3-26-2015

Network Routing Using the Network Tasking Order, a Chron Approach

Nicholas J. Paltzer

NETWORK ROUTING USING THE NETWORK TASKING ORDER, A CHRON APPROACH

THESIS

Nicholas J. Paltzer, Captain, USAF

AFIT-ENG-MS-15-M-059

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the United States Government. This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

AFIT-ENG-MS-15-M-059

NETWORK ROUTING USING THE NETWORK TASKING ORDER, A CHRON APPROACH

THESIS

Presented to the Faculty

Department of Electrical and Computer Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Computer Engineering

Nicholas J. Paltzer, B.S.C.P., B.S.C.S.

Captain, USAF

March 2015

AFIT-ENG-MS-15-M-059

NETWORK ROUTING USING THE NETWORK TASKING ORDER, A CHRON APPROACH

Nicholas J. Paltzer, B.S.C.P., B.S.C.S.

Captain, USAF

Committee Membership:

Dr. Kenneth Hopkinson
Chair

Dr. Timothy Lacey
Member

Maj Thomas Dube, PhD
Member

AFIT-ENG-MS-15-M-059

Abstract

This thesis promotes the use of the Network Tasking Order (NTO), in collaboration with the Air Tasking Order (ATO), to optimize routing in Mobile Ad hoc Networks (MANET). The network topology created by airborne platforms is determined ahead of time and network transitions are calculated offline prior to mission execution. This information is used to run maximum multi-commodity flow algorithms offline to optimize network flow and schedule route changes for each network node. These calculations and timely route modifications increases network efficiency. This increased performance is critical to command and control decision making in the battlefield. One test scenario demonstrates near a 100% success rate when route scheduling and splitting network traffic over an emulated MANET compared to Open Shortest Path First (OSPF) which only achieved around a 71% success rate, and Mesh Made Easy (MME) which achieved about 50% success. Another test scenario demonstrates that the NTO can experience degradation due to schedule delay. Overall, if executed and planned properly, the NTO can significantly improve network Quality of Service (QoS).

Acknowledgments

I would like to express my appreciation to my advisor, Dr. Kenneth Hopkinson, for his outstanding support during this academic process. His experience and guidance are greatly appreciated. Additional thanks to my committee members Dr. Timothy Lacey and Major Thomas Dube, this could never have been accomplished without them.

Nicholas J. Paltzer

Table of Contents

List of Figures

Page

List of Tables

Page

List of Abbreviations

		Page
AOC	Air Operations Center	7
AODV	Ad Hoc On Demand Distance Vector Protocol	10
ATO	Air Tasking Order	iv
BATMAN	Better Approach to Mobile Ad-hoc Networks	13
BFS	Breadth First Search	31
CAOC	Combined Air Operations Center	17
CARMA	Context Aware Routing Management Architecture	4
CSAR	Combat Search and Rescue	17
DoD	Department of Defense	1
DRQC	Dynamic Routing Queue Controller	16
GIG	Global Information Grid	1
GPS	Global Positioning System	23
HWMP	Hybrid Wireless Mesh Protocol	43
LSA	Link State Advertisements	12
MANET	Mobile Ad hoc Networks	iv
MDR	MANET Designated Router	12
MME	Mesh Made Easy	iv
NTO	Network Tasking Order	iv
NTP	Network Time Protocol	22
OSPF	Open Shortest Path First	iv
QoS	Quality of Service	iv

RouterOS	MikroTik Router Operating System	35
TADIL	Tactical Data Links	7
UAV	Unmanned Aerial Vehicles	1
UDP	User Datagram Protocol	22

NETWORK ROUTING USING THE NETWORK TASKING ORDER, A CHRON APPROACH

1 Introduction

The importance of computer and information networks continues to increase. Society, businesses, and the military rely heavily on networks for information exchange between personnel and control systems. More specifically, the military has become dependent on computer networks for more than just day to day business, but also for real-time war fighting situational awareness. The integration of computer networks has expanded deeply into combat assets from unmanned aerial vehicles (UAVs) to most recently, surveillance, attack, and fighter aircraft. At this time all aircraft platforms have the capability to support a computer network and share data with other aircraft or ground stations. The Department of Defense (DoD) defines the complete network comprised of all military bases, sites, aircraft, and other end points as the Global Information Grid (GIG). It is assumed that the standard protocols that are used for stationary GIG nodes do operate efficiently enough to route network traffic over mobile nodes. This document presents other routing methods that are presumed to operate much more efficiently than these standard protocols in mobile airborne environments. Furthermore, multiple routing methods are implemented in test scenarios to analyze the performance characteristics of these routing methods.

1.1 Problem

Traditional networks have been in use for years. Network traffic decision algorithms have been in place since the creation of the Internet. These algorithms are successful in redirecting network traffic as the network changes. The problem is that these algorithms do not handle mobile networks where connections have lower bandwidth and are constantly changing. In mobile networks the amount of changes that must occur is substantially greater than the typical Internet connections between cities or corporations. These protocols also depend on the network, or at least a portion of the network, to pass information among nodes to update routing tables. This overhead traffic is not always an option in airborne networks. In airborne networks, the ability to send updates is limited due to how often the network changes and, more importantly, the need to use the bandwidth for operational data instead of network control data.

1.2 Background

During a military operation, the network connections among aircraft are viewed as a complex network. Two different aircraft may maintain a connection for a given period of time, but they must be within a certain distance to maintain a connection. The closer the two aircraft are together, the faster the link operates. As the aircraft separate, the link speed between them decreases. The network that is established during the course of a day in a heavily occupied airspace becomes dynamic and changes often. Each aircraft acts like a network node or router. Therefore each aircraft has the potential to connect to any other aircraft in that airspace. As the links between the aircraft are established or removed from the network a decision is made to change the way data flow from one end of the network to the other. This decision making is critical to network optimization or, more importantly, basic network operation. Figure 1.1 from NSA depicts a global environment where various platforms are able to communicate over network links. These links are established between ground nodes, naval ships, satellites, aircraft, and tactical mobile ground units.

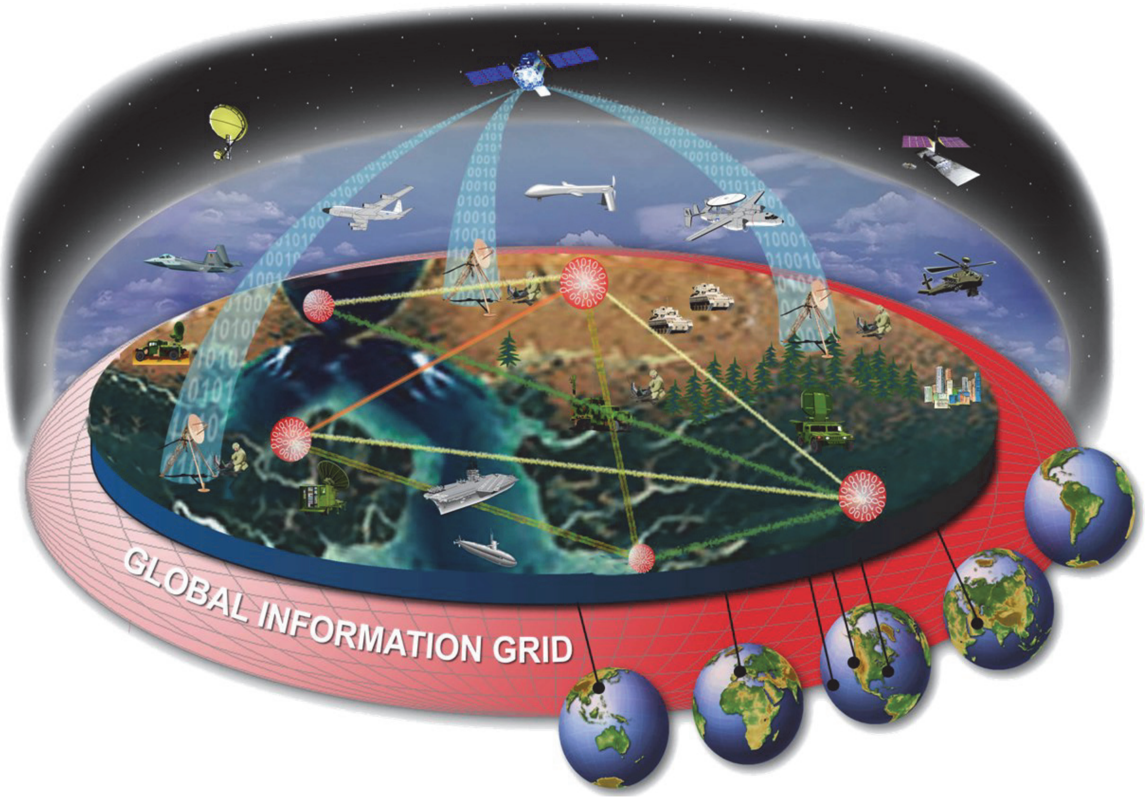


Figure 1.1: Global Information Grid [1] image showing connections between various network platforms.

1.3 Contributions

The ability to use this network by using one aircraft to make a connection for another through the battle space is previously researched. The state of the network must be known ahead of time to be any use to the operation. The air tasking order (ATO) directs the aircraft to a given location at a given time. Using this location and time, the state of the network is determined. The ability to task airborne assets and use the interconnections between these aircraft as a network to pass traffic is known as the network tasking order (NTO) [2]. Although the overall intent of the NTO is to have a method of tasking assets to provide a network to support daily missions, it is further used to predict the network state and make routing decisions to pass traffic through the network. Using the known aircraft location to route network traffic is researched and summarized in a thesis titled, "context aware routing management architecture for airborne networks" (CARMA) [3].

The previous research has had favorable results showing that knowing aircraft location can greatly assist in network traffic flow prediction and routing determination. However, the previous research is conducted using network simulators. These simulators lack many characteristics of an actual network since they are implemented and executed within a single program that simulates many different network nodes and associated interconnections. The network nodes may not operate independently. Additionally, the simulators received control signals from a central control server that had a connection to the network nodes over an independent network connection. In order to determine operational effectiveness, these NTO scenarios are implemented on an operational network.

1.4 Objectives

The goal of this thesis is to implement the previous findings in an operational network. While a router performs a specific function on the network there are many different ways to implement the function of a router. Any standard operating system can act as a router. There are software firewalls, virtual appliances, and even some of the mainstream hardware vendors distribute virtual machines to emulate one of their products. A collection of these devices are established in a virtual environment where the links between them are controlled to emulate a network that would follow the characteristics of an airborne network. Doing so provides independence between the various network devices and proves feasibility in an actual airborne environment.

1.5 Thesis Overview

This chapter provided a summary of airborne networks, the limitations of airborne networks, the limitations of previous research, and the various challenges associated with this topic. Additionally, the objectives of this thesis are summarized. The following is a summary of the contents of this document:

- Chapter 2 provides a literature review of previous relevant research in this area of expertise. This literature is not limited to airborne networks, but also network protocols, virtualization, synchronization, and any other pertinent information.
- Chapter 3 covers the methodology used to assess the airborne network emulation and how success or failure is measured.
- Chapter 4 discusses the results of the experiments.

- Chapter 5 summarizes the experiments, provides any and all conclusions, and provides recommendations for future work.

2 Literature Review

2.1 Chapter Overview

This chapter focuses on previous research or background information already present that pertains to airborne network routing and traffic management. The following background information is presented in a top to bottom approach detailing how an airborne network operates, how the traffic can traverse a constantly changing dynamic network, some methods to control this dynamic network, as well as some critical underlying protocols that assist with the process. First, mobile ad hoc networks (MANET) and network routing protocols are discussed. Then, some advanced prediction techniques that are utilized for traffic routing and management as well as some synchronization techniques are presented. Finally, this chapter focuses on network topology prediction and how priori knowledge is used to enhance network throughput and performance.

2.2 Mobile Ad Hoc Networks

Today, modern technology makes it possible for aircraft to have a network connection to either a ground station, or another aircraft. Modern military aircraft utilize tactical data links (TADIL) as a basic form of network communication. TADILs are radio interconnections between aircraft that allow the sharing of information such as position and other aviation data. The capability to provide increased network connectivity for additional purposes has been developed and implemented. These aircraft use military grade radios to pass mission data to an air operation center (AOC). As these aircraft (nodes) move into range of each other or ground stations they create a connection. If a node is within range of multiple other nodes, either ground or airborne, multiple connections are established. Multiple nodes connected together form a network. These wireless connections between independent systems are typically short lived. They may not have a connection to the Internet as connectivity is isolated within the systems connected. This system of systems is known as a MANET. Figure 2.1 depicts a complicated MANET involving many different types of systems all connected together. MANETs take advantage of the fact that although a system is not able to connect directly to a node that is out of range, also known as the “hidden node problem” in wireless networks. Instead, it can forward the traffic destined for the out of reach system to another system in between. The system in between can then store and forward the data to the destined system. MANET systems typically operate using batteries and, as such, power consumption is a concern. Quality of service (QoS) is another concern with MANETs. QoS is measured

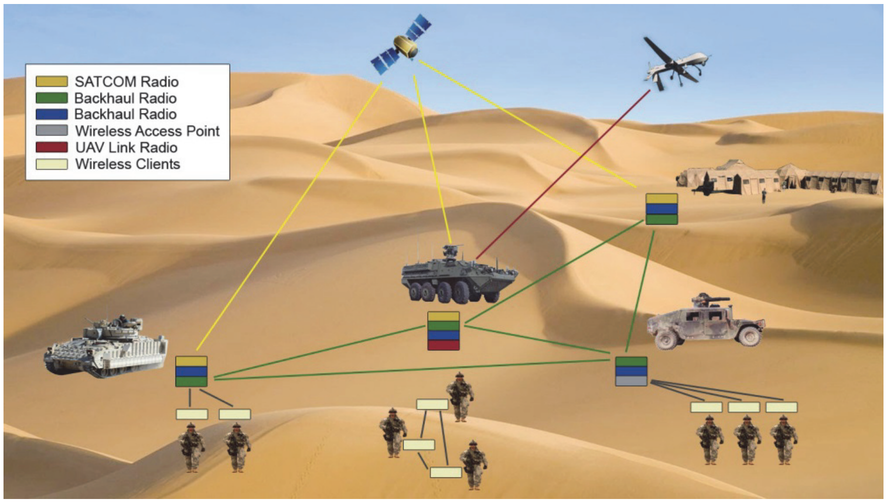


Figure 2.1: Mobile Ad Hoc Network [4]

by the quantity of data that can traverse a network successfully from source to destination.   The limitations of MANETs are summarized by the following [5]:

- The quickly changing network topology that occurs inhibits traffic routing performance as connections are made based on proximity to other devices in the network and routing decision changes are more frequently necessary.
- While wireless connections are error prone, MANETs further complicate this error rate with additional noise, interference, and fading.
- Routing decision algorithms must be extremely lightweight and efficient to minimize CPU, memory, and RF transmission due to the power limitations in the MANET environment.
- MANET wireless connections are not only low bandwidth, but the bandwidth fluctuates as proximity/signal strength increases or decreases.

To ensure optimal Quality of Service, there are three factors that must be determined and re-determined as necessary [6]:

- A loop-free route from source to destination must be established with enough bandwidth to support the required data transfer.
- Upon topology changes, a route can be quickly established to support any current QoS obligations.
- As resources change, such as link speed between nodes, QoS obligations can be maintained by ensuring bandwidth is sufficient for the requirement.

MANETs are becoming extremely popular. Originally they were developed for the military and disaster recovery but have propagated into society for business or personal use. This thesis focuses on MANETs for use by the military and more specifically as an airborne network in battlefield scenarios.

2.3 Dynamic Routing

Routing network traffic in a MANET is challenging and has been researched for many years and some routing protocols have been developed that have had significant improvements in providing a better level of QoS. Some of these protocols are new versions of older internet routing protocols and some have been specifically developed for use in a MANET.

