higher throughputs that link very well with the delay results. In each speed category, the proposed protocol demonstrates acceptable results even when it reaches the bottleneck performance level that is after the third speed category.

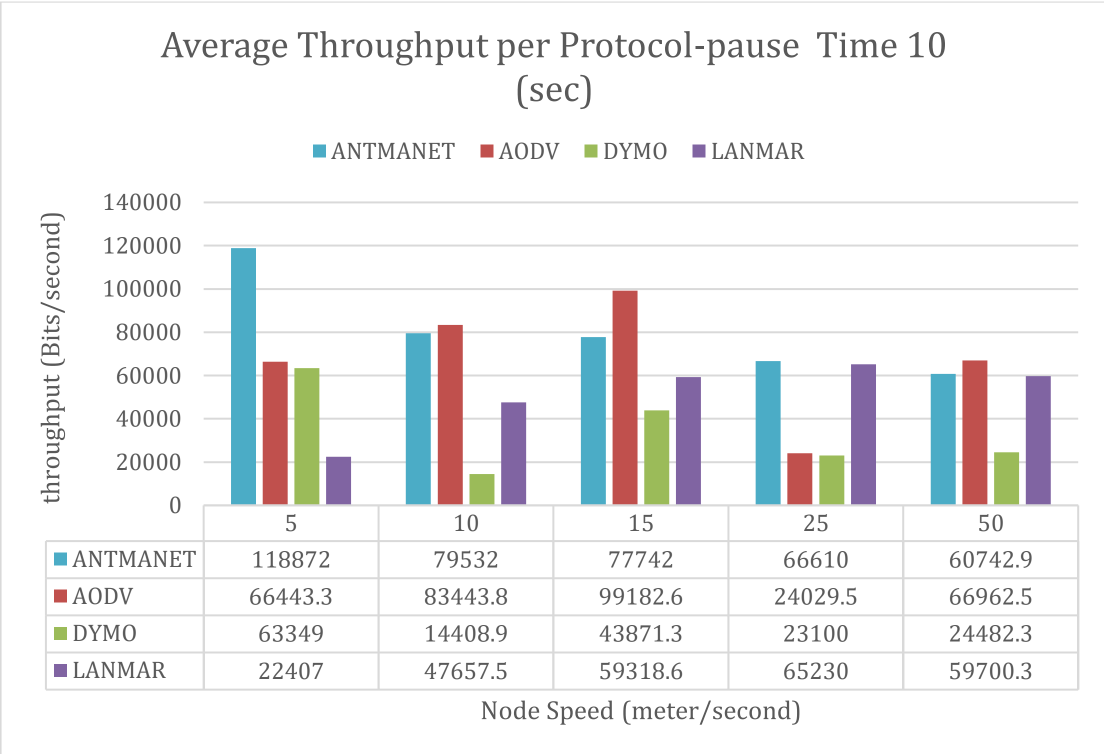


	5	10	15	25	50
ANTMANET	61340	59640	91440	89700	99033
AODV	60403	75858	90166	21845	60875
DYMO	57590	13099	39883	21000	85893
LANMAR	20370	43325	53926	59300	54273

Figure 42: Average Throughput Vs Node Speed per Protocol-Pause time 5(sec)

Figure 42 illustrates the average throughput results of the network of each of the tested protocol with a 5 (sec) pause time. In this case, it is a high mobility condition where the nodes remain stationary for only 5 (sec). The higher throughput in this case, wouldn't necessary reflect that it is a good throughput. More or less it will mean that there is more packets get routed in the network, which means the use of the control packets is higher which will allow the protocol to converge fast enough to maintain fresh routing information in its routing tables. This is a paramount performance metrics as it measures the main functionality of any network that is delivering packets.

The total network traffic is 61340 (bits/sec) that is generated by the 30 nodes over 3600 (sec). each node that is operating ANTMANET has delivered 51340 (bits/sec) that is a 12% of the total network traffic, which is considered -when compared to the other protocol's results- as a satisfactory result specially within the rapid high mobility situation. While AODV was successful in delivering only 10% of the total network traffic and LANMAR came last delivering only 5% of the total network traffic. ANTMANET has illustrated 5% better performance than AODV and 15% better than LANMAR. Looking at the third-speed category ANTMANET and AODV have similar throughputs and that represents the performance bottleneck of all tested protocols.

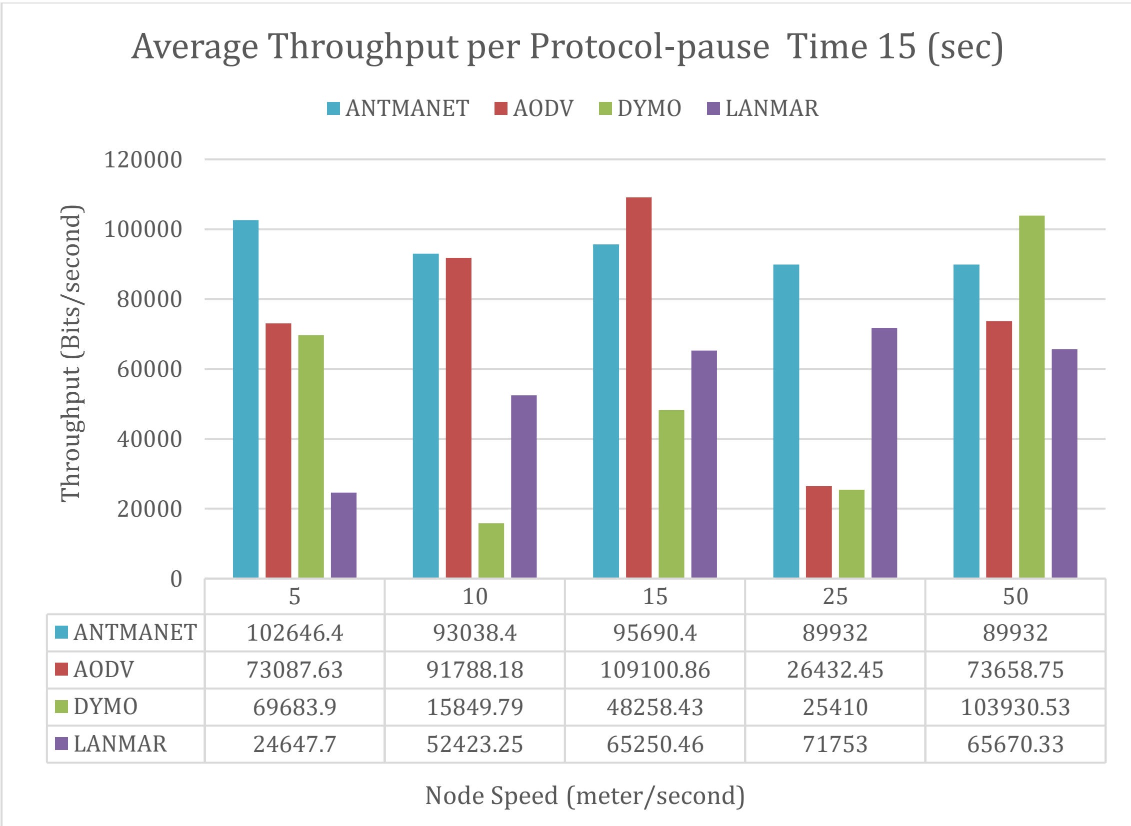


	5	10	15	25	50
ANTMANET	118872	79532	77742	66610	60742.9
AODV	66443.3	83443.8	99182.6	24029.5	66962.5
DYMO	63349	14408.9	43871.3	23100	24482.3
LANMAR	22407	47657.5	59318.6	65230	59700.3

Figure 43: Average Throughput Vs Node Speed per Protocol-Pause time 10 (sec)

The increase of the pause time results in a more relaxed network. As the nodes remain stationary for 10 (sec), which has a large impact on the throughput results that is evident in Figure 43. ANTMANET performance has increased delivering around 50% of the total network traffic. Where AODV and DYMO did not show noticeable improvement. Although DYMO is showing the lowest throughputs and its performance has clear instability.

Under this network conditions, ANTMANET shows a clear advantage over the standard MANET protocols. ANTMANET has demonstrates 50% better performance than AODV and 65% better than LANMAR in the first speed category. Looking at the third-speed category ANTMANET still clearly has the better performance as it has maintained steady and robust performance unlike and AODV and DYMO. The performance bottleneck of the proposed protocol is now in the fourth-speed category where it shows a huge drop of the measurement, yet it is still better than the standard tested protocols.



	5	10	15	25	50
ANTMANET	102646.4	93038.4	95690.4	89932	89932
AODV	73087.63	91788.18	109100.86	26432.45	73658.75
DYMO	69683.9	15849.79	48258.43	25410	103930.53
LANMAR	24647.7	52423.25	65250.46	71753	65670.33

Figure 44: Average Throughput Vs Node Speed per Protocol-Pause time 15 (sec)

Figure 44 demonstrate the performance of the tested protocols while the pause time increased to 15 (sec). Looking at the third category, it is noticeable that AODV has outperformed the ANTMANET by 20%. But in the same time, firstly, ANTMANET has outperformed AODV in all the other speed categories. Secondly its performance has enhanced by 5% compared to the previous network conditions. LANMAR has clearly improved in its overall performance and the reason of this comes to the nature of the algorithm that it is based on which uses the landscape information to calculate the routing cost. Another protocol that shows noticeable performance improvement is DYMO, where AODV has only a slight improvement.

Looking at the first and the second category, AODV performance has dropped leaving ANTMANET in the lead. Most importantly ANTMANET has an evident advantage of all tested protocols in the fourth category as it represents the performance bottleneck of all three protocols where ANTMANET has an 30% better performance than LANMAR the second-best protocol in this category.

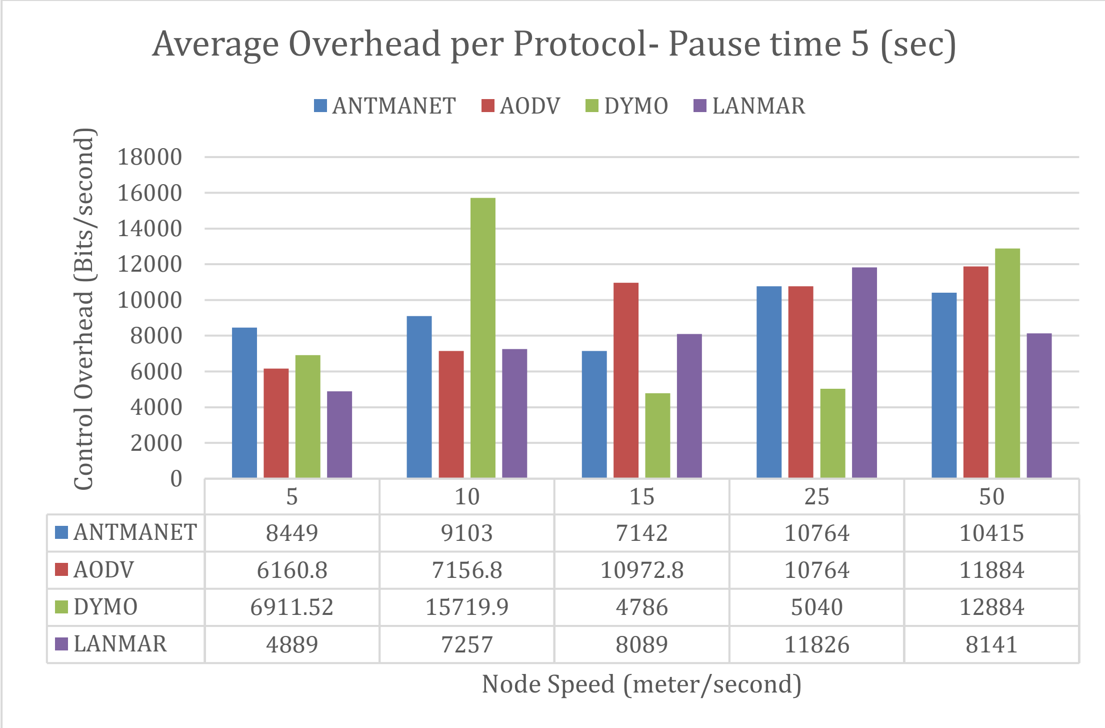
Now throughputs metrics measures all delivered packets, and in any network, there are two types of packets (data packets and control packets). The high throughput is a good indication that the information is being delivered to the desired destination unless most of, many of the measured metrics are control packets. Therefore, there is another performance metrics - network overhead - needs to be considered to adequately prove that the throughputs measured in this section is “good throughputs”. The network overhead discussed in the following section.

5.3.1.3 Network overhead

Network overhead is an important concept to understand when it comes to evaluating any network performance. It is basic to comprehend the methodology employed by various routing protocols to deliver information from one node to another, and the costs involved. Network overhead refers to the network routing information sent by the protocol, which uses a portion of the available bandwidth of the communication medium. This extra data, making up the protocol headers and this application-specific information is referred to as overhead. Since it does not contribute to the content of the message, using a higher rate of control packets will cause fewer data packets to be delivered and that is not acceptable since delivering information is the main function of any network. This section illustrates ANTMANET performance in terms of the network overhead.

Figure 45, Figure 46 and Figure 47 consist of Five sub- figures showing the tested protocols following the same colour code as in the previous figures –legend is in the top of the graph- illustrating the routing message overhead that is calculated as the total number of control packets transmitted. The increase in the routing message overhead reduces the performance of the network as it consumes portions from the bandwidth available to transfer data between the nodes.

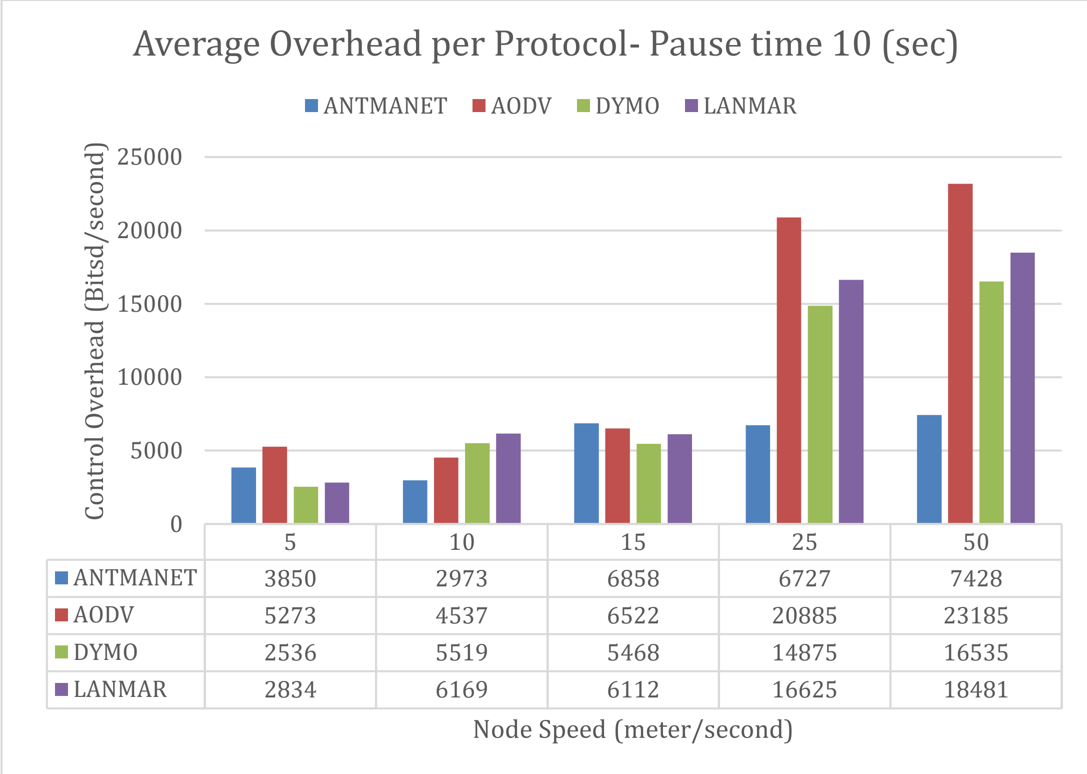
Overall, ANTMANET has been successful in maintaining a high level of stability and robustness in terms of the network overhead results. It has shown the lowest use of control packets in each category in all three experimental conditions. ANTMANET performance has improved while the pause time increased. The proposed protocol has shown steady behaviour especially in the fourth and fifth speed category, which implies that the node speed did not force the protocol to use more control packets to maintain routing information.



	5	10	15	25	50
ANTMANET	8449	9103	7142	10764	10415
AODV	6160.8	7156.8	10972.8	10764	11884
DYMO	6911.52	15719.9	4786	5040	12884
LANMAR	4889	7257	8089	11826	8141

Figure 45: Average Overhead Vs Node Speed per Protocol- Pause time 5 (sec)

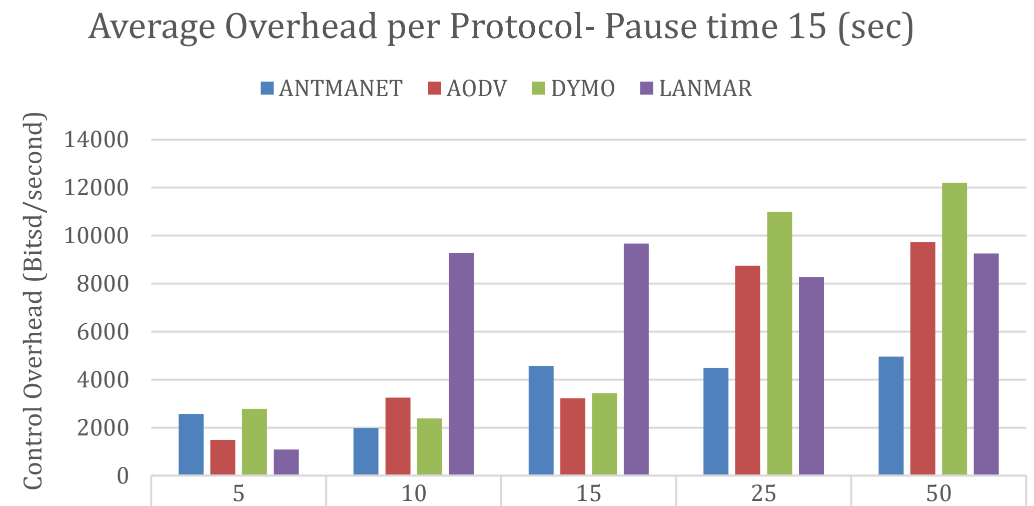
Figure 45 illustrates the network Overhead results of the tested protocols when the pause time is 5 (sec). This condition is the extreme scenario of all the proposed scenarios. That is because it stresses the network to the highest limit examining different levels of speeds from low to extremely high speed. ANTMANET has shown a steady behaviour throughout each category. ANTMANET has generated control packets over all 15% less than AODV and this percentage increased to approximately 35% during the second, fourth and the fifth category. ANTMANET has a clear advantageous point when compared to LANMAR and DYMO. As it has outperformed both by an average of 25%.



	5	10	15	25	50
ANTMANET	3850	2973	6858	6727	7428
AODV	5273	4537	6522	20885	23185
DYMO	2536	5519	5468	14875	16535
LANMAR	2834	6169	6112	16625	18481

Figure 46: Average Overhead Vs Node Speed per Protocol- Pause time 10 (sec)

Figure 46 illustrates the network Overhead results of the tested protocols when the pause time is 10 (sec). Again, ANTMANET has scored a steady performance in each category. ANTMANET has outperformed LANMAR by 45%, AODV by 25% and DYMO by 30%. The tested protocols have performed as expected, where LANAMR performance has shown some improvement in its performance. This is expected as LANAMR becomes more effective when the nodes stay stationary for a longer period. It has improved by 15% when compared with its performance in Figure 45. AODV and DYMO have witnessed improvement as well.



	5	10	15	25	50
ANTMANET	2567	1982	4572	4485	4952
AODV	1492	3247	3217	8750	9727
DYMO	2775.5	2388	3433	10992.5	12203
LANMAR	1090	9269	9669	8272	9249

Figure 47: Average Overhead Vs Node Speed per Protocol- Pause time 15 (sec)

Figure 47 illustrates the network Overhead results of the tested protocols when the pause time is 15 (sec). Generally, all protocols have followed the same behaviour patterns in each category as in Figure 45 and Figure 46. However, in this scenario, they all have propagated less control packets than the previous scenarios since this scenario has the longest pause time. This allows all protocols to reduce the usage of the communication medium and improve their behaviour. ANTMANET has illustrated its best behaviour in this scenario. It has improved its performance by 20% and has maintained steady performance in all categories. This is a vital result coupled with the throughputs result as it clearly proves that the proposed protocol is delivering data packets more than control packets.

