Air Force Institute of Technology

AFIT Scholar

3-2007

Performance Evaluation of Ad Hoc Routing in a Swarm of Autonomous Aerial Vehicles

Matthew T. Hyland

Recommended Citation

**PERFORMANCE EVALUATION OF AD HOC ROUTING PROTOCOLS IN A
SWARM OF AUTONOMOUS UNMANNED AERIAL VEHICLES**

THESIS

Matthew T. Hyland, Captain, USAF

AFIT/GCS/ENG/07-07

**DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY**

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GCS/ENG/07-07

PERFORMANCE EVALUATION OF AD HOC ROUTING PROTOCOLS IN A SWARM OF AUTONOMOUS UNMANNED AERIAL VEHICLES

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 Systems

Matthew T. Hyland, BS

Captain, USAF

March 2007
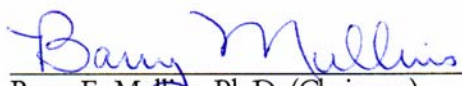
AFIT/GCS/ENG/07-07

PERFORMANCE EVALUATION OF AD HOC ROUTING PROTOCOLS IN A SWARM OF AUTONOMOUS UNMANNED AERIAL VEHICLES

Matthew T. Hyland, BS

Captain, USAF

Approved:
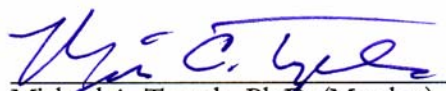
_____   28 Feb 07
Barry E. Mullins, Ph.D. (Chairman)   Date

_____   28 Feb 47
Rusty O. Baldwin, Ph.D. (Member)   Date

_____   27 Feb 47
Michael A. Temple, Ph.D. (Member)   Date

Abstract

This thesis investigates the performance of three mobile ad hoc routing protocols in the context of a swarm of autonomous unmanned aerial vehicles (UAVs). It is proposed that a wireless network of nodes having an average of 5.1774 log n neighbors, where n is the total number of nodes in the network, has a high probability of having no partitions. By decreasing transmission range while ensuring network connectivity, and implementing multi-hop routing between nodes, spatial multiplexing is exploited whereby multiple pairs of nodes simultaneously transmit on the same channel.

The proposal is evaluated using the Greedy Perimeter Stateless Routing (GPSR), Optimized Link State Routing (OLSR), and Ad hoc On-demand Distance Vector (AODV) routing protocols in the context of a swarm of UAVs using the OPNET network simulation tool. The first-known implementation of GPSR in OPNET is constructed, and routing performance is observed when routing protocol, number of nodes, transmission range, and traffic workload are varied. Performance is evaluated based on proportion of packets successfully delivered, average packet hop count, and average end-to-end delay of packets received.

Results indicate that the routing protocol choice has a significant impact on routing performance. While GPSR successfully delivers 50% more packets than OLSR, and experiences a 53% smaller end-to-end delay than AODV when routing packets in a swarm of UAVs, increasing transmission range and using direct transmission to destination nodes with no routing results in a level of performance not achieved using any of the routing protocols evaluated.

To my Mother and Father

Acknowledgments

I would like to thank my thesis advisor, Dr. Barry Mullins, for his support, guidance and motivation throughout the course of this thesis effort. His patience and experience were always appreciated.

Matthew T. Hyland

Table of Contents

List of Figures

List of Tables

PERFORMANCE EVALUATION OF AD HOC ROUTING PROTOCOLS IN A SWARM OF AUTONOMOUS UNMANNED AERIAL VEHICLES

I. Introduction

It is often said that "the whole is greater than the sum of its parts." The concept described by this idiom is synergy, referring to the "phenomenon in which two or more discrete influences or agents create an effect greater than that predicted by knowing only the separate effects of the individual agents" [Wik07b]. One concludes that in order for synergy to exist, there needs to be some level of communication and collaboration between individual agents.

1.1 Motivation

There are a variety of proposals to use a group of small, inexpensive unmanned aerial vehicles (UAVs) to perform some task in a more stealthy, efficient, or safe manner than using traditional manned aircraft or other assets [AuM05], [LAN03], [USA05], [YPH06]. To facilitate effective synergies in these groups of UAVs, they need to be able to communicate effectively and efficiently. A computer simulation environment with the capability to evaluate and compare various methods of communication within a group of UAVs is an inexpensive step towards actually employing the tasks that have been proposed.

1.2 Overview and Goals

The goal of this research effort is to determine appropriate measures of communication effectiveness in the context of a group of UAVs, and then to evaluate the performance of several of communication methods under a variety of configurations to determine which is most effective.

1.3   Organization and Layout

In this chapter, the research topic is described and the motivation behind the effort is presented. Chapter 2 reviews important background information and discusses related research. In Chapter 3, the methodology used to perform the experiments is outlined. Chapter 4 presents the results of the experiment and provides some discussion and analysis of the outcome. Chapter 5 offers conclusions drawn from the results and describes some ideas for future work in this research area. Model implementation specifics and details about preliminary studies are provided in the appendix.

II. Background and Literature Review

2.1 Introduction

This chapter discusses Unmanned Aerial Vehicles (UAVs), Mobile Ad hoc Networks (MANETs), routing protocols, and recent research related to this effort. Section 2.2 defines UAVs and describes the synergy that can be attained in a UAV swarm. In Section 2.3, MANETs are defined, and various routing protocols designed for their use are described. In addition, relevant mobility models are presented. Section 2.4 describes other recent research efforts in the analysis of routing protocols for UAV swarms.

2.2 Unmanned Aerial Vehicles

The Department of Defense defines an Unmanned Aerial Vehicle as follows:

> A powered, aerial vehicle that does not carry a human operator, uses aerodynamic forces to provide vehicle lift, can fly autonomously or be piloted remotely, can be expendable or recoverable, and can carry a lethal or non-lethal payload. Ballistic or semi-ballistic vehicles, cruise missiles, and artillery projectiles are not considered unmanned aerial vehicles. Also called UAV. [DoD01]

With no human operator, unmanned aircraft do not need to carry life support systems, human-operable flight controls, or even windows. Without this additional equipment, the UAVs can be built smaller, lighter and cheaper than manned aircraft. Furthermore, since flight characteristics such as acceleration and duration do not need to be constrained to the limitations of the human body, unmanned aircraft can be designed to perform maneuvers that a human pilot could not withstand, and mission durations can exceed human endurance.

2.2.1 Current UAV Usage

Unmanned aerial vehicles have proven to be worthwhile assets in real-world scenarios around the world, most recently during Operations ENDURING FREEDOM and IRAQI FREEDOM. The RQ-4 Global Hawk provided surveillance for time-sensitive targeting operations in the Iraqi missile engagement zone during combat operations. Although the Global Hawk flew only 5% of the high-altitude missions, it accounted for over half of the time-sensitive targeting intelligence used to combat Iraqi air defense equipment [USA05].

2.2.2 UAV Swarms

The successful use of unmanned aircraft for intelligence, surveillance and reconnaissance (ISR) to date has largely relied on small numbers of aircraft transmitting information over a dedicated channel. While these aircraft are usually operated remotely by an Air Force pilot, great potential exists for using swarms of autonomous unmanned aircraft to perform similar tasks.

In [AuM05], Augeri and Mullins propose a swarm they call a Host of Armed Reconnaissance Vehicles Enabling Surveillance and Targeting, or HARVEST. As shown in Figure 1, HARVEST is a heterogeneous collection of unmanned aircraft which differ in sensor function as well as in their ability to communicate. A large number of small sensor UAVs, which may each have different sensing capabilities, gather sensor data. These sensor UAVs transmit using low-power radios to roving swarm monitors which provide guidance to the swarm in addition to routing sensor data. One or more Unmanned Combat Aerial Vehicles (UCAVs) can engage in air-to-air or air-to-ground combat to protect the swarm or effect offensive operational objectives. One or more edge-access UAVs carry high-power

radios and act as access nodes to an air- or ground-based station for extracting intelligence data and passing queries and instructions to the swarm.
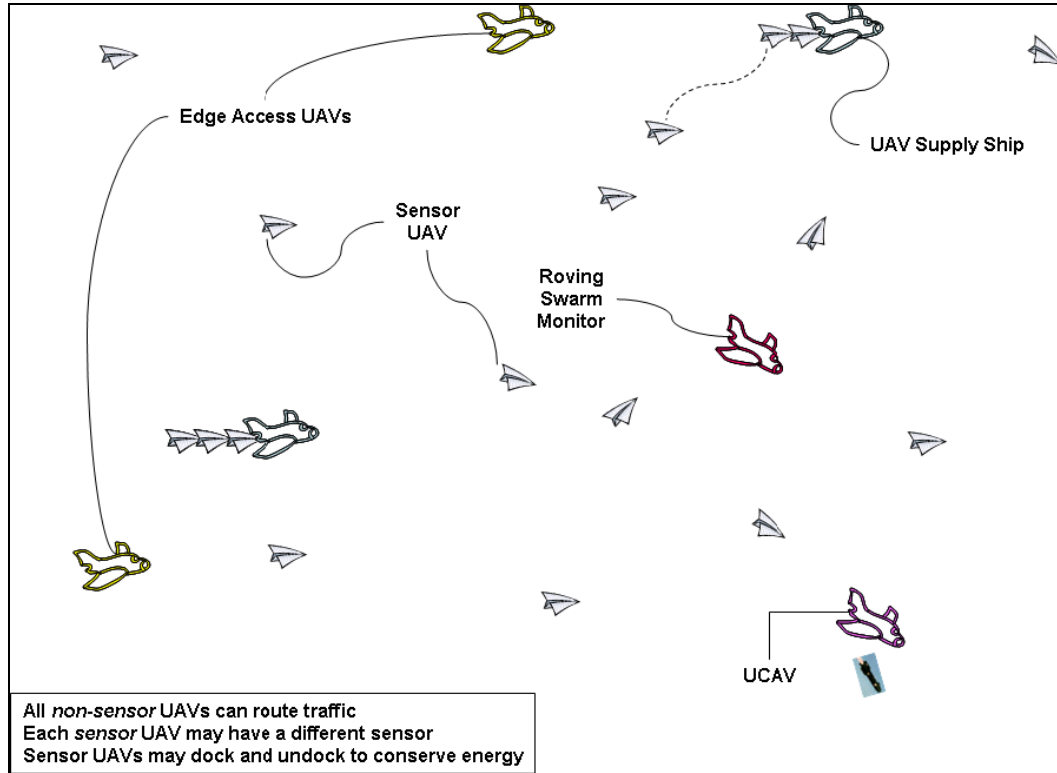
Figure 1. HARVEST configuration [AuM05]