2.3.1 Ad Hoc On-Demand Distance Vector Protocol (AODV)

AODV is a protocol that automates data routing within a MANET. AODV's features are [7]:

- Built for mobile networks
- Creates routes on-demand
- Loop free with quick convergence
- Scales well
- Fits easily in the existing protocol stack

Because AODV has these features, it is easy to implement and therefore the most widely used MANET protocol. Another advantage to AODV is its low memory and processor utilization when computing routes. AODV uses hello messages for nodes to gain knowledge of neighbors. When a request to pass traffic is initiated, AODV initiates a path discovery process. A route request (RREQ) packet is broadcasted by the source node to its immediate neighbors. The neighbors can either fulfil the RREQ by sending a route reply (RREP) or send the RREQ on to its own neighbors. This can result in some intermediate node(s) receiving the RREQ twice, in which case the node discards the duplicate RREQ. Each intermediate node that cannot satisfy the original request still retains the original request information to assist in setting up the connection in the case that the RREP is transmitted back through. Figure 2.2 provides a visualization of the RREQ/RREP process. The information retained are the source and destination IP addresses, the broadcast ID, the expiration time, and the source node's sequence number. It is the retention of the data, and the lack of computation of complicated algorithms, that allows AODV to run with low memory and processor overhead. However, AODV has some disadvantages. The many broadcast RREQs that are utilized consume additional bandwidth to request status across the network. In addition, several routing update requests are sent often due to the highly dynamic airborne network. [8]

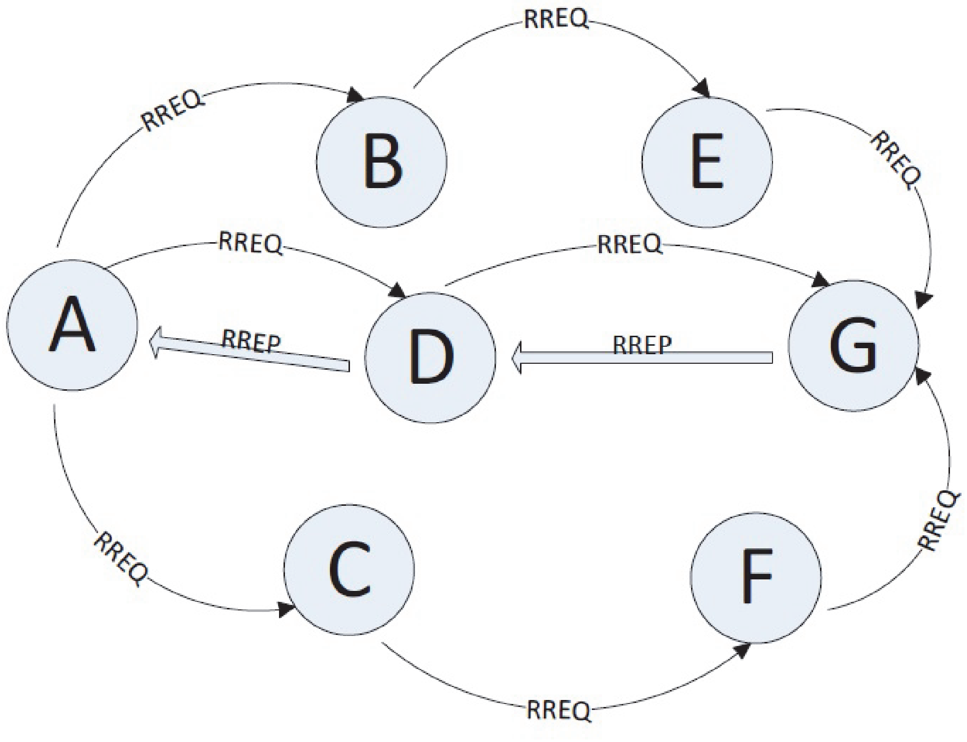


Figure 2.2: Ad Hoc On-Demand Distance Vector Protocol – Node A requests routing information for node G [3].

2.3.2 Open Shortest Path First (OSPF)

OSPF is a de facto standard for use with the original Internet Protocol version 4 (IPv4) networks in enterprise, campus, local area network (LAN), and metropolitan area network (MAN) environments. OSPF is a link state routing protocol, originally introduced as an upgrade to Routing Information Protocol (RIP), a distance-vector protocol, in the mid-90s. OSPF uses areas to define administrative boundaries between other OSPF networks. The advanced features of OSPF make it a more desirable interior routing protocol for many organizations. Version 2 of OSPF (OSPFv2) is the most common occurrence of OSPF for IPv4. OSPF version 3 (OSPFv3) was developed to support Internet Protocol version 6 (IPv6). OSPF was originally designed for wired networks. Since OSPFv3 has been implemented, modifications have been suggested to provide support for MANETs and airborne platforms [9]. Boeing, with the collaboration of the Navy, has developed OSPF using MANET designated routers (MDR), or OSPF-MDR. OSPF-MDR is specifically for use in MANET environments. The MDR version of OSPF reduces flooding of link state advertisements (LSAs). Instead MDRs only flood new LSAs out of receiving interfaces. Retransmission of LSAs to adjacent neighbors is accomplished to ensure reliability. Adjacencies are limited to a subset of neighbors providing better scalability in a much more populated MANET environment. Furthermore, hello packets are transmitted only when reporting changes in neighbor states limiting overuse of scarce network bandwidth. Figure 2.3 shows how complicated a MANET can become and the adjacencies that are formed using OSPF-MDR [10].

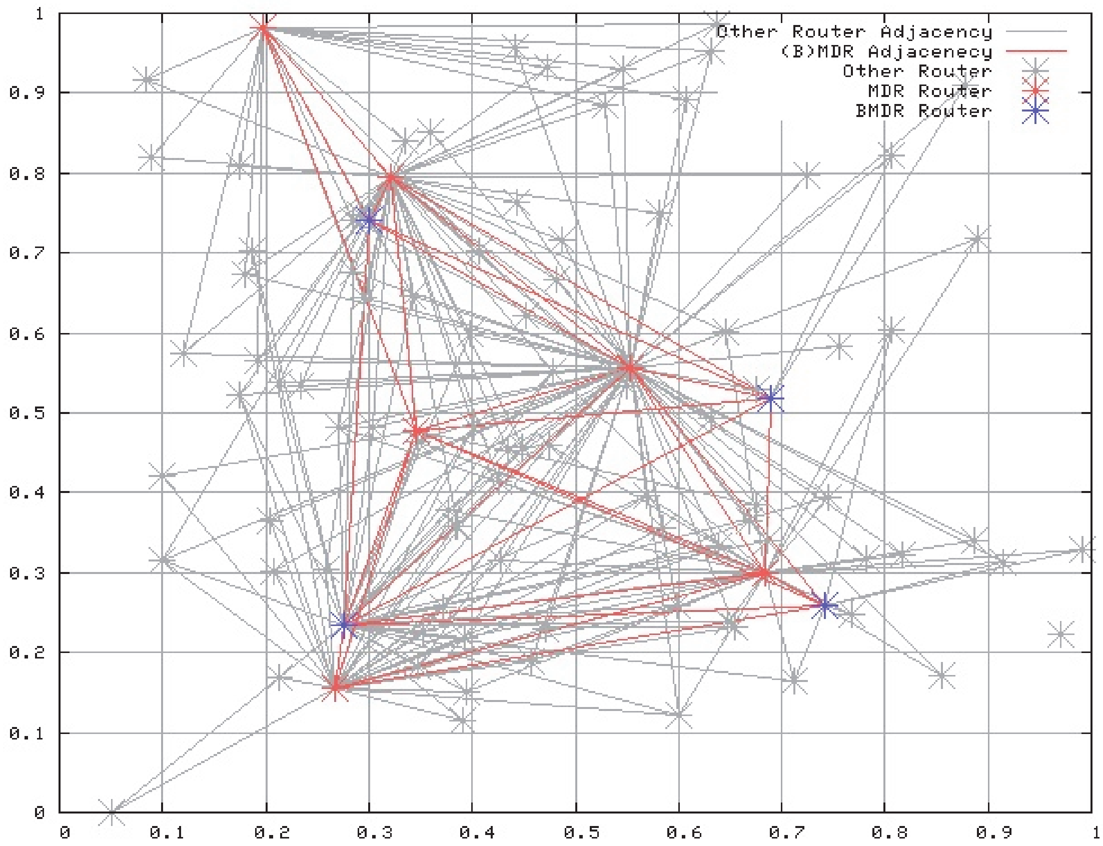


Figure 2.3: A model of a complex MANET forming router adjacencies using OSPF-MDR [11]

2.3.3 Mesh Made Easy (MME)

MME is a proprietary routing protocol developed by MicroTik. MME has been developed to directly support IP level routing in wireless mesh networks. MicroTik has created MME based on the Better Approach to Mobile Ad-hoc Network (BATMAN) routing protocol. MME never retains information regarding the topology of the network, nor does it determine a routing table. Instead, MME tracks sequence numbers from its own generated messages to determine packet loss. It then gathers statistics of lost packets from neighbors or originators to find the best path to the destination. The extra bandwidth consumed can have an adverse effect on the network but is balanced with a constantly changing network where this protocol can have an advantage over OSPF or other link state routing protocols. MME has a single packet format used in the originator message. This message contains the following [12]:

- Originator IP
- Current time to live (TTL) value
- Sequence number
- Gateway class
- Protocol version
- Host and network announcements

These messages are broadcasted throughout the network and there are a set of rules to follow to prohibit rebroadcasting. Using these messages, MME makes routing decisions on no more than the last 64 messages received.

2.4 MANET Prediction Routing

2.4.1 Kalman Filter

The Kalman filter was first conceptualized by Dr. Rudolph E. Kalman in 1960 [13]. The Kalman filter is used to compute an optimal estimate for linear filtering and prediction. Since then, the filter has been used for many research scenarios from radar to computer vision and proven useful for estimating a state of a process. For the purpose of network prediction, the Kalman filter offers a feasible solution as it can predict the future network state with little computational power. Figure 2.4 displays a flow diagram of the Kalman filter being used for measurement prediction. The Kalman filter is a recursive algorithm that can make a future prediction based on the most recent past measurement.

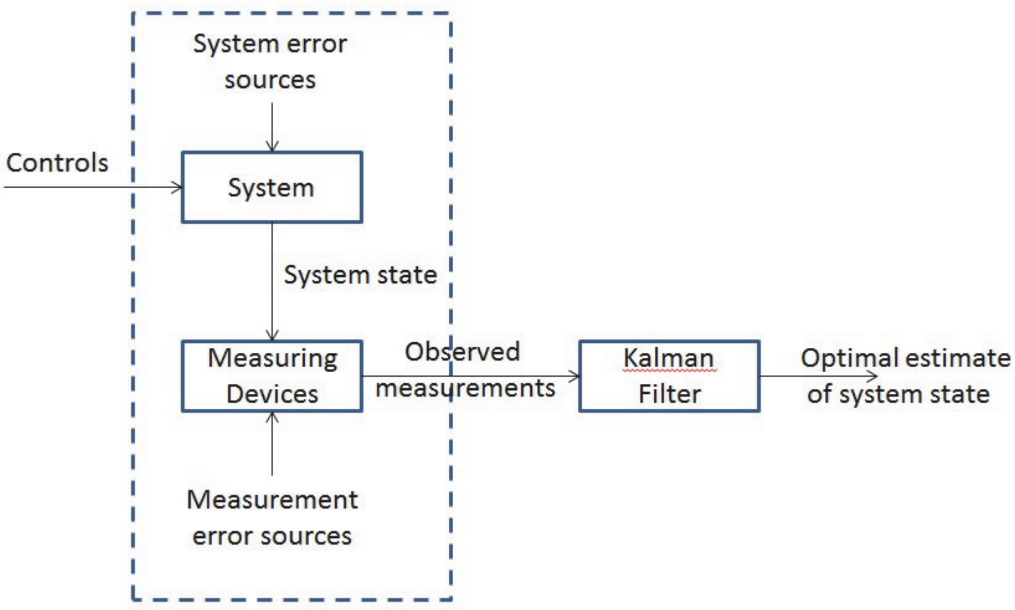


Figure 2.4: Kalman filter flow chart, from Stuckey [14]

2.4.2 Network Weatherman

The Network Weatherman algorithm was developed by Nathan C. Stuckey [14]. The Network Weatherman can predict the future network state based on the current network state by using the Kalman filter across multiple network nodes using stochastic estimation. This algorithm is based on router queue size prediction considering future state only seconds before the predicted state is to occur. The Network Weatherman uses two Kalman filter estimates, one for the size of the network queue and another for the packet arrival rate. These two estimates are used as inputs into a network queue controller which computes the desired packet transmission rate to maintain the desired queue size. Using those predictions, network control is manipulated to route traffic accordingly. Stuckey used OPNET modeler version 11.5 to simulate the Network Weatherman prediction technique. MATLAB, version 2006a, was utilized to compute the linear model for the Kalman filter to function. His results demonstrated that the Kalman filter and stochastic estimation were successful in network prediction and traffic routing. The Network Weatherman was further investigated by Muflih Alqahtani [15]. Alqahtani extended the research conducted by Stuckey implementing the Network Weatherman method in a virtual environment using VMware Workstation. This demonstrated that the Kalman filter and the Network Weatherman predictions are implemented successfully in a live network rather than inside of a controlled network simulator environment such as OPNET.

2.4.3 Dynamic Routing Queue Controller (DRQC)

The concept of the dynamic routing queue controller was proposed by James Haught at the Air Force Institute of Technology. The DRQC implements a central controller into the network that monitors queue predictions by using the Network Weatherman prediction method (section 2.4.2) and watches for potential network congestion. When network congestion is detected, the DRQC modifies the flow of the traffic to prevent congestion from occurring. The DRQC also tracks network flow priorities and attempts to maintain a higher level of QoS for higher priority traffic. Additionally, the DRQC is capable of splitting flows between two low bandwidth paths to meet the bandwidth requirements for a network flow. The DRQC was found successful in improving QoS in network environments [16].

2.5 Network Tasking Order (NTO)

When network prediction is desired for an airborne network, the military has an extreme advantage over most random scenarios. Every military operation involving aircraft has been preplanned and the location of aircraft in a given day is predefined rather than calculated on-the-fly. Assuming each aircraft acts like a router, network state and topology are predicted with precision by using the air tasking order (ATO) that was developed for that day’s flight plans. The ATO is a planning document developed at the air operation center (AOC) for tasking military aircraft in a warzone. For example, the Combined Air Operations Center (CAOC) at Al Udeid Air Base in Qatar controls all of the aircraft in the Middle East region dedicated to United States Central Command. For each day, the CAOC develops an ATO which directs flying squadrons to fly aircraft on various missions. These ATOs provide times and locations that the aircraft must meet. The following is a snippet from a sample ATO [17]:

```
1-TSKCNTRY/US//
2-SRCVTASK/F//
3-TASKUNIT/555FS/ICAO:ETAD//
4-AMSNDAT/C2342/CSAR//
        /DE/TGT-ID/LOCATION/TOT
        /01/-/294248N0473106E/241200ZJAN
        /02/-/294300N0473896E/241215ZJAN
        /03/-/294300N0473805E/241233ZJAN
        /04/-/294236N0473106E/241303ZJAN
5-MSNACFT/1/ACTYP:F16C/SANDY01/2MK-82
        /1654/3322//
6-AMSNLOC/AGL200/1//
```

The first line indicates that the United States is responsible. The second line assigns the Air Force to this mission. The third line specifies the 555th Fighter Squadron is tasked to fill the requirement. The fourth line indicates that this is a combat search and rescue (CSAR) mission. The fifth line is a header for the following four lines where the location is defined as well as the time-on-target (TOT) or time the aircraft needs to arrive at that location. For example, the fifth line indicates a location of 29° 42’ 48”N, 47° 31’ 06”E and a time-on target of 1200 Zulu on the 24th of January. A typical ATO is much larger and identify numerous aircraft and corresponding locations and times.

Compton developed the concept of using this ATO information for anticipating network topology at a given time of day and thereby routing based on that topology [2]. This concept has been called the network tasking order (NTO). Compton further proposed that network requirements as airborne assets could be planned through the cyber community such that the NTO could potentially affect the ATO. Either way, the NTO concept determines location at a given time and calculates the distance between aircraft. If the aircraft are within proximity, a link is assumed to be established and the bandwidth available on that link is determined. Network routing is then modified directly by scheduling route changes when necessary and not recalculate every time a data packet is transmitted. The only time the network routing tables would change is when a link is dropped or added. This process is simulated by Betances and its effectiveness has demonstrated effectiveness in improving QoS [3]. However, it is implemented in a simulator where all procedures are discrete-event based and are controlled strictly by a centralized program.

2.6 Commodity Flow

As most routing protocols determine the shortest path as the optimal route, they typically do not have any way of dealing with dynamic networks such as MANETs. In some cases, a single network link in a MANET may not be sufficient for a desired network flow. For these cases, the network flow may need to be split between two separate network connections to satisfy bandwidth requirements. An algorithm best suited for this is the multi-commodity flow. Figure 2.5 shows a graph with two sources and two sinks (a), the solution for S_1 to T_1 (b), and the solution for S_2 to T_2 (c) [18].