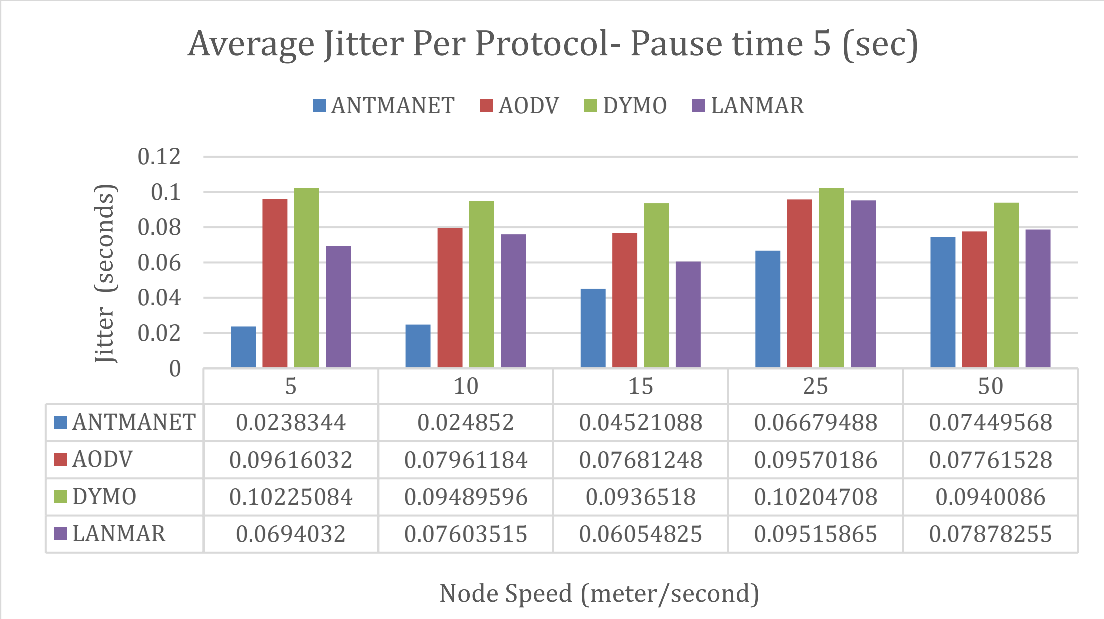
In general, the network metrics are similar and are related to desired outcomes. Defining network metrics that matter to an organisation depends on the use case is a key point in evaluating any system. In this research's case, the delivery of the collected information is very important, known as the bulk data movement, and often it is desirable to have a path of low delay and network overhead. However, there are other desirable features that affect the delay such as jitter. The following section describes the performance results of the proposed protocol in terms of its jitter delay.

5.3.1.4 Jitter

Jitter metrics represents the variation in the delay of received packets in a flow. It is an important metrics, especially for real-time applications. This section illustrates the performance measurements of the proposed protocol in terms of jitter.

Figure 48, Figure 49 and Figure 50 consist of Five sub- figures showing the tested protocols following the same colour code as in the previous figures –legend is in the top of the graph- illustrating the variation in the delay of received packets.

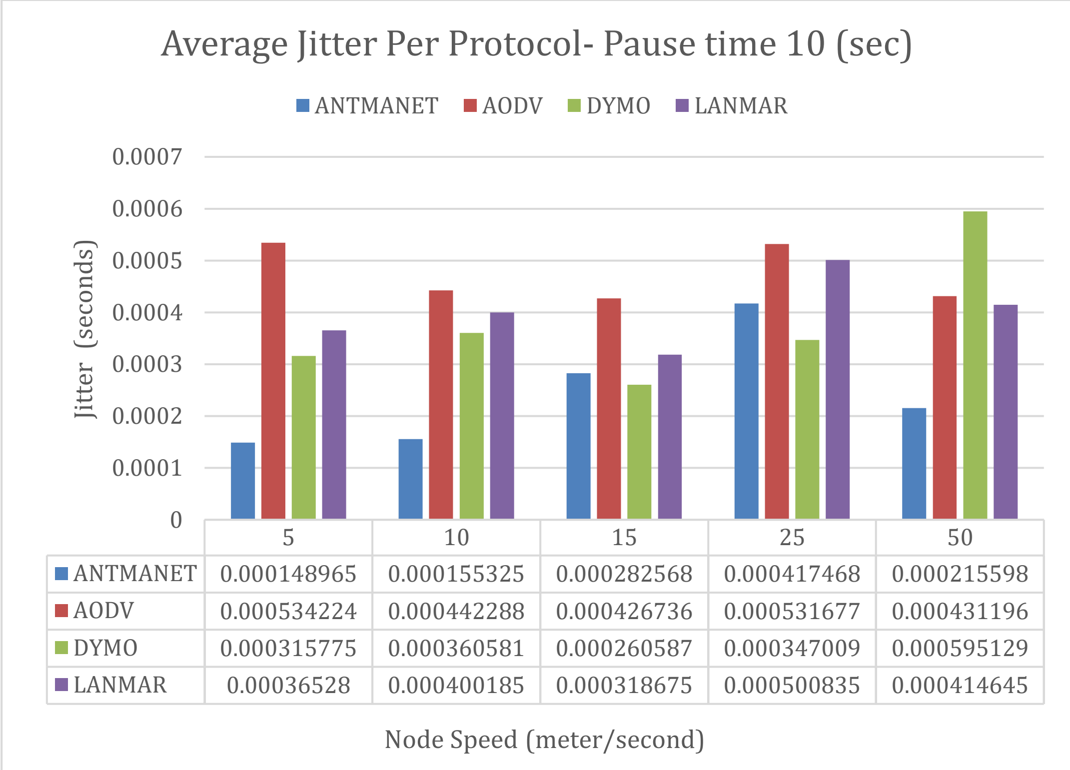
Overall, in terms of the jitter measurement ANTMANET has shown the lowest measurement compared to the tested protocols. As in E-to E delay measurement the proposed protocol has shown stability and robustness even in the extreme network situation. These results represent a clear evidence that the proposed protocol can handle to operating real-time applications.



	5	10	15	25	50
ANTMANET	0.0238344	0.024852	0.04521088	0.06679488	0.07449568
AODV	0.09616032	0.07961184	0.07681248	0.09570186	0.07761528
DYMO	0.10225084	0.09489596	0.0936518	0.10204708	0.0940086
LANMAR	0.0694032	0.07603515	0.06054825	0.09515865	0.07878255

Figure 48: Average Jitter Vs Node Speed per Protocol- Pause time 5 (sec)

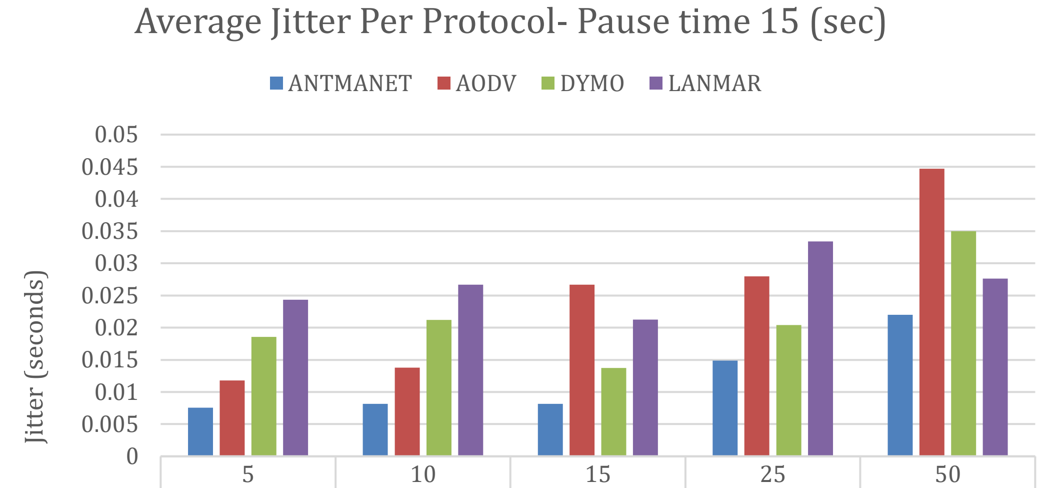
Figure 48 illustrates the jitter measurements for the tested protocols when the pause time is 5 (sec). Jitter is a crucial network performance indicator as it directly affects the buffering requirements for all real-time applications. A higher value of jitter can lead to many issues ranging from lip-sync errors in video traffic to the loss of packets because of either buffer overflow or underflow (Jacobson 2000). By examining the first speed component, ANTMANET has low jitter (0.005 sec), 20 % less than AODV, 55% less than DYMO and 75% less than LANMAR. The performance of ANTMNET degraded by 8% while the speed of the nodes increases. This is considered as a better performance compared to 12% in DYMO. AODV shows unexpected performance fluctuating in infrequent behaviour. This indicates that AODV is not suitable for real-time application in such stressed networks. AODV and DYMO had the highest jitter displaying 0.04 and 0.03 (sec) respectively.



	5	10	15	25	50
ANTMANET	0.000148965	0.000155325	0.000282568	0.000417468	0.000215598
AODV	0.000534224	0.000442288	0.000426736	0.000531677	0.000431196
DYMO	0.000315775	0.000360581	0.000260587	0.000347009	0.000595129
LANMAR	0.00036528	0.000400185	0.000318675	0.000500835	0.000414645

Figure 49: Average Jitter Vs Node Speed per Protocol- Pause time 10 (sec)

Figure 49 Figure 48 illustrates the jitter measurements for the tested protocols when the pause time is 10 (sec). While the pause time increased to 10 (sec) the performance of all tested protocols has improved. ANTMANET witnessed around a 10% improvement in most speed categories. The proposed protocol has illustrated lower jitter in each category and the performance bottleneck is clear in the fourth speed category as the jitter has doubled. In the same time, all tested protocols had the same behaviour in the same category.



	5	10	15	25	50
ANTMANET	0.007539	0.008175	0.008175	0.014872	0.021972
AODV	0.011792	0.013765	0.026671	0.027983	0.044676
DYMO	0.018575	0.02121065	0.0137151	0.0204123	0.0350076
LANMAR	0.024352	0.026679	0.021245	0.033389	0.027643

Node Speed (meter/second)

Figure 50: Average Jitter Vs Node Speed per Protocol- Pause time 15 (sec)

Figure 50 illustrates the jitter measurements for the tested protocols when the pause time is 15 (sec). As expected ANTMANET has the lowest jitter in all categories. It stays at the lowest measurement in the first speed category with 0.0075 (sec) and then as the speed increases the jitter degraded as well. In the very high-speed condition, ANTMANET jitter is 0.02 (sec) compared to AODV that starts at 0.01 (sec) delay, which is 10% more than ANTMANET and ends with 4 (sec).

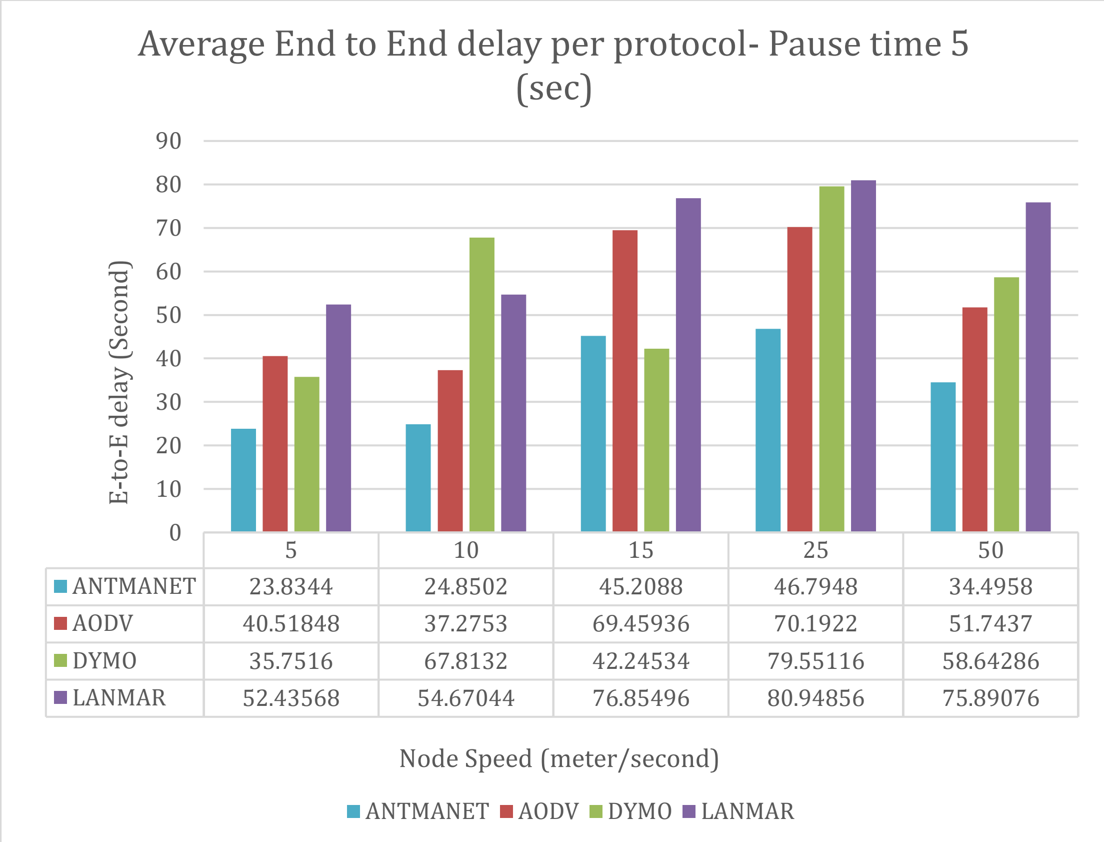
All the above results considered a low data rate network, where the proposed protocol has an improved performance. The following section illustrates the second scenario where the number of generated messages per second is doubled.

5.3.2 Scenario B

The second condition examined is when the CBR application is generating traffic at the rate of 8 packets/sec. This is tested using three different pause times (10, 20 and 30 seconds).

5.3.2.1 End to End Delay

Overall the charts in Figure 51, Figure 52 and Figure 53 shows that the average End to End delay of ANTMANET – denoted in blue- was lower in each category than all tested protocols, which indicates that the proposed protocol was successful in outperforming the standard protocols even when there is more load on the network. It also illustrates the same level of stability in ANTMANET's performance. When changing node speeds and pause time the proposed protocol has demonstrated good behavioural performance up to the point where the performance starts to degrade in the high speed that is represented in the last two speed categories.

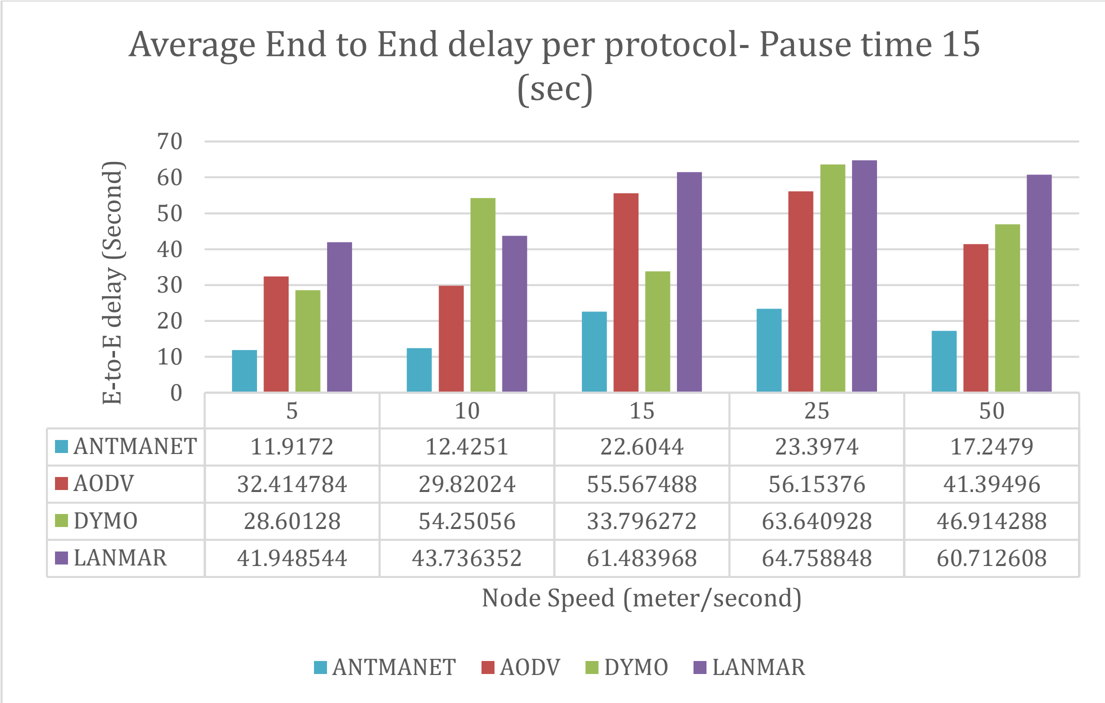


	5	10	15	25	50
ANTMANET	23.8344	24.8502	45.2088	46.7948	34.4958
AODV	40.51848	37.2753	69.45936	70.1922	51.7437
DYMO	35.7516	67.8132	42.24534	79.55116	58.64286
LANMAR	52.43568	54.67044	76.85496	80.94856	75.89076

Figure 51: Average End to End delay vs Node Speed per Protocol- Pause time 5 (sec)

In Figure 51, in the first-speed category ANTMANET demonstrates the best performance by generating a delay that is less than the AODV by 35%. AODV is one of the established protocols that are well tested and designed for MANET generally and has been used widely in MANET. AODV is performing better that DYMO in most categories, although DYMO is a derivation of AODV and this is basically due to the same reasons mentioned in the previous section.

LANMAR network shows the highest delay of the tested protocols within most categories. As previously mentioned, higher speed will cause higher delay rates. Consequently, all protocols have witnessed an increase in their delay rates. ANTMANET has managed to stabilise its performance until it reached its performance bottleneck that occurred in the fourth speed category.

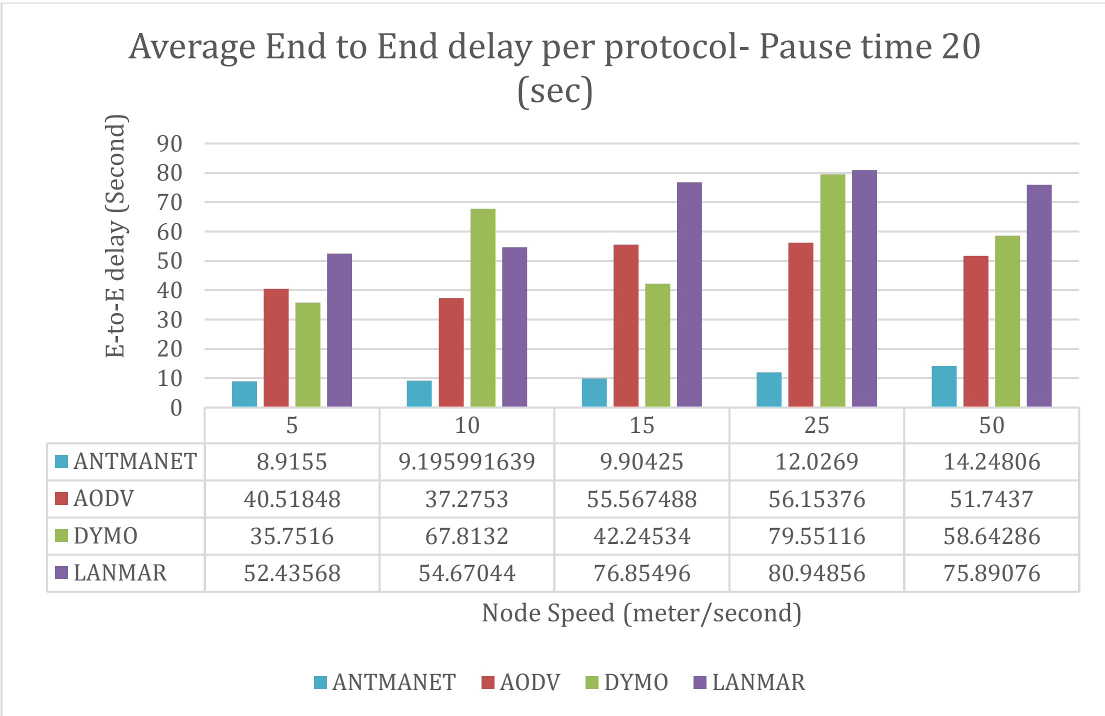


	5	10	15	25	50
ANTMANET	11.9172	12.4251	22.6044	23.3974	17.2479
AODV	32.414784	29.82024	55.567488	56.15376	41.39496
DYMO	28.60128	54.25056	33.796272	63.640928	46.914288
LANMAR	41.948544	43.736352	61.483968	64.758848	60.712608

Figure 52: Average End to End delay vs Node Speed per Protocol- Pause time 15 (sec)

Figure 52 illustrates the amount of the average E-to-E delay while the pause time is increased to 15 Seconds, this increase has clearly affected the performance of all protocols and decreased the load on the network infrastructure. ANTMANET has the lowest average E-to-E delay that is illustrated in all node speed categories. Although DYMO operates like its predecessor i.e. AODV and does not add any extra modifications to the existing functionality but operation is moreover quite simpler, that is purely based on sequence numbers assigned to all the packets. For this reason, it shows significantly high delay of all protocols. DYMO was second after LANMAR in the previous chart showing highest delay as well, but in this chart, LANMAER performance has slightly improved, for the pause time has increased, which allowed it to have more time to update its routing tables.