2.3   Mobile Ad hoc Networks

In [Rot99], Royer and Toh define a Mobile Ad hoc Network, or MANET, as:

> …a collection of mobile nodes that are dynamically and arbitrarily located in such a manner that the interconnection between nodes are capable of changing on a continual basis.

HARVEST, described in Section 2.2.2, exhibits the properties of a MANET as described above; the mobility of aircraft within the swarm necessarily move nodes outside of the transmission range of some nodes, and into the transmission range of others. This limitation on transmission range has the effect of partitioning the network, and allows nodes on one side of the swarm to communicate at the same time an on identical frequencies as

5

nodes on the other side of the swarm. Thus, spatial separation allows them to share the medium using what is called spatial multiplexing.

This partitioning, however, means that data must be relayed from one node to another to forward information to edge-access nodes which transmit it out of the swarm. To facilitate this packet forwarding, a routing protocol must be used to discover and manage efficient routes between nodes in the swarm.

2.3.1 OSI Model

The Open Systems Interconnection (OSI) Network Reference Model was developed jointly by the International Standards Organization (ISO) and the International Telecommunications Union (ITU) in 1984 to serve as a framework for developing various standards for interconnecting systems [ISO94]. Such a framework ensures that disparate development efforts could be compatible with each other, so long as they adhere to the framework.

The model itself divides the job of communicating information over a network into seven layers of responsibility. Each relies only on the services of the layer immediately below it, and provides services only to the layer immediately above [Wik06]. This division of responsibility allows seamless communication between millions of computers connected to the Internet, despite the fact that they are produced by different manufacturers and may use vastly different means by which to connect to the Internet, such as a dial-up modem, wireless connection, or a high-speed cable modem.

The seven layers of the OSI model are shown in Figure 2. The physical layer at the bottom of the diagram is layer 1, and the application layer at the top is layer 7. The Internet implements the OSI model in four groups; application protocols such as Hypertext Transfer

Protocol (HTTP) and File Transfer Protocol (FTP) correspond to the application, presentation and session layers. Two transport protocols, Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) transport data segments across a heterogeneous network to an application at the destination host. Transport protocols manage the efficient use of network resources, and may provide reliable data transfer to the layers above. Both TCP and UDP use the Internet Protocol (IP) at the network layer. Network layer protocols handle routing and relay considerations to deliver data segments to the appropriate transport-layer protocol at the destination. Technologies such as Ethernet, wireless Ethernet and dial-up modems implement both the data link and physical layers, and deliver IP packets across a single network link. The physical and data link layers are typically implemented in hardware [ISO94].
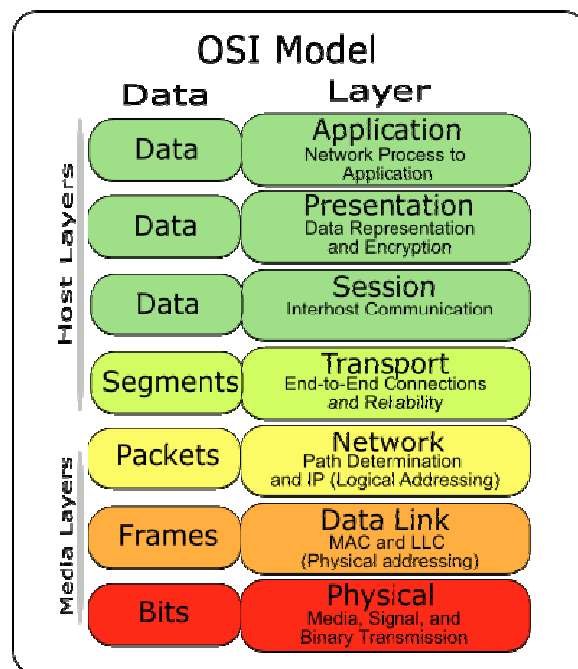
Figure 2. OSI Network Reference Model [Wik06]

7

2.3.2 IEEE 802.11 Standard

Communication with Unmanned Aerial Vehicles is a rather specialized task which typically does not rely on the interoperability of different components developed by different organizations, and may not implement every layer of the OSI model explicitly. It is certainly an economic benefit to use commercially-available hardware based on accepted standards and to leverage best practices from a large user base. The Institute of Electrical and Electronics Engineers (IEEE) 802.11 Wireless Ethernet standard [IEE03a], commonly known as Wi-Fi, has become the de facto standard for connecting to the Internet wirelessly. The IEEE 802.11 standard specifies both the medium access control (MAC) and physical (PHY) layers of the OSI model.

Wi-Fi networks can be configured as either an infrastructure-mode network or an ad hoc wireless network [KuR05]. In an infrastructure-mode network, a wireless access point (AP), typically connected to a wired network with Internet connectivity, coordinates network membership and all packet transmissions. In fact, wireless nodes in an infrastructure-mode network cannot transmit packets directly to another wireless node; packets are received by the AP and re-transmitted to the intended wireless destination node. Ad hoc networks, on the other hand, have no AP and wireless nodes transmit packets directly between each other.

2.3.2.1 Medium Access Control (MAC)

The primary function of the 802.11 MAC is to coordinate access to the shared medium (the wireless channel) and to ensure reliable transmission of packets across a single wireless link [KuR05]. To minimize interference on the radio channel, Wi-Fi employs a distributed coordination function (DCF) called Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) which is similar to the Carrier Sense Multiple Access with Collision

Detection (CSMA/CD) protocol used in wired Ethernet networks. An optional point coordination function (PCF) is specified in the standard [IEE03a] for use in infrastructure mode networks. Since this research assumes ad hoc networks, PCF will not be discussed.

Link-level reliability, or ensuring that a packet is successfully received by the intended node, is accomplished by acknowledgement (ACK) frames. Upon receiving a unicast data frame (broadcast and multicast frames are not acknowledged) and performing a CRC error check, a node waits for a short period of time known as the Short Inter-Frame Space (SIFS, described below) and then sends an acknowledgement frame back to the sending node. If a sending node does not receive an ACK frame within a specified amount of time, it re-transmits the frame. Data retransmission will continue until either an ACK is received, or a maximum number of retransmissions occur without an ACK, at which time the frame is dropped [KuR05].

The basic operation of CSMA begins with a node that has data to transmit. The node monitors the channel and if the channel is sensed idle, the node is free to transmit its data. With Ethernet CSMA/CD, transmitting nodes can immediately sense collisions and stop transmitting. Assuming half-duplex operation, wireless radios cannot detect collisions due to the fact that the signal being transmitted is far stronger than any signal received from another node; if two nodes transmit simultaneously to a third node, the sending nodes will not detect the collision. For this reason, 802.11 networks employ collision avoidance techniques. Collision avoidance is managed using a variety of delays called Inter-Frame Spaces (IFS) and random backoff delays. The relevant IFS times are specified by the specific PHY in use, and are [Bre97]:

- <u>Short Inter-Frame Space (SIFS):</u> At least long enough so that a transmitting radio has enough time to switch to receive mode to detect an acknowledgement frame

- <u>Slot time:</u> slightly longer than the SIFS; defined by the PHY such that a node can determine if any other node has begun transmitting during the previous slot, which is at least as long as the longest one-way propagation time possible in the network