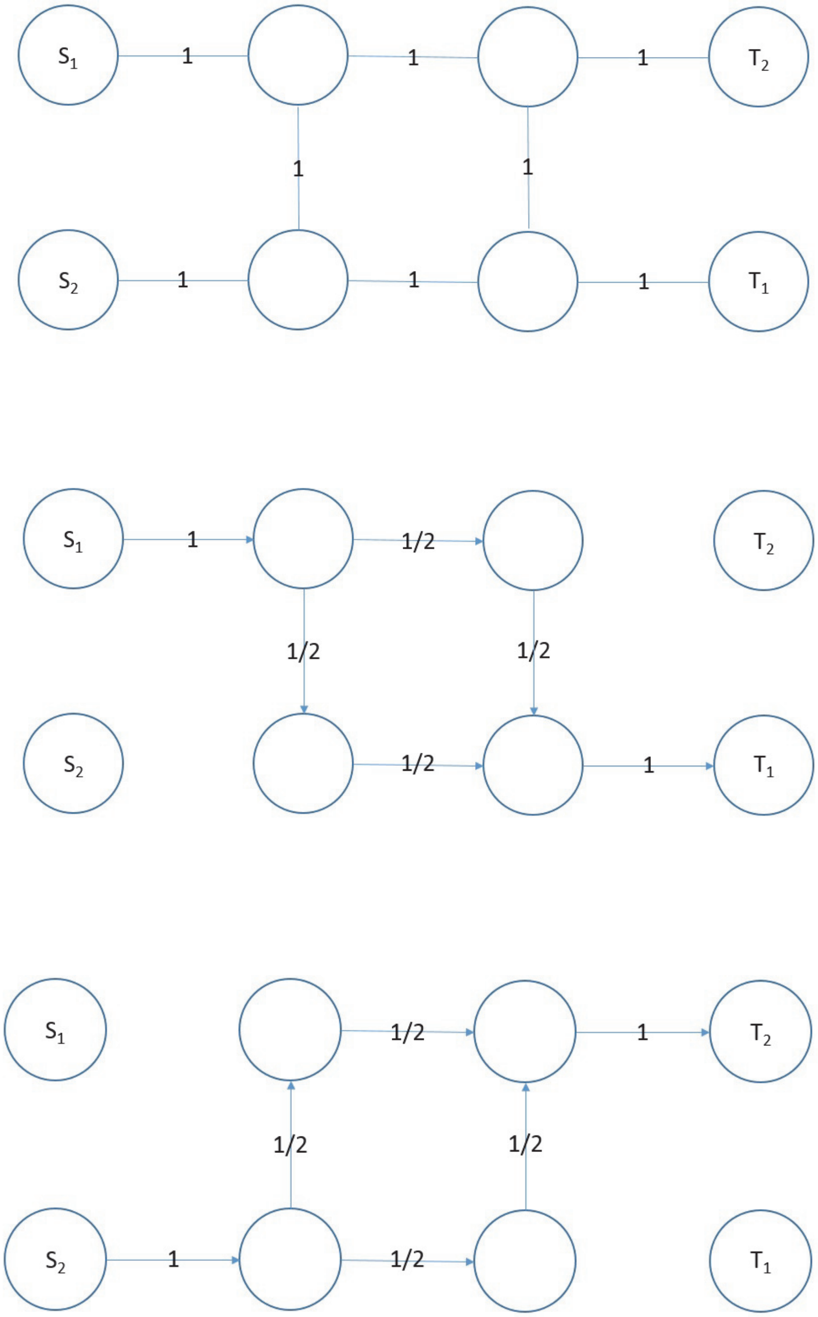


Figure 2.5: Multi-commodity Flow solution for (S_2, T_2) [18].

In a multi-commodity flow there can be multiple sources and destinations (or sinks). These sources and sinks are specified as ordered pairs of vertices such as S_1 and T_1. The goal of the multi-commodity flow is to maximize the flow from all sources to all sinks.

The commodity flow is crucial in providing an appropriate level of QoS in MANETs. The ability to route traffic onto two different paths is the only way to get the data from the source to the destination due to other network flows, or simply because the bandwidth required exceeds any single link in the network.

2.7 Synchronization Tools

An important element when attempting to use the NTO concept is time synchronization. Aside from the computers maintaining a timing signal for data transmission, in this case the actual time of day is important. Consider the fact that the NTO is purely a scheduled based procedure, then certainly all network devices making changes based on time must be able to produce the same clock time within fractions of a millisecond. Otherwise, routing changes may have adverse effects on the flow of mission data. Imagine two routers routing data but the NTO algorithm requires the path to reverse the flow of data between them to maintain overall routing success. If the router that is now to receive is one second behind the other router on the internal clock, the currently receiving router starts sending the traffic to the now receiving router which sends it back to the now transmitting router creating a routing loop. This may result in a large amount of packets lost during that time period. The bigger the difference in time between the devices, the larger the amount of data lost. In order to maintain time synchronization there are a couple of methods that can be employed.

2.7.1 Network Time Protocol (NTP)

NTP is an internet protocol that provides “time of day” synchronization between networked systems and devices. NTP operates at the application layer, uses User Datagram Protocol (UDP) at the transport layer (port 123), and can use either Internet Protocol Version 6 (IPV6) or Internet Protocol Version 4 (IPV4) at the network layer. NTP has three different modes of operation; client/server, symmetric, or broadcast. For the purpose of this thesis only client/servers are discussed. However, the other two methods should not be ignored for use in MANETs as they may assist in time accuracy and network efficiency. In client/server, not only does the client request time from the server, but more times than not, the server requests time from another server. The number of upstream servers corresponds to what is considered the “stratum” of the client or server. Primary servers are considered stratum one. Each lower level server increases by a factor of one. For example, a client or server that has three consecutive upstream servers is considered stratum four. Time accuracy is directly related to the stratum number. The higher the stratum number the less accurate the time server is. Other negative effects on NTP are security and propagation delay. Security concerns around NTP are abundant. The ability for an intruder to spoof a server’s IP address and provide a different time, day, or year can have some significant effects on a systems security logs or system scheduling. As such, NTP has some built-in security measures such as authentication and encryption, but both may affect time accuracy as processing time for either is increased during time transmission. Propagation delay is handled through the client’s NTP synchronization algorithm by calculating a new time based on the current time and the received update difference. This allows the protocol to slowly implement an accurate time and reduce time jitter. [19]

2.7.2 Global Positioning System (GPS)

GPS is capable of not only providing a position on the globe, as the acronym states, but also capable of providing time of day [20]. GPS uses time embedded in its signal to determine location. That same time signal can be used to synchronize a local device clock with extreme accuracy and precision. This time source can minimize local time drift to within one nanosecond per day. GPS may provide time even when location is not available. The navigation feature of GPS requires four satellites, three for location and an additional for time offset. If only time is desired one satellite is sufficient for an accurate signal. As GPS is already integrated to aircraft and most military vehicles, this time source is available where these MANET devices operate during military operations. Figure 2.6 depicts GPS receivers receiving signals from the GPS satellite constellation.

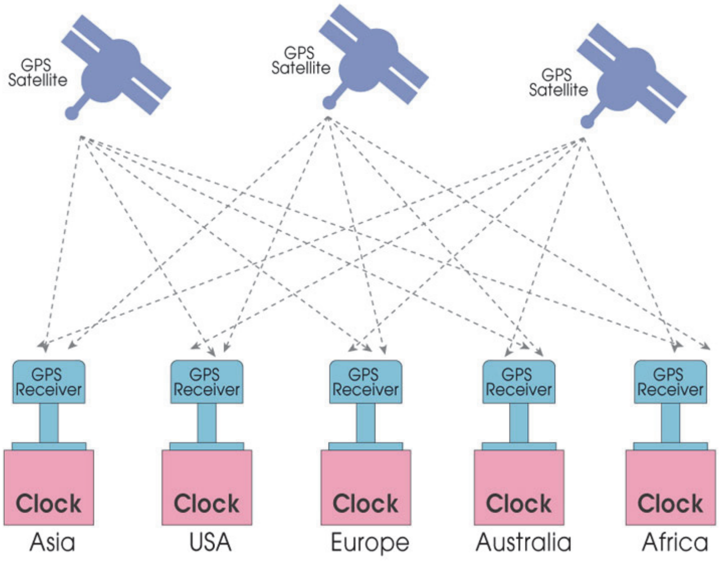


Figure 2.6: GPS satellites providing time around the world [21]

2.7.3 GPS and NTP Hybrid

While GPS is significantly more precise than NTP, there are many situations in the network community where NTP is the only method of time synchronization. In these cases, it is a better practice to use a GPS time source as an NTP server on the local network. All devices using NTP within the network should synchronize time with the GPS time source. All devices should be stratum two or better.

2.8 Summary

In this chapter, the concepts of MANETs, MANET routing prediction methods, and synchronization tools have been presented. These MANET concepts are used to develop a model network to determine effectiveness and quality of service performance of the NTO, derived from the ATO, routing method.

3 Methodology

3.1 Introduction

This chapter focuses on the methodology used to determine the applicability of the NTO concept as a routing protocol compared to other MANET routing protocols and quality of service characteristics. This chapter defines the research goals and the hypotheses of this thesis. The two simulations that are used to assess the NTO concept are explained. Finally, the comparisons that are accomplished to demonstrate which routing protocol or method performed best in these simulations are defined. Any limitations or assumptions are provided.

3.2 Research Objectives

The objective of this research is to demonstrate that the NTO concept operates on a decentralized network. As of now, the NTO concept has been simulated using OPNET or NS2 simulations and the results have concluded that the NTO concept is useful in network routing in MANETs. The simulators used have been in a controlled discrete-event environment where changes are implemented directly through each network component with no network overhead. In a live network, the components are decentralized and may act erratically in comparison to other nodes. In order to demonstrate that the NTO process can be used on a decentralized network, a model network is constructed. This model network is built in a virtual environment and the algorithm provided by Betances [3] is implemented on the model network. The field data that is analyzed are data packets to and from end systems on the virtual network. For the network to function optimally, the data packets must reach their destinations. The optimal network routing algorithm is the method that delivers the greatest number of packets over the duration of the data transmissions.

3.3 Research Hypothesis

There are three hypotheses that are analyzed:

- The NTO routing method will drop the fewest number of data packets when compared to either the MME or the OSPF dynamic routing protocols.
- The NTO routing method will have the greater network throughput when compared to either the MME or OSPF dynamic routing protocols.
- The NTO routing method is time critical and will demonstrate success only when the scheduled NTO is executed as planned. Any delay or advance in aircraft scheduling will have adverse effects on network performance.

3.4 Measurements

The scenarios are assessed by comparing the number of dropped data packets between the various MANET routing protocols and compared to the NTO scheduled routing method. The number of network connections that data packets have to traverse to reach their destination are assessed. The optimal protocol has the highest number of successfully delivered data packets. The highest number of delivered data packets resembles a higher overall network availability and overall throughput. Therefore, this indicates a higher level of quality of service.

3.5 NTO Implementation

An advantage of the NTO concept is that all calculations are conducted offline, before the aircraft depart the base, and route changes are scheduled to take place at whatever time they are necessary. Before scheduling is accomplished, network prediction is calculated for the duration of the NTO.

3.5.1 Network Prediction

The algorithm required to predict a MANET based on the NTO has been previously derived and described in Betances' thesis [3]. The distance between two nodes is computed utilizing the Euclidean formula. For any node, find the latitude, longitude, and altitude. The distance between two nodes is the square root of the sum of the squared difference of these three dimensions. The scenarios in this thesis are subcomponents of an example NTO and determination is made for all waypoints where link state changes based on this formula. In this model, link state is defined as link established, link disconnected, or change in bandwidth.

To accomplish the task of determining route changes, the calculations are made for all aircraft at a given time. These calculations provide all possible link states between all aircraft for that time and a relationship table is created. The table is created for incremental amounts of time throughout the start time and end time of the NTO. Because an NTO is derived from an ATO and the ATO is a plan for a given day, the calculations for the link states may have a duration of up to 24 hours. For each calculation there are $O(n^2)$ possible connections between aircraft. Analysis is made to determine the best time increment for each calculation. Using a one second interval for calculations requires 86400 permutations of this algorithm for a 24 hour scenario.

The choice of using a one second interval versus a one minute interval greatly depends on how many aircraft are involved and how close they are in proximity during the operation. More granular control can be determined by calculating every millisecond or hundredth of a second instead of every second, but this is quite extreme depending on the scenario as link states do not change that quickly. Consider two aircraft approaching each other at 600 mph, the closing speed is 1200 mph. If link states change at 30 mile proximity distance intervals as a worst case, then it would take 90 seconds between link states changes. Therefore, if distances are calculated for every discrete second during the mission, route changes occur in a timely enough manner to minimize data loss.

As the NTO may eventually affect the planning of the ATO, a good planning practice would require overlap of aircraft in proximity to other aircraft to maintain connections so the route changes can take place while the two aircraft are both in proximity ensuring a connection always exists. The longer the overlap, the less susceptible to aircraft scheduling delay the network is.

For each calculation interval, the connection between each aircraft is calculated using the Euclidean formula. Consider a scenario where the number of nodes, either aircraft, ground stations, or satellites, equals 50. This means that there are potentially 2500/2 (divide by two because each node needs to be considered only once) combinations that are analyzed. A total of 108 million vector combinations are calculated for each 24 hour NTO scenario. Action is taken only when a difference in link state is calculated from one interval to the next. If no change is calculated from one interval to another, it is disregarded. When a link state change is calculated from one interval to the next, a route change is scheduled in the affected routers. Keep in mind that a change in link state may affect more than neighboring nodes. A new flow is calculated through the entire network from all sources to all destinations.

3.5.2 Example Topology Prediction

An example for predicting network topology is provided by Betances. Table 3.1 is provided by Betances for an example air tasking order scenario. During this scenario, link state is calculated using the following:

- If the distance is less than or equal to 15 miles, the connection is 8 Mbps
- If the distance is greater than 15 miles and less than or equal to 25 miles, the connection is 4 Mbps.
- If the distance is greater than 25 miles, but less than or equal to 45 miles, the connection is 2 Mbps
- If the distance exceeds 45 miles, the connection is out of range, and considered disconnected.

These distance to link state ratios are notional and for example only, derived from a fictional ATO scenario. Using the Euclidean formula, the distances between all nodes are calculated. The distances for this particular time interval are provided in Table 3.2. Using these distances, the link states are determined. For example, the table shows that node one and two are 44 miles apart, which implies that they have a 2 Mbps connection between them. Table 3.3 shows the link states between all nodes in the network at this snapshot in time. [3]

Table 3.1: Node position at a given time– Information is notional only, not derived from any real or practice scenario. [3]

Node ID	Time (Seconds)	X Position(Feet)	Y Position(Feet)	Z Position(Feet)
1	8,100	174,449	709,160	6,096
2	8,100	436,748	769,637	6,705
3	8,100	577,265	777,840	6,096
4	8,100	788,872	609,752	6,096
5	8,100	1,081,194	695,136	6,096
6	8,100	333,360	348,673	18,288
7	8,100	763,127	623,568	18,897
8	8,100	494,884	467,064	3,048
9	8,100	851,892	550,564	3,657
10	8,100	872,555	1,031,564	4,572
11	8,100	923,453	1,021,378	5,486
12	8,100	972,761	1,008,174	4,572
13	8,100	872,555	168,532	4,572
14	8,100	923,453	178,718	5,486
15	8,100	972,761	191,922	4,572
16	8,100	296,320	694,500	0
17	8,100	481,520	666,720	0
18	8,100	1,037,120	972,300	0

Table 3.2: Node distances at a given time– Information is notional only, not derived from any real or practice scenario. [3]