In terms of, ANTMANET performance it has the lowest delay in all categories as opposed to exactly 24.6 seconds delay of AODV network and 56.5-seconds delay of DYMO, which is (55% and 70% delay increase respectively) in the first category. And in the second category, the delay increased as expected in all protocols, where the delay increase of ANTMANET was 25% of the previous category opposed to 68% delay increase of the AODV network and 70% increase in the LANMAR network delay.



	5	10	15	25	50
ANTMANET	8.9155	9.195991639	9.90425	12.0269	14.24806
AODV	40.51848	37.2753	55.567488	56.15376	51.7437
DYMO	35.7516	67.8132	42.24534	79.55116	58.64286
LANMAR	52.43568	54.67044	76.85496	80.94856	75.89076

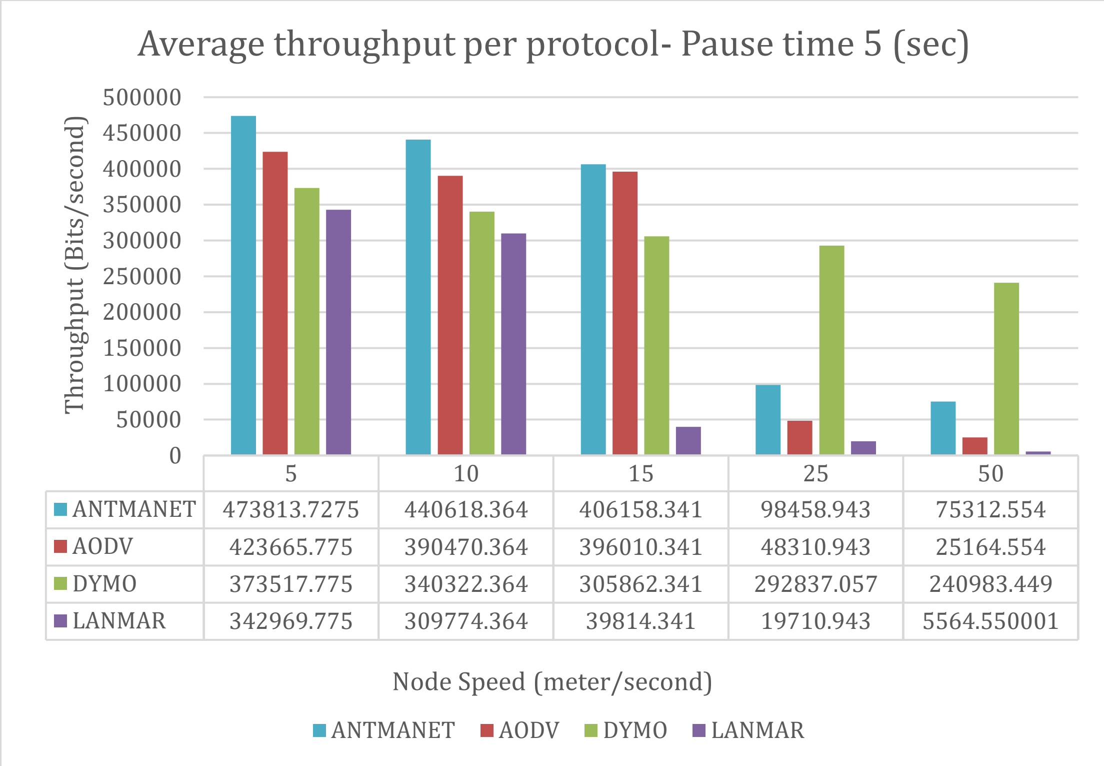
Figure 53: Average End to End delay vs Node Speed per Protocol- Pause time 20 (sec)

Figure 53 highlights the expenditure of the average E-to-E delay while the pause time has increased again to 30 seconds, again the effect of this increase is quite clear on the performance of the proposed protocols. LANMAR has illustrated a major progress of its delay performance as the pause time increased to 20 (sec), yet was not enough to outperformed ANTMANET which shows significantly low delay. The effect of the longer pause time is clear on the proposed protocol's performance as in each speed category the measurements have improved by around 25% in all three charts.

5.3.2.2 Throughputs

Overall, ANTMANET has offered and effective routing, as mentioned in section 5.3.1.2 good throughputs means increasing data delivery while keeping delay and overhead to the lowest possible. Furthermore, it is evident that the proposed routing protocol can operate to keep average delay per packet as low as possible for any given level of mobility. While this is by no means simple, it provides a clear-cut objective, which is expressed by its structure and design (refer to chapter 4). This section illustrates the proposed protocol's performance in terms of the throughputs.

Figure 54, Figure 55 and Figure 56 consist of Five sub- figures showing the tested protocols following the same colour code as in the previous figures –legend is in the top of the graph- illustrating the number of packets that been received by the destination. ANTMANET shows higher throughputs that link very well with the delay results. In each speed category, the proposed protocol demonstrates well to acceptable results even when it reaches the bottleneck performance level that is after the third speed category.



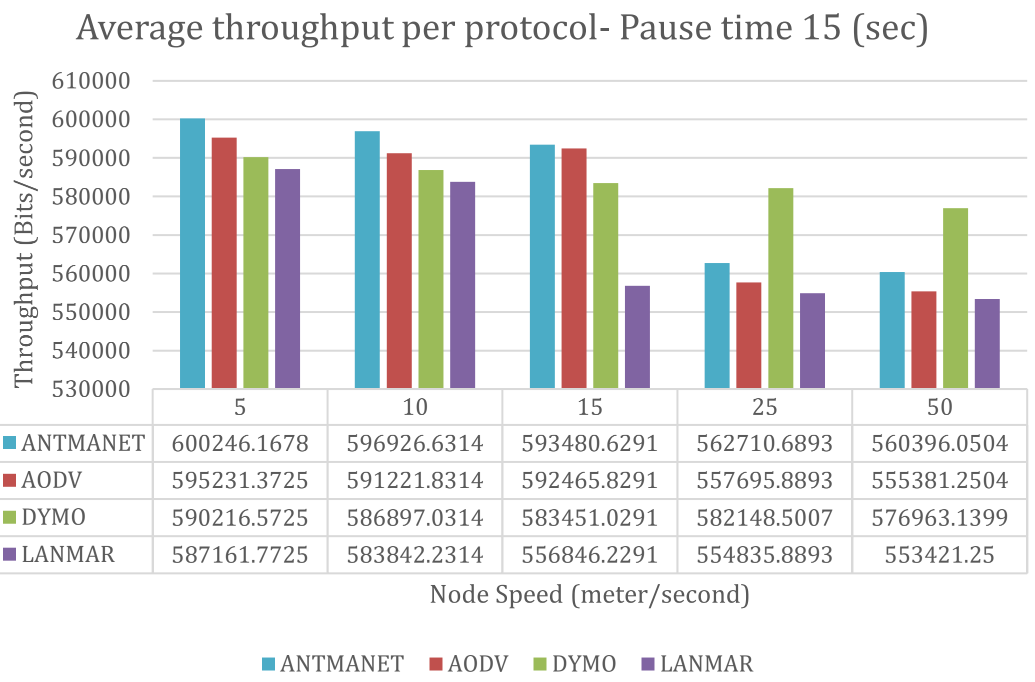
	5	10	15	25	50
ANTMANET	473813.7275	440618.364	406158.341	98458.943	75312.554
AODV	423665.775	390470.364	396010.341	48310.943	25164.554
DYMO	373517.775	340322.364	305862.341	292837.057	240983.449
LANMAR	342969.775	309774.364	39814.341	19710.943	5564.550001

Figure 54: Average Throughput vs Node Speed per Protocol- Pause time 5 (sec)

Figure 54 illustrates the average throughput results of the network of each of the tested protocol with a 5 (sec) pause time. Now this is a high mobility condition where the nodes remain stationary for only 5 (sec). The higher throughputs in this case, mean more packets gets routed in the network, which mean the protocol can converge fast enough to maintain fresh routine information in its routing tables. This is an important performance metric as it measures the main functionality of any network that is delivering packets.

ANTMANET has 45% packets of the total network traffic, which is considered as a good result compared with the standard protocols within the rapid high mobility situation. While AODV was successful in delivering only 30% of the total network traffic and LANMAR came last delivering only 20% of the total network traffic.

ANTMANET has illustrated 15% better performance than AODV and 25% better than LANMAR. Looking at the third speed category ANTMANET and AODV similar throughputs and that represents the performance bottleneck of all tested protocols.

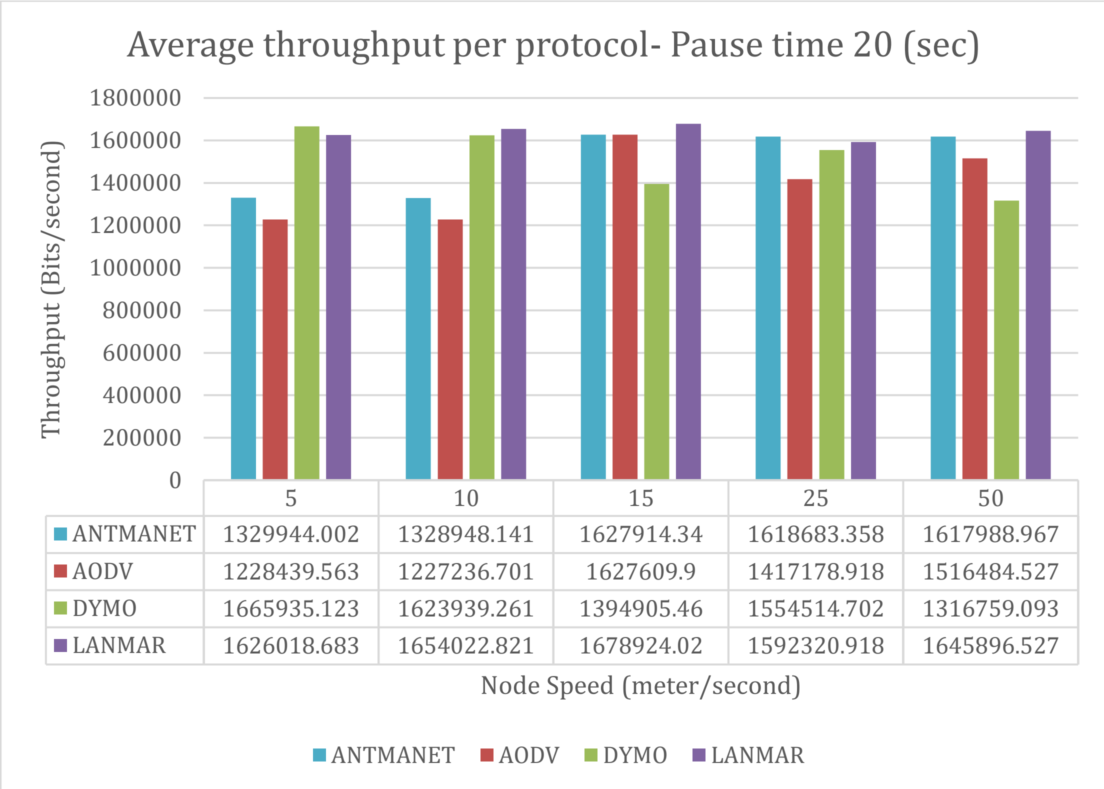


	5	10	15	25	50
ANTMANET	600246.1678	596926.6314	593480.6291	562710.6893	560396.0504
AODV	595231.3725	591221.8314	592465.8291	557695.8893	555381.2504
DYMO	590216.5725	586897.0314	583451.0291	582148.5007	576963.1399
LANMAR	587161.7725	583842.2314	556846.2291	554835.8893	553421.25

Figure 55: Average throughput vs Node Speed per protocol- Pause time 15 (sec)

The increase of the pause time results in a more stable network. The nodes remain stationary for 15 (sec), which has a large impact on the throughput results that is evident in Figure 55. ANTMANET is now delivering around 55% of the total network traffic. AODV also showed noticeable improvement, 7% more delivered packets compared to the previous chart. LANAMR has improved its performance and is no longer showing the lowest throughputs. DYMO has shown the worst measurements of all tested protocols and when the speed is very high it breaks down.

Under this network condition, ANTMANET shows a clear advantage on the standard MANET protocols. ANTMANET has illustrated 10% better performance than AODV and 15% better than DYMO. Looking at the third-speed category ANTMANET has clearly improved where it has maintained steady and robust performance. Unlike DYMO the performance bottleneck of the proposed protocol is now in the fourth-speed category where it witnesses a huge drop of the measurement, yet performs better than the standard tested protocols.



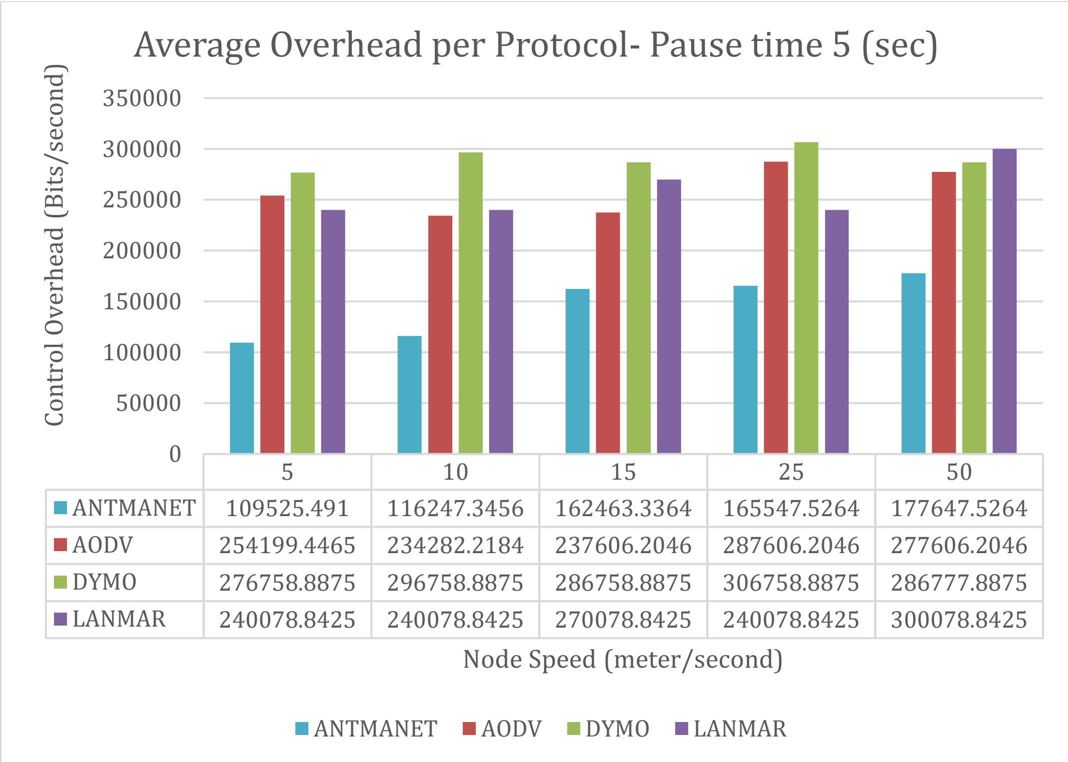
	5	10	15	25	50
ANTMANET	1329944.002	1328948.141	1627914.34	1618683.358	1617988.967
AODV	1228439.563	1227236.701	1627609.9	1417178.918	1516484.527
DYMO	1665935.123	1623939.261	1394905.46	1554514.702	1316759.093
LANMAR	1626018.683	1654022.821	1678924.02	1592320.918	1645896.527

Figure 56: Average throughput vs Node Speed per protocol- Pause time 20 (sec)

Figure 56 show the throughputs performance of the tested protocols while the pause time increased to 20 (sec). ANTMANET has managed to score some high throughputs in each of the speed categories in this chart. The longer pause time did improve the performance and the speed did not greatly impact its performance. Although the network load was higher in this scenario is higher but in the fourth category ANTMANET managed to outperform AODV by 15%, DYMO and LANMAR by 5%.

5.3.2.3 Network Overhead

Figure 57, Figure 58 and Figure 59 consist of Five sub- figures showing the tested protocols following the same colour code as in the previous figures –legend is in the top of the graph- illustrating the routing message overhead that is calculated as the total number of control packets transmitted. The increase in the routing message overhead reduces the performance of the Ad-Hoc network as it consumes portions from the bandwidth available to transfer data between the nodes.



	5	10	15	25	50
ANTMANET	109525.491	116247.3456	162463.3364	165547.5264	177647.5264
AODV	254199.4465	234282.2184	237606.2046	287606.2046	277606.2046
DYMO	276758.8875	296758.8875	286758.8875	306758.8875	286777.8875
LANMAR	240078.8425	240078.8425	270078.8425	240078.8425	300078.8425

Figure 57: Average Overhead Vs Node Speed per Protocol- Pause time 5 (sec)

Figure 57 illustrates the network Overhead results of the tested protocols when the pause time is 5 (sec). This condition is the extreme scenario of all the proposed scenarios. That is because it stresses the network to the limit examining different levels of speeds from low to extremely high. ANTMANET has shown a very steady behaviour throughout each category. ANTMANET has generated 60% fewer control packets than AODV and this percentage increased to 65% during the third, fourth and the fifth category. ANTMANET has clear advantage point when to compare LANMAR and DYMO, outperforming both by an average of 50%.