- <u>Distributed Inter-Frame Space (DIFS):</u> SIFS plus two slot times

The random backoff scheme used is an exponential backoff algorithm [Bre97]. A node determines a backoff delay by choosing a random integer between zero and a value known as the Contention Window (CW), initially set to the minimum CW value as specified by PHY. The backoff value is decremented by one for each idle slot time that passes; when the channel is sensed busy, decrementing ceases. The channel is considered idle if no signal is detected for a DIFS period; only then does the countdown begin. If a node's transmission is not successful, the CW value is doubled until it reaches a maximum CW value as specified by the PHY and a new backoff value is chosen. Once the CW reaches the maximum value, it remains at that value until it is reset due to a successful transmission or the frame is dropped. Figure 3 shows the exponential increase of the CW for each subsequent re-transmission of a frame until the maximum CW value is reached.
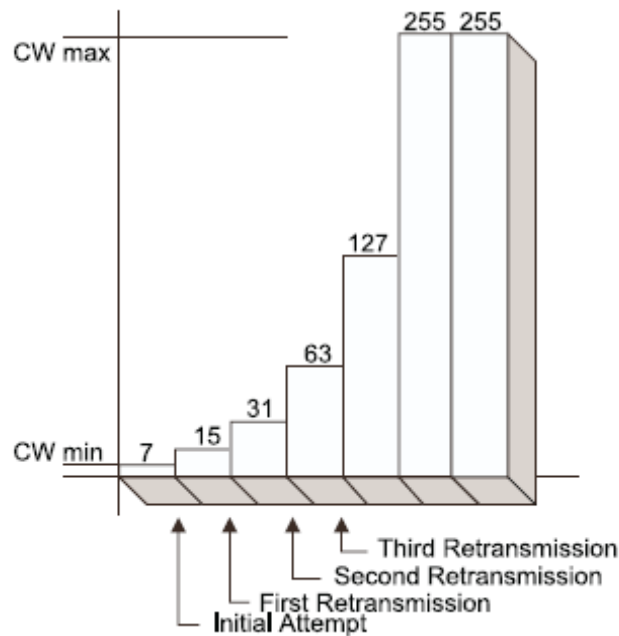
Figure 3. Exponential increase of CW [IEE03a]

Assume a node has received data for transmission from the network layer. The operation of the CSMA/CA protocol is as follows [KuR05]:

- If the channel is idle for a DIFS period, the packet is transmitted

- If the channel is not idle for a DIFS period, a random backoff value is chosen according to the exponential backoff algorithm, and the counter is decremented for each idle slot time that passes (after an idle DIFS)

- Once the counter reaches zero, the packet is transmitted and the node waits for an ACK

- If the ACK is not received, the CW is doubled and the decrement process repeats

Figure 4 outlines the basic CSMA/CA operation, as well as portrays the relationship between the different IFS values.
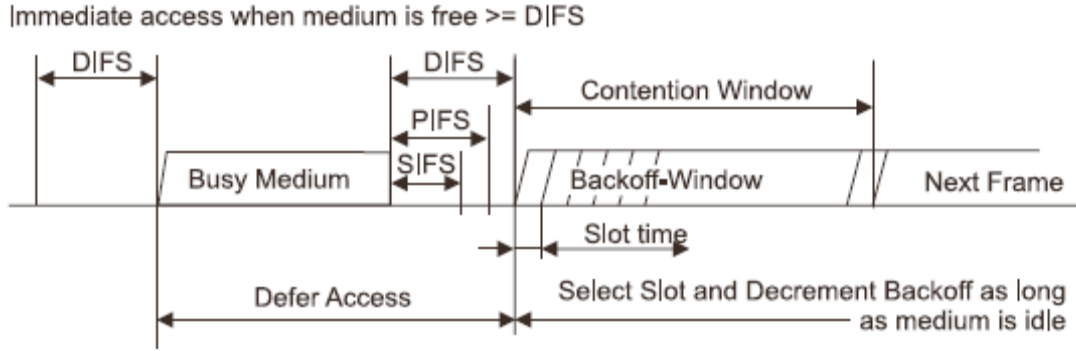
Figure 4. CSMA/CA Operation [IEE03a]

2.3.2.2 Physical Layer (PHY)

The original IEEE 802.11 standard specified three PHY implementations; an infrared (IR) physical layer capable of transmitting and receiving at up to 2 megabits per second (Mbps); a direct sequence spread spectrum (DSSS) physical layer in the 2.4 GHz frequency range capable of up to 2 Mbps; and a frequency-hopping spread spectrum (FHSS) physical layer in the same frequency band capable of up to 4.5 Mbps [IEE03a].

The 802.11b supplement specifies a high-rate extension to the original DSSS specification capable of up to 11 Mbps [IEE03b]. The high-rate DSSS PHY specifies a 20 μs slot time and 10 μs SIFS. Recall from Section 2.3.2.1 that the slot time must be at least as long as the longest one-way signal propagation time; the 20 μs slot time specified in the IEEE standard is sufficient for signal propagation of up to approximately 6,000 meters. In order for the WLAN MAC to operate as intended, the slot time needs to be increased if transmission ranges longer than 6,000 meters are used.

2.3.2.3 Hidden and Exposed Terminals

The CSMA/CA protocol used by 802.11 is not without its weaknesses; consider the scenario portrayed in Figure 5, where the shaded circles represent the transmission range of each the wireless node at its center. When H1 is transmitting to AP, H2 cannot detect the

signal since it is outside the transmission range of H1. Incorrectly assuming the channel is idle, H2 could begin transmission, causing a collision at AP and neither signal will be received without error. This is known as the "hidden terminal problem" [KuR05].
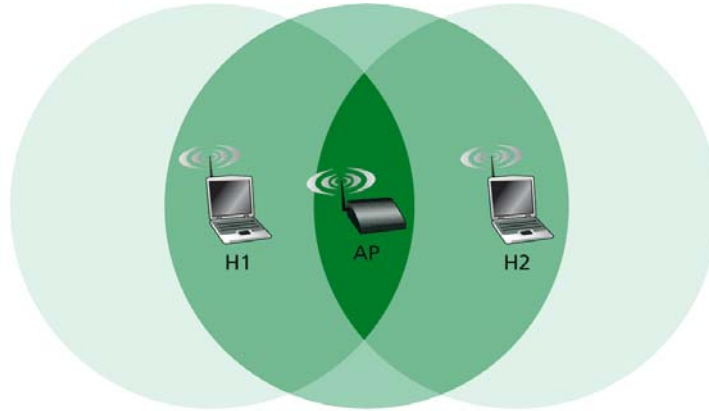
Figure 5. Hidden terminal problem [KuR05]

A similar weakness, known as the "exposed terminal problem," is portrayed in Figure 6. Node S1 is transmitting to node R1, and S2 has a frame to transmit to R2. Since R1 is outside of the range of S2's radio, S2 could successfully transmit without interfering with R1's ability to receive S1's transmission; but since S2 detects S1's transmission, S2 will needlessly defer transmission to R2 until S1's transmission has completed [Wik07a].
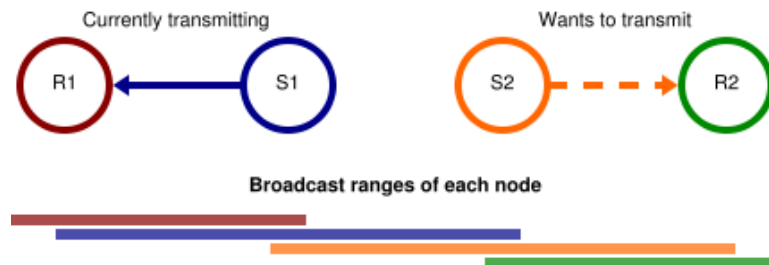
Figure 6. Exposed terminal problem [Wik07a]

2.3.3 Routing Protocols

The Network layer has the responsibility to deliver packets to the destination node across a heterogeneous network. A common mathematical analogy to computer networks uses graph theory to model and analyze the connectivity of nodes [KuR05]; each node in the network is represented by a node in the graph, and an edge connecting two nodes in the graph is added if those two nodes can communicate directly in the network. Though it is not necessarily the case, all communication links are assumed bi-directional for simplicity, so the graph is undirected.

There may be one or more distinct paths via intermediate nodes from the source to the destination, and there may be no path if the source and destination are in disconnected sub-graphs. Algorithms designed to find and store these paths are called routing protocols.

The simplest method used for routing, called Static Routing, uses a table stored in each node that contains the directly connected nodes and destination nodes that the node can route traffic to [KuR05]. Such a scheme would be unwieldy, or even impossible to implement in the dynamic environment of a UAV swarm; thus, a class of routing protocols called Dynamic Routing is required.

Dynamic routing protocols can be divided into table-driven (or proactive) and demand-driven (or reactive) protocols [RoT99]. In table-driven protocols, information necessary for routing packets to other nodes in the network is stored at each node. Typically, the address of the next-hop router is all that is required, though some algorithms store the address of every node along the path to the destination. In highly-dynamic networks, table-driven protocols generate a large amount of control traffic to establish and

maintain routes which may never be needed, decreasing the bandwidth available for useful data transmission.

The Internet primarily uses table-driven protocols, including Routing Information Protocol (RIP) which uses a distance-vector (DV) shortest-path algorithm, and Open Shortest Path First (OSPF) which uses a link-state (LS) shortest-path algorithm. [KuR05] presents an excellent primer on RIP and OSPF and their underlying LS and DV algorithms; some salient details are provided below.

The link-state shortest path algorithm used by OSPF is a variant of Dijkstra's algorithm, named after its inventor [KuR05]. Given a list of all available links in a network, each node employs Dijkstra's algorithm to iteratively compute the shortest path to each node in the network. A routing protocol that employs a link-state algorithm must broadcast the status of all network links (hence the link-state name) to every node in the network upon initialization, and any changes in link status must be similarly broadcast. In a wireless network, each node has many neighbors, and the required link-state messages can become excessive.