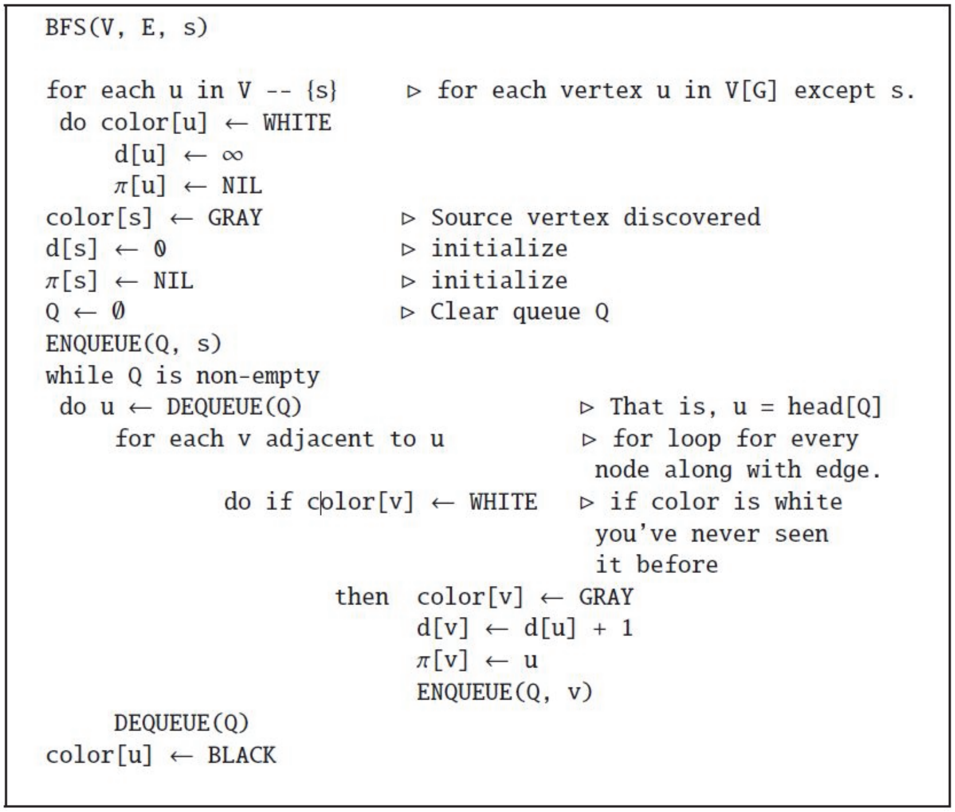
Nodes	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	0	44	67	102	149	65	98	66	115	127	134	140	145	151	157	20	51	148
2	44	0	23	64	107	71	59	51	77	84	90	97	122	126	130	26	19	104
3	67	23	0	44	84	81	40	53	59	64	70	75	111	114	116	48	24	82
4	102	64	44	0	50	86	5	54	14	71	71	72	74	74	75	82	51	72
5	149	107	84	50	0	136	54	104	45	65	60	55	93	89	85	129	99	46
6	65	71	81	86	136	0	84	33	92	143	147	151	94	101	108	57	58	155
7	98	59	40	5	54	84	0	51	19	70	71	72	77	78	79	78	47	73
8	66	51	53	54	104	33	51	0	60	112	115	119	79	85	91	50	33	122
9	115	77	59	14	45	92	19	60	0	79	78	78	63	62	62	94	64	76
10	127	84	64	71	65	143	70	112	79	0	9	17	142	141	139	110	88	29
11	134	90	70	71	60	147	71	115	78	9	0	8	141	139	137	116	93	20
12	140	97	75	72	55	151	72	119	78	17	8	0	139	137	134	123	98	12
13	145	122	111	74	93	94	77	79	63	142	141	139	0	9	17	128	104	135
14	151	126	114	74	89	101	78	85	62	141	139	137	9	0	8	134	108	132
15	157	130	116	75	85	108	79	91	62	139	137	134	17	8	0	139	112	129
16	20	26	48	82	129	57	78	50	94	110	116	123	128	134	139	0	31	130
17	51	19	24	51	99	58	47	33	64	88	93	98	104	108	112	31	0	104
18	148	104	82	72	46	155	73	122	76	29	20	12	135	132	129	130	104	0

Table 3.3: Calculated link states at a given time in Mbps. Absence of speed indicates no connection. Information is notional only, not derived from any real or practice scenario. [3]

Nodes	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1		2														4		
2	2		4													2	4	
3		4		2			2										4	
4			2				8		8									
5									2									
6								2										
7			2	8					4									
8						2											2	
9				8	2		4											
10											8	4						2
11										8		8						4
12										4	8							8
13														8	4			
14													8		8			
15													4	8				
16	4	2															2	
17		4	4					2								2		
18										2	4	8						

3.5.3 Network Prediction Algorithm

The data from Table 3.3 is the key to calculating a possible end to end connection in a network. Consider the nodes as vertices in a graph and the available connections between them as edges. Using these data, a breadth first search (BFS) is conducted to determine connected groups in the network. Figure 3.1 provides the BFS algorithm to identify the connected groups. Each connected group established, if more than one, requires its own outside network connection if such a connection is desired. If a ground station is part of the connected group, then outside connectivity is most likely established there. Figure 3.2 shows the predicted network topology based on the breadth first search. From this visualization, it is determined that there are three separate connected groups.



```
BFS(V, E, s)

for each u in V -- {s}      ▷ for each vertex u in V[G] except s.
 do color[u] ← WHITE
     d[u] ← ∞
     π[u] ← NIL
color[s] ← GRAY             ▷ Source vertex discovered
d[s] ← 0                    ▷ initialize
π[s] ← NIL                  ▷ initialize
Q ← ∅                       ▷ Clear queue Q
ENQUEUE(Q, s)
while Q is non-empty
 do u ← DEQUEUE(Q)                         ▷ That is, u = head[Q]
     for each v adjacent to u              ▷ for loop for every
                                            node along with edge.
             do if color[v] ← WHITE        ▷ if color is white
                                            you've never seen
                                            it before
                     then  color[v] ← GRAY
                           d[v] ← d[u] + 1
                           π[v] ← u
                           ENQUEUE(Q, v)
     DEQUEUE(Q)
color[u] ← BLACK
```

Figure 3.1: BFS Algorithm, WHITE identifies *undiscovered* state, GRAY identifies *discovered* but *not fully explored* state and BLACK identifies *fully explored* state. [3]

The next step in the NTO method is to calculate the maximum concurrent multi-commodity flow. As in the previous scenario, if the graph is not complete, a new connection is completed by adding an edge to the closest node in each connected group to the main connected group. This simulates a satellite connection tying the two connected groups together, which is the only option when two or more connected groups are out of range from another and any one of them does not have a ground station to provide outside connectivity. This new edge gives a flow rate of 1 Mbps, equal to what a satellite connection can supply. With a completed graph, the multi-commodity flow algorithm is conducted on the graph from all sources and destinations (sources and sinks). The max concurrent flow algorithm may require significant processing resources. However, the algorithm is conducted offline, before NTO execution, minimizing run time concerns. If algorithm run time becomes an issue, an approximation may be sufficient using a fraction of each connection.

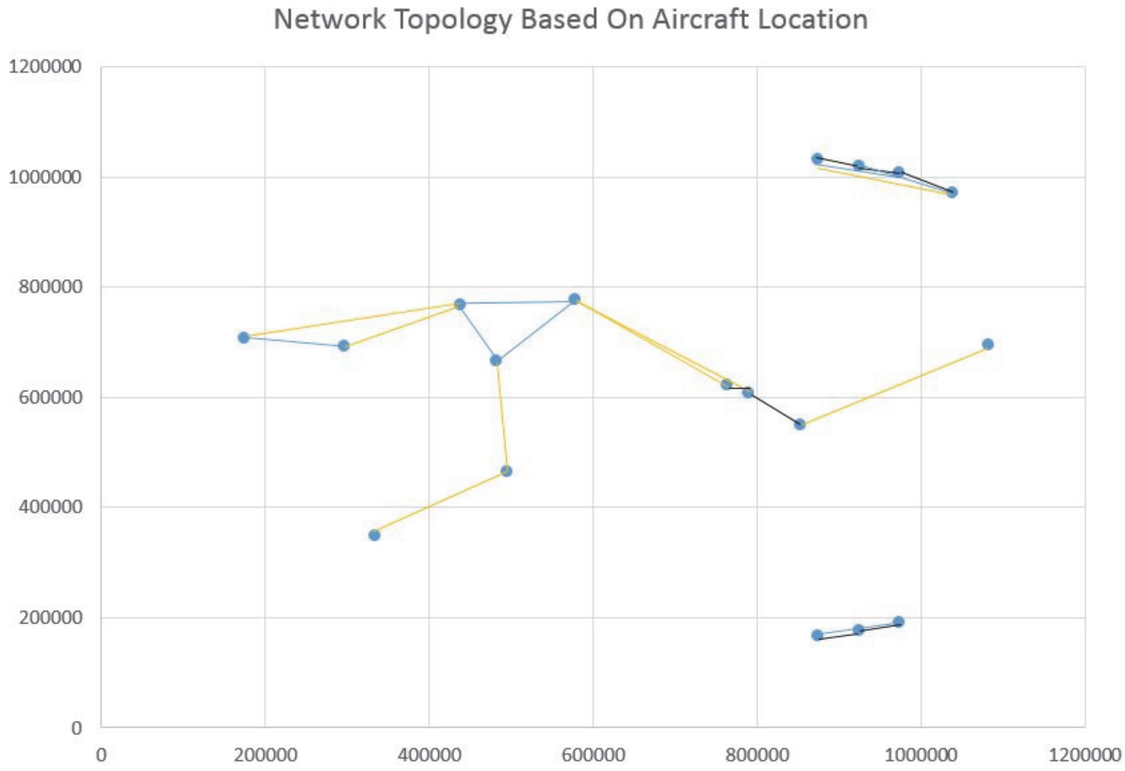
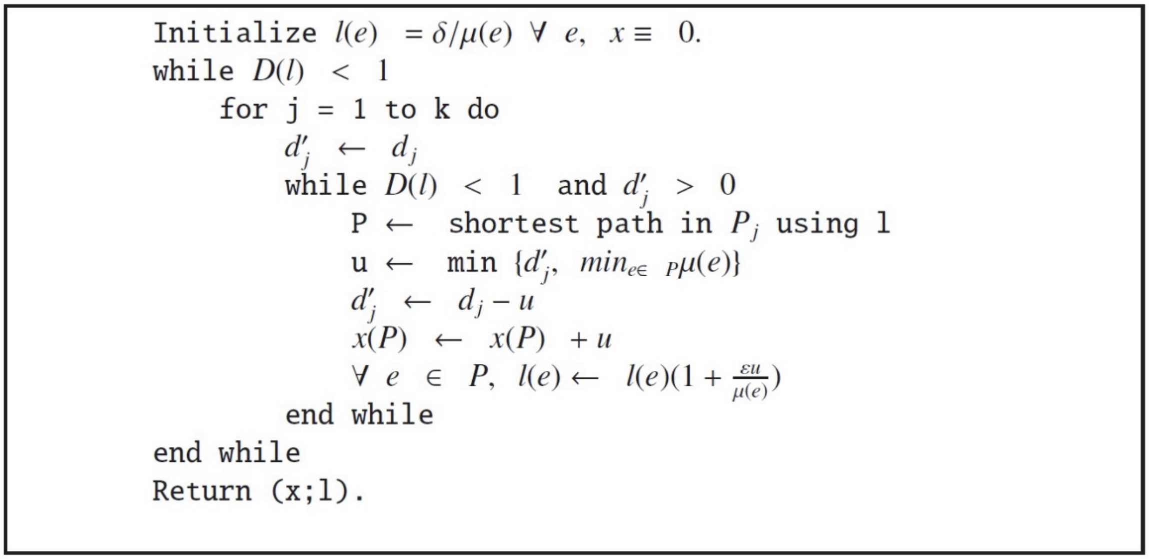


Figure 3.2: Predicted topology based on aircraft location. The black line indicates an 8 Mbps connection, blue indicates a 4 Mbps, and each yellow line is a 2 Mbps connection.

The multi-commodity flow algorithm follows the following steps. First, all edge lengths in the graph are initialized with $\frac{\delta}{\mu(e)}$. Although this value is small, it is exponentially incremented as edges become congested. Next, the shortest path route is determined and edges are lengthened based on commodity requirements. After path selection, edge lengths are increased preventing overutilization of any given edge. Finally, the algorithm exits once the length of all edges exceeds a value of one. The pseudocode of this algorithm is provided in Figure 3.3. [3]



```
Initialize $l(e) = \delta/\mu(e)\ \forall\ e,\ x \equiv 0$.
while $D(l) < 1$
    for j = 1 to k do
        $d'_j \leftarrow d_j$
        while $D(l) < 1$ and $d'_j > 0$
            P $\leftarrow$ shortest path in $P_j$ using l
            u $\leftarrow$ min $\{d'_j,\ min_{e\in\ P}\mu(e)\}$
            $d'_j \leftarrow d_j - u$
            $x(P) \leftarrow x(P) + u$
            $\forall\ e \in P,\ l(e) \leftarrow l(e)(1 + \frac{\varepsilon u}{\mu(e)})$
        end while
end while
Return (x;l).
```

Figure 3.3: Pseudo code of the multi commodity flow algorithm. [3]

3.5.4 Network Emulation Development

For this experiment, the network environment is virtualized to reduce the amount of equipment required to develop each scenario. Virtualization allows the implementation of multiple operating systems on one physical machine. The host machine for this experiment has the following specifications:

- AMD FX-8350 eight core processor
- 32 GB of DDR3 1866 (PC3 14900) RAM
- SSD hard drive with SATA 3 Gbps interface

This large amount of physical memory and multi-core processor allows for many virtual machines with less worry about memory thrashing or input and output (I/O) contention.

The virtualization platform running on top of the host machine is VMware Workstation 10.0.4. Within VMware Workstation, four virtual machines are created. Two OpenSuse machines for use as end clients on the network, a Debian system for use as an NTP server, and VMware's ESXi virtualization platform for the routers. Although virtualizing a virtual environment is not typically a desired scenario, in this case it allows for snapshots of the network topology for faster changes in network configurations. Running the end clients in VMware Workstation provides a graphical interface to run the necessary scripts to pass traffic through the network. Within the ESXi platform, four MikroTik Router Operating Systems (RouterOS) are installed. Within ESXi, the network connections are established to facilitate the topology for the desired scenarios. The network topology that are established in the virtual environment is shown in Figure 3.1. No experimental data traverses the management network. The management network is strictly for NTP and router management interface to access the configurations. The management network is established using the 192.168.46.0/24 network addressing scheme while the experimental network uses the 10.0.0.0/8 network. When dynamic protocols are in use, only the 10.0.0.0/8 network is configured. Route scheduling using the NTO method is accomplished using only 10.0.0.0/8 interface addresses for the next hop to reach the destination. NTP is configured for use on the management network as NTP traffic would not have to traverse the actual MANET as most airborne routers could introduce GPS as their primary timing source to ensure accurate route scheduling. Accurate timing is critical to the NTO concept as route changes must occur simultaneously throughout the network. A level of atomicity is observed if downstream routers update network routes prior to upstream routers. Synchronization is critical.

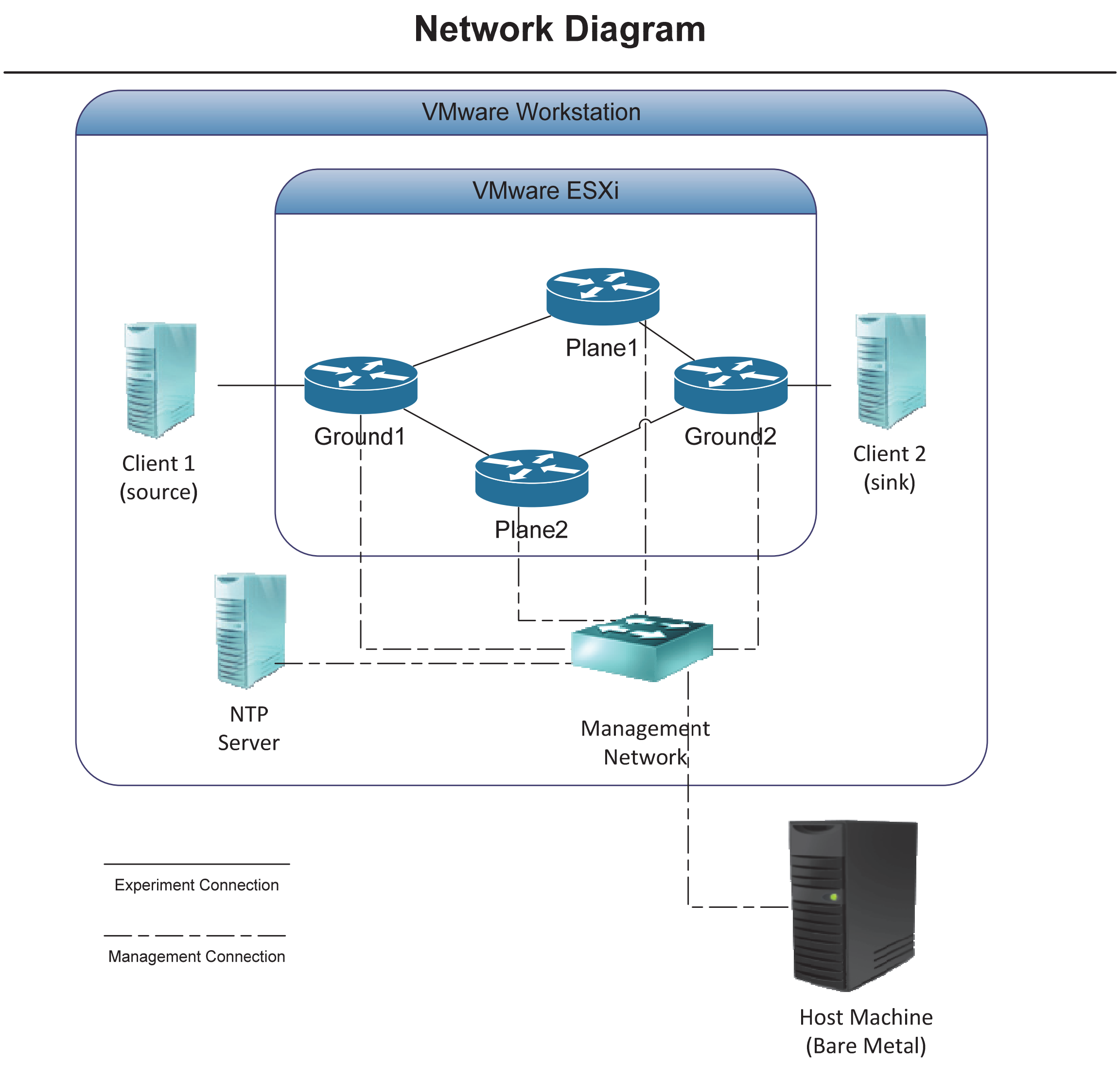


Figure 3.4: Network diagram used for both scenarios

3.6 Scenario One – Commodity Flow

This scenario is intended to test the NTO commodity flow algorithm. Using the topology that has been established in section 3.6, a network with two aircraft orbiting between two ground stations is established. These aircraft simulate flying in a circle with an 80 mile diameter. Both planes are on opposite sides of the circle and maintain that distance. For the implementation of the commodity flow, a 1 Mbps radio connection is simulated between both aircraft and not change state for the course of the experiment. Both aircraft have a connection to ground station one and ground station two which dynamically change based on the following parameters:

- If the distance is less than or equal to 88 miles, the connection is 1 Mbps
- If the distance is greater than 88 miles and less than or equal to 126 miles, the connection is 768 kbps.
- If the distance is greater than 126 miles, but less than or equal to 156 miles, the connection is 256 kbps.
- If the distance exceeds 156 miles, the connection is out of range, and considered disconnected.