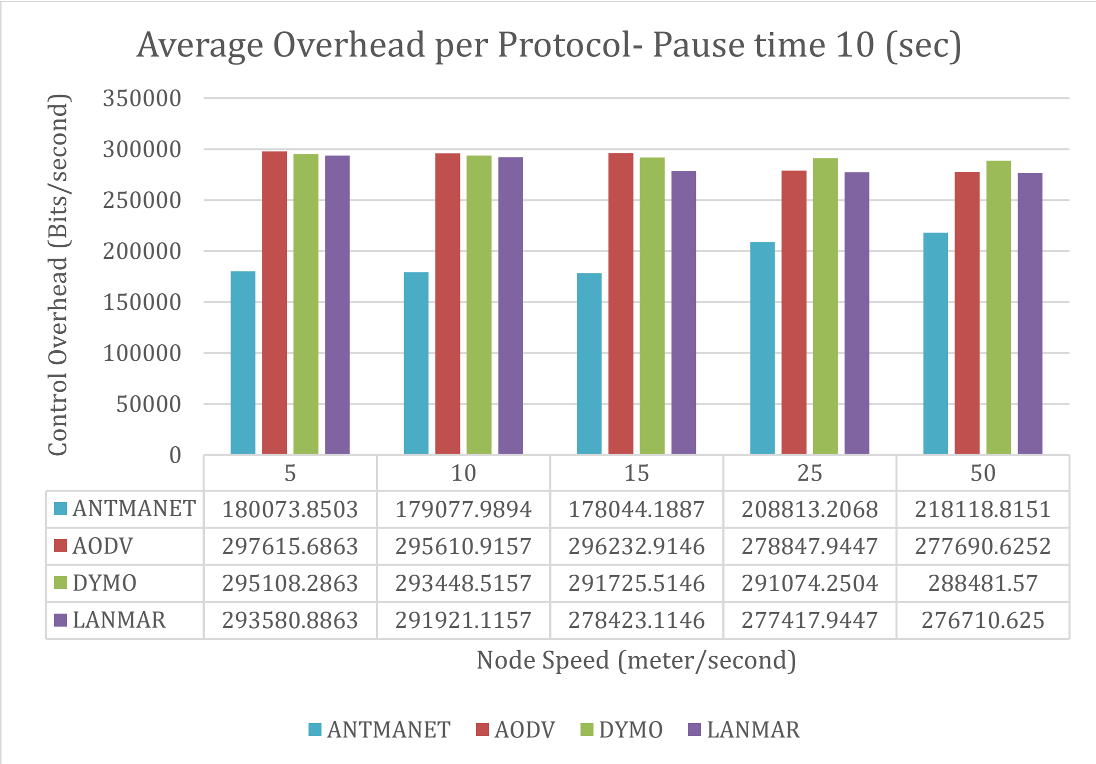


Figure 58: Average Overhead Vs Node Speed per Protocol- Pause time 20 (sec)

Figure 58 illustrates the network overhead results of the tested protocols when the pause time is 20 (sec). Again, ANTMANET has scored a steady performance in each category. ANTMANET has outperformed LANMAR by 45%, AODV by 55% and DYMO by 40%. The tested protocols have performed as expected, where LANAMR performance has shown some improvement in its performance. This is expected as LANAMR becomes more effective when the nodes stay stationary for longer. It has improved by 15% when compared with its performance in Figure 45. AODV and DYMO have also displayed improvement.

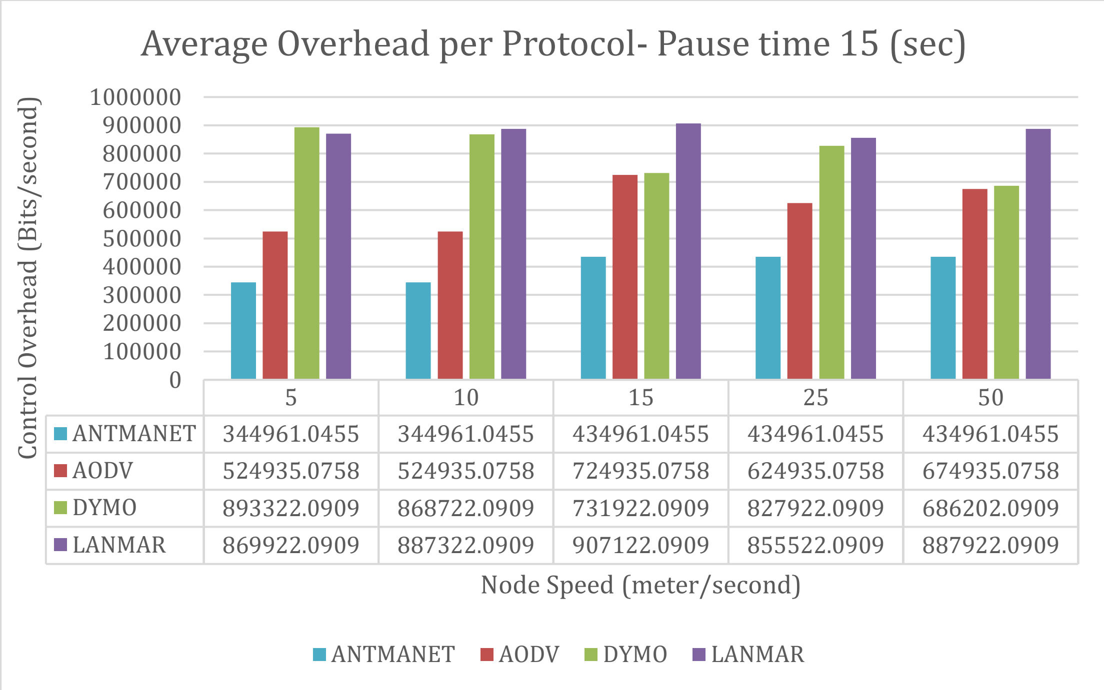


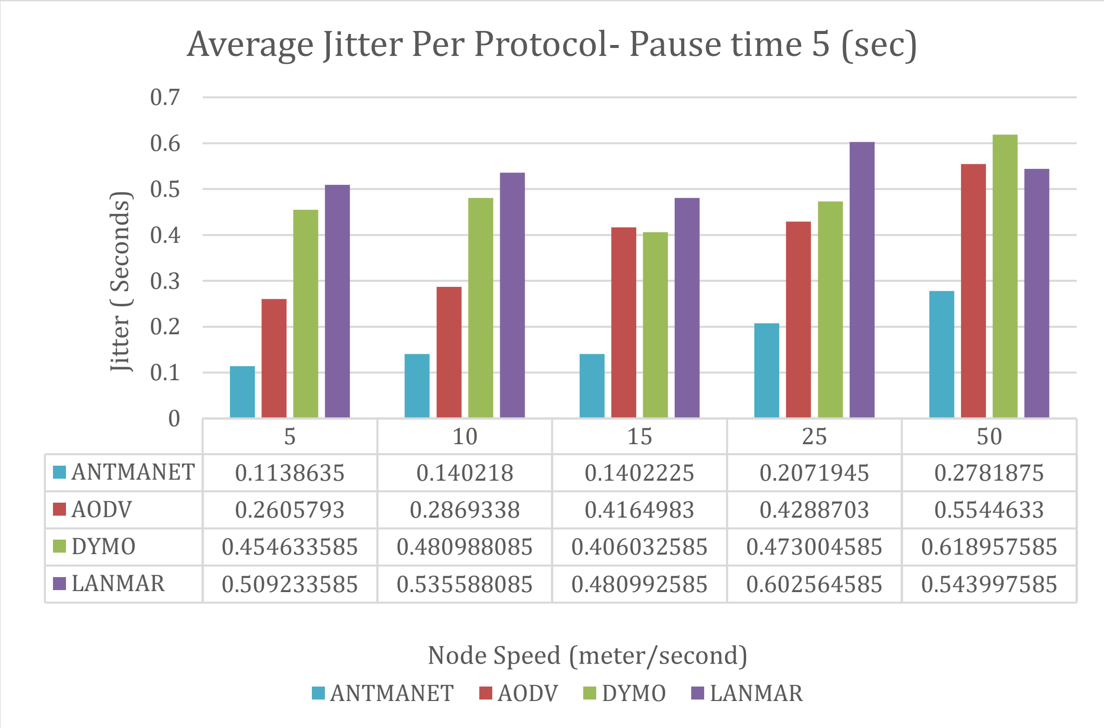
Figure 59: Average Overhead Vs Node Speed per Protocol- Pause time 15 (sec)

Figure 59 illustrates the network overhead results of the tested protocols when the pause time is 15 (sec). Generally, all protocols have followed the same behaviour patterns in each category as in Figure 45 and Figure 46 but they all has sent in this scenario less control packets than the previous scenarios, this is due to the fact that this scenario has the longest pause time. This allows all protocols to reduce the usage of the communication medium and improve its behaviour. ANTMANET has illustrated its best behaviour in this scenario. ANTMANET has improved its performance by 20% and has maintained steady performance in all categories. This is important, in that, along with the throughputs results, it proves that the proposed protocol is delivering data packets and generating fewer control packets.

5.3.2.4 Jitter

Figure 60, Figure 61and Figure 62 consist of five sub-figures showing the tested protocols following the same colour code as in the previous figures – the legend is in the top of the graph – illustrating the variation in the delay of received packets.

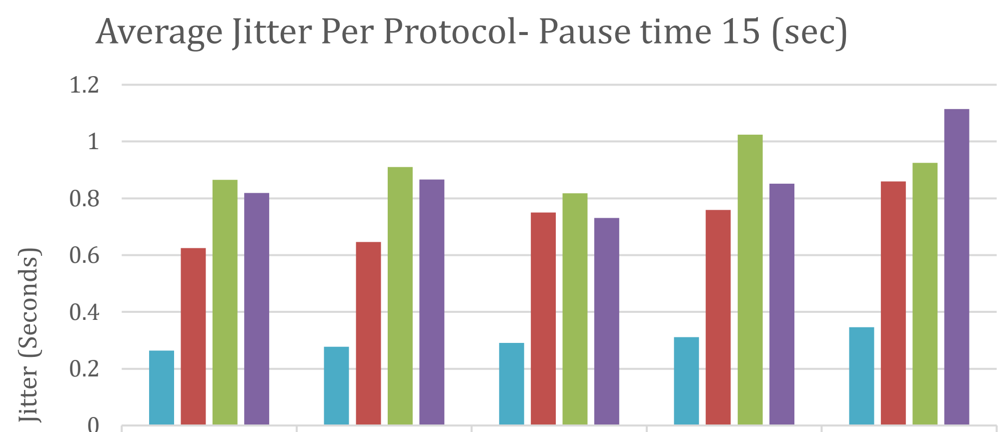
Overall, in terms of the jitter measurement ANTMANET has shown the lowest measurement compared to the tested protocols. As in E-to E delay measurement the proposed protocol has shown stability and robustness even in the extreme network situation. These results represent evidence that the proposed protocol can operate in real-time applications.



	5	10	15	25	50
ANTMANET	0.1138635	0.140218	0.1402225	0.2071945	0.2781875
AODV	0.2605793	0.2869338	0.4164983	0.4288703	0.5544633
DYMO	0.454633585	0.480988085	0.406032585	0.473004585	0.618957585
LANMAR	0.509233585	0.535588085	0.480992585	0.602564585	0.543997585

Figure 60: Average Jitter Vs Node Speed per Protocol- Pause time 5 (sec)

Figure 60, illustrates the jitter measurements for the tested protocols when the pause time is 5 seconds. Jitter is very important and crucial network performance indicator as it directly affects the buffering requirements for all real-time applications. As mentioned previously, a higher value of jitter can lead to many problems ranging from lip-sync errors to the loss of packets because of buffer overflow or underflow (Hakak, Anwar, et al. 2014; Aweya et al. 2001). By examining the first speed component, ANTMANET has low jitter that is 0.11 (sec) this result is 40 % less than AODV, 55% less than DYMO and 60% less than LANMAR. The performance of ANTMNET degraded by 10% while the speed of the nodes increases. This is considered as a good performance compared to 15% in DYMO and unexpected performance of AODV as the results increases and decreases for no justified reason. This indicates that AODV is not suitable for real-time application in such stressed network. DYMO and LANMAR had the highest jitter displaying 0.61 and 0.54 (sec) respectively.



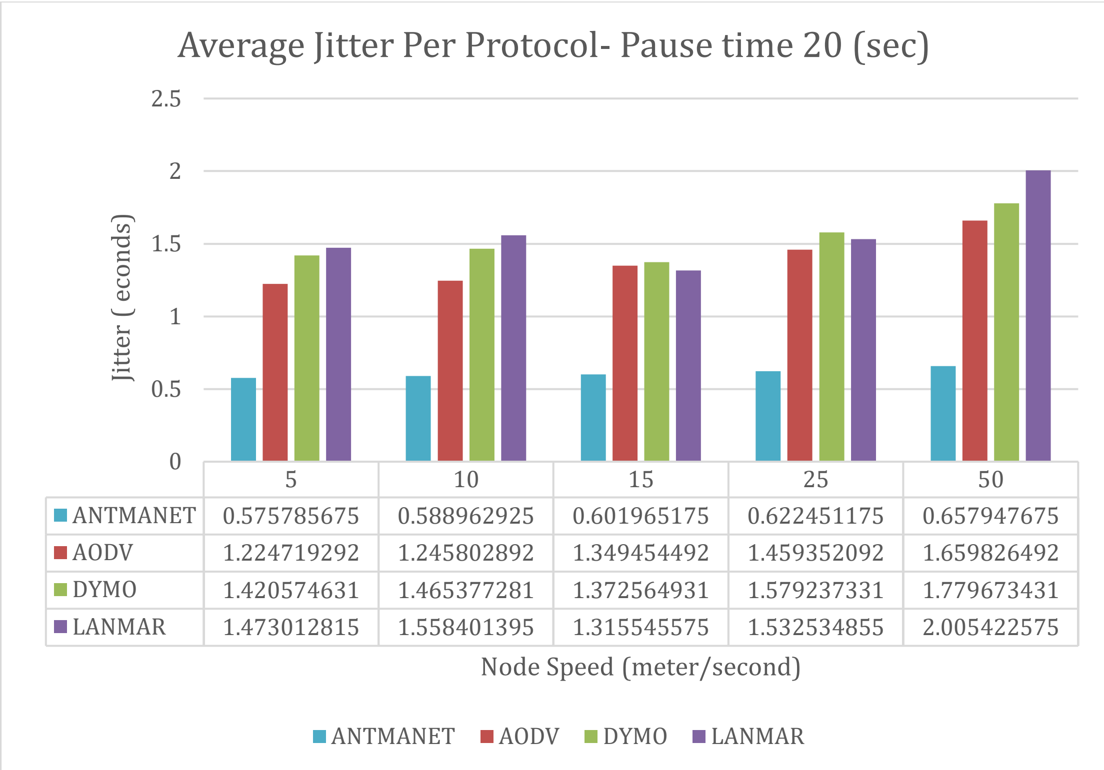
	5	10	15	25	50
ANTMANET	0.26412625	0.2773035	0.29030575	0.31079175	0.34628825
AODV	0.62496174	0.64604534	0.74969694	0.75959454	0.86006894
DYMO	0.865697095	0.910499745	0.817687395	1.024359795	0.924795895
LANMAR	0.818340453	0.865778553	0.730858653	0.851408253	1.114123653

Node Speed (meter/second)

ANTMANET AODV DYMO LANMAR

Figure 61: Average Jitter Vs Node Speed per Protocol- Pause time 15 (sec)

Figure 61 illustrates the jitter measurements for the tested protocols when the pause time is 15 (sec). While the pause time increased to 15 (sec) the performance of all tested protocols as improved. ANTMANET witnessed around 50% improvement in most speed categories. The proposed protocol has illustrated lower jitter in each category and the performance bottleneck is clear in the fourth speed category as the jitter has doubled. However, all tested protocols have the same behaviour in this category.



	5	10	15	25	50
ANTMANET	0.575785675	0.588962925	0.601965175	0.622451175	0.657947675
AODV	1.224719292	1.245802892	1.349454492	1.459352092	1.659826492
DYMO	1.420574631	1.465377281	1.372564931	1.579237331	1.779673431
LANMAR	1.473012815	1.558401395	1.315545575	1.532534855	2.005422575

Figure 62: Average Jitter Vs Node Speed per Protocol- Pause time 20 (sec)

Figure 62 illustrates the jitter measurements for the tested protocols when the pause time is 15 (sec). As expected ANTMANET has the lowest jitter in all categories. It starts at the lowest measurement in the first speed category with 0.5 (sec) and then as the speed increases the jitter depredates as well. In the high-speed jitter is 1.4 (sec). Compared to AODV that starts with 1.2 (sec) delay, which is 40% more than ANTMANET. LANMAR has lowest jitter 1.5 (sec) and scored 3 (sec) as the highest measurement.

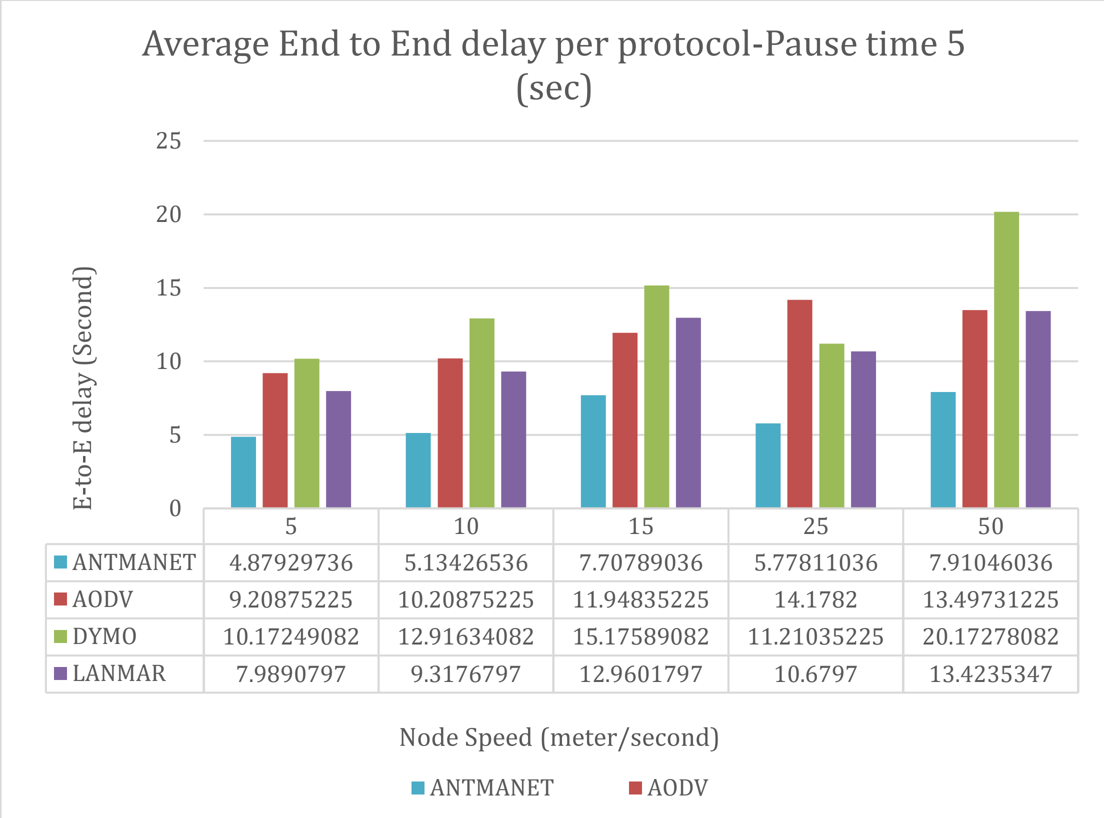
All the above results considered a low data rate network, where the proposed protocol has performed very well. The following section illustrates the second scenario where the number of generated messages per second is increased to 12 packets.

5.3.3 Scenario C

The third condition examined is when the CBR application is generating traffic at the rate of 12 packets/sec. this is considered as a high network load when compared to the normal load of the MANET. The importance of fully understanding the performance of the proposed protocol range of circumstances reliable and available networks and services.

5.3.3.1 End to End Delay

Overall the charts in Figure 63, Figure 64 and Figure 65 shows that the average End to End delay of ANTMANET – denoted in blue- was lower in each category than all tested protocols, which indicates that the proposed protocol was successful in outperforming the standard protocols. It also shows that ANTMANET has a level of stability in its performance while varying node speeds and pause time, the bottleneck of ANTMANET performance along with all four tested protocols is in exists within the high-speed categories.



	5	10	15	25	50
ANTMANET	4.87929736	5.13426536	7.70789036	5.77811036	7.91046036
AODV	9.20875225	10.20875225	11.94835225	14.1782	13.49731225
DYMO	10.17249082	12.91634082	15.17589082	11.21035225	20.17278082
LANMAR	7.9890797	9.3176797	12.9601797	10.6797	13.4235347

Figure 63: Average End to End delay vs Node Speed per Protocol-Pause time 5 (sec)

In Figure 63, at the first category of 5 (m/sec) speed, ANTMANET network shows a delay that is less than the AODV and DYMO networks by 45%. The late one by far it shows the worst performance of the tested protocol. LANMAR is not performing as good as well and the clear reason for this poor performance of LANMAR is because it uses the Landscape details to calculate the routing cost. The nodes are moving with the lowest speed of the running simulation but still they are rapidly moving giving the pause time is very short. Consequently, LANMAR did not have enough time to converge. On the other hand, ANTMANET has performed better as the ACO algorithm is a distributed routing algorithm. All nodes share their view of the network and each node need to know at least one neighbour towards its destination, which helps in speeding up the convergence.

While the speed of the nodes increases, the E-to-E delay increases as well. As moving faster causes more packets to queue for updated information regarding their destination to be obtained. The delay in the ANTMANET network has increased by 5% moving to the second category as the node's speed increased to 10 (m/sec) and 10% when the speed increased again the following category. ANTMANET sudden delay drop in the third category this indicates different behaviour pattern that is unexpected, this is because of the high network load and mobility level. The remaining protocols show higher delays while the speed increases, for instance, 25% delay increase in AODV network and 30% delay increase in the LANMAR and DYMO networks. For the last categories, the speed is high so the increase in the delay showing a big jump in DYMO and again ANTMANET kept its consistency with the lower E-to-E delay.