Protocols such as RIP which use a distance-vector shortest-path algorithm are based on the classic Bellman-Ford equation [Bel58]. A given node's distance vector is simply a list of the estimated shortest-path distance to every node in the network. Nodes with no known path are considered to have an infinite entry in the distance vector. Rather than flood the state of every link in the network to all nodes, nodes in a network employing a distance-vector protocol simply exchange their distance vector with each of their neighbors. Upon receipt of a new distance vector from a neighbor, nodes update their routing tables using the shortest-path information learned from their neighbors.

In demand-driven protocols, routes are discovered as needed and maintained only as long as necessary. While this eliminates much of the route-maintenance overhead incurred by table-driven protocols, a route-discovery delay is introduced to each session. Given the highly-dynamic nature of a UAV swarm, however, this tradeoff may prove beneficial.

Several routing protocols, both table-driven and demand-driven, are presented below. In most cases, these protocols have been designed or modified specifically for use in MANETs.

2.3.3.1 Destination-Sequenced Distance-Vector Routing (DSDV)

Originally presented by Perkins and Bhagwat [PeB94], DSDV is an adaptation of RIP which uses a modified version of the Bellman-Ford distance vector algorithm suitable for ad hoc networks. Like the RIP protocol it is based on, DSDV is a table-driven routing protocol, in which each node in the network maintains a table with the number of hops to each possible destination node. Table entries also contain a sequence number, set by the destination node, which is used to discard old routes and prevent routing loops [RoT99].

Table updates are either *full dump* or *incremental* mode. Full dump updates transmit the entire routing table, but may not be necessary during a period with relatively few network topology changes. Incremental updates transmit only those routes which have changed since the last full dump.

All updates contain a node-specific sequence number to identify the age of routes terminating at the node that originated the update. Newer routes are always preferred, and when routes have the same sequence number, the shortest path is chosen. Because of the preference for "fresh" routes over route length, route lengths will fluctuate until a route

which is both fresh and optimal (short) is received. This "settling time" is used to set the update frequency to reduce update traffic.

2.3.3.2 Wireless Routing Protocol (WRP)

Another tabled-driven protocol, WRP, was designed to reduce the latency in route discovery incurred by DSDV [MuG96]. WRP also maintains route information at each node in the network, consisting of a distance table, routing table, link-cost table and message retransmission list (MRL) table [RoT99].

The distance table at node i maintains the distance from every neighbor of i to every other node in the system, along with that neighbor's next-hop node to the destination node. The routing table maintains the shortest path to every known destination by recording the distance to that destination, predecessor and successor nodes, and an update marker. The link-cost table lists the path cost for each destination node; nodes out of range are labeled infinity. The MRL contains the sequence number of an update message, a retransmission counter that is decremented each time a new update is sent, an *ack-required* flag for each neighbor which records whether the neighbor has acknowledged the update message, and a list of the actual updates sent during that update. The MRL keeps track of updates that need to be re-transmitted due to transmission errors.

Updates are sent when a node detects a change in link status, and when processing updates from its neighbors. If a node does not send any messages for a specified *HelloInterval*, the node will send a *hello* message. When a *hello* message is received, the sending node is added to the receiving node's routing table and a copy of the routing table is returned to the sender.

Update messages can include acknowledgements to previous updates as well as new updates, in addition to a list of nodes which should acknowledge the current update. This response list, in conjunction with the MRL of the update's sender, eliminates multiple acknowledgements to the same update. Distance table information is updated using the information contained in the update message, and the new distance table is examined for any changes to the routing table. The path-finding algorithm, described in detail in [MuG96], is a modified distance-vector algorithm which uses the predecessor and successor information from the routing table to eliminate routing loops.

Due to the update scheme used by WSR, the percentage of network traffic consumed by updates after a link failure is lower than protocols such as DSDV which transmit the entire routing table with each update. It is still, however, a table-based protocol which inherently maintains information on paths which may never be needed.

2.3.3.3 Optimized Link-State Routing (OLSR)

Developed by the French National Institute for Research in Computer Science and Control (INRIA) and originally published in [JMC01], OLSR is a table-driven proactive routing protocol. Like OSPF, OLSR is built around a link-state shortest-path routing algorithm, but was designed specifically for use in mobile ad hoc wireless networks. An experimental specification for OLSR has been published by the Internet Engineering Task Force as RFC 3626 [ClJ03]. OLSR optimizes the link-state algorithm for the wireless environment by reducing the size of control packets and minimizing the flooding of broadcast messages by using multipoint relay nodes.

In a multipoint relay (MPR) scheme, only nodes designated as multipoint relays for a transmitting node retransmit that node's broadcast messages [JMC01]. An example of

normal flooding and MPR flooding is shown in Figure 7. On the left, the original message broadcast by the node in the center is retransmitted 24 times to reach all three-hop neighbors. In the MPR flooding example on the right, (MPR nodes are shown in black), only 12 retransmissions are needed to reach all three-hop neighbors.
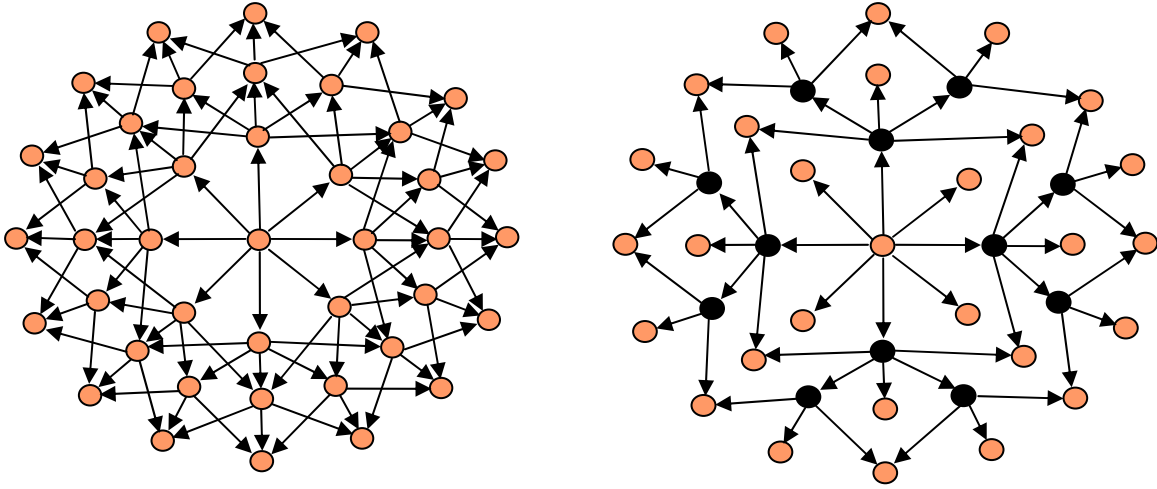
Figure 7. MPR Flooding Example [Ton06]

The OLSR specification defines four major protocol functions: neighbor sensing, multipoint relay selection, multipoint relay information declaration, and routing table calculation.

The neighbor sensing function defines how each node detects neighbors it can communicate with by having every node broadcast periodic *HELLO* messages which are received by each node's one-hop neighbors and are not rebroadcast. Each *HELLO* message contains a list of nodes the sending node has a bi-directional link with, and a list of nodes the sending node has received a *HELLO* message from. Links are annotated uni-directional, bi-directional, or multi-point relay (MPR). MPR links indicate which neighbors the sending node has chosen as its multipoint relay nodes. When a node itself is listed in a

HELLO message, it records the link between itself and the sending node as bi-directional. Each node uses these *HELLO* messages to construct a table containing information on all one-hop and two-hop neighbors they can reach.

The MPR selection function defines how nodes select the subset of nodes from their list of one-hop neighbors which become the multipoint relay set for that node. It is important to note that the only requirement for the MPR set of any given node is that all two-hop neighbors of that node are reachable through an MPR. The precise method by which the MPR set is determined is an open research item; a proposed heuristic is given in [ClJ03].

The MPR information declaration function uses MPR flooding to broadcast *Topology Control (TC)* messages throughout the entire network announcing the MPR Selector set for each node. A given node's MPR selector set is the set of neighbors which have chosen that node as an MPR. *TC* messages are used by each node to generate a network topology table consisting of the address of a potential destination (an MPR selector from the *TC* message) and the address of that node's MPR (the sender of that particular *TC* message). It is assumed that data for the potential destination node can be sent to the MPR and will be re-broadcast to the destination node.

Once each node constructs a topology table, the routing table calculation function uses a shortest-path algorithm similar to Dijkstra's algorithm to develop a next-hop node for all potential destinations in the network. This next-hop table is used by IP to forward data packets as necessary.

Through multipoint relay forwarding, OLSR requires significantly less control traffic overhead than its non-optimized OSPF predecessor, though as it is a proactive routing

protocol, many routes are calculated and maintained which may never be used. In addition, the frequent network topology changes due to the high mobility of a UAV swarm requires constant adjusting of the routing tables to adapt to topology changes, as with any proactive protocol.

2.3.3.4 Dynamic Source Routing (DSR)

Originally developed by Johnson and Maltz in [JoM96], the Internet Engineering Task Force (IETF) MANET Working Group is currently developing a standardized design for DSR, a demand-driven protocol. The latest draft was released in July 2004 [BJM04]. Royer and Toh present an excellent summary of and analysis of DSR in [RoT99].