The aircraft fly at approximately 500 mph (502.4 mph to be precise), such that it takes exactly 30 minutes to complete an orbit. This allows the scenario to be cyclical to run concurrent experiments without rescheduling. Using the above distances to modify link state, the links change state every five minutes. Figure 3.2 shows the first four stages of this scenario. The scenario continues through these stages indefinitely. Using this scenario, the routing methods OSPF, MME, and the NTO routing concept is evaluated. Each routing method has four different tests. Each test uses a different size datagram to validate packet size effects on each routing method. The goal is to achieve as close to the maximum flow as possible without exceeding it. Exceeding the maximum flow results in

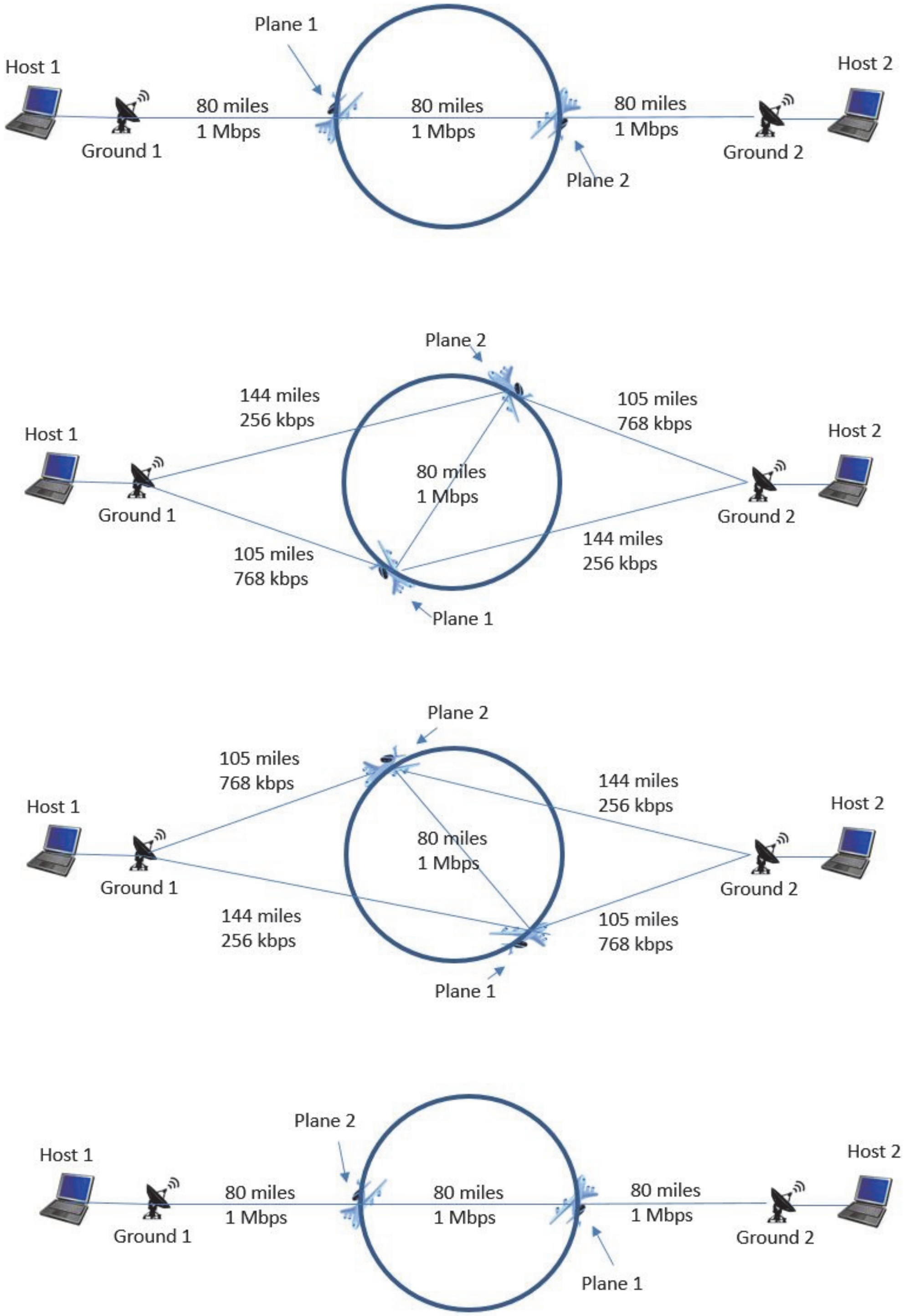


Figure 3.5: First four cycles of scenario one, each cycle has a five minute duration.

the unintentional loss of datagrams. The datagrams are delayed to achieve the desired traffic load on the network. To prove the effectiveness of the commodity flow, the traffic must consume more than what is available on a single link, but less than what is available from point to point. Datagrams are created using four different sizes such that a traffic generator demands around 950 kbps of bandwidth using the following rules:

- A datagram is sent 125 times per second.
- A datagram is sent 250 times per second.
- A datagram is sent 500 times per second.
- A datagram is sent 1000 times per second.

These packets are sent using a Python script on the source virtual machine “Host 1”. A Python script is also running on the destination virtual machine “Host 2”. A sequence number is injected into the payload of the datagram. This sequence number is inspected by the receiving Python script. If any datagram is missing, the receiving script detects a jump in sequence numbers. The number of jumps in sequence numbers are added as well as the number of successfully received datagrams. The number of received divided by the total is the achieved success rate. An anomaly that is possible when using the multi-commodity flow is that datagrams may arrive out of order. The application layer may be able to handle out of order datagrams, so this is not a negative effect depending on the application. This is a unique design consideration when adopting a multi-commodity flow scenario. Datagrams are buffered and placed in proper order when received. If the application is streaming video, then reassembly is irrelevant and undesired after a certain amount of time. A buffer and time delay is established to allow for this anomaly to occur. For this reason, the number of out of order datagrams are also tracked. The datagram is considered successfully delivered if received within 100 datagrams of the current sequence number, meaning it was successfully delivered within one tenth of a second when using the 1000 datagram per second test scenario. If the sequence number history exceeds 100, it is considered discarded and unsuccessful.

3.7 Scenario Two – Fast Switching

The NTO routing method is effective in achieving a high level of QoS. The advantage is knowing the location of the aircraft ahead of time. However, aircraft are sometimes delayed, show up too early, get rerouted, or remain on the ground due to mechanical issues. This scenario implementation is to determine when the NTO routing method may become less effective. More importantly, how a scheduling delay affects the NTO routing method performance. Using the same topology, this scenario models an aircraft flying in a line adjacent to two ground stations. The two ground stations are out of range of each other and require the aircraft to communicate between them. Considering a properly planned and executed NTO, before the aircraft is out of range, another aircraft approaches and establishes connectivity. This is used to determine routing efficiency when a quick route path change needs to take effect. Link speeds maintain a constant 256k. Only interface state is modified to emulate the environment. Figure 3.6 shows the first stage of this scenario where only Plane 1 is in range. Figure 3.7 demonstrates the topology when both Plane 1 and Plane 2 are in range. Figure 3.8 shows how the topology changes after Plane 1 is out of range, but Plane 2 remains in range. Each plane is in range for one minute and five seconds.

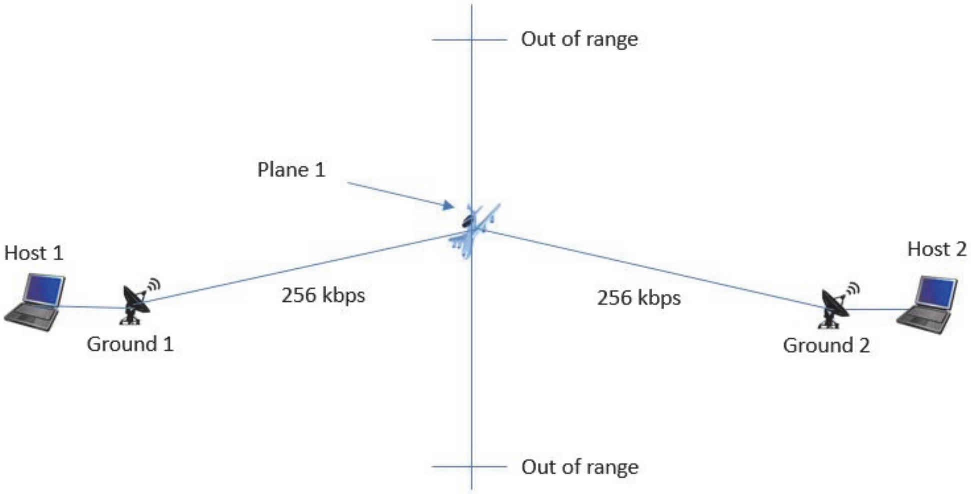


Figure 3.6: Scenario two while Plane 1 is in range of the ground stations.

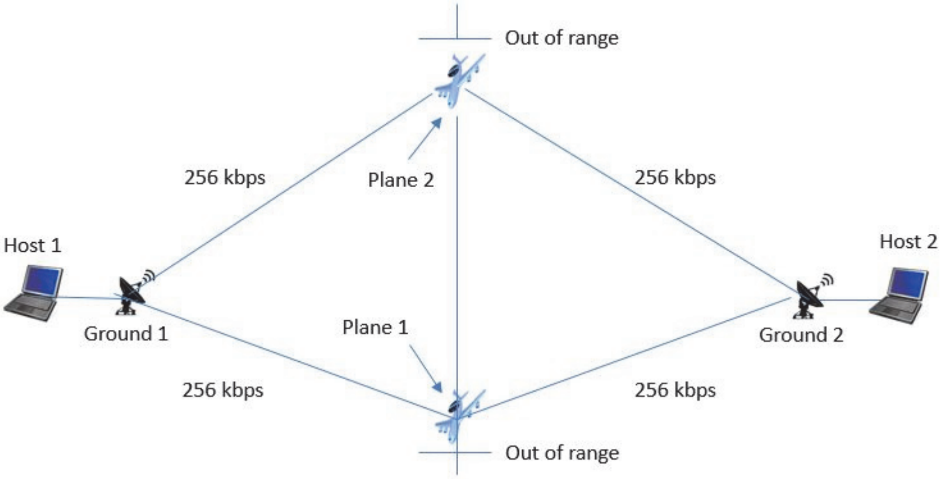


Figure 3.7: Scenario two while Plane 1 and Plane 2 are both in range of the ground stations.

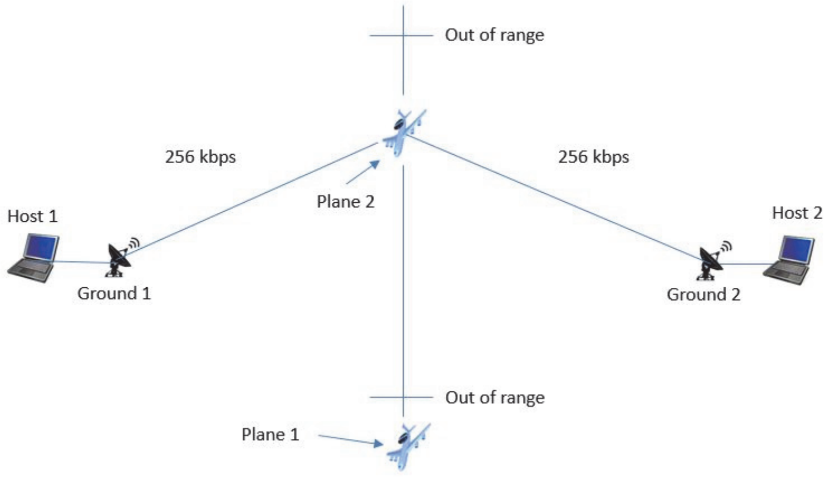


Figure 3.8: Scenario two while Plane 1 has just gone out of range and lost connectivity but Plane 2 is still in range.

The next plane follows the previous plane by exactly one minute. This provides a five second overlap where the routes can change without dropping data. As with the first scenario, OSPF, MME, and the NTO routing method are assessed. For each routing method, datagrams are transmitted from source to destination to nearly saturate the 256 kbps max flow. The datagram size is static at 102 bytes. Smaller datagrams allow for a more precise measurement as the quantity sent per second is higher than larger datagrams. The loss of a single smaller datagram has less effect on the overall success rate than a larger datagram. The NTO routing method is implemented using many additional tests simulating an aircraft delay. The delay increases at five second intervals until network performance is less than the dynamic routing protocols OSPF and MME. This demonstrates the amount of delay that can be tolerated before a dynamic routing protocol may be more beneficial than the NTO routing method.

3.8 Limitations and Assumptions

Modeling a MANET in a virtual environment overlooks many complications that can occur within an actual wireless MANET. Events like frequency jamming, interference, multi-path delay, weather, and radio protocol overhead cannot be integrated into this modeled environment. Although these elements are important to MANETs, for the purpose of this experiment they are not considered. The demonstrated effectiveness of the NTO concept is to determine efficient routing of data packets when the airborne network environment dynamically changes due to aircraft position. These elements are not part of the NTO concept decision making and therefore not a concern. Propagation delay is another element which could impact the NTO concept. However, the result is in the millisecond region and we are dealing with time intervals spanning minutes, so propagation delay is ignored due to its minimal impact.

The MikroTik RouterOS was used due to its ability to change link speeds to emulate a MANET. However, MikroTik had limited routing protocols to closely emulate a modern MANET protocol such as AODV. The MikroTik RouterOS did support a protocol known as Hybrid Wireless Mesh Protocol (HWMP) which closely resembles AODV. The use of HWMP was attempted, but since HWMP is a layer 2 bridged routing protocol, it bypassed the queues used for bandwidth limiting and provided inaccurate results.

For this research, the test results are based specifically on UDP data transfer. TCP may be desired in a MANET, but determining whether or not data is successfully transferred when TCP auto corrects provides insignificant results. To determine bidirectional success, UDP transmission in both directions provides better overall results and determine directionality problems. TCP has too many limitations for the purpose of this research.

Another limitation of the MikroTik RouterOS is that it does not split UDP transmissions from a single source to a single destination when multipath routes exist. To test the function of the commodity flow, multiple source IP addresses were used, in a round-robin fashion, to force the RouterOS to split the transmissions. UDP is a connectionless protocol where routers do not follow these rules. If incorporated into a functional network, the ability to split the UDP traffic from source to destination should be permitted in the implemented hardware.

3.9 Summary

This chapter defined how a MANET is modeled and how the various MANET routing protocols are utilized on this environment. The routing protocol that drops the fewest data packets is determined as the optimal routing protocol. The hypothesis is that the NTO routing concept prevails as the optimal routing method because it can predict network state based on a priori knowledge.

4 Results and Analysis

4.1 Introduction

This chapter defines the experiments that were performed to implement and compare the NTO process to other network protocols. These experiments were conducted using a time scheduler to introduce bandwidth delay into network links between routers to emulate a real world airborne network environment. For each scenario, the following routing methods were used to determine which was more efficient for the two dynamic network scenarios:

- The first set of iterations utilized the MME protocol. This protocol was utilized to create a baseline comparison for dynamic protocols in an airborne environment.
- The next set of simulations utilizes OSPF to demonstrate how standard routing protocols operate in an airborne environment.
- The last set of experiments utilized the NTO concept for route scheduling and multi-path commodity flow base on network planning for scenario one. The second scenario used time offsets to introduce scheduling conflicts where the aircraft may have been late to demonstrate how this would affect the network traffic.