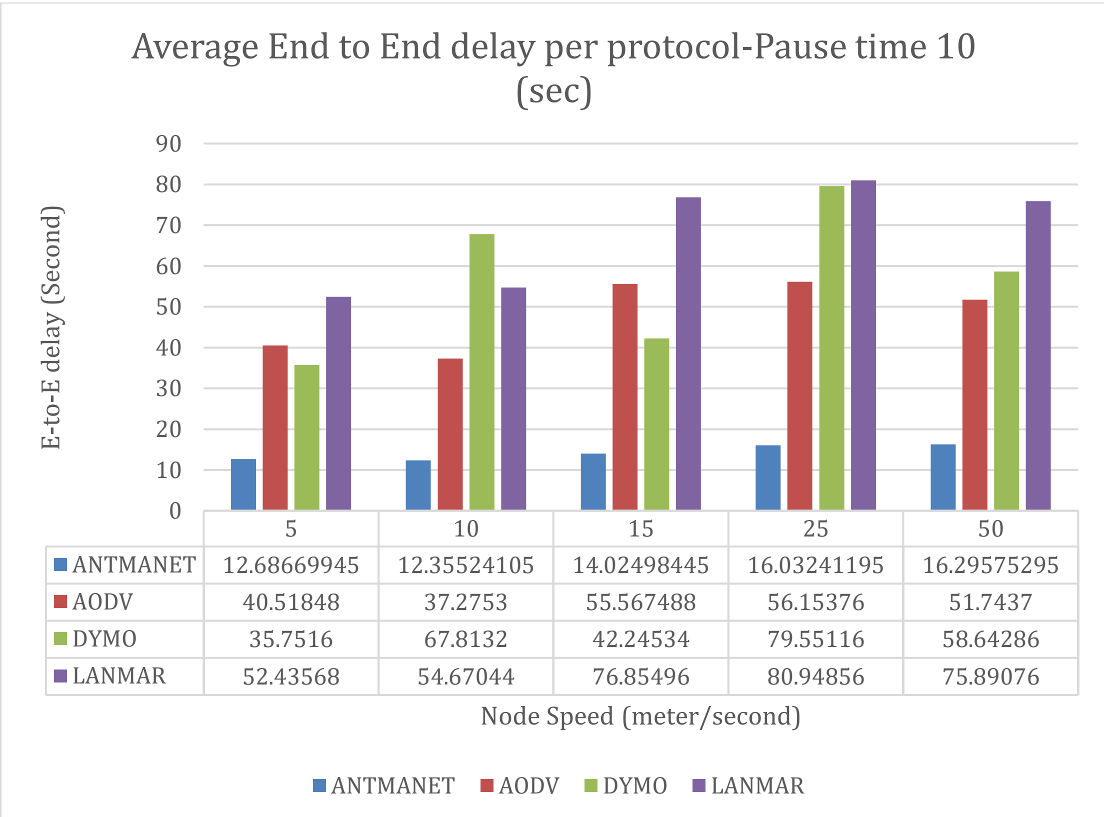
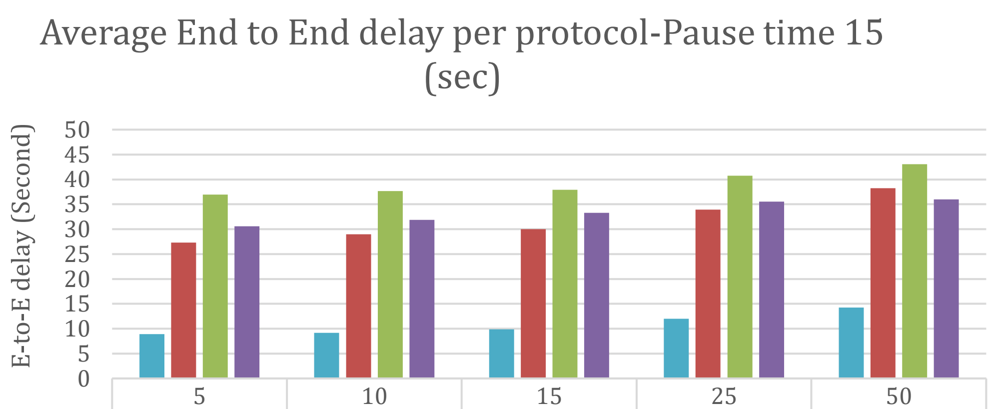


Figure 64: Average End to End delay vs Node Speed per Protocol-Pause time 10 (sec)

Figure 64 illustrates the amount of the average E-to-E delay while the pause time is increased to 10 Seconds, this increase has clearly effected the performance of all protocols and decreased the load on the network infrastructure. ANTMANET has the lowest average E-to-E delay that is illustrated in all node speed categories. Although DYMO has a behaviour like AODV but is still shows high delay of all protocols in most categories.

In terms of, ANTMANET performance it has the lowest delay in all categories on this as opposed to exactly 37 seconds delay of AODV network and 64 seconds delay of DYMO, which is (50% and 65% delay increase respectively) in the first category. And in the second category, the delay increased as expected in all protocols, where the delay increase of ANTMANET was 30% of the previous category opposed to 25% increase in the delay of the AODV network and 30% increase in the LANMAR network delay.



	5	10	15	25	50
ANTMANET	8.9155	9.195991639	9.90425	12.0269	14.24806
AODV	27.27435405	28.98074528	30.01395405	33.90589315	38.2438018
DYMO	36.94583857	37.66041097	37.90469	40.75504657	43.063208
LANMAR	30.55860178	31.88720178	33.24922208	35.52970178	35.99305678

Figure 65: Average End to End delay vs Node Speed per Protocol-Pause time 15 (sec)

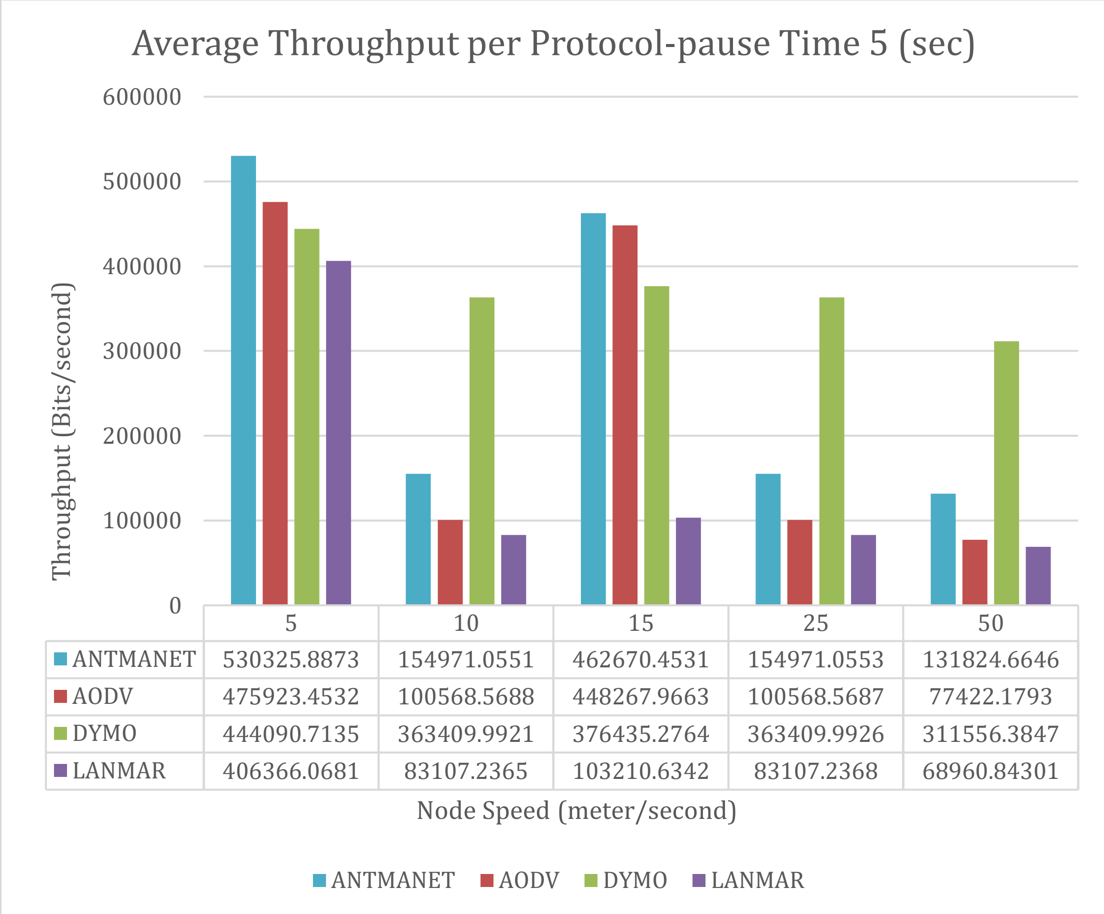
Figure 65 highlights the expenditure of the average E-to-E delay while the pause time has increased again to 15 seconds, again the effect of this increase is quite clear on the performance of the proposed protocols. LANMAR has illustrated a significant improve of it delay performance as the pause time increased to 15 (sec), but this was not enough to outperformed ANTMANET that is showing significantly low delay. In every experiment, this scenario has shown the best result ANTMANET offered.

5.3.3.2 Throughput

Generally, it is difficult to achieve both high throughput and low packet delay. Theoretically in mobile Ad-Hoc networks. Gupta and Kumar (Gupta & Kumar 2000) show the average available throughput per node decreases as $1/\sqrt{n}$ or $1/\sqrt{(n\ lg\ n)}$ in a static Ad-Hoc network, where n is the number of nodes.

Overall, ANTMANET has offered and effective routing by achieving the objective through exploiting the patterns in the mobility of nodes. Furthermore, it is evident that the proposed routing protocol can operate to keep average delay per packet as low as possible for any given level of mobility. While this is easier said than done, it provides a clear-cut objective, which is expressed by its structure and design (refer to chapter 4). This section illustrates the proposed protocol’s performance in terms of the throughputs.

Figure 66, Figure 67 and Figure 68 consist of Five sub- figures showing the tested protocols following the same colour code as in the previous figures –legend is in the top of the graph- illustrating the number of packets that been received by the destination. ANTMANET shows higher throughputs that goes along with the delay results. In each speed category, the proposed protocol demonstrates well to acceptable results even when it reaches the bottleneck performance level that is after the third speed category.



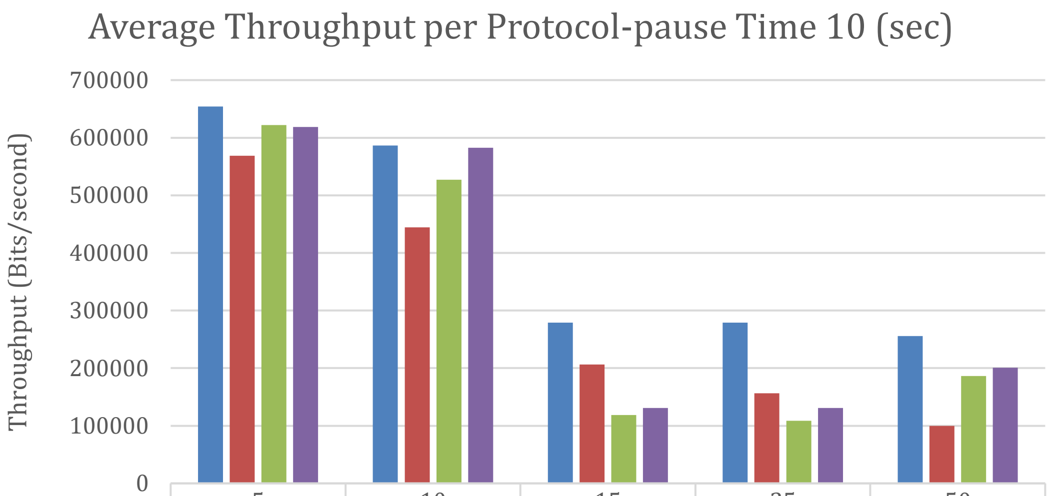
	5	10	15	25	50
ANTMANET	530325.8873	154971.0551	462670.4531	154971.0553	131824.6646
AODV	475923.4532	100568.5688	448267.9663	100568.5687	77422.1793
DYMO	444090.7135	363409.9921	376435.2764	363409.9926	311556.3847
LANMAR	406366.0681	83107.2365	103210.6342	83107.2368	68960.84301

Figure 66: Average Throughput vs Node Speed per Protocol-Pause Time 5 (sec)

Figure 66 illustrates the average throughput results of the network of each of the tested protocol with a 10 (sec) pause time. ANTMANET throughputs initially show and maintain its value.

ANTMANET has delivered 53023.8873 (packets/sec) that is a 40% of the total network traffic, which is seen as a good result within the high mobility situation. While AODV was successful in delivering only 30% of the total network traffic and LANMAR came last delivering only 15% of the total network traffic.

ANTMANET has illustrated 25% better performance than AODV and 45% better than LANMAR. Looking at the third-speed category ANTMANET and AODV have very similar throughput where LANMAR has a very low throughput.



	5	10	15	25	50
ANTMANET	654302.7314	586647.2973	278947.8992	278947.8996	255801.5121
AODV	568912.4952	444494.8876	206350.1304	156350.1304	99545.1802
DYMO	621726.9941	527009.3864	118773.9888	108773.9888	186178.9376
LANMAR	618700.4234	582748.3558	130739.1384	130739.1384	200648.8327

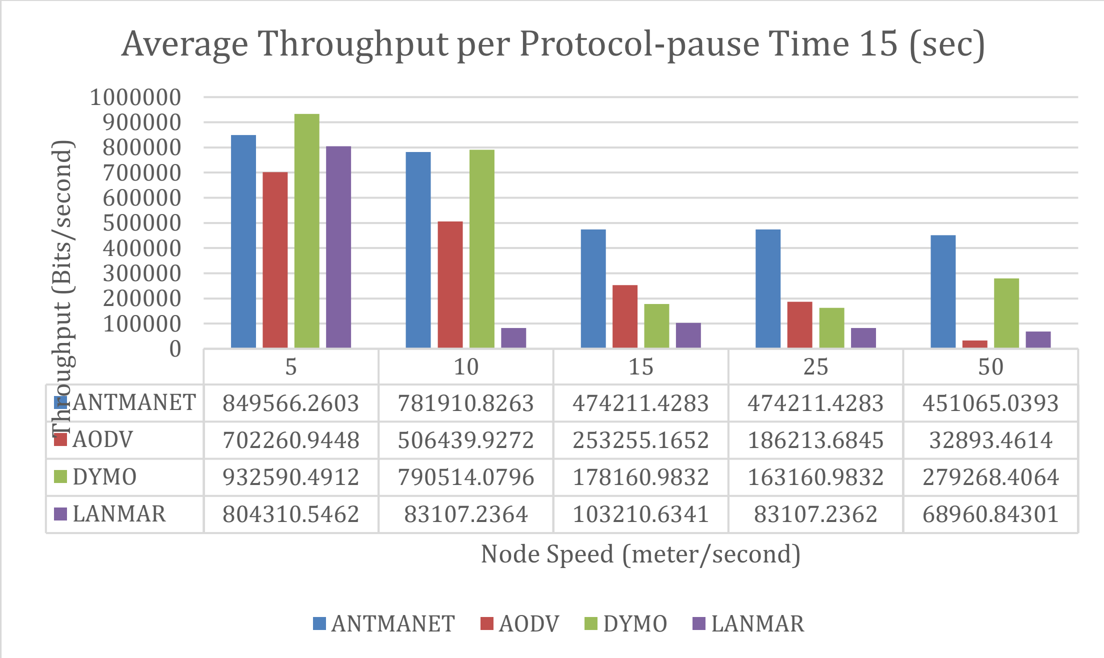
Node Speed (meter/second)

ANTMANET AODV DYMO LANMAR

Figure 67: Average Throughput vs Node Speed per Protocol-Pause Time 20 (sec)

Figure 67 illustrates the average throughput results of the network of each of the tested protocol with a 10 (sec) pause time.

Under this network conditions, ANTMANET shows visible advantage on the standard MANET protocols. ANTMANET has illustrated 30% better performance than AODV and 25% better than LANMAR. Looking at the third-speed category ANTMANET has clearly improved where it has maintained steady and robust performance unlike and AODV and DYMO the performance bottleneck of the4 proposed protocol is now in the fourth-speed category where it witnesses a huge drop of the measurement, yet it is still better than the standard tested protocols.



	5	10	15	25	50
ANTMANET	849566.2603	781910.8263	474211.4283	474211.4283	451065.0393
AODV	702260.9448	506439.9272	253255.1652	186213.6845	32893.4614
DYMO	932590.4912	790514.0796	178160.9832	163160.9832	279268.4064
LANMAR	804310.5462	83107.2364	103210.6341	83107.2362	68960.84301

Figure 68: Average Throughput vs Node Speed per Protocol-Pause Time 15 (sec)

Figure 68 illustrates the average throughput results of the network of each of the tested protocol with a 15 (sec) pause time.

Looking at the first category, it is noticeable that DYMO has outperformed the proposed protocol by 5%. But, at the same time, ANMANET performance has enhanced by 7% compared to the previous network conditions. LANMAR performance has clearly improved in the first category and the performance has fallen behind, one reason of this poor performance comes to the nature of the algorithm that is based on using the landscape information to calculate the routing cost. On the other hand, DYMO performance has noticeable improved.

Looking at the second and the third category, LANMAR performance has dropped leaving ANTMANET in the lead. Most importantly ANTMANET has an evident advantage of all tested protocols in the third category as it represents the performance bottleneck of all three protocols where ANTMANET has 60% better performance that LANMAR.

5.3.3.3 Network Overhead

Figure 72, Figure 73, Figure 74 consist of Five sub- figures showing the tested protocols following the same colour code as in the previous figures –legend is in the top of the graph- illustrating the routing message overhead that is calculated as the total number of control packets transmitted. The increase in the routing message overhead reduces the performance of the Ad-Hoc network as it consumes portions from the bandwidth available to transfer data between the nodes.

Overall, ANTMANET has been successful in maintaining a high level of stability and robustness in terms of the network overhead results. It has shown the lowest use of the control packets in each category in all three experimental conditions. ANTMANET performance has improved while the pause time increased. The proposed protocol has shown steady behaviour especially in the fourth and fifth speed category, which implies that the node speed did not force the protocol to use more control packets to maintain routing information.