In DSR, mobile nodes maintain a cache of known routes that expire after a specified time. The two major phases of DSR are route discovery and route maintenance. When a node has a packet to send to some destination node *j,* it checks the route cache to see if an unexpired route is already known. If so, the packet is transmitted along the route; if not, the node enters route discovery.

In route discovery, the source node broadcasts a *route request* packet consisting of the ID of the source and destination nodes as well as a unique identification number for the route request. Each node that receives the route request checks its cache for a valid route; if no valid route is present, it appends its own ID to the route record of the route request packet and re-broadcasts the request. The route record in each request packet contains a list of nodes on a path back to the originating node. The intermediate node will only re-transmit the request packet if it has not done so already, based on the unique identification number and the presence of its ID in the route record.

When the route request packet reaches either the destination node, or an intermediate node which has a valid route to the destination, the complete route is forwarded back to the originating node in a route reply packet, using the route record to find a path back to the source. Each intermediate node also records in its cache the route from itself to the destination using the data in the route record.
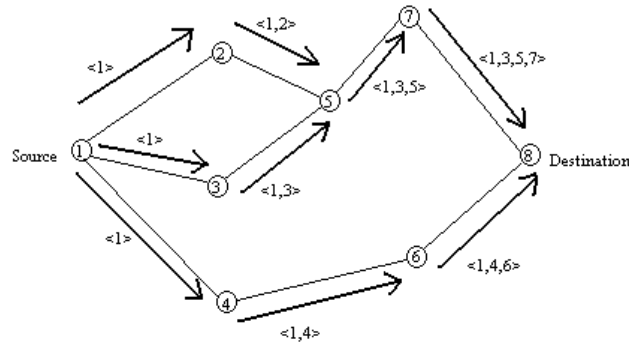
(a) Building Record Route during Route Discovery
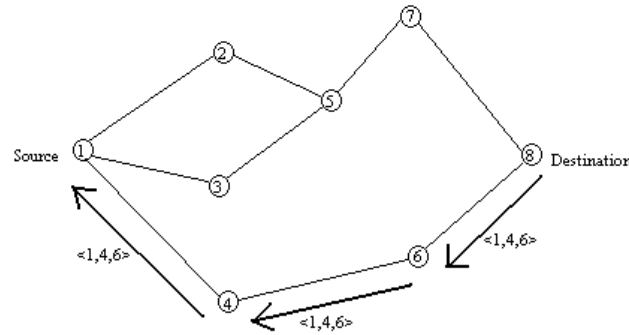
(b) Propogation of Route Reply with the Route Record

Figure 8. Creation of the route record in DSR [Mis99]

Figure 8 shows the construction of the route record in panel a, and its return to the requesting node in panel b. The source node floods a route request packet to each of its neighbors, which in turn append their node ID to the request, and forward the packet to their neighbors. Node 5 arbitrarily chooses one route request to forward, since they both have the same path length to the source. The destination receives two route request packets;

one from node 7, and one from node 6. Since the request form node 6 has a shorter path to the source, the request from node 7 is discarded, and a route reply packet is sent back to the source by reversing the path <1, 4, 6> stored in the route request packet.

If a node detects a packet transmission failure, typically as a notification from the data link layer that a link-level acknowledgement was not received, the node enters a route maintenance phase. The purpose of route maintenance is to notify the source of the data that the particular link has failed. The node deletes all cached paths that use the broken link, then sends a route error packet containing the ID of the source and destination nodes on that link to the originating node of the failed data packet using the route information from the packet in error. Upon receiving an error packet from a neighboring node, each intermediate node deletes any cached routes which contain this hop, and forwards the packet one hop towards the source along the original path. Upon receiving an error packet, the source node will re-initiate route discovery if the data needs to be re-transmitted.

While the DSR protocol does not create any network traffic to maintain routes that are not used, it does increase overhead by requiring that entire route path be stored in each data and route reply packet. On the other hand, nodes using DSR can maintain multiple routes to a given destination; if one route fails, the data can be re-transmitted using an alternate unexpired route.

2.3.3.5 Ad Hoc On-demand Distance Vector Routing (AODV)

Like DSDV, the AODV protocol uses sequence numbers to prevent routing loops. Originally presented by Perkins (one of the creators of DSDV) and Royer in [PeR99], the protocol is also very similar to DSR [RoT99].

23

AODV has path discovery and path maintenance phases, roughly analogous to route discovery and route maintenance phases in DSDV. Path discovery follows the same mechanics as route discovery in DSDV; if a source has no route to a destination, it broadcasts a route request (RREQ) packet containing the source ID and sequence number, destination ID and sequence number, broadcast ID and hop count. The broadcast ID is incremented each time the node initiates an RREQ, the hop count begins at zero and is incremented at each intermediate node along the path to the destination.

Each intermediate node identifies an RREQ packet by the pair <source ID, broadcast ID> to avoid acting on the same RREQ more than once, even if the RREQ is received multiple times. If the node cannot satisfy the RREQ, it keeps track of the destination ID, source ID, broadcast ID, reverse path expiration and source sequence number to use to return the route reply (RREP) and set up the forward path once a route is determined. Each node need only remember the first hop towards the RREQ originator for the reverse path; the source sequence number is used to determine how fresh the path back to the source is.

An RREP is generated when the RREQ reaches the destination or an intermediate node that has a route to the destination with a higher destination sequence number than the RREQ. The RREP is forwarded back to the RREQ originator using the reverse path setup during the forwarding of the RREQ. The RREP contains source ID, destination ID, destination sequence number, hop count and lifetime fields. As the RREP is forwarded back to the source, each node along the path sets up a forward pointer to forward any packets destined for the destination along the correct path. Destination sequence numbers are used to update paths if multiple RREP packets are received.

If a link failure is detected by the link layer, a link failure notification message is propagated upstream to the source node, indicating to all nodes along the way to delete the route. The source node can re-initiate path discovery if a route is still required to the destination.

Since each packet does not contain the entire route, but only the destination address, per-packet overhead is smaller than DSR, though as with most demand-driven routing protocols, path discovery causes some latency at the start of each data session. Unlike DSR, AODV can support multicast operations, and the absence of periodic updates reduces overall routing overhead.

2.3.3.6 Greedy Perimeter Stateless Routing (GPSR)

An entirely different class of routing protocols is location-based routing protocols. In location-based routing, forwarding decisions are based on the relative location of the destination rather than a topology-based route. Since there is no need for the network to maintain route information, location-based routing protocols scale well even in highly mobile networks [MWH01]. Such a routing scheme would be useful in a UAV swarm (as described in Section 2.2.2) to deliver data to a location that is known to have connectivity to the network edge for transmission, for example, to a ground station. A survey of several location-based routing protocols is presented by Mauve, Widmer and Hartenstein in [MWH01].

In some mobile networks, location-based routing is difficult if nodes do not know the geographic location of all other nodes in the network. Several methods to address this issue have been developed, including a location service that resolves addresses to locations [MWH01][KaK00], relative location determination based on beacon signals [RRP03], and a

method to learn locations over time [JPS01]. In a UAV swarm, however, every node knows its location via GPS; by including source and destination (when known) locations in the packet, the locations of specific UAVs can be learned over time. Location records at each node can be time stamped and discarded after an expiration period. Furthermore, in many instances data may not necessarily be destined for a specific node, but rather to any node located at a specific location.

Once such location-based routing protocol is the Greedy Perimeter Stateless Routing protocol (GPSR) [KaK00]. In greedy forwarding algorithms, a packet is forwarded to the neighboring node geographically closest to the destination. For example, in Figure 9 node y is node x's neighbor closest to D. The dotted line represents x's transmission range, and the dashed line is on the circle centered at D with radius equal to the distance from y to D. Any node inside the intersection of the two circles is a neighbor of x closer to D than y. Successive greedy forwarding hops are made until the packet reaches the destination.
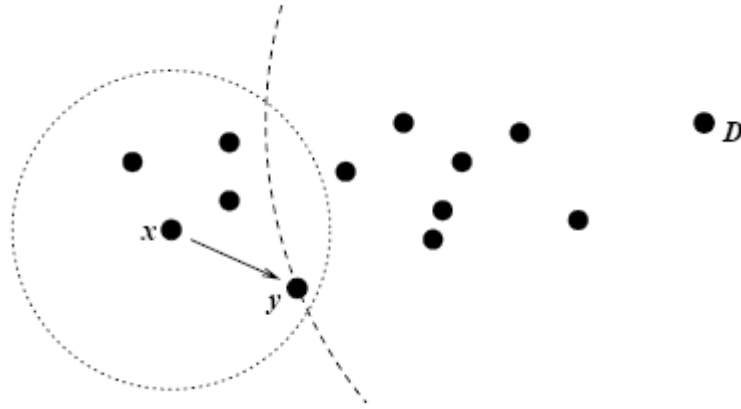
Figure 9. Greedy forwarding example [KaK00]

Greedy forwarding fails, however, when none of *x*'s neighbors are closer to *D* than *x*, as seen in Figure 10. When this "dead end" is reached, the packet must be forwarded to a node further away from the destination until a node closer to *D* is reachable.
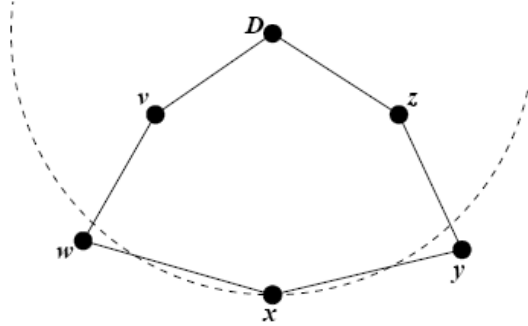
Figure 10. Greedy forwarding failure [KaK00]

Figure 10 clearly shows a path from *x* to *D*; GPSR includes an algorithm to find and exploit this path called perimeter routing. In perimeter routing, each GPSR node maintains a planar graph representation of all nodes within its transmission range. In a planar graph, no edges cross; however, a graph representation of a mobile wireless network certainly has crossing edges, so GPSR uses an algorithm that reduces the full network graph to a planar graph such that the graph is not disconnected during the reduction. Two well known algorithms for creating such planarized graphs, the Relative Neighborhood Graph (RNG) [Tou80] and the Gabriel Graph (GG) [GaS69], satisfy this connectedness property. While GPSR implementations may use any graph planarization algorithm, [KaK00] uses RNG.