4.2 Network Environment Validation

Prior to each series of tests performed during each of these scenarios, the network is validated to ensure ICMP, UDP, and TCP all function properly on the network topology. Furthermore, verify that each node on the network synchronizes system clocks with the NTP server.

4.3 Scenario One – Commodity Flow

NTO Determined Links Speeds NTO Determined Links Speeds

In this scenario, the network environment is created as described in section 3.8. Two airplanes orbit in a circle that is 80 miles in diameter and each aircraft is at opposite ends of the circle, equidistant from each other on the border of the circle. In order for the NTO method to function and routes scheduled a couple of procedures are implemented. First, the NTO table is constructed to find where link states change over time. Next, the commodity flow is calculated and a routing table developed for each time link state changes. This section is broken down into those two steps and then the results of the data transmissions is analyzed.

4.3.1 NTO Route Scheduling

Using the link specifications from section 3.8, the NTO link states are calculated. These link states are notional and created strictly for the use of testing this type of dynamic network. The link states for this scenario are as follows:

- If the distance is less than or equal to 88 miles, the connection is 1 Mbps
- If the distance is greater than 88 miles and less than or equal to 126 miles, the connection is 768 kbps.
- If the distance is greater than 126 miles, but less than or equal to 156 miles, the connection is 256 kbps.
- If the distance exceeds 156 miles, no connection exists.

Table 4.1 shows the calculated tables of link states. With this information and using the

Table 4.1: Calculated links speeds based on NTO for period of time

	Ground 1	Plane 1	Plane 2	Ground 2
Ground 1	-	1 Mbps	N/C	N/C
Plane 1	1 Mbps	-	1 Mbps	N/C
Plane 2	N/C	1 Mbps	-	1 Mbps
Ground 2	N/C	N/C	1 Mbps	-

t = 0 - 300 seconds

	Ground 1	Plane 1	Plane 2	Ground 2
Ground 1	-	768 kbps	256 kbps	N/C
Plane 1	768 kbps	-	1 Mbps	256 kbps
Plane 2	256 kbps	1 Mbps	-	768 kbps
Ground 2	N/C	256 kbps	768 kbps	-

t = 301 - 600 seconds

NTO Determined Links Speeds

	Ground 1	Plane 1	Plane 2	Ground 2
Ground 1	-	256 kbps	768 kbps	N/C
Plane 1	256 kbps	-	1 Mbps	768 kbps
Plane 2	768 kbps	1 Mbps	-	256 kbps
Ground 2	N/C	768 kbps	256 kbps	-

t = 601 - 900 seconds

NTO Determined Links Speeds

	Ground 1	Plane 1	Plane 2	Ground 2
Ground 1	-	N/C	1 Mbps	N/C
Plane 1	N/C	-	1 Mbps	1 Mbps
Plane 2	1 Mbps	1 Mbps	-	N/C
Ground 2	N/C	1 Mbps	N/C	-

t = 901 - 1200 seconds

NTO Determined Links Speeds

	Ground 1	Plane 1	Plane 2	Ground 2
Ground 1	-	256 kbps	768 kbps	N/C
Plane 1	256 kbps	-	1 Mbps	768 kbps
Plane 2	768 kbps	1 Mbps	-	256 kbps
Ground 2	N/C	768 kbps	256 kbps	-

t = 1201 - 1500 seconds

NTO Determined Links Speeds

	Ground 1	Plane 1	Plane 2	Ground 2
Ground 1	-	768 kbps	256 kbps	N/C
Plane 1	768 kbps	-	1 Mbps	256 kbps
Plane 2	256 kbps	1 Mbps	-	768 kbps
Ground 2	N/C	256 kbps	768 kbps	-

t = 1501 - 1800 seconds

Euclidean formulas, the link states are calculated. The link states change on five minute intervals. There are six intervals that occur during each cycle of this network scenario.

The six intervals discovered, in seconds, are: 1 – 300, 301 – 600, 601 – 900, 901 – 1200, 1201 – 1500, 1501 – 1800. Since this is cyclical, the 1800th and zero seconds are equivalent as the cycle repeats. For each of these intervals the commodity flow is calculated.

4.3.2 Commodity Flow Calculations

After investigating the intervals of this scenario, it is determined that there is always a single flow from end to end of at least 768 kbps. However, using a commodity flow, it is capable of achieving up to 1 Mbps if traffic is split amongst multiple connections. Figure 4.1 shows how the commodity flow is implemented for the second interval of this scenario where time falls between 301 and 600 seconds. The other five intervals have similar results. Based on each of these flows, the routing table is

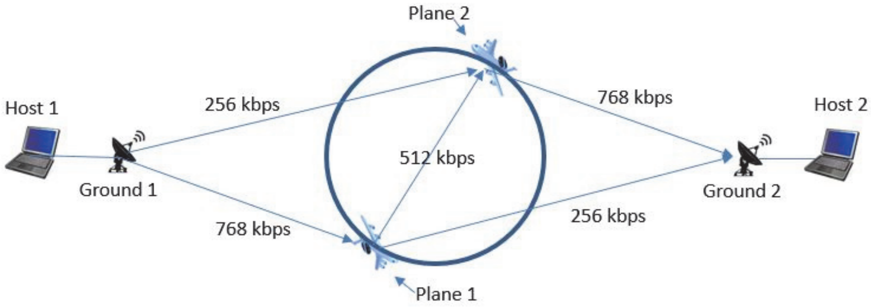


Figure 4.1: Calculated flow for second cycle of scenario one.

Table 4.2: Established routing table for commodity flow per time increment.

t = 1 to 300	
Ground 1	100% to Plane 1
Plane 1	100% to Plane 2
Plane 2	100% to Ground 2

t = 301 to 600	
Ground 1	75% to Plane 1, 25% to Plane 2
Plane 1	66% to Plane 2, 33% to Ground 2
Plane 2	100% to Ground 2
t = 601 to 900	
Ground 1	75% to Plane 2, 25% to Plane 1
Plane 1	100% to Ground 2
Plane 2	66% to Plane 1, 33% to Ground 2
t = 901 to 1200	
Ground 1	100% to Plane 2
Plane 1	100% to Ground 2
Plane 2	100% to Plane 1
t = 1201 to 1500	
Ground 1	75% to Plane 2, 25% to Plane 1
Plane 1	100% to Ground 2
Plane 2	66% to Plane 1, 33% to Ground 2
t = 1501 to 1800	
Ground 1	75% to Plane 1, 25% to Plane 2
Plane 1	66% to Plane 2, 33% to Ground 2
Plane 2	100% to Ground 2

established where fractions of data are routed as appropriate to reach the Host 2 from the Host 1. Table 4.2 shows the routing table for each interval of time that are scheduled in the routers. These routes are scheduled at the precise time in each router so that the routes change at the proper time.

4.3.3 Scenario One, Traffic Load One

Iteration one involves the use of 1000 datagrams transmitted every second to achieve approximately 900 kbps of traffic generation to test the various routing methods.

By generating 82 bytes of data and adding the UDP header of eight bytes, the IP header of 20 bytes, and the 802 header of 14 bytes, a total datagram size of 124 bytes is place on the wire. A wait time is established to send approximately 1000 per second. Each 124 byte datagram, when converted, is 992 bits. This generated 992,000 bits per second and when divided by 1024 to achieve kilobits, approximately 968 kilobits per second were transmitted from Host 1 to Host 2. Table 4.3 provides the percentage of datagrams that were received by Host 2.

Table 4.3: Success rate using 1000 datagrams per second.

	MME	OSPF	NTO
1	51.82%	74.91%	99.89%
2	50.74%	74.27%	97.71%
3	50.47%	74.70%	98.61%
4	50.29%	75.12%	97.72%
5	52.19%	74.72%	98.62%
6	51.49%	74.92%	99.00%
7	50.30%	74.36%	99.88%
8	53.43%	74.94%	99.00%
9	51.50%	74.49%	98.99%
10	51.08%	73.86%	99.94%
Average	51.33%	74.63%	98.94%
Standard Deviation	0.009862	0.00383	0.008173

There is a large difference in the success rate of these protocols. MME averaged a success rate of 51.33%, far below the success rate of OSPF which averaged 74.63%. The NTO routing method was extremely successful due to the implementation of the commodity flow averaging 98.94% success. There is very little variance to the success rates for each iteration of this scenario. Figure 4.2 displays the average utilization and associated 95% confidence intervals when compared to each other.

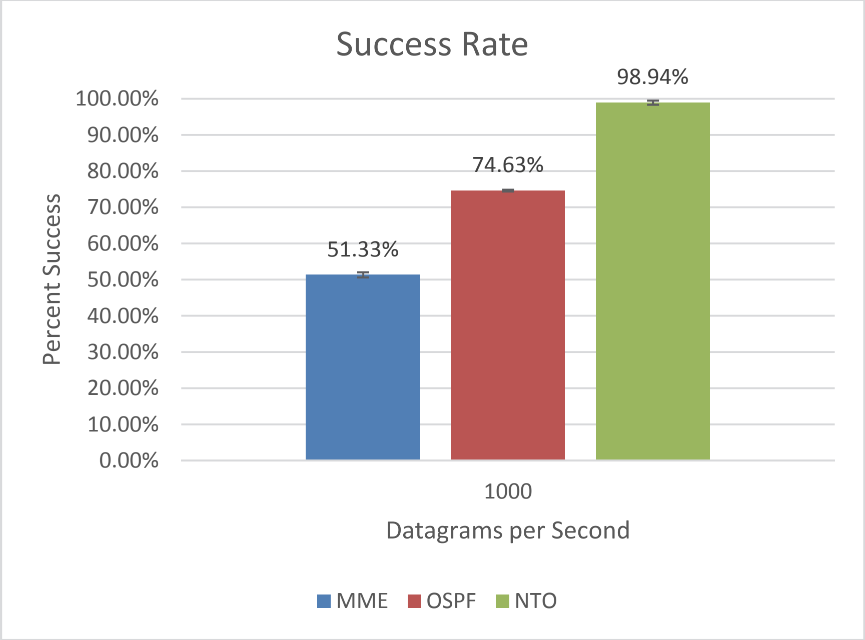


Figure 4.2: Success rates of each routing method at 1000 datagrams per second.

T-tests were conducted to compare the three routing methods. MME compared to OSPF resulted in a p-value of 6.176 x 10^{-17}. MME compared to NTO has the smallest p-value of 5.518 x 10^{-27}. Finally, when OSPF is compared to NTO, the p-value is 2.807 x 10^{-19}. None of these p-values demonstrate that these results are closely related as expected based on the large disparity between average success rates.

Another element of this experiment that is observed is the overall bandwidth that is attainable using these routing methods. Although the actual traffic generated for each of these test iterations varies, the maximum achievable data rate is calculated by taking the ratio of attempted throughput divided by the achieved throughput and multiplied by the maximum link speed attainable which is 1024 kbps. For this trial of 1000 datagrams per second, the calculated maximum possible data rate is 525 kbps for MME, 764 kbps for OSPF, and 1013 kbps for NTO. These rates are calculated and not confirmed and therefore presented as maximum possibilities.

4.3.4 Scenario One, Traffic Load Two

Iteration two involves the use of 500 datagrams transmitted every second to achieve approximately 900 kbps of traffic generation to test the various routing methods. By generating 205 bytes of data and adding the UDP header of eight bytes, the IP header of 20 bytes, and the 802 header of 14 bytes, a total datagram size of 247 bytes is place on the wire. This time, the wait time is established to send approximately 500 per second. Each 247 byte datagram, when converted, is 1976 bits. This generated 988,000 bits per second and when divided by 1024 to achieve kilobits, approximately 964 kilobits per second were transmitted from Host 1 to Host 2. Table 4.4 provides the percentage of datagrams that were received by Host 2.

Again, there is a large difference in the success rate of these protocols. Under this traffic load MME averaged a success rate of 52.96%, OSPF averaged 76.07%, and the NTO routing method proved is optimal again at 99.25% success. The values appear to maintain their same linear relationship meaning the values are not rising or falling. The 500 datagram per second results are shown in Figure 4.3 displays the average utilization and associated 95% confidence intervals when compared to the three other datagram sizes.

Table 4.4: Success rate using 500 datagrams per second.

	MME	OSPF	NTO
1	52.82%	72.56%	98.45%
2	53.30%	73.54%	99.26%
3	51.55%	78.39%	99.93%
4	52.04%	75.16%	98.80%
5	54.86%	78.81%	99.94%
6	53.31%	75.76%	99.93%
7	52.86%	76.13%	99.93%
8	53.04%	76.00%	97.79%
9	51.38%	76.17%	99.95%
10	54.43%	78.17%	98.49%
Average	52.96%	76.07%	99.25%
Standard Deviation	0.011257	0.020263	0.008086

The T-tests on this dataset provide similar results to the first. MME compared to OSPF resulted in a p-value of 9.323×10^{-15}. MME compared to NTO again has the smallest p-value of 6.810×10^{-25}. OSPF compared to NTO has a p-value of 2.229×10^{-13}. Similarly, none of these p-values demonstrate that these results are closely related.

Maximum data rate for this trial of 500 datagrams per second is also calculated. The maximum possible data rate is 542 kbps for MME, 778 kbps for OSPF, and 1016 kbps for NTO. Similar relative ratio to the 1000 datagram per second averages.

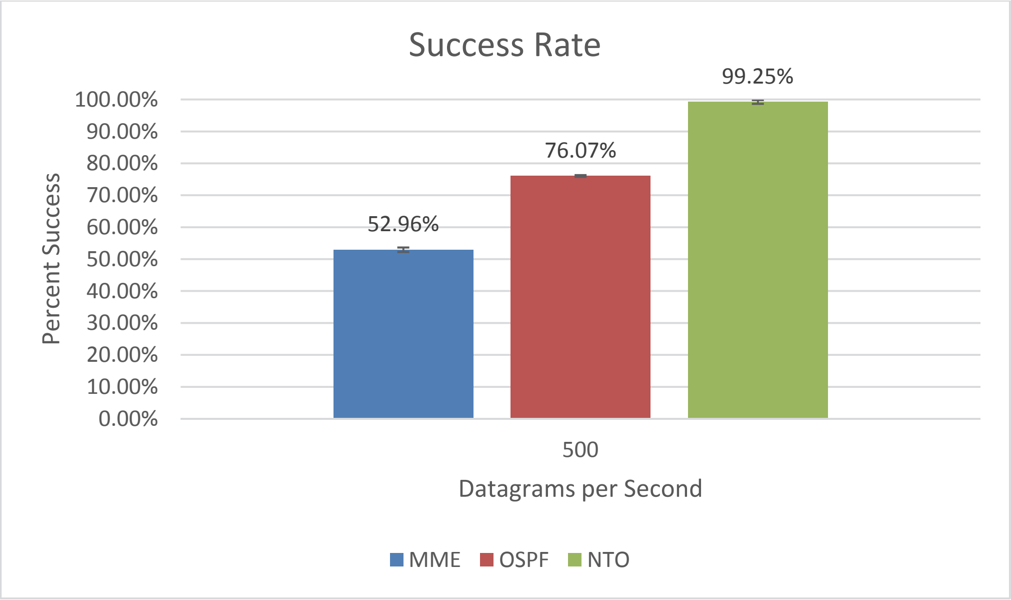


Figure 4.3: Success rates of each routing method at 500 datagrams per second.

4.3.5 Scenario One, Traffic Load Three

This experiment involves the use of 250 datagrams transmitted every second to achieve approximately 900 kbps of traffic generation to test the various routing methods. The payload is now set to 425 bytes of data and again adding the UDP header of eight bytes, the IP header of 20 bytes, and the 802 header of 14 bytes, the total datagram size is 467 bytes. The wait time is established to send approximately 250 per second. Each 467 byte datagram, when converted, is 3736 bits. This generated 934,000 bits per second and when divided by 1024 to achieve kilobits, approximately 912 kilobits per second were transmitted from Host 1 to Host 2. Table 4.5 provides the percentage of datagrams that were successfully received by Host 2.

Similar results were provided using this traffic load. MME averaged a success rate of 49.80%, OSPF averaged 73.52%, and the NTO achieved 98.84% success. Figure 4.4 displays the average utilization and associated 95% confidence intervals when compared to the other three datagram sizes. Again, there is very little variance to the success rates for each iteration of this scenario.

Table 4.5: Success rate using 250 datagrams per second.