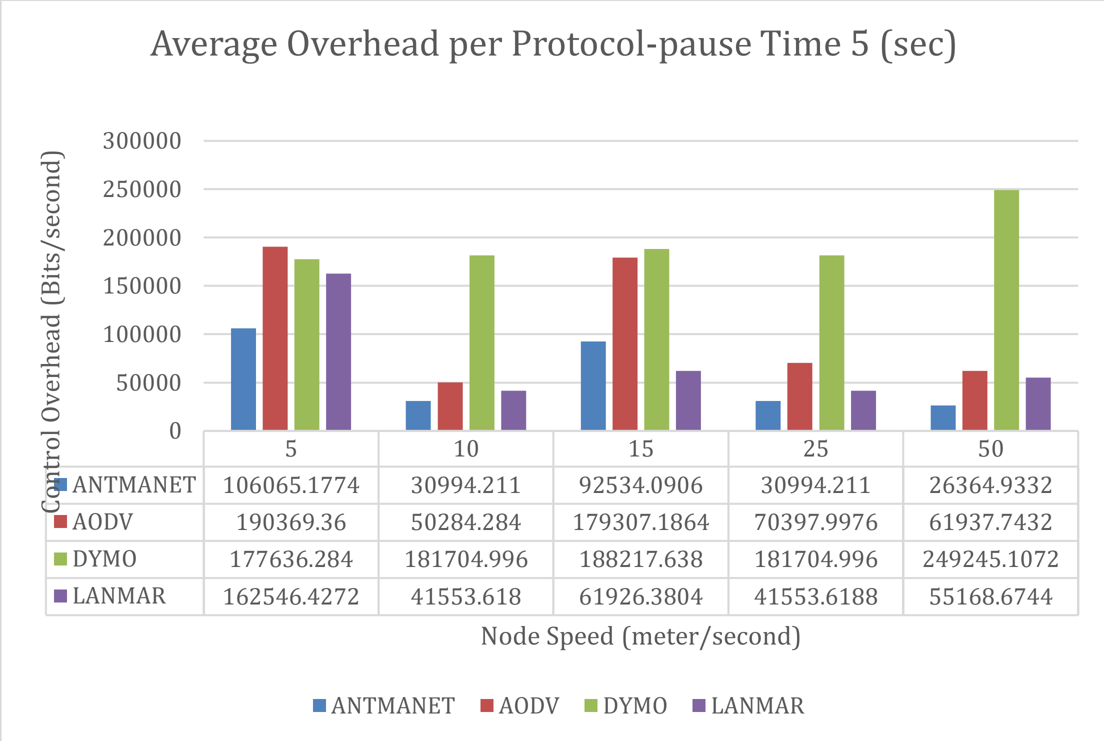
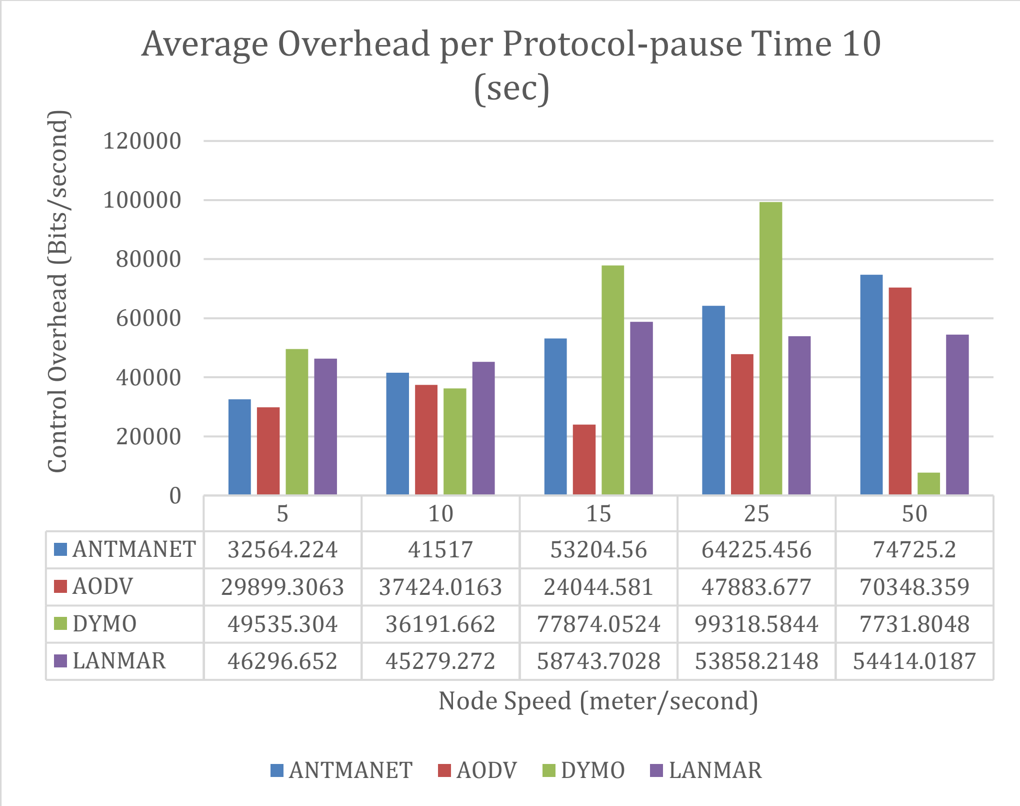


Figure 69: Average Overhead Vs Node Speed per Protocol-Pause time 10 (sec)

Figure 69 illustrates the network Overhead results of the tested protocols when the pause time is 5 (sec). This condition is the extreme scenario of all the proposed scenarios. That is because it stresses the network to the limit examining different levels of speeds from low to extremely high. ANTMANET has shown a very steady behaviour throughout each category. ANTMANET has generated control packets 30% less than AODV and this percentage increased to 55% during the third, fourth and the fifth category. ANTMANET has clear advantage point when to compare LANMAR and DYMO.



	5	10	15	25	50
ANTMANET	32564.224	41517	53204.56	64225.456	74725.2
AODV	29899.3063	37424.0163	24044.581	47883.677	70348.359
DYMO	49535.304	36191.662	77874.0524	99318.5844	7731.8048
LANMAR	46296.652	45279.272	58743.7028	53858.2148	54414.0187

Figure 70: Average Overhead Vs Node Speed per Protocol-Pause time 10 (sec)

Figure 70 illustrates the network Overhead results of the tested protocols when the pause time is 10 (sec). once more, ANTMANET has scored a steady performance in each category. ANTMANET has outperformed LANMAR by 35% and DYMO by 15%. The tested protocols have performed as expected, where LANAMR performance has shown some improvement in its performance. This is expected as LANAMR becomes more effective when the nodes stay stationary for longer. It has improved by 15% when compared with its performance in Figure 45.

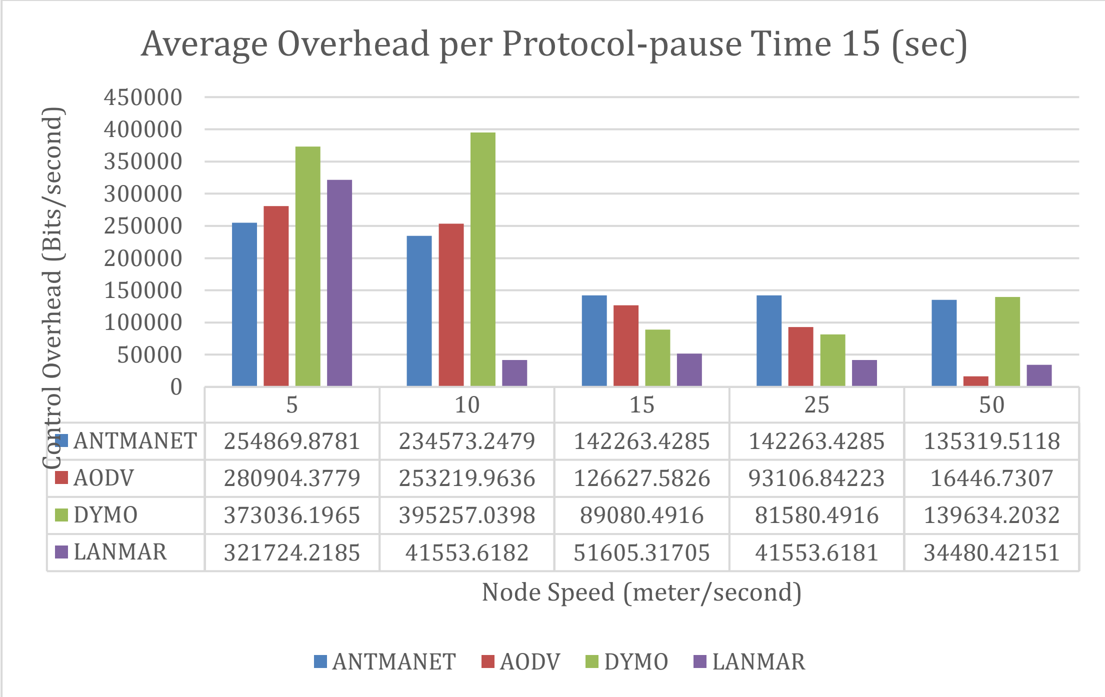


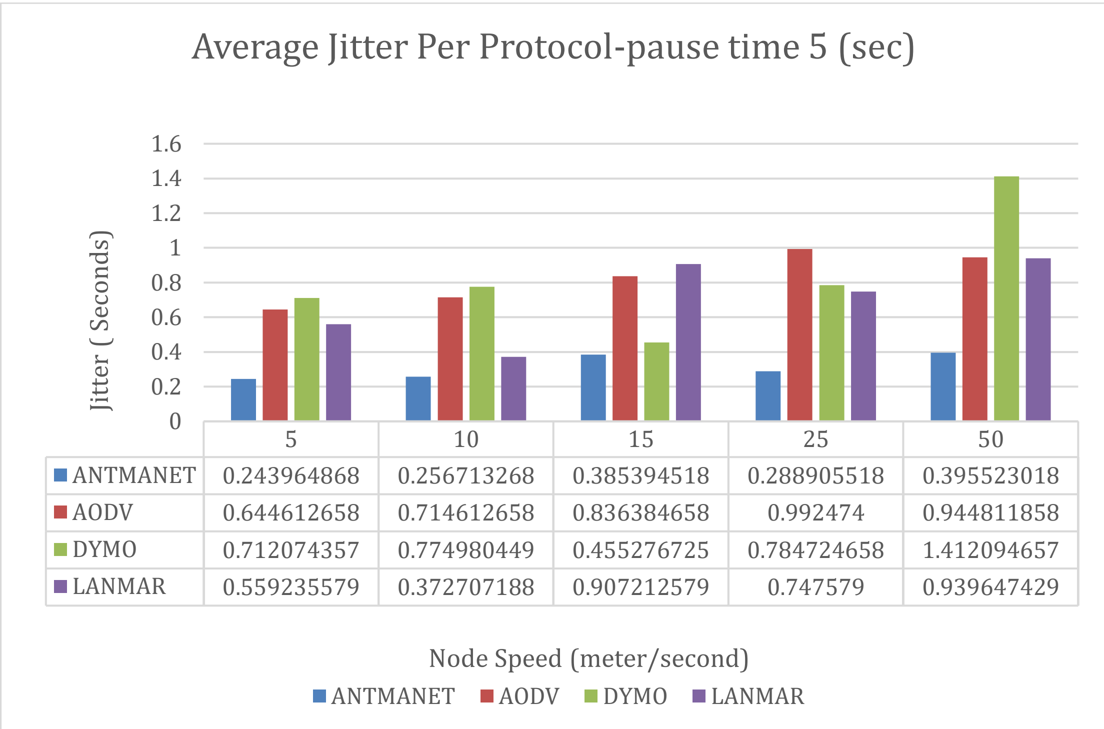
Figure 71: Average Overhead Vs Node Speed per Protocol-Pause time 15 (sec)

Figure 71 illustrates the network Overhead results of the tested protocols when the pause time is 15 (sec). Generally, all protocols have followed the same behaviour patterns in each category as in Figure 45 and Figure 46 but they all has sent in this scenario less control packets than the previous scenarios, this is due to the fact that this scenario has the longest pause time. This allows all protocols to reduce the usage of the communication medium and improve its behaviour. ANTMANET has illustrated its best behaviour in this scenario. ANTMANET has improved its performance by 15% and has maintained steady performance in all categories. This is very important along with the throughputs results as it clearly proves that the proposed protocol is delivering data packets more than control packets.

5.3.3.4 Jitter

Figure 72, Figure 73 and Figure 74 consist of five sub- figures showing the tested protocols following the same colour code as in the previous figures –legend is in the top of the graph- illustrating the variation in the delay of received packets.

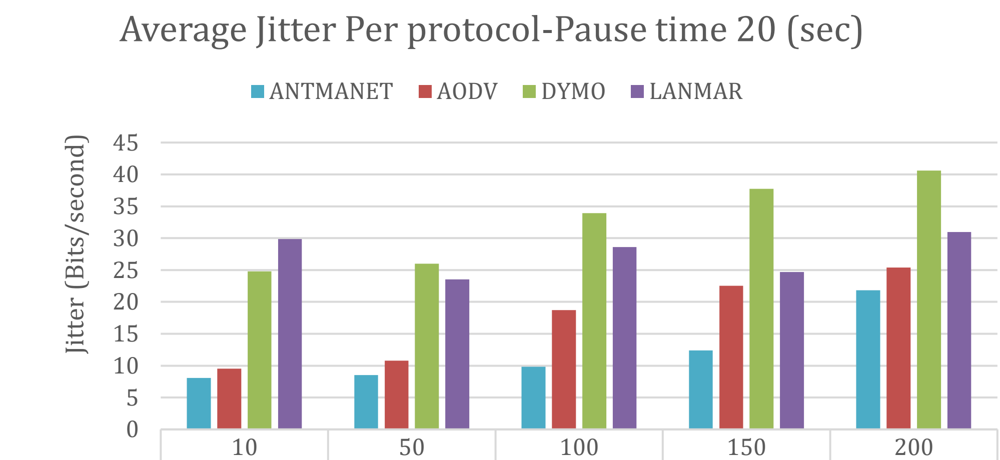
Overall, in terms of the jitter measurement ANTMANET has shown the lowest measurement compared to the tested protocols. As in E-to E delay measurement the proposed protocol has shown stability and robustness even in the extreme network situation. These results represent a clear evidence that the proposed protocol can handle to operate in real-time applications.



	5	10	15	25	50
ANTMANET	0.243964868	0.256713268	0.385394518	0.288905518	0.395523018
AODV	0.644612658	0.714612658	0.836384658	0.992474	0.944811858
DYMO	0.712074357	0.774980449	0.455276725	0.784724658	1.412094657
LANMAR	0.559235579	0.372707188	0.907212579	0.747579	0.939647429

Figure 72: Average Jitter Vs Node Speed per Protocol-Pause time 10 (sec)

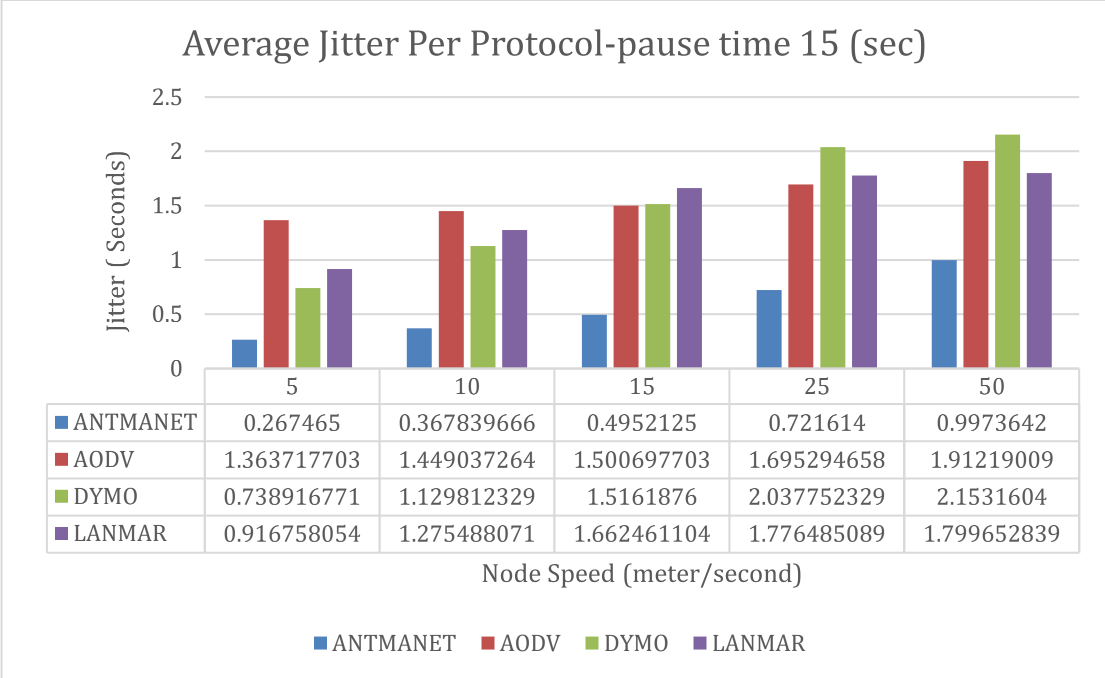
Figure 72 Higher value of jitter can lead to many problems ranging from lip-sync errors to the loss of packets because of buffer overflow or underflow (Hakak, Latif, et al. 2014). By examining the first speed component, ANTMANET has low jitter that is 0.2 (sec) this result is 5 % less than AODV, 6% less than DYMO and 75% less than LANMAR. The performance of ANTMNET depredates by 5 % while the speed of the nodes increases. This is considered as a good performance compared to 10% in DYMO and unexpected performance of AODV as the results increases and decreases for no justified reason. This indicates that AODV is not suitable for real-time application in such stressed network.



	10	50	100	150	200
ANTMANET	8.06203797	8.523500861	9.838566917	12.41414594	21.82637888
AODV	9.544633	10.770776	18.709616	22.50675652	25.37609452
DYMO	24.77028971	25.99643271	33.93527271	37.73241323	40.60175123
LANMAR	29.87718298	23.52520948	28.62528598	24.66380998	30.97069098

Figure 73: Average Jitter Vs Node Speed Per Protocol-Pause time 20 (sec)

Figure 73 illustrates the jitter measurements for the tested protocols when the pause time is 20 (sec). As expected ANTMANET has the lowest jitter in all categories. It stats at the lowest measurement in the first speed category with 0.63 (sec) and then as the speed increases the jitter depredates as well. In very high-speed jitter is 0.35 (sec). Compared to AODV that starts with 2 (sec) delay, which then experience a 50% increase in the third category.



	5	10	15	25	50
ANTMANET	0.267465	0.367839666	0.4952125	0.721614	0.9973642
AODV	1.363717703	1.449037264	1.500697703	1.695294658	1.91219009
DYMO	0.738916771	1.129812329	1.5161876	2.037752329	2.1531604
LANMAR	0.916758054	1.275488071	1.662461104	1.776485089	1.799652839

Figure 74: Average Jitter Vs Node Speed Per Protocol-Pause time 30 (sec)

Figure 74 illustrates the jitter measurements for the tested protocols when the pause time is 15 (sec). As expected ANTMANET has the lowest jitter in all categories. It starts at the lowest measurement in the first speed category with 0.2 (sec) and then as the speed increases the jitter degrade as well. In the high speed jitter is 0.9 (sec). Compared to AODV that starts with 1 (sec) delay, which is 50% more than ANTMANET and ends with 4 (sec). LANMAR has lowest jitter 2.5 (sec) and scored 3 (sec) as the highest measurement.

5.4. **Summary**

In this chapter, the evaluation experiment design was discussed in detail and the significant of the simulation attributes is illustrated. the experiment was based on many simulation scenarios to benchmark the ANTMANET against some of the standard well-defined MANET protocols. The performance results of the proposed protocol were satisfactory and explained in the chapter. The following tables summarise how well the proposed protocol done compared to the protocols in question in all cases.

Table 17: Scenario A- 4 packets- Performance Parameters - ANTMANET vs the Standard Protocols -

Standard protocol	Performance metrics	Leve 1	Level 2	Level 3
AODV	Delay	57%	50%	31%
	throughputs	72%	32%	32%
	Overhead	42%	32%	27%
	Jitter	50%	51%	45%
DYMO	Delay	49%	25%	25%
	throughputs	55%	46%	52%
	Over head	40%	37%	30%
	Jitter	53%	60%	39%
LANMAR	Delay	40%	26%	43%
	throughputs	36%	38%	52%
	Overhead	37%	33%	31%
	Jitter	43%	70%	50%

Table 18: Scenario B- 8 packets- Performance Parameters - ANTMANET vs the Standard Protocols -

Standard protocol	Performance metrics	Leve 1	Level 2	Level 3
AODV	Delay	64%	42%	52%
	throughputs	72%	58%	54%
	Overhead	27%	21%	57%
	Jitter	45%	41%	44%
DYMO	Delay	45%	48%	24%
	throughputs	63%	33%	36%
	Overhead	32%	18%	47%
	Jitter	35%	33%	40%
LANMAR	Delay	41%	72%	26%
	throughputs	36%	78%	33%
	Overhead	23%	26%	50%
	Jitter	33%	34%	39%

Table 19: Scenario C- 12 packets- Performance Parameters - ANTMANET vs the Standard Protocols -

Standard protocol	Performance metrics	Leve 1	Level 2	Level 3
AODV	Delay	53%	45%	45%
	throughputs	75%	66%	58%
	Overhead	80%	64%	54%
	Jitter	39%	71%	68%
DYMO	Delay	46%	37%	36%
	throughputs	61%	68%	56%
	Overhead	60%	53%	45%
	Jitter	51%	36%	40%
LANMAR	Delay	58%	41%	43%
	throughputs	52%	73%	32%
	Overhead	76%	75%	66%
	Jitter	45%	44%	50%

The following chapter illustrates a simulation comparison to study the performance of the proposed protocol when compared to another ACO based protocol.