Construction of the RNG is depicted in Figure 11. Starting with the full network graph, every edge is considered for removal. For an edge (*u,v*) to be included in the RNG, the shaded area must not contain any other node *w*. If such a node appears in the shaded

area, the edge is removed from the RNG. Note that the deletion of edge (*u,v*) does not disconnect the graph because the path is replaced by the two shorter edges (*u,w*) and (*w,v*).
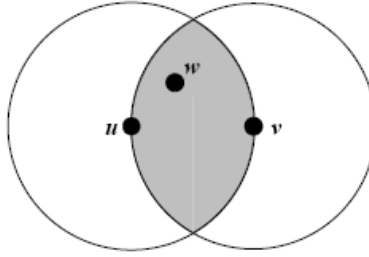
Figure 11. Constructing the RNG [KaK00]

Once a planar representation of the network is achieved, a node which has a packet to forward but does not have any neighbors closer to the destination begins perimeter routing. In perimeter routing, the node marks the packet for perimeter mode including setting the perimeter location (Lp) field in the packet to the location where the packet entered perimeter routing. The node then forwards the packet around the perimeter of the RNG using the right-hand rule. The right-hand rule forwards the packet along the first edge encountered by sweeping counter-clockwise around the node from the incoming edge the packet was received on. An example of perimeter forwarding is shown in Figure 12. If the edge selected by the right-hand rule for forwarding crosses the line between Lp and D, the node updates Lp to that intersection point and continues perimeter-mode routing, using the right-hand rule starting from the line between Lp and itself. If a node receives a packet in perimeter mode, but determines that it is closer to the destination than the location where the packet entered perimeter mode, the packet is removed from perimeter mode and greedy forwarding resumes.
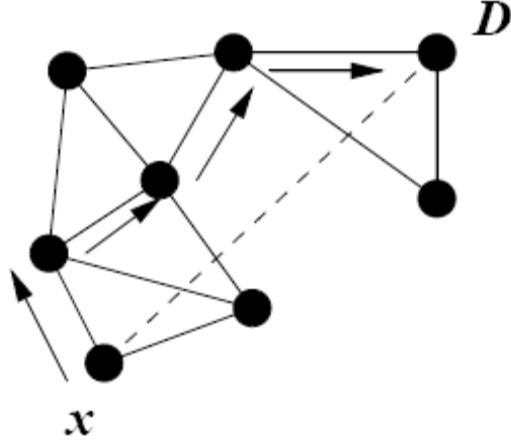
Figure 12. Perimeter forwarding example [KaK00]

Note that each node need only determine the RNG which includes those nodes within its transmission range. To do so, it needs to know the existence and location of each of its neighbors. Including location information in every transmitted packet and broadcasting beacon packets containing location information in the absence of data transmissions ensures that nodes in the network have the required information about their neighbors to form the RNG.

To optimize the protocol's neighbor maintenance function, failed data packet transmissions can be used as evidence that a neighbor has moved beyond range. While this requires direct feedback from the MAC layer, it will prevent GPSR from forwarding additional packets to an unreachable node earlier than that node's entry in the neighbor table would naturally expire. This optimization is described as having a "profound impact" on packet delivery by the protocol's creators.

A variation of GPSR which uses a hybrid of greedy and face routing is presented in [LGS04]. Another version which limits power usage can be found in [YGE01], and has

interesting possibilities for implementation in a swarm which uses very lightweight or battery-powered UAVs.

2.3.4   Mobility Models

When evaluating a routing protocol for a MANET, the environment in which a routing protocol is evaluated can have as much impact on its performance as the choice of protocol itself [CBD02]. One environmental factor of significance is the movement of the nodes in the network, often defined by a mobility model. One type of mobility model, called a trace, is a recorded history of mobility as observed in an actual system. Without any currently operating UAV swarms from which to record mobility, this research instead uses a synthetic model which simulates the behavior of mobile nodes mathematically.

Camp, Boleng and Davies define two classes of mobility models: entity and group mobility models [CBD02]. In an entity mobility model, all nodes move independently of one another; in a group mobility model, the movement of each node is dependent on the other mobile nodes in the group.

While there is no limit to the number and types of mobility models which could be devised, those that are most relevant to swarms of UAVs are presented here. In all of the models presented, there is a simulation boundary which encloses all nodes in the system and represents the operating boundary of the UAV swarm.

2.3.4.1   Random Walk

In the Random Walk Mobility Model, sometimes referred to as Brownian Motion, nodes travel from their current location by choosing a random speed and direction within a predefined range, and traveling either for a constant amount of time, or for a constant distance [CBD02]. If the node reaches the simulation boundary, it changes direction at the

30

boundary at an angle equal to the angle at which it approached, much like a laser beam bouncing off of a mirror.

The Random Walk model can be applied to a UAV swarm by limiting the range of speeds to a realistic range around the efficient cruising speed of the UAV. Limiting the direction to a realistic range around the current direction of travel and 90° in each direction lends a degree of aerodynamic possibility to the model, as it would prevent nodes from reversing direction instantaneously.

2.3.4.2 Random Waypoint

In the Random Waypoint Mobility Model, a mobile node chooses a random destination within the simulation boundary and travels to that point [CBD02]. Once the destination is reached, a new destination is chosen and the node departs for the new destination after pausing for a random period. As in the Random Walk model, speed is also chosen at random from a specified range for each waypoint.

Since the probability of choosing a destination near the center of the simulation space, or a destination which causes the node to travel through the center of the simulation space, is high, nodes tend to pass through the middle of the simulation space with higher frequency than the edges, as seen in Figure 13 [CBD02]. A cooperative application using a swarm of UAVs might prefer a more uniform distribution of nodes if uniform coverage of the operational space is desired.
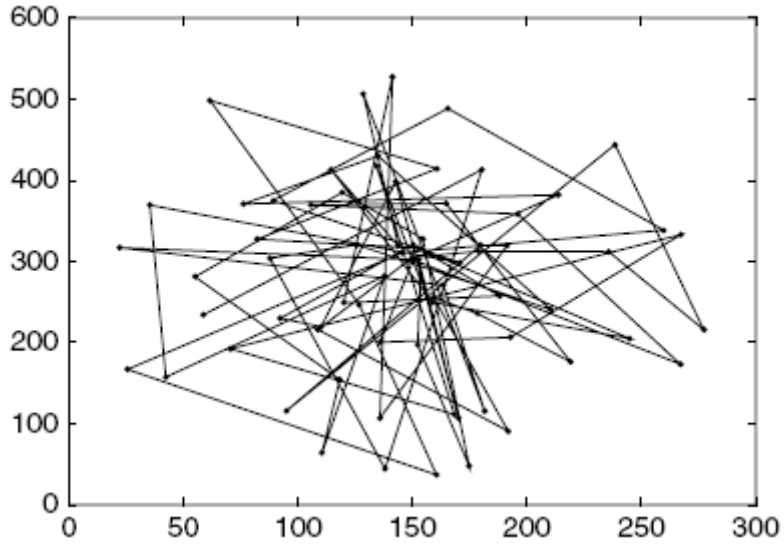
Figure 13. Traveling pattern of a node using the Random Walk model [CBD02]

Since a UAV cannot "pause" at its destination, a zero pause time is used to simulate a UAV swarm. As in the Random Walk model, speed of travel is chosen from a realistic range of speeds for the UAV under consideration.