	MME	OSPF	NTO
1	50.44%	73.66%	99.92%
2	51.25%	73.83%	99.88%
3	49.91%	74.29%	99.89%
4	49.32%	73.21%	96.78%
5	48.43%	74.36%	99.92%
6	49.77%	72.40%	96.91%
7	49.66%	72.76%	99.91%
8	49.90%	73.64%	98.23%
9	50.08%	74.29%	97.03%
10	49.22%	72.83%	99.91%
Average	49.80%	73.52%	98.84%
Standard Deviation	0.007511	0.007013	0.014303

T-tests results are again insignificant. MME compared to OSPF resulted in a p-value of 6.270 x 10^{-24}, the smallest p-value using this traffic load. MME compared to NTO is not the smallest in this case with a p-value of 5.701 x 10^{-21}. OSPF compared to NTO has a p-value of 1.153 x 10^{-16}. Similarly, none of these p-values demonstrate that these results are closely related.

Maximum data rate for this trial of 250 datagrams per second is also calculated. The maximum possible data rate is 510 kbps for MME, 753 kbps for OSPF, and 1012 kbps for NTO. These values are closely related to the 1000 and 500 datagram results.

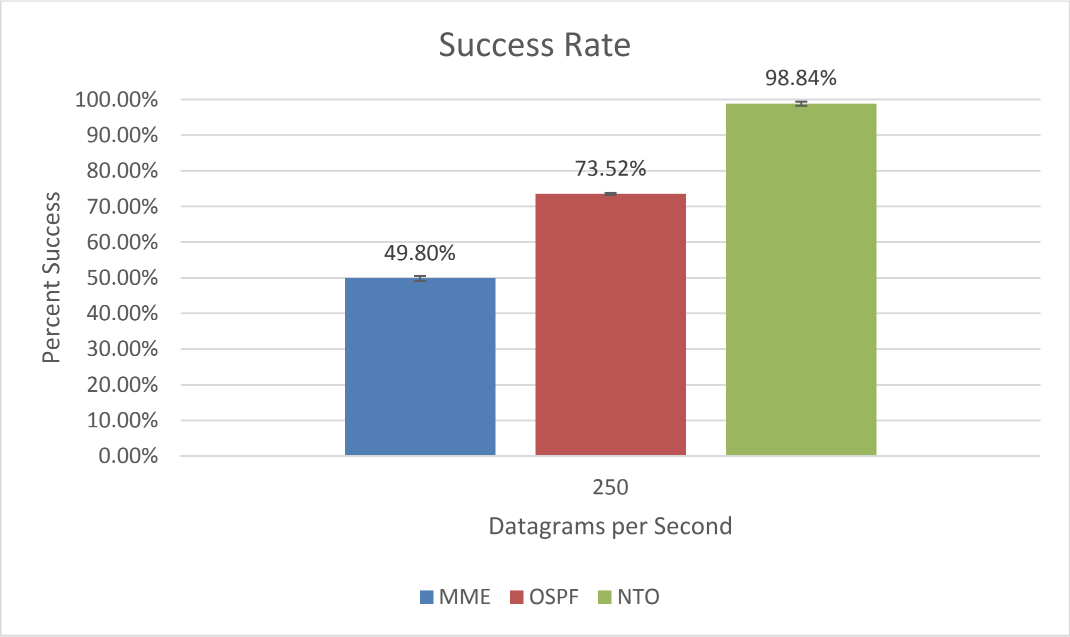


Figure 4.4: Success rates of each routing method at 250 datagrams per second.

4.3.6 Scenario One, Traffic Load Four

This experiment involves the use of 125 datagrams transmitted every second to achieve approximately 900 kbps of traffic generation to test the various routing methods. The payload is now set to 906 bytes of data and again adding the UDP header of eight bytes, the IP header of 20 bytes, and the 802 header of 14 bytes, the total datagram size is 948 bytes. The wait time is established to send approximately 125 per second. Each 948 byte datagram, when converted, is 7584 bits. This generated 948,000 bits per second and when divided by 1024 to achieve kilobits, approximately 926 kilobits per second were transmitted from Host 1 to Host 2. Table 4.6 provides the percentage of datagrams that were received by Host 2.

Success rates did not vary significantly under this traffic load. MME achieved 49.41% success, OSPF averaged 72.23%, and the NTO achieved the highest rate at

99.50% success. Again, this is seen in Figure 4.5 which displays the average utilization and associated 95% confidence intervals when compared to the other three datagram sizes.

Table 4.6: Success rate using 125 datagrams per second.

	MME	OSPF	NTO
1	50.97%	71.13%	99.96%
2	50.17%	71.56%	99.95%
3	49.35%	71.73%	96.93%
4	47.91%	71.76%	99.96%
5	48.25%	73.27%	99.95%
6	49.08%	72.56%	98.39%
7	50.80%	72.64%	99.96%
8	48.59%	72.30%	99.96%
9	52.09%	73.30%	99.95%
10	46.86%	72.06%	99.95%
Average	49.41%	72.23%	99.50%
Standard Deviation	0.016004	0.007201	0.010276

T-tests conducted show that MME compared to OSPF has a p-value of 5.054 x 10^{-15}. MME compared to NTO again has the smallest p-value of 4.209 x 10^{-22}. OSPF compared to NTO has a p-value of 1.247 x 10^{-21}. None of these p-values demonstrate that these results have any relationship.

Maximum data rate for this trial of 125 datagrams per second is also calculated. The maximum possible data rate is 506 kbps for MME, 740 kbps for OSPF, and 1019 kbps for NTO. Similar ratio to the 1000, 500, and 250 datagram per second averages.

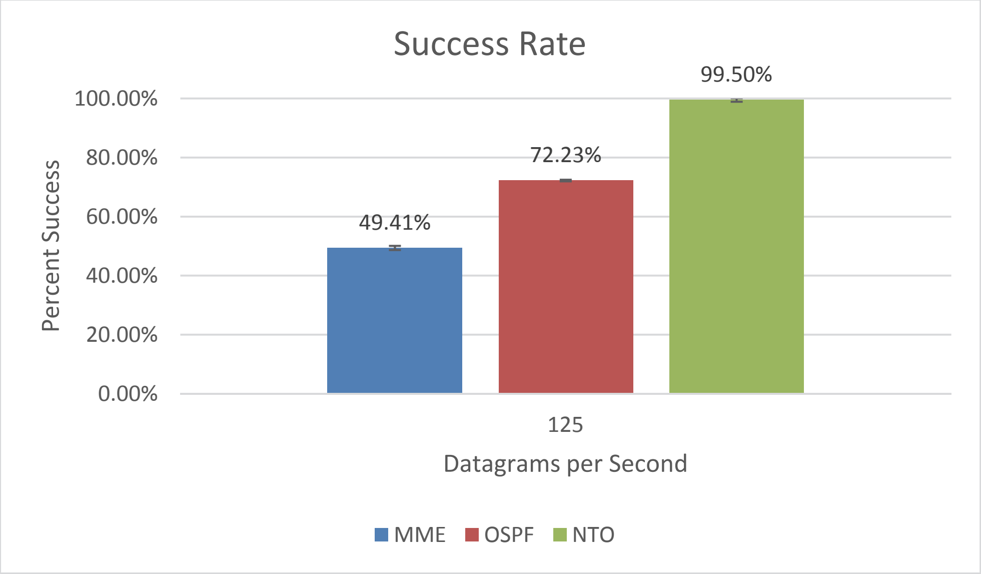


Figure 4.5: Success rates using each routing method at 125 datagrams per second.

4.3.7 Scenario One Comparisons

Previously discussed is the calculated limit at which these protocols could pass traffic with an overall connection of 1024 kbps from end to end as in this scenario. Figure 4.6 compares all four traffic load results showing all were similar.

It is assumed that traffic load has little to no effect on performance using any of these routing methods. However, after further investigation it is determined that MME and OSPF react differently to various traffic conditions. T-tests were conducted on the data from the same protocol over the various traffic conditions. P-values for MME show no correlation to data from other traffic loads as shown in Table 4.7. OSPF p-values also show no probability of relationship from one traffic load to another. Table 4.8 shows the corresponding p-values for each traffic load. Only the NTO routing method shows relationship between the results with p-values greater than 0.05 for all comparisons. The NTO results are shown in Table 4.9.

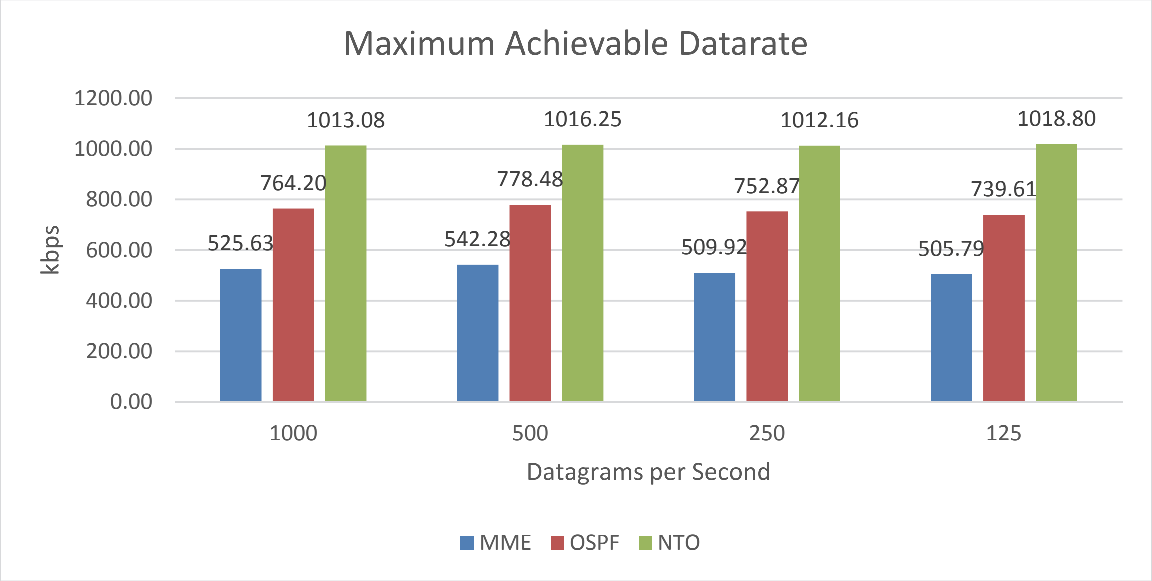


Figure 4.6: Maximum bandwidth possible.

After running all protocols using all four different traffic loads, it is apparent that the NTO method is extremely effective in network routing. The commodity flow is a critical component in MANET performance and efficiency. The MikroTik router performed well under these conditions. A screenshot of the MikroTik router using the commodity flow is shown in Figure 4.7. In the screenshot, the amount of traffic received on *ether4* is approximately 963 kbps and the outbound is split amongst both *ether2* and *ether3* at 241 kbps and 722 kbps respectively.

Table 4.7: P-Values when comparing MME using different traffic loads.

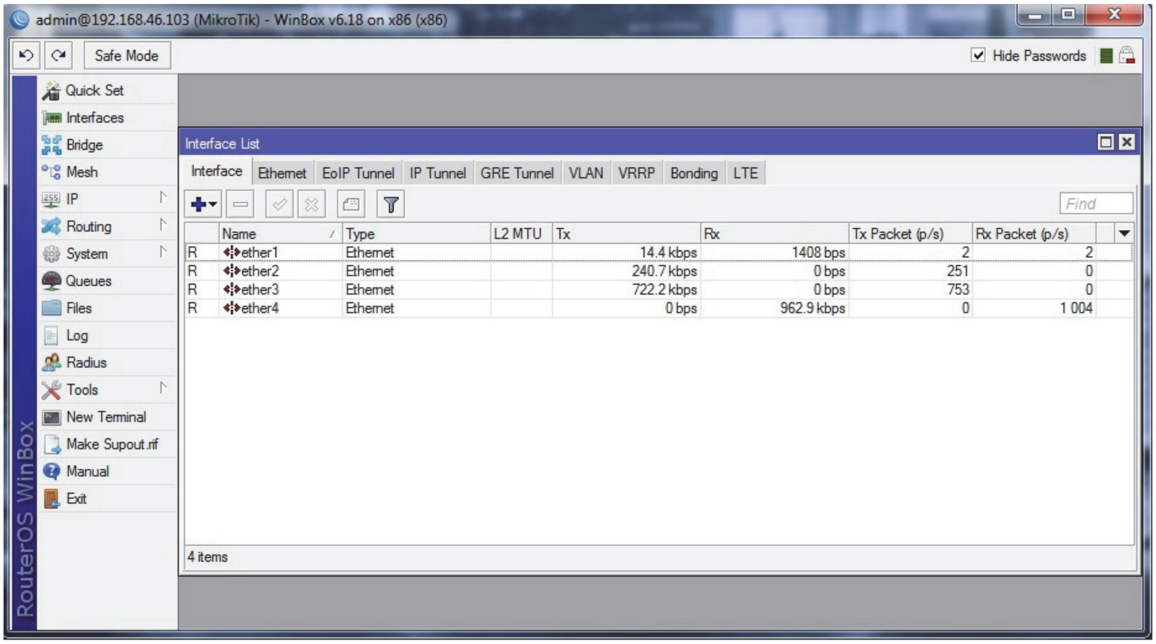
MME P-Value Comparisons				
	1000	500	250	125
1000		0.001493893	0.000567	0.00277
500	0.001494		8.61E-07	1.46E-05
250	0.000567	8.60581E-07		0.249309
125	0.00277	1.46428E-05	0.249309	

Table 4.8: P-Values when comparing OSPF using different traffic loads.

OSPF P-Value Comparisons				
	1000	500	250	125
1000		0.026419764	0.000322	1.35E-07
500	0.02642		0.00157	6.95E-05
250	0.000322	0.00156964		0.000361
125	1.35E-07	6.94874E-05	0.000361	

Table 4.9: P-Values when comparing NTO using different traffic loads.

NTO P-Value Comparisons				
	1000	500	250	125
1000		0.201371174	0.427466	0.097098
500	0.201371		0.222302	0.277114
250	0.427466	0.222302414		0.127159
125	0.097098	0.277113662	0.127159	



	Name	Type	L2 MTU	Tx	Rx	Tx Packet (p/s)	Rx Packet (p/s)
R	ether1	Ethernet		14.4 kbps	1408 bps	2	2
R	ether2	Ethernet		240.7 kbps	0 bps	251	0
R	ether3	Ethernet		722.2 kbps	0 bps	753	0
R	ether4	Ethernet		0 bps	962.9 kbps	0	1 004

4 items

Figure 4.7: MikroTik Router interface showing commodity flow in action.

4.4 Scenario Two – Fast Switching

This scenario is designed to test how the NTO compares to other protocols when the schedule is not followed. Aircraft are delayed or disabled with little or no notice. This scenario emulates time delays with the aircraft to determine when the NTO is just as ineffective as OSPF or MME. The delays occur at five second intervals until the NTO proves worse than both dynamic routing protocols.

4.4.1 NTO Route Scheduling

For this simulation, it is assumed that the NTO planners have planned aircraft schedules to meet the desired topology. From section 3.9, the topology is implemented such that a stream of aircraft fly in a line, one after the other at one minute intervals. They are in range of the two ground stations providing a network path between them. The aircraft are in range for one minute and five seconds. This provides a five second overlap where two aircraft are in range. The route schedule can take place anywhere during that five second interval. The middle of the overlap is best, but changes have to take place on the second since that is the granularity of the router chron scheduler. Choosing the 57th second of every minute is sufficient. Therefore, the routes change according to this schedule. All links are 256 kbps and do not change speeds.

4.4.2 Results

This experiment involves the use of 250 datagrams transmitted every second to achieve approximately 240 kbps of traffic generation to test the various routing methods. The payload is set to 103 bytes of data and upon adding the UDP header of eight bytes, the IP header of 20 bytes, and the 14 byte Ethernet header, the total datagram size is 145 bytes. The wait time is established to send approximately 250 per second. Each 145 byte datagram, when converted, is 1160 bits. This generated 290,000 bits per second and when divided by 1024 to achieve kilobits, approximately 283 kilobits per second were transmitted from Host 1 to Host 2. Even though this is higher than what the network connections are capable of, it was found that this datagram size averaged out to 240 kbps due to additional processing delay at the source Host 1. This was confirmed by analysis of the Python script running on Host 1. Table 4.9 provides the percentage of datagrams that were received by Host 2.

The NTO routing method demonstrated as the optimal routing method. Figure 4.8 shows how the delay in aircraft affects the performance of the NTO routing method. The graph displays the following correlation:

- NTO-Prime – Aircraft arrive on time, routes updated during overlap.
- NTO-0 – Aircraft connection created at same time route changes.
- NTO-5 – Simulates 5 second delay for all aircraft.
- NTO-10 – Simulates 10 second delay for all aircraft.
- NTO-15 – Simulates 15 second delay for all aircraft.
- NTO-20 – Simulates 20 second delay for all aircraft.
- NTO-25 – Simulates 25 second delay for all aircraft.
- NTO-30 – Simulates 30 second delay for all aircraft.
- NTO-35 – Simulates 35 second delay for all aircraft.