Chapter 6. : ACO Based Protocols Comparison

Overview

This chapter illustrates a comparison study of the ANTMANET and another ACO based protocol. The protocols in question are faced with the challenge of producing better routing solution under a high mobility environment. In recent years, a number of new ACO based protocols of different styles have been proposed for Ad-Hoc networks. However, systematic performance evaluations and comparative analysis of these protocols in a common realistic environment have not yet been performed. In this chapter, a set of simulation scenarios representative to a mobile MANET is conducted in order to benchmark the proposed protocol against another ACO based protocol.

The remainder of this chapter is organised as follows: section 6.2 illustrates detailed experiment a design, followed by section 6.3 that presents details on the performance experiment. A chapter summary is given in section 6.4

6.1. Introduction

The measurement of packet level performance in terms of packet loss or delays is a challenging open problem in the computer networks medium as it facilitates a better understanding of network and application characteristics. The proposed protocol has been evaluated and benchmarked against several standard protocols and it is showing advantage point over them. It is therefore important to implement a network performance experiment to evaluate the proposed protocol performance against another ACO based protocol, ANTHOCNET. This experiment has been designed based on standard simulation attributes used by the majority of (Kumar & Rajesh 2009; Mbarushimana & Shahrabi 2007; Gopi et al. 2015; Loo et al. 2016). The following sections describe the simulation attributes and experiments.

6.2. **Experiment Design**

The network scenario that is considered in this experiment consists of all nodes able to passively move and they cannot control their movements. The communication strategy used in this simulation considers different paths for each pair source-destination nodes and the best path is selected to be used for data communication. The choice of the best path is based on a metric. Specifically, in this context we consider the path whose nodes must travel the total minimum distance to reach the evenly spaced positions on the straight line between the source-destination pair. All the simulation attributes are discussed in detail in Chapter 5.

This experiment considers a network of 30 nodes placed randomly within an area of $1500(m^2)$.[4] The data application used is the Constant Bit Rate (CBR) with realistic network load of 4 bits/sec. all the 30 have the same hardware aspects and are involved in providing the necessary support for routing and data forwarding over the on-going communication session, the following Table 20 summarises the simulation attributes.

Table 20: Scenario attributes.

Parameters	Value
Experiment time	3 H
Number of nodes	30
Terrain size	1500 m x1500 m
Application	CBR
Packet Size	512 bit
Number of packets (packet/s)	4,8,12
Mobility Model	Random Way Point
Pause time (sec)	30
Speed (m/sec)	5,10,15,25,50
Propagation model	Free Space
Channel frequency	2.4 GHz
Radio type	Accumulated noise model
Network protocol	IPv4

[4] These choices were based on the testbed in reference (Anon 2012)

6.3. **Results and Analysis**

The network condition examined is when the CBR application is generating traffic at the rate of 4 packets/sec, which is considered the most realistic packet rate of a MANET network because of the low bandwidth and the energy restraints of such network (Paul 2016). To explain the data collected of the simulation exterminate many bar graphs are charts used to represent the performance of all tested protocols. Bar graphs are one type of data representation that is different from the histograms. These graphs have x-axis that represents a different category of data- in this case, the different five node speeds- (5, 10, 15, 25, 50 (m/sec)) and y-axis that is the numerical values which, represents the collected data- in this case the average performance metrics of ten different seed simulations. Each category displays the performance of the tested protocols within its conditions. The results are represented in four different coloured bars. Denoted in the blue bar in all graphs is the proposed protocol ANTMANET. The legend of the graph is located on the top under the graph title. All results are presented in the following sections.

6.3.1 Throughputs

Figure 75 shows the throughputs comparison of AODV, ANTMANET and ANTHOCNET. Whereas AODV shows the lowest throughputs in all category, as expected, ANTMANET and ANTHOCNET has very close results.

Overall, ANTMANET has offered an effective routing, achieving the objective through exploiting the patterns of the nodes mobility modules.

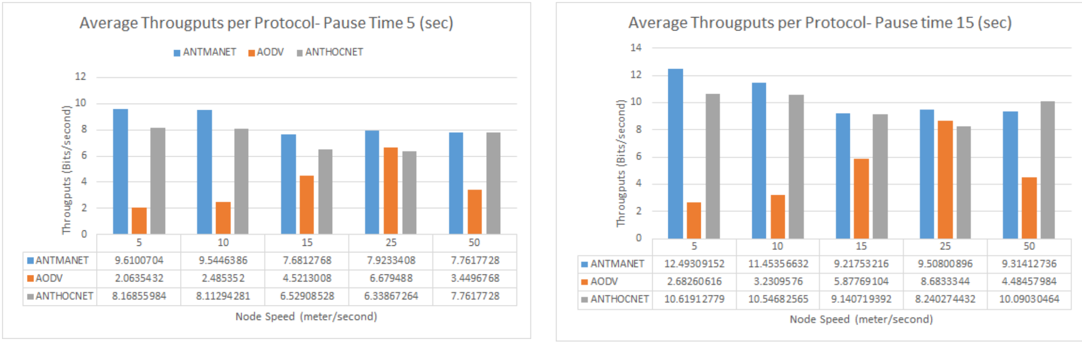


Figure 75: Data delivery ratio vs Node Speed per Protocol-Pause time 15 second.

6.3.2 Control Overhead.

The average network overhead is shown in Figure 76. ANTMANET and ANTHOCNET have yet again shown similar performance. ANTMANET is still at the lead but the behaviour is very like ANTHOCNET. However, coupling with the previous measurement. ANTMANET can outperform ANTHOCNET in overall period.

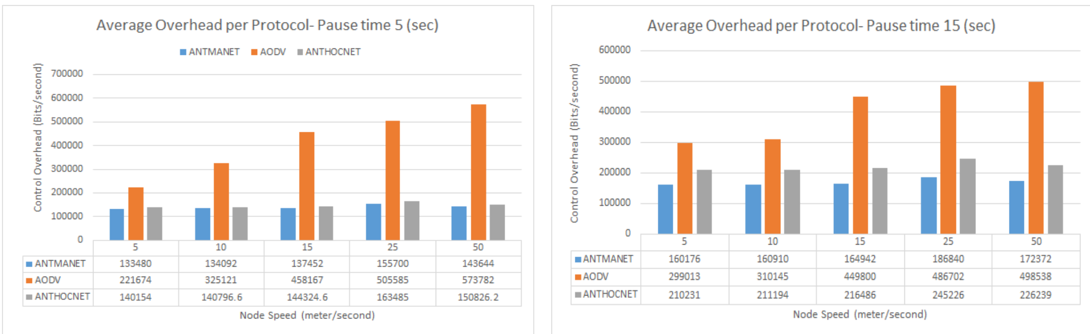
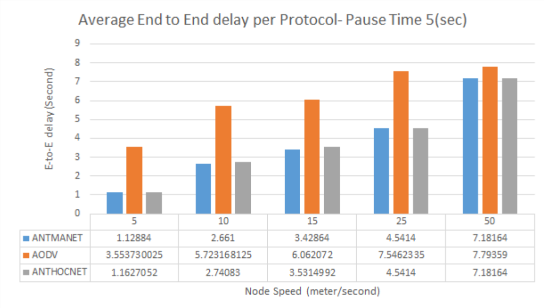


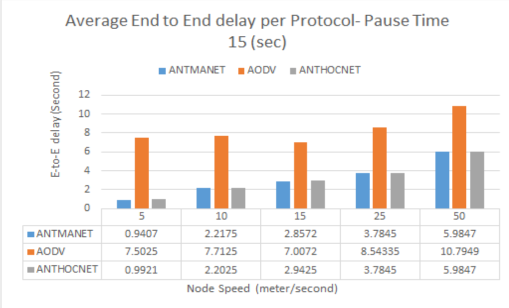
Figure 76: Network Control Overhead Vs Node Speed per Protocol-pause time 5 and 15 seconds

6.3.3 End to End delay

Figure 77 shows End-to-End delay. ANTMANET and ANTHOCNET are using the same routing algorithm to calculate the shortest path. It is noticeable that the delays of the routes chosen by the probability equation are very similar. Consequently, optimising the ACO algorithm and the mechanism of choosing the shortest path would significantly improve the performance of the ANTMANET, potentially providing it with the edge in performance when compared with any ACO based protocols.



	5	10	15	25	50
ANTMANET	1.12884	2.661	3.42864	4.5414	7.18164
AODV	3.553730025	5.723168125	6.062072	7.5462335	7.79359
ANTHOCNET	1.1627052	2.74083	3.5314992	4.5414	7.18164



	5	10	15	25	50
ANTMANET	0.9407	2.2175	2.8572	3.7845	5.9847
AODV	7.5025	7.7125	7.0072	8.54335	10.7949
ANTHOCNET	0.9921	2.2025	2.9425	3.7845	5.9847

Figure 77: End to End Delay Vs Node Speed per Protocol-pause time 5 and15 seconds

6.4. Summary

This chapter presented a full simulation experiment to evaluate the performance of the proposed protocol, with various network conditions. The network conditions variated three simulation parameters, node speed, pause time and the number of traffic packets sent. The proposed protocol has been benched marked against an ACO based protocol “ANTHOCNET”. The results have shown some advantage but both had a very close performance in most comparison components. This performance is satisfactory to one stand but an optimisation to the ACO algorithm is needed to enhance and improve the performance of the proposed protocol to create the performance edge that is required.

Chapter 7. Conclusion and Future Work

7.1. Thesis Overview

The Mobile Ad-Hoc networks (MANET) architecture has enriched wireless networks with new technologies and mechanisms to facilitate communications between people and devices. However, existing literature has outlined many problems associated with higher level of mobility in MANETs. This thesis has addressed some essential issues occurring when both source and destination node are moving rapidly. These issues are represented in higher network overhead and higher delay.

The solution proposed by this thesis is implementing a new routing protocol based on an Ant algorithm that imitates the behaviour of Ants in the real world. Two techniques embedded in this protocol are Local Zone technique and the North Neighbour Table both takes an advantage of the fact that the nodes can obtain their location information by any means. Both techniques reduced the network overhead during the route discovery phase and reduced the size of the routing table to guarantee faster convergence.

7.2. Novel Contributions

This thesis has proposed ANTMANET that is an ACO based routing protocol for mobile MANET.

ANTMANET performance has been evaluated in a wide range of testing conditions. Experiment conditions a varied number of attributes such as the number of packets generated by nodes, Pause time and node speed. The performance metrics used to evaluate the proposed protocol are end-to-end delay, jitter, network overhead and throughput. All results were collected via a very sophisticated simulation system called QualNet and they corresponded well to the expectations of the protocol designs.

All evaluation results were presented in Chapter (5), in terms of the best performance this category can be divided into four groups, with respect to End-to-End delay- Overall scenarios- the best case was when ANTMANET has had 35% less delay than AODV and LANMAR, 40% less delay than DYMO. Looking at the poorest performance of ANTMANET it shows 10% less delay than AODV, 15% less delay than DYMO and 20% less delay than LANMAR.

In terms of throughputs, ANTMANET in its best performance has delivered 35% more packets than AODV and LANMAR, 20% more than DYMO Looking at the poorest performance of ANTMANET it has delivered 10% more packets than AODV and LANMAR, 8% more than DYMO.

With respect to the network overhead results, ANTMANET has illustrated 45% less overhead than AODV, 25% less than DYMO and 30 % less than LANMAR.

In regard to the Jitter, ANTMANET at its best has shown 30% less jitter than AODV, 27% jitter less than DYMO and 25% less jitter than LANMAR. After the proposed protocol, has demonstrated a huge success in the first stage of the evaluation experiment, the second stage was necessary to understand the advantages of the unique design of the proposed protocol when compared to the existing ACO based protocols.

The proposed protocol has shown a measurable advantage over ANTHOCNET, an alternative ACO based protocol, based on the simulation results. In terms of the End-to- End delay the proposed protocol has shown 7% less delay than ANTHOCNET and at its poorest performance is still shown 2% less delay than its rival protocol. In regard to the network overhead the proposed protocol has shown 10% less overhead than ANTHOCNET and at its worse, it has had 5% less overhead. In terms of the throughputs ANTMANET in its best performance has delivered 25% more packets than ANTHOCNET and at its lowest delivered 10% more packets than the other protocol.

In terms of the End-to-End delay, ANTMANET has shown 15% less delay than ANTHOCNET at its best and 10% less delay in the worst case. In respect of the network overhead the proposed protocol has demonstrated 15% less overhead than it is rival and at the poorest performance it showed 5% less network overhead than ANTHOCNET. Throughputs wise the proposed protocol has delivered 17% more packets than ANTHOCNET in the best case and 8% in the worst case. Optimising the algorithm has improved the performance of the proposed protocol by at least 2% in each metrics and 15% at its best.

7.3. **Future Work**

Here we point out some future research directions are indicated that are relevant for the work presented in this thesis. These concerns the deployment and testing of ANTMANET in hardware testbeds, the support of energy efficiency issues in MANET, the use of the ANTMANET in other types of networks, and the application of other ideas from the Internet of Things (IoT) field.

7.3.1 Algorithm optimisation

The Ant Colony Optimisation (ACO) heuristic is a very promising area of research in which the behaviour of a single agent, called an artificial ant or ant for short in the following, is inspired by imitating the behaviour of real ants (Dorigo & Di Caro 1999a). To improve the performance of the proposed protocol an optimization technique needs to be applied on the Ant Colony Optimisation (ACO) heuristic to adjusted the parameters to improve areas in the performance as convergence speed and accuracy.

7.3.2 Energy Efficiency

Energy efficiency is one main attribute any battery powered devices are concerned with. Sensor devices are no exception as they in most forms consist of a battery on board with the sensor nodes and it is often extremely complicated to change or recharge batteries for these sensor nodes. Sometimes it is helpful to replace the sensor node rather than recharging them, which comes with a high cost. For this reason, implementing an efficient method to manage the energy consumption is vital to this type of networks, reducing the power wastage in scenarios such as monitoring unattended area can affect the performance of the network and increase the worth of the information gathered.

There are several reasons for wastage of energy in wireless sensor networks, such as:

- Collision: - Sometimes the packet gets corrupted during transmission these packets need to be discarded and re-sent, these lead to increased energy consumption.
- Control Packet Overhead:- Energy is also required for Sending and receiving control packets due to this less useful data packets can be transmitted.
- Idle Listening: - Extra energy is also consumed for Listening to receive possible traffic which is not sent.

7.3.3 Future applications

Distributed monitoring allows new categories of control and evaluation. The recent advances in very-large-scale integration (VLSI), and the micro-electro-mechanical systems (MEMS), as well as in wireless communication technology, have made it possible to manufacture sensor networks where a very large number of very small nodes are scattered across some environment to sense and report to a central node (sink). Such networks have many applications. In military applications, they are used for battlefield surveillance and object tracking. They are used for seismic data collection and reporting, in addition to factories and warehouses for tracking and monitoring. It is also used in monitoring weakness in building structure or vehicles and aeroplanes. The proposed protocol can improve the reliability of the MANET networks in many new applications such as:

More examples are:

- Disaster relief operations
- Biodiversity mapping
- Vehicular ad-hoc networks for high mobility vehicle
- Machine surveillance and preventive maintenance

References

Abdullah, M. & Ehsan, A., 2014. Routing Protocols for Wireless Sensor Networks: Classifications and Challenges. *Journal of Electronics and Communication Engineering Research*, (2.2), pp.05–15.

Abido, M.A., 2005. Analysis and assessment of STATCOM-based damping stabilizers for power system stability enhancement. *Electric Power Systems Research*, 73(2), pp.177–185.

Abido, M.A. & Abdel-Magid, Y.L., 2003. Coordinated design of a PSS and an SVC-based controller to enhance power system stability. *International journal of electrical power & energy systems*, 25(9), pp.695–704.

Abuhmida, M., Radhakrishnan, K. & Wells, I., 2015a. Evaluating the Performance of ANTMANET Protocol for MANET. In *IEEE International Conference on Internet Technologies & Applications*,.

Abuhmida, M., Radhakrishnan, K. & Wells, I., 2015b. Performance Evaluation of Mobile Adhoc Routing Protocols on Wireless Sensor Networks for Environmental Monitoring. In *IEEE UKSim-AMSS 17th International Conference on Computer Modelling and Simulation*. Cambridge, United Kingdom.

AbuKhalil, T., Sobh, T. & Patil, M., 2015. Survey on Decentralized Modular Robots and Control Platforms. In *Innovations and Advances in Computing, Informatics, Systems Sciences, Networking and Engineering*. Springer, pp. 165–175.

Agrawal, D. & Zeng, Q.-A., 2015. *Introduction to wireless and mobile systems*, Cengage Learning.

Akyildiz, I.F. & Kasimoglu, I.H., 2004. Wireless sensor and actor networks: research challenges. *Ad hoc networks*, 2(4), pp.351–367.

Akyildiz, I.F. & Vuran, M.C., 2010. *Wireless sensor networks*, John Wiley & Sons.

Alemdar, H. & Ersoy, C., 2010. Wireless sensor networks for healthcare: A survey. *Computer Networks*, 54, pp.2688–2710.

Anastasi, G. et al., 2009. Energy conservation in wireless sensor networks: A survey. *Ad hoc networks*, 7(3), pp.537–568.

Anon, 2012. Floating Sensor Network. *University of California Berkeley*. Available at: http://float.berkeley.edu.

Arabshahi, P. et al., 2001. Adaptive routing in wireless communication networks using swarm intelligence.

Arora, R. & Rizvi, N.F., A Behavioral Comparison of LAR with AODV And DSR Routing Protocols.

Aweya, J., Ouellette, M. & Montuno, D.Y., 2001. A control theoretic approach to active queue management. *Computer networks*, 36(2), pp.203–235.

Babaoglu, O. et al., 2006. Design Patterns from Biology for Distributed Computing. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, 1(1).

Babaoglu, O. et al., 2005. Design Patterns from Biology for Distributed Computing. In *Proceedings of the European Conference on Complex Systems (ECCS)*. Paris, France.

Bandyopadhyay, S. & Coyle, E.J., 2003. An energy efficient hierarchical clustering algorithm for wireless sensor networks. In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*. IEEE, pp. 1713–1723.

Bansal, M., Rajput, R. & Gupta, G., 1999. Mobile ad hoc networking (MANET): Routing protocol performance issues and evaluation considerations. *The Internet Society*.

Bellavista, P. et al., 2013. Convergence of MANET and WSN in IoT urban scenarios.

Sensors Journal, IEEE, 13(10), pp.3558–3567.

Bettstetter, C., Resta, G. & Santi, P., 2003. The node distribution of the random waypoint mobility model for wireless ad hoc networks. *Mobile Computing, IEEE Transactions on*, 2(3), pp.257–269.

Black, U.D., 2000. *IP routing protocols: RIP, OSPF, BGP, PNNI, and Cisco routing protocols*, Prentice Hall Professional.

Bonabeau, E., Dorigo, M. & Theraulaz, G., 2000. Inspiration for optimization from social insect behaviour. *Nature*, 406(6791), pp.39–42.

Bonabeau, E., Dorigo, M. & Theraulaz, G., 1999. *Swarm intelligence: from natural to artificial systems*, Oxford university press.

Bonabeau, E. & Théraulaz, G., 2000. Swarm smarts. *Scientific American*, 282(3), pp.54–61.

Bruno, E.A. et al., 2015. Global Positioning System Derived Performance Measures Are Responsive Indicators of Physical Activity, Disease and the Success of Clinical Treatments in Domestic Dogs. *PloS one*, 10(2), p.e0117094.

Di Caro, G.A., Ducatelle, F., Heegarden, P., et al., 2004. Models for basic services in Ad-hoc, peer-to-peer and grid networks.

Di Caro, G.A. & Dorigo, M., 1998a. *Ant Colony Routing*,

Di Caro, G.A. & Dorigo, M., 1998b. AntNet: Distributed Stigmergetic Control for Communications Networks. *Journal of Artificial Intelligence Research (JAIR)*, 12(3 & 4), pp.317–365.

Di Caro, G.A., Ducatelle, F. & Gambardella, L.M., 2004a. {AntHocNet}: an Ant-Based Hybrid Routing Algorithm for Mobile Ad Hoc Networks. In *Proceedings of Parallel Problem Solving from Nature (PPSN) VIII*. Lecture Notes in Computer Science. Springer-Verlag, pp. 461–470.

Di Caro, G.A., Ducatelle, F. & Gambardella, L.M., 2008a. *Ant Colony Optimization for Routing in Mobile Ad Hoc Networks in Urban Environments*, Lugano (Switzerland).