2.3.4.3 Random Direction

To mitigate the clustering of nodes at the center of the simulation space when using the Random Waypoint model, the Random Direction Mobility Model has each node choose a direction at random and travel to the simulation boundary at a random speed [CBD02]. Once the boundary is reached, it chooses a new direction at random and travels to the next boundary. This model provides a more uniform distribution of nodes throughout the simulation space, but can also deliver sharp, sudden turns at the boundaries, as seen in Figure 14, which is unrealistic in the context of a UAV swarm.
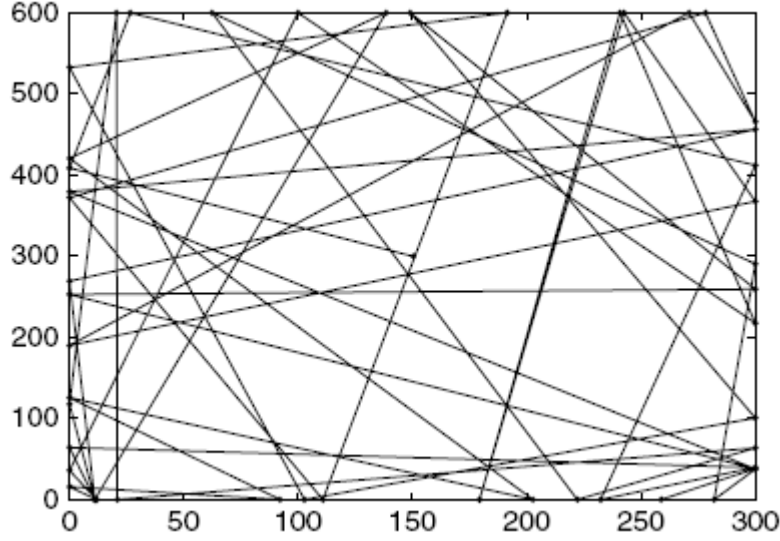
Figure 14. Traveling pattern of a node using the Random Direction model [CBD02]

2.3.4.4 Gauss-Markov

In the Gauss-Markov Mobility Model, the speed and direction of a node is calculated from the current speed and direction and from a random number [CBD02]. The name comes from the distribution from which the random number is chosen (Gaussian) and from the fact that the next "state" of the node is dependent only on the characteristics of the current state, as in a Markov process. The speed and direction are calculated using:

$$s_n = \alpha s_{n-1} + (1-\alpha)\bar{s} + \sqrt{(1-\alpha^2)s_{x_{n-1}}} \tag{1}$$

$$d_n = \alpha d_{n-1} + (1-\alpha)\bar{d} + \sqrt{(1-\alpha^2)d_{x_{n-1}}} \tag{2}$$

where s_n and d_n are the new speed and direction of the node, $\bar{s}$ and $\bar{d}$ are the mean value of the speed and direction over time, $s_{x_{n-1}}$ and $d_{x_{n-1}}$ are random variables from a Gaussian distribution, and α is a "tuning" parameter used to vary the randomness. Note that by setting α to 0 we get totally random, or "Brownian" motion and with $\alpha = 1$ the speed and direction never change.

33
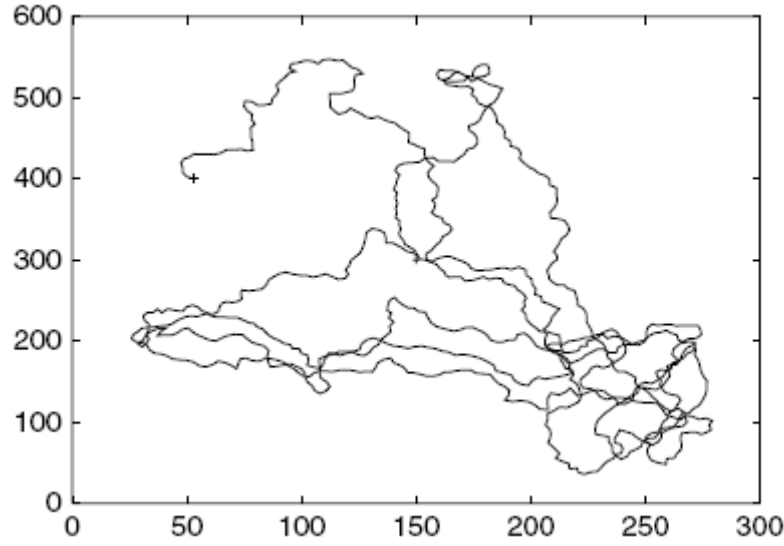
Figure 15. Traveling pattern of a node using the Gauss-Markov model [CBD02]

The Gauss-Markov model is quite appealing for an application modeling aircraft, unmanned or otherwise, as it more accurately models an airborne vehicles tendency to travel in a straight direction at a constant speed. As seen in Figure 15, the Gauss-Markov model exhibits no sudden changes in speed or direction.

2.3.4.5 Pursue Mobility Model

The only group mobility model presented here, the Pursue Mobility Model, simulates a group of mobile nodes following a mobile target [CBD02]. In the pursue model, the next position of each mobile node is set by:

$$pos_{new} = pos_{old} + acceleration(target - pos_{old}) + rand\_vector \qquad (3)$$

where pos_{new} and pos_{old} are the new and old position, respectively, of the mobile node; $acceleration(target - pos_{old})$ is a function of the distance to the target which is used to impart physical feasibility of the movement; and $rand\_vector$ simply imparts some randomness to the motion.

The pursue model, with appropriately selected parameters for the acceleration and random functions, can be used to effectively simulate a subset of the UAV swarm tracking a mobile ground target. York, Pack and Harder propose such a mechanism in [YPH06] where three UAVs track a mobile target in a circle formation around the target's estimated location.

2.4   Related Research

There is seemingly no limit to the amount of research in mobile, ad hoc routing protocols. Broch, et al. performed a simulation comparison of the DSDV, DSR, AODV and Temporally-Ordered Routing Algorithm (TORA) routing protocols using the open source discrete event simulator *ns*, developed by the University of California at Berkeley and the Virtual InterNetwork Testbed (VINT) project [BMJ98]. While this effort produced recommendations on the appropriateness of the three protocols evaluated, the mobility model (random waypoint with pause times of up to 900 seconds) and node traveling speed (maximum of 20 m/s) used in simulation do not map well to the dynamics of a UAV swarm.

Another DoD-sponsored research effort studied the Hierarchical State Routing (HSR) protocol [GPL00] and its variations [GGL01] to implement a multi-level network which uses UAVs as mobile routers to enhance connectivity and provide routing between mobile base stations on the ground. It leveraged UAVs to enable better network connectivity for ground-based ad hoc networks, and not on communication between UAVs in a swarm.

In [LAN03] and [MMP06], cooperative efforts of UAV swarms were studied and communication was assumed to be broadcast, not routed. It was accepted that not all nodes in the swarm would receive the transmission.

A 2002 research effort at the Naval Postgraduate School [Bla02] began with many of the same goals of this research: to evaluate the performance of various ad hoc routing protocols when implemented in a UAV swarm. Blackshear found, however, that the protocol models, as implemented, were insufficient at the time to adequately simulate the UAV environment.

2.5 Summary

In this chapter, background on swarming UAVs and their applications was presented. Next, Mobile Ad hoc Networks and some common ad hoc routing protocols were described. Mobility models for simulating MANET node movement and their appropriateness for simulating a UAV swarm was also presented. Finally, related research efforts were described.

III. Methodology

3.1 Problem Definition

3.1.1 Goals and Hypothesis

One way to transmit data through a swarm of mobile nodes is to simply increase the transmission power at every node so the data can be received directly by every other node in the network. Aside from the issues of limited power, this scheme precludes the exploitation of "spatial multiplexing," whereby many pairs of nodes simultaneously communicate using the same transmission channel since the distance between each of the pairs is large enough to prevent interference.

The goals of this research are to:

- Determine whether multi-hop routing with reduced transmission range increases throughput compared to a no-hop broadcast scheme, and to

- Evaluate and compare several ad hoc routing protocols

Although a shorter transmission range requires multiple transmissions of the same data packet to relay it from source to destination, it is hypothesized that the corresponding increase in aggregate throughput due to spatial multiplexing will more than make up for the loss due to retransmissions.

As transmission delay is typically the dominant factor in end-to-end packet delay, retransmitting packets several times before reaching the intended destination will necessarily increase the time from packet origination to delivery. This increase is the cost for achieving higher aggregate network throughput, and it is hypothesized that an increase in network throughput can be achieved with an acceptable increase in end-to-end delay.

3.1.2 Approach

Several methods of transmitting data through the network are analyzed and compared with the baseline (no relay or routing mechanism) with an emphasis on those methods which exploit spatial multiplexing. By exploiting simultaneous transmission and reception on the same wireless channel in several different areas of the swarm, the aggregate network capacity is increased without changing the underlying transmission scheme. By definition, multi-hop routing protocols facilitate the routing of packets beyond the transmission range of a single node. Not all routing protocols are appropriate for use in this highly-mobile environment, however. Proactive protocols which maintain a table of routes to all network destinations require significant overhead in the form of control and update packets. In addition, the dynamic network topology resulting from node mobility causes these routes to expire on a regular basis, causing more control and update packets to be sent. Reactive routing protocols designed specifically for dynamic network topologies are more appropriate for this environment; protocols in both classes are performance tested in the environment under consideration.

3.2 System Boundaries

The System Under Test (SUT) is defined as the UAV Swarm Data Routing System, and consists of the Routing Protocol, (Intra-Swarm) Wireless Network, UAVs and (Inter-Swarm) External Network.

The environment is assumed to present no obstacles to UAV travel or communication. Furthermore, mechanisms to avoid mid-air collisions are outside the scope of this effort; multiple nodes are allowed to occupy and pass through the same space.