NTO Prime provided a 100% delivery rate. NTO-0 takes place at the same time the link to the new aircraft is established achieving a success rate of 99.39%. Changing routes at the same time the link is being established or disconnected can cause a race condition between the two events and therefore lost less than one percent of data. NTO-5 incurs a five second delay, NTO-10 a ten second delay, and so forth. Each five second delay causes an approximate 9% in data loss. OSPF achieved a 63.03% average success rate providing better results than a 25 second delay in schedule with the NTO. This implies that if the schedule is not guaranteed and aircraft may arrive over 25 seconds late, OSPF is the better option. MME achieved a 44.51% success rate which was worse than a 30 second delay but better than a 35 second delay. As in the case with OSPF, if the aircraft are 35 seconds late, MME would perform better. The 95% confidence intervals are displayed on the chart.

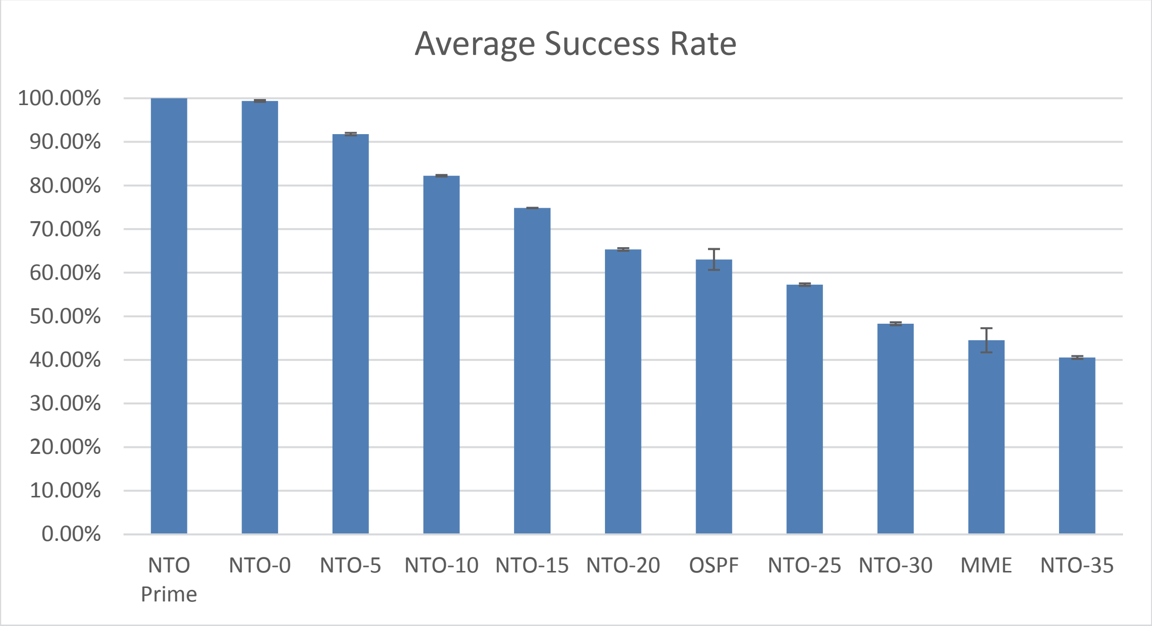


Figure 4.8: Percentage of success rate based on routing method. The number following the NTO is the amount of delay incurred by the aircraft for that trial.

A t-test was also performed on these results. MME was compared to OSPF, and each NTO test run was compared to both MME and OSPF. MME and OSPF proved to have no significant correlation. The p-value between MME and OSPF is 7.032×10^{-10}. Most NTO results had extremely small p-values below 7×10^{-7} except NTO-20 which had a value of 0.0294. Although this is under the standard 0.05 probability, this is the where the p-value demonstrates the closest relationship. This means that NTO-20 and OSPF are the closest related of all of the NTO test runs. MME and NTO-30 have a smaller correlation with a p-value of 0.00659 and another similar value at NTO-35 of 0.00526. Based on the average results, the two routing methods share a similar result somewhere between those two time intervals. Figure 4.9 displays the graph of these p-values and where they peak in relationship to their associated NTO time delay.

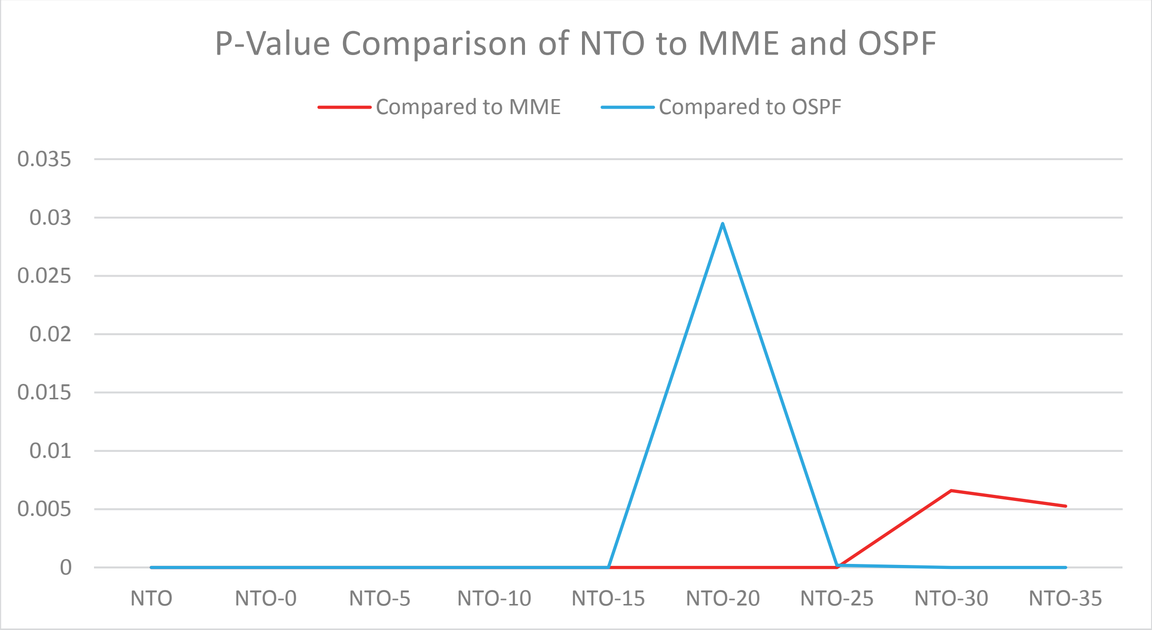


Figure 4.9: P-values comparing MME to NTO and OSPF to NTO.

4.5 Conclusions

This chapter presented the network modeling and traffic load tests conducted on these models using different levels of traffic. The results indicate that the NTO is the optimal routing method when aircraft arrive according to the schedule. Any delay in aircraft has an impact on the NTO routing performance and therefore should be avoided.

If too much schedule delay occurs, a dynamic routing protocol may provide better results.

Test results further indicated a higher success rate with OSPF over MME.

5 Conclusions

Military operations rely heavily on MANETs to perform routine, exercise, and real-world operations. A high level of QoS is necessary to ensure mission data successfully arrives across that network. Typical routing algorithms have their place, but MANETs require a more unique method to ensure network integrity. MANETs are extremely dynamic and constantly changing. A method must be implemented to ensure no bandwidth is wasted when a demand is present. Using the NTO to predict aircraft positions and leverage it for network routing can improve network QoS as long as the schedule has minimal deviation. If there is any chance of deviation in aircraft schedule, the impact may have severe consequences to MANET Quality of Service. A method should be implemented to adjust for aircraft delay or unavailability.

5.1 Research Impact

This research demonstrates that the military has a great advantage when routing data on MANETs. The military schedules all aircraft for arrival at precise locations throughout the day. Using this priori knowledge to advance network routing decisions can assist in achieving mission success.

The ability for the NTO to affect ATO scheduling may also be of greater significance. To achieve a high level of QoS, scheduling delays must be eliminated. More importantly overlap in aircraft schedules to provide connectivity can also be deemed important. This is not typically a concern with ATO planners, but should be considered such if the NTO is to be implemented. The elimination of scheduling delay has additional adverse effects on the mission which must be considered, it reduces flexibility, the key to air power.

5.2 Contributions

The goal of this research was to demonstrate concepts previously performed in simulation using NS2 or OPNET. While these simulations are effective, they are contained in a single discrete-event network simulator. This research has advanced these concepts into a virtual environment using routers and real operating systems. This provides a more realistic environment to assess the feasibility of these routing methods. The NTO must accomplish the following:

- Predetermine network connectivity and topology based on aircraft location provided in the NTO.
- Calculate an optimal commodity flow to meet bandwidth requirements.
- Minimize data lost when comparing to either MME or OSPF.

Several tests were executed to demonstrate the performance of the NTO routing method with multiple test runs for each scenario using different patterns of network traffic. The NTO constantly achieved around a 99% success rate while OSPF and MME were much lower at 74% and 50% respectively.

5.3 Future Work

The following are suggestions for future work within this topic:

- Due to the significant impact of aircraft delay, future work should include adaptation to these delays. Embed an algorithm or control system that can watch for problems in the network and dynamically adapt on the fly.

- Implement the NTO using wireless connections and protocols over hardware radios outside of the virtual environment.
- Evaluate effects of the NTO commodity flow with a dynamic routing protocol to provide weights on network connections within that routing protocol.
- Compare to other MANET protocols like AODV or OSPF-MDR.

5.4 Summary

This research demonstrated the effectiveness of the NTO in a dynamic network environment. However, aircraft availability must be guaranteed. The unavailability of an aircraft for minutes or even seconds can negatively affect QoS. Otherwise, the NTO has the capability to use pre-determined knowledge to make decisions including splitting network flows and timely route changes. Furthermore the algorithms are run offline before they are implemented and scheduled as necessary. Simulations demonstrate that this knowledge has a significant impact on network QoS providing less packet loss and an overall increase in network throughput.

Bibliography

[1] NSA, "Global Information Grid," 23 April 2012. [Online]. Available: http://www.nsa.gov/ia/programs/global_information_grid/index.shtml. [Accessed 2014].

[2] M. D. Compton, K. Hopkinson and S. Graham, "The Network Tasking Order (NTO)," in *IEEE, Military Communications Conference*, 2008.

[3] J. A. Betances, Context Aware Routing Management Architecture For Airborne Networks, Wright-Patterson Air Force Base, Ohio, 2012.

[4] Cisco Systems, "Mobile Ad Hoc Networking Revamps Military Communications," November 2011. [Online]. Available: http://www.cotsjournalonline.com/articles/view/102158. [Accessed 22 January 2015].

[5] B.-N. Cheng and S. Moore, "A Comparison of MANET Routing Protocols on Airborne Tactical Networks," in *Military Communications Conference*, Orlando, FL, 2012.

[6] M. Karimi and D. Pan, "Challenges for Quality of Service (QoS) in Mobile," in *Wireless and Microwave Technology Conference (WAMICON)*, 2009.

[7] I. D. Chakeres and L. Klein-Berndt, "AODVjr, AODV Simplified," *ACM SIGMOBILE Mobile Computing and Communications Review,* vol. 6, no. 3, pp. 100-101, 2002.

[8] S. Das, C. Perkins and E. Royer, "Ad Hoc On Demand Distance Vector (AODV) Routing," in *Mobile Ad-hoc Network (MANET) Working Group, IETF*, 2002.

[9] D. Kiwior, E. G. Idhaw and S. V. Pizzi, "Quality of service (QoS) sensitivity for the OSPF protocol in the airborne networking environment," in *Military Communications Conference*, Atlantic City, NJ, 2005.

[10] R. Ogier and P. Spagnolo, "Mobile Ad Hoc Network (MANET) Extension of OSPF Using Connected Dominating Set (CDS) Flooding," August 2009. [Online].

Available: <http://www.rfc-editor.org/info/rfc5614>.

[11] T. Henderson, "OSPFv3 MANET MDR," Boeing, 27 July 2010. [Online]. Available: http://hipserver.mct.phantomworks.org/ietf/ospf/. [Accessed 12 September 2014].

[12] MicroTik, "MicroTik RouterOS Manual," 4 November 2014. [Online]. Available: http://wiki.mikrotik.com/wiki/Manual:TOC.

[13] R. M. Du Plessis, "Poor Man's Explanation of Kalman Filtering," *Autonectics Division of North American Rockwell Corporation,* 1967.

[14] N. Stuckey, Stochastic Estimation and Control of Queues within a Computer Network, Wright Patterson AFB: Air Force Institute of Technology, 2007.

[15] M. Alqahtani, "Stochastic Prediction and Feedback Control of Routers Queue Size in Virtual Network Environment," Air Force Institue of Technology, Wright Patterson AFB, OH, 2014.

[16] J. D. Haught, K. M. Hopkinson and J. M. Hemmes, "The modeling and simulation of a network controller based on router queue-size predictions," *Journal of Defense Modeling and Simulation,* pp. 105-115, 2012.

[17] M. D. Compton, K. M. Hopkinson, G. L. Peterson and J. T. Moore, "Using Modeling and Simulation to Examine the Benefits of a Network Tasking Order," *The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology,* pp. 205-217, 2010.

[18] C. Scheideler, "Network Flows III - Multicommodity Flows," 2003. [Online].

[19] D. Mills, J. Burbank, Ed., J. Burbank and W. Kasch, "Network Time Protocol Version 4: Protocol and Algorithms Specification," [Online]. Available: <http://www.rfc-editor.org/info/rfc5905>.

[20] W. Lewandowski, J. Azoubib and W. J. Klepczynski, "GPS: Primary Tool for Time Transfer," in *IEEE*, 1999.

[21] San Jose Technology, Inc., "SANAV," 2015. [Online]. Available: http://es.sanav.com/uploadfiles/9/proimages/VP_200T_GPS_Time_Transfer_R.j

pg.

[22] C. Kapoor and G. Sharma, "To Improve the QoS in Manets Through Analysis Between Reactive and Proactive Routing Protocols," *Computer Science & Engineering: An International Journal (CSEIJ),* 2011.

[23] N. McKeown, "OpenFlow: Enabling Innovation in Campus Networks," Open Networking Foundation, Stanford University, 2008.

[24] F. Bai and A. Helmy, "A Survey of Mobility Models in Wireless Adhoc Networks".

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 074-0188*

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of the collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to an penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.
PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE *(DD-MM-YYYY)* 26-03-2015	**2. REPORT TYPE** Master's Thesis	**3. DATES COVERED** *(From – To)* March 2014 – March 2015

4. TITLE AND SUBTITLE Network Routing Using the Network Tasking Order, a Chron Approach	**5a. CONTRACT NUMBER**
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S) Paltzer, Nicholas J., Captain, USAF	**5d. PROJECT NUMBER** 15G-453B
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S) Air Force Institute of Technology Graduate School of Engineering and Management (AFIT/EN) 2950 Hobson Way, Building 640 WPAFB OH 45433-8865	**8. PERFORMING ORGANIZATION REPORT NUMBER** AFIT-ENG-MS-15-M-059
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Air Force Office of Scientific Research 875 N Randolph St Ste 325 Rm 3112, Arlington AFB, VA, 22203 426-9545, robert.bonneau.1@us.af.mil ATTN: Mr. Robert Bonneau	**10. SPONSOR/MONITOR'S ACRONYM(S)** **AFOSR/RTC**
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
DISTRUBTION STATEMENT A. APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES
This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

14. ABSTRACT
This thesis promotes the use of the network tasking order (NTO), in collaboration with the air tasking order (ATO), to optimize routing in Mobile Ad hoc Networks (MANET). The network topology created by airborne platforms is determined ahead of time and network transitions are calculated offline prior to mission execution. This information is used to run maximum multi-commodity flow algorithms offline to optimize network flow and schedule route changes for each network node. These calculations and timely route modifications increases network efficiency. This increased performance is critical to command and control decision making in the battlefield. One test scenario demonstrates near a 100% success rate when route scheduling and splitting network traffic over an emulated MANET compared to Open Shortest Path First (OSPF) which only achieved around a 71% success rate, and Mesh Made Easy (MME) which achieved about 50% success. Another test scenario demonstrates that the NTO can experience degradation due to schedule delay. Overall, if executed and planned properly, the NTO can significantly improve network Quality of Service (QoS).

15. SUBJECT TERMS
MANET, NTO, Airborne Networks

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON** Kenneth Hopkinson, AFIT/ENG
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			**19b. TELEPHONE NUMBER** *(Include area code)*
U	U	U	UU	84	(937) 255-6565, ext 4579 (Kenneth.hopkinson@afit.edu)

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39-18