Di Caro, G.A., Ducatelle, F. & Gambardella, L.M., 2004b. *AntHocNet: probabilistic multi-path routing in mobile ad hoc networks using ant-like agents*, Lugano, Switzerland.

Di Caro, G.A., Ducatelle, F. & Gambardella, L.M., 2008b. Theory and practice of {Ant Colony Optimization} for routing in dynamic telecommunications networks. In N. Sala & F. Orsucci, eds. *Reflecting interfaces: the complex coevolution of information technology ecosystems*. Idea Group, Hershey, PA, USA, pp. 185–216.

Di Caro, G.A. & Vasilakos, T., 2000. *Ant-SELA: Ant-agents and stochastic automata learn adaptive routing tables for QOD routing in ATM networks*,

Di Caro, G., Ducatelle, F. & Gambardella, L.M., 2005. AntHocNet: an adaptive nature-inspired algorithm for routing in mobile ad hoc networks. *European Transactions on Telecommunications*, 16(5), pp.443–455.

Chakeres, I.D. & Belding-Royer, E.M., 2004. AODV routing protocol implementation design. In *Distributed Computing Systems Workshops, 2004. Proceedings. 24th International Conference on*. pp. 698–703.

Chauhan, R.K. & Dahiya, A., 2012. AODV extension using ant colony optimization for scalable routing in VANETs. *Journal of Emerging Trends in Computing and Information Sciences*, 3(2).

Chun, W. & Tang, W., 2006. Multicasting in Wireless Sensor Networks. In *Optical Internet and Next Generation Network, 2006. COIN-NGNCON 2006. The Joint International Conference on*. IEEE, pp. 251–253.

Clausen, T. et al., 2003. Optimized link state routing protocol (OLSR).

Coiro, A. et al., 2013. Energy-aware traffic engineering: A routing-based distributed solution for connection-oriented IP networks. *Computer Networks*, 57(9), pp.2004–2020.

Divecha, B. et al., 2007. Impact of node mobility on MANET routing protocols models. *Journal of Digital Information Management*, 5(1), p.19.

Dorigo, M., 2006. *Ant Colony Optimization and Swarm Intelligence: 5th International Workshop, ANTS 2006, Brussels, Belgium, September 4-7, 2006, Proceedings*, Springer.

Dorigo, M. et al., 2011. ANTS 2010 Special Issue -- Editorial. *Swarm Intelligence*, 5(3--4), pp.143–147.

Dorigo, M. et al. eds., 2010. Swarm Intelligence, Proceedings of the 7th International Conference ANTS 2010. In LNCS. Springer.

Dorigo, M., Bonabeau, E. & Theraulaz, G., 2000. Ant algorithms and stigmergy. *Future Generation Computer Systems*, 16(8), pp.851–871.

Dorigo, M. & Di Caro, G.A., 1999. Ant Colony Optimization : A New Meta-Heuristic. In *Proceedings of CEC99 - Congress on Evolutionary Computation*. Washington DC.

Dorigo, M., Caro, G. & Gambardella, L., 1999. Ant algorithms for discrete optimization. *Artificial life*, 5(2), pp.137–172.

Dou, C., Chang, Y.H. & Ruan, J.S., 2014. An Area-Based WCDS-Induced Routing Protocol for Large-Scale Wireless Sensor/Mesh Networks. *International Journal of Future Computer and Communication*, 3(5), p.295.

Ducatelle, F. et al., 2009. New task allocation methods for robotic swarms. In *Proceedings of the 9th IEEE/RAS Conference on Autonomous Robot Systems and Competitions (ROBOTICA), Castelo Branco, Portugal, May 8--9*.

Ducatelle, F., Di Caro, G.A. & Gambardella, L.M., 2010. Cooperative self-organization in a heterogeneous swarm robotic system. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2010)*. Portland, Oregon, USA, July 7--11.

Elshakankiri, M. & El-darieby, M., 2016. Implementation of a Random Wireless Sensor Network in an Irregular Shape Large Building. *Journal of Wireless Networking and Communications*, 6(2), pp.40–46.

Farooq, M. & Di Caro, G.A., 2008. Routing Protocols for Next Generation Networks Inspired by Collective Behaviors of Insect Societies: An Overview. In C. Blum & D. Merckle, eds. *Swarm Intelligence: Introduction and Applications*. Natural Computing. Springer.

Farooq, M.U. & Tapus, N., 2014. CAMT: Core assisted multicast tree for ad hoc networks. In *RoEduNet Conference 13th Edition: Networking in Education and Research Joint Event RENAM 8th Conference, 2014*. IEEE, pp. 1–5.

Furnas, G.W., 1981. *The FISHEYE view: A new look at structured files,* Citeseer.

Ghassemian, M. & Aghvami, A.H., 2008. Mobility to improve efficiency of data collection in wireless sensor networks. In *PIMRC*. pp. 1–5.

Gholami, F. & Mahjoob, M.J., 2007. An Investigation of Parameters in Ant Colony Optimization for a Path Optimization Algorithm. In *Mechatronics and Automation, 2007. ICMA 2007. International Conference on*. pp. 463–468.

Giusti, A. et al., 2012. Distributed consensus for interaction between humans and mobile robot swarms. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS), Demonstration track*. Valencia, Spain, June 4--8, pp. 1503–1504.

Golberg, D.E., 1989. Genetic algorithms in search, optimization, and machine learning.

Addion wesley, 1989.

Gopi, A.P. et al., 2015. Designing an Adversarial Model Against Reactive and Proactive Routing Protocols in MANETS: A Comparative Performance Study. *International Journal of Electrical and Computer Engineering*, 5(5).

Guo, J., 2012. Sink Mobility Schemes in Wireless Sensor Networks for Network Lifetime Extension.

Gupta, P. & Kumar, P.R., 2000. The capacity of wireless networks. *Information Theory, IEEE Transactions on*, 46(2), pp.388–404.

Hakak, S., Latif, S.A., et al., 2014. Impact of Key Factors on Average Jitter in MANET. In *Proceedings of the 2014 First International Conference on Systems Informatics, Modelling and Simulation*. IEEE Computer Society, pp. 219–223.

Hakak, S., Anwar, F., et al., 2014. Impact of packet size and node mobility pause time on average end to end delay and Jitter in MANET's. In *Computer and Communication Engineering (ICCCE), 2014 International Conference on*. IEEE, pp. 56–59.

Hao, Z.-F., Cai, R. & Huang, H., 2006. An adaptive parameter control strategy for ACO. In *Machine Learning and Cybernetics, 2006 International Conference on*. IEEE, pp. 203–206.

Heinzelman, W.R. et al., 2000. Energy-efficient communication protocol for wireless microsensor networks. In *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*. pp. 3005–3014. Available at: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=926982.

Hua, E.Y. & Haas, Z.J., 2015. Mobile-Projected Trajectory Algorithm With Velocity-Change Detection for Predicting Residual Link Lifetime in MANET. *IEEE Transactions on Vehicular Technology*, 64(3), pp.1065–1078.

Hui, B.H., 1992. Defense Advanced Research Projects Agency.

Husain, A. & Sharma, S.C., 2015. Analysis of Location and Zone Based Routing in Vanet Using IEEE 802.11 P.

Jacobson, A., 2000. Metrics in ad hoc networks. *Lulea University of Technology*.

Jacquet, P. et al., 2001. Optimized link state routing protocol for ad hoc networks. In *Multi Topic Conference, 2001. IEEE INMIC 2001. Technology for the 21st Century. Proceedings. IEEE International*. pp. 62–68.

Jaikaeo, C. & Shen, C.-C., 2005. Qualnet Tutorial. *Retrieved Jan*, 6, p.2006.

Jamatia, A. et al., 2015. Performance Analysis of Hierarchical and Flat Network Routing Protocols in Wireless Sensor Network Using Ns-2. *International Journal of Modeling and Optimization*, 5(1), p.40.

Johnson, D.B. & Maltz, D.A., 1996. Dynamic source routing in ad hoc wireless networks. In *Mobile computing*. Springer, pp. 153–181.

Jolfaei, M.G. et al., 2016. A hybrid PSS–SSSC GA-stabilization scheme for damping power system small signal oscillations. *International Journal of Electrical Power & Energy Systems*, 75, pp.337–344.

Jubin, J. & Tornow, J.D., 1987. The DARPA packet radio network protocols. *Proceedings of the IEEE*, 75(1), pp.21–32.

Kamali, S. & Opatrny, J., 2007. Posant: A position based ant colony routing algorithm for mobile ad-hoc networks. In *Wireless and Mobile Communications, 2007. ICWMC'07. Third International Conference on*. p. 21.

Karaboga, D. & Akay, B., 2009. A survey: algorithms simulating bee swarm intelligence. *Artificial Intelligence Review*, 31(1–4), pp.61–85.

Karl, H. & Willig, A., 2006. *Protocols and Architectures for Wireless Sensor Networks*,

Kaveh, A. & Khayatazad, M., 2013. Ray optimization for size and shape optimization of truss structures. *Computers & Structures*, 117, pp.82–94.

Kennedy, J., Kennedy, J.F. & Eberhart, R.C., 2001. *Swarm intelligence*, Morgan Kaufmann.

Kim, E.-H. et al., 2014. Probability-based spray and wait protocol in Delay Tolerant Networks. In *Information Networking (ICOIN), 2014 International Conference on*. IEEE, pp. 412–416.

Kim, K., Min, P.S. & Yan, P.Y., 2000. Packet switching apparatus with multi-channel and multi-cast switching functions and packet switching system using the same.

Ko, Y.-B. & Vaidya, N.H., 2000. Location-Aided Routing (LAR) in mobile ad hoc networks. *Wireless Networks*, 6(4), pp.307–321.

Kuila, P. & Jana, P.K., 2012. Energy efficient load-balanced clustering algorithm for wireless sensor networks. *Procedia Technology*, 6, pp.771–777.

Kumar, M.K.J. & Rajesh, R.S., 2009. Performance analysis of MANET routing protocols in different mobility models. *IJCSNS International Journal of Computer Science and Network Security*, 9(2), pp.22–29.

Le, D. Van, Oh, H. & Yoon, S., 2013. RoCoMAR: robots' controllable mobility aided routing and relay architecture for mobile sensor networks. *Sensors*, 13(7), pp.8695–8721.

Li, J., Liu, S. & Wu, S., 2012. A Design of Remote Computer House Monitoring and Control System Based on ZigBee WSN. *International Journal of Advancements in Computing Technology*, 4(12).

Loo, J., Mauri, J.L. & Ortiz, J.H., 2016. *Mobile ad hoc networks: current status and future trends*, CRC Press.

López-Ibánez, M., Stützle, T. & Dorigo, M., 2015. Ant Colony Optimization: A Component-Wise Overview.

Mahajan, R.M. & Bang, A.O., 2015. Study of Various Geographic Position Based Routing Protocols in VANETS. In *National Conference "CONVERGENCE*. p. 28.

Mbarushimana, C. & Shahrabi, A., 2007. Comparative study of reactive and proactive routing protocols performance in mobile ad hoc networks. In *Advanced Information Networking and Applications Workshops, 2007, AINAW'07. 21st International Conference on*. pp. 679–684.

Misra, P. and Enge, P., 2006. Global Positioning System: Signals, Measurements and Performance , Second Edition.

Mojana, M. et al., 2011. An algorithm combining linear programming and an ant system for the sequential ordering problem. In *Proceedings of the Second Annual International Conference on Advanced Topics in Artificial Intelligence (ATAI)*. Singapore.

Moraes, F.G. et al., 2003. A Low Area Overhead Packet-switched Network on Chip: Architecture and Prototyping. In *VLSI-SOC*. pp. 318–323.

de Morais Cordeiro, C. & Agrawal, D.P., 2011. *Ad hoc and sensor networks: theory and applications*, World Scientific.

Moussaoui, A., Semchedine, F. & Boukerram, A., 2014. A link-state QoS routing protocol based on link stability for Mobile Ad hoc Networks. *Journal of Network and Computer Applications*, 39, pp.117–125.

Muralishankar, V.G. & Raj, D.E.G.D.P., 2014. Routing Protocols for MANET: A Literature Survey. *International Journal of Computer Science and Mobile Applications*, 2(3), pp.18–24.

Nagi, J. et al., 2015. Wisdom of the swarm for human-swarm interaction. In *Proceedings of the IEEE International Conference on Robotics and Automation*

(ICRA). Seattle, USA, May 26 -- 30.

Nallaperuma, S., Wagner, M. & Neumann, F., 2015. Analyzing the Effects of Instance Features and Algorithm Parameters for Max Min Ant System and the Traveling Salesperson Problem. *Frontiers in Robotics and AI*, 2, p.18.

Navidi, W. & Camp, T., 2004. Stationary distributions for the random waypoint mobility model. *Mobile Computing, IEEE Transactions on*, 3(1), pp.99–108.

Ni, L.M. & McKinley, P.K., 2000. A survey of wormhole routing techniques in direct networks. In *Readings in computer architecture*. Morgan Kaufmann Publishers Inc., pp. 492–506.

Nikolidakis, S.A. et al., 2013. Energy efficient routing in wireless sensor networks through balanced clustering. *Algorithms*, 6(1), pp.29–42.

Okdem, S. & Karaboga, D., 2009. Routing in wireless sensor networks using an ant colony optimization (ACO) router chip. *Sensors*, 9(2), pp.909–921.

Panda, S. & Padhy, N.P., 2008. Comparison of particle swarm optimization and genetic algorithm for FACTS-based controller design. *Applied soft computing*, 8(4), pp.1418–1427.

Pandey, M.A., 2015. Link Stability in Mobile Ad Hoc Network. *International Journal of Emerging Trends in Science and Technology*, 2(4).

Paul, S., 2016. *Introduction to MANET and Clustering in MANET*, Anchor Academic Publishing.

Pei, G., Gerla, M. & Hong, X., 2000. LANMAR: landmark routing for large scale wireless ad hoc networks with group mobility. In *Proceedings of the 1st ACM international symposium on Mobile ad hoc networking & computing*. pp. 11–18.

Perkins, C.E. & Royer, E.M., 1999. Ad-hoc on-demand distance vector routing. In *Mobile Computing Systems and Applications, 1999. Proceedings. WMCSA'99.*

Second IEEE Workshop on. pp. 90–100.

Rahman, A.H.A. & Zukarnain, Z.A., 2009. Performance comparison of AODV, DSDV and I-DSDV routing protocols in mobile ad hoc networks. *European Journal of Scientific Research*, 31(4), pp.566–576.

Raju, G.S.V.P. & Murthy, K.S.R., 2015. An Overview of Classification in AD HOC Routing Protocols. *International Journal*, 3(5).

Ramírez, J.M. & Castillo, I., 2004. PSS and FDS simultaneous tuning. *Electric Power Systems Research*, 68(1), pp.33–40.

Ravindranath, K. & Rao, K.R., 2015. Energy Optimized Link Selection Algorithm for Mobile Cloud Computing. *Int. J. Advanced Networking and Applications*, 6(5), pp.2446–2451.

Reichenbach, F., Handy, M. & Timmermann, D., 2005. Monitoring the ocean environment with large-area wireless sensor networks. In *Proceedings of the 8th EUROMICRO Conference on Digital System Design*. pp. 57–58.

Reinelt, G., 1991. TSPLIB—A traveling salesman problem library. *ORSA journal on computing*, 3(4), pp.376–384.

Roberts, A.A. & Das, A.X., 2013. Comparison of Performance Parameters of Reactive, Proactive and Hybrid Routing Protocols in RWP Mobility Model through Qualnet. *Journal of Advances in Computer Network*, 1(3).

Romer, K. & Mattern, F., 2004. The design space of wireless sensor networks. *Wireless Communications, IEEE*, 11(6), pp.54–61.

Royer, E.M. & Toh, C.-K., 1999. A review of current routing protocols for ad hoc mobile wireless networks. *Personal Communications, IEEE*, 6(2), pp.46–55.

Saeed, N.H., Abbod, M.F. & Al-Raweshidy, H.S., 2012. MANET routing protocols taxonomy. In *Future Communication Networks (ICFCN), 2012 International*

Conference on. IEEE, pp. 123–128.

Santi, P., 2012. *Mobility Models for Next Generation Wireless Networks: Ad Hoc, Vehicular and Mesh Networks*, John Wiley & Sons.

Santos, R.A. et al., 2015. Analysis of Topological and Geographical Multicast Routing Algorithms onWireless Ad Hoc Networks. *Elektronika ir Elektrotechnika*, 82(2), pp.23–28.

Sarikaya, B., 1993. *Principles of protocol engineering and conformance testing*, Prentice Hall.

Shah, R.C. et al., 2003. Data mules: Modeling and analysis of a three-tier architecture for sparse sensor networks. *Ad Hoc Networks*, 1(2), pp.215–233.

Shanmugam, M.P. et al., 2015. Energy Efficient Mobile Element Based Data Gathering in Wireless Sensor Networks. *Energy*, 3(11).

Sharma, R.M., 2010. Performance Comparison of AODV, DSR and AntHocNet Protocols. *Department of Computer Engineering, NIT Kurukshetra*.

Silva, R., Silva, J.S. & Boavida, F., 2014. Mobility in wireless sensor networks–Survey and proposal. *Computer Communications*, 52, pp.1–20.

Singh, G., 2014. Design and Development of ACO Routing Protocol for MANETs.

Singh, G., Kumar, N. & Verma, A.K., 2014. ANTALG: An Innovative ACO based Routing Algorithm for MANETs. *Journal of Network and Computer Applications*, 45, pp.151–167.

Singh, V.P., 1997. *Kinematic wave modeling in water resources, environmental hydrology*, John Wiley & Sons.

Skiena, S., 1990. Dijkstra's algorithm. *Implementing Discrete Mathematics: Combinatorics and Graph Theory with Mathematica, Reading, MA: Addison-Wesley*, pp.225–227.

Stützle, T. et al., 2012. Parameter adaptation in ant colony optimization. In *Autonomous Search*. Springer, pp. 191–215.

Subramaniam, A., 2003. Power Management In Zone Routing Protocol (ZRP). *University of Central England, Birmingham*.

Sutariya, D. & Kamboj, P., 2014. A SURVEY OF ANT COLONY BASED ROUTING ALGORITHMS FOR MANET. *European Scientific Journal*, 9(10).

Vandenberghe, W., Moerman, I. & Demeester, P., 2011. On the feasibility of utilizing smartphones for vehicular ad hoc networking. In *ITS Telecommunications (ITST), 2011 11th International Conference on*. IEEE, pp. 246–251.

Varšek, A., Urbančič, T. & Filipič, B., 1993. Genetic algorithms in controller design and tuning. *Systems, Man and Cybernetics, IEEE Transactions on*, 23(5), pp.1330–1339.

Vecchio, M. & López-Valcarce, R., 2015. A greedy topology design to accelerate consensus in broadcast wireless sensor networks. *Information Processing Letters*, 115(3), pp.408–413.

Vodnala, D., Aluvala, S. & Kumar, S.P., 2015. A Backbone based multicast routing protocol for route recovery in MANETs. In *Electronics and Communication Systems (ICECS), 2015 2nd International Conference on*. IEEE, pp. 237–241.

Wang, J. et al., 2009. HOPNET: A hybrid ant colony optimization routing algorithm for mobile ad hoc network. *Ad Hoc Networks*, 7(4), pp.690–705.

Wang, Q. & Yang, W., 2007. Energy consumption model for power management in wireless sensor networks. In *2007 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*. IEEE, pp. 142–151.

Yang, X. et al., 2002. Design of a wireless sensor network for long-term, in-situ monitoring of an aqueous environment. *Sensors*, 2(11), pp.455–472.

Yoshikawa, M. & Otani, K., 2010. Ant Colony Optimization Routing Algorithm with Tabu Search. In *proceedings of the International Multiconference of Engineers and Computer Scientists*. pp. 17–19.

Yousefi, S., Mousavi, M.S. & Fathy, M., 2006. Vehicular ad hoc networks (VANETs): challenges and perspectives. In *ITS Telecommunications Proceedings, 2006 6th International Conference on*. IEEE, pp. 761–766.

Yuan, C., Billington, J. & others, 2006. *A Coloured Petri Net Model of the Dynamic MANET on Demand Routing Protocol*. University of Aarhus.

Zhang, Y. et al., 2015. Pathological brain detection in magnetic resonance imaging scanning by wavelet entropy and hybridization of biogeography-based optimization and particle swarm optimization. *Progress in Electromagnetics Research*, 152, pp.41–58.