Finally, the UAVs are assumed to have an infinite amount of fuel and can fly for the duration of the simulation without refueling.
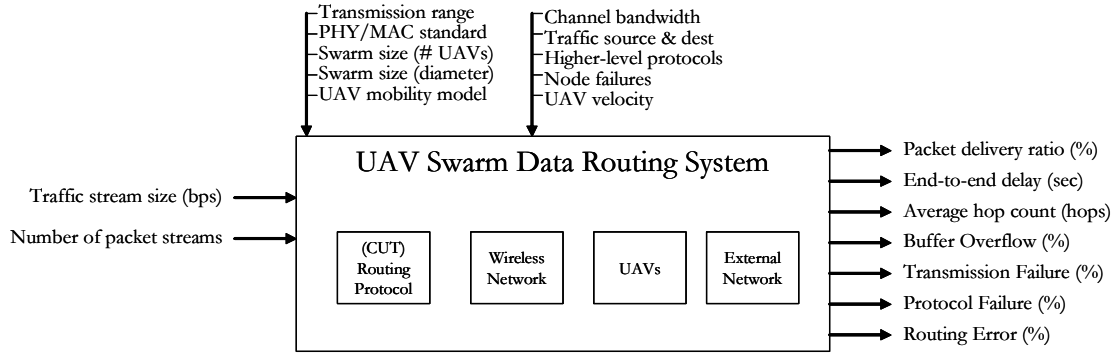
Figure 16. UAV Swarm Data Routing System

3.3 System Services

The UAV Swarm Data Routing System provides a Packet Delivery Service which transports a data packet from a source node to a destination node. Potential service outcomes are:

- Success: The packet is successfully relayed (if necessary) through the swarm and delivered to the destination node.

- Destination unreachable: The destination node is unreachable, either due to the failure of the destination node or a network partition.

- Delivered with errors: The packet is delivered to the destination, but one or more errors are detected in the data.

- Dropped packet: The packet was dropped at an intermediate node between the source and destination.

3.4   Workload

The workload of the system is comprised of a number of traffic streams flowing through the system. The underlying physical layer network standard determines the absolute maximum rate at which data can be transmitted from node to node; channel contention, control frames and bit errors reduce useful throughput.

All workloads consist of eight generating nodes transmitting packets to a randomly chosen destination node. Destination nodes are randomly chosen at simulation start, and each generating node transmits to the same destination node for the duration of the simulation.   Packet sizes are exponentially distributed with a 1024-byte mean and exponentially-distributed inter-arrival times which are varied to simulate different traffic loads. Five mean inter-arrival times are used to present five workload levels to the system:

- Minimal (25% of baseline): 0.5-second inter-arrival time (total 128 kbps),

- Low (50% of baseline): 0.25-second inter-arrival time (total 256 kbps),

- Medium (baseline): 0.125-second inter-arrival time (total 512 kbps),

- High (125% of baseline): 0.1-second inter-arrival time (total 640 kbps),

- Overload (250% of baseline): 0.05-second inter-arrival time (total 1280 kbps).

3.5   Performance Metrics

The performance of a particular ad hoc routing protocol under each load is evaluated on the basis of the following metrics:

- Packet Delivery Ratio: Defined as the percentage of all generated packets which are successfully delivered to their intended destination, packet delivery ratio (PDR) is measured by taking the sum of all packets successfully delivered to their intended destinations and dividing by the sum of all generated packets.

40

- End-to-end delay: This is the per-packet average of the time from origination at the source node to delivery at the destination node; a lower delay is better.

- Average hop count: The number of transmission hops taken by each successfully delivered packet is counted at destination nodes and the average is calculated over all packets delivered; discarded packets are not counted. In conjunction with the transmission range and average distance between source and destination nodes, this metric measures the efficiency of the routing protocol's underlying path selection algorithm.

- Failure Mode: The percentage of generated packets dropped by each of four failure modes, these metrics are measured by dividing the sum of all packets dropped by each failure mode by the total number of packets generated

    o Buffer Overflow: If the network layer fills the MAC buffer with packets for transmission faster than the MAC successfully transmits them into the channel, additional packets from the network layer are dropped.

    o Transmission Failure: If the MAC is unable to successfully transmit a frame, the frame is dropped.

    o Protocol Failure: When a routing protocol fails to find a route to the destination, the packet is dropped by the routing protocol. Protocol failures can occur when there is no path to the destination (e.g., network partition) or when there is a path which the routing algorithm fails to discover.

    o Routing Failure: Packets have been processed by the routing protocol but fail to reach their destination and are discarded by IP. Routing failures are different than protocol failures. Packets discarded due to a routing failure

have been processed by the routing protocol but an incorrect routing decision was made by the protocol.

3.6 Parameters

Parameters are the characteristics of the system under test which affect system performance when varied. Parameters are divided into system parameters and workload parameters. Workload parameters are the characteristics of user requests to the system, such as packet arrivals; all other parameters are considered system parameters

3.6.1 System

- Transmission range: The maximum distance over which two nodes can successfully communicate directly. Nodes separated by at least twice this distance can transmit simultaneously without collision.

- PHY/MAC standard: IEEE 802.11, IEEE 802.16 and HIPERLAN are but a few of the myriad possible standards which define channel access and data encoding, modulation and transmission. Since it is the most widely used technology in similar studies, IEEE 802.11b is chosen to facilitate comparison with similar research efforts.

- Swarm size (# of UAVs): The number of UAVs in the swarm can impact the number of traffic flows, in addition to the number of possible paths through which to route packets across the swarm.

- Swarm size (diameter): For a swarm of a given number of UAVs, an increase in swarm diameter increases the possibility of a network partition; decreasing the swarm diameter, conversely, decreases the potential for a partition. A reasonable area given

the expected mission for a HARVEST, the swarm boundary is fixed to a 10 kilometer by 10 kilometer square.

- UAV mobility model: The mobility of each UAV changes the topology of the network and the available paths upon which to route packets. A highly-dynamic environment places more demands on the routing protocol than one in which UAVs loiter in the same general area for long periods of time. This study uses a random waypoint mobility model.

- UAV velocity: In conjunction with the mobility model, UAV velocity impacts how fast and to what degree the network topology changes. A reasonable velocity given the expected size and maneuverability of a typical UAV in HARVEST is 25 meters per second.

- Channel bandwidth: Bandwidth is constrained by the physical-layer standard selected; changing the channel modulation rate with fixed transmission power varies the effective range and bit error rate. Channel bandwidth for this study is the maximum bandwidth for the 802.11b standard, 11 Mbps.

- Higher-level protocols: Choice of higher-layer protocol impacts the amount of overhead, in the form of headers, retransmissions and acknowledgements. This study does not model protocols above the network layer and directly specifies network-layer (IP) packet payload length to simulate data.

- Node failures: The failure of a node can cause network partitions or otherwise undelivered packets. No node failures are simulated in this experiment; it is assumed that some mechanism exists to quickly and automatically replace failed nodes.

3.6.2 Workload

- <u>Traffic stream size (kbps):</u> The size of each traffic stream flowing through the network; values used in this study are specified in paragraph 3.4 above.

- <u>Number of packet streams:</u> The number of traffic streams currently flowing through the network; in this experiment there are eight traffic streams.

- <u>Stream source & destination:</u> Source and destination nodes are necessarily different for each traffic stream (nodes do not generate packet addressed to themselves). Eight traffic generating sources each choose one random destination node for the duration of the experiment.

Table 1 specifies the settings used for all fixed parameter values.

Table 1. Fixed parameter values

Parameter	Value
PHY/MAC Standard	IEEE 802.11b
Swarm diameter	10km x 10km
Mobility model	Random waypoint
UAV velocity	25 meters/sec
Channel bandwidth	11 Mbit/sec
Higher-layer protocol	N/A
Node failures	none

3.7 Factors

- <u>Routing protocol:</u> AODV and OLSR are widely-used ad hoc routing protocols, and validated simulation models are available for the OPNET Modeler. GPSR exploits geographic routing, but the OPNET model needs to be built from scratch; GPSR is tested with (GPSR) and without (Greedy) perimeter-mode routing enabled. A baseline scenario with no routing is also simulated.

  o Levels: AODV, OLSR, Greedy, GPSR, None

- Transmission range: Changes in the transmission range vary the potential for spatial multiplexing and the necessity (and hop count) of multi-hop routing. Note that while this is a discussion about varying transmission range, the physical property which is actually varied to achieve different transmission ranges is transmit power; see Appendix A for a discussion of how transmission range relates to transmit power. For a network to successfully route data between any source/destination pair, there must be a path between all pairs of nodes in the network, i.e., the network must not be partitioned. If each node in the network has at least 5.1774 log n neighbors (where n is the total number of nodes in the network) the network is connected with high probability [XuK04]. In a less-connected network there is a high probability of partition, and in a more-connected network, contention for the medium among more nodes decreases total throughput. This result is used to approximate the optimal transmission range in a network of uniformly located nodes which ensures a low probability of network partition while minimizing contention among nodes as

$$\frac{\pi r^2}{A} \approx \frac{x}{n} \tag{4}$$

where r is the transmission range in meters, A is the total network area in square meters, x is the desired number of connected neighbors, and n is the total number of nodes in the network. With a uniformly distributed network, the ratio of reachable transmission space to the entire network area is proportional to the ratio of the number of immediate neighbors to the total number of nodes. Substituting the minimum number of neighbors from (4) and solving for r we have

$$r \approx \sqrt{\frac{xA}{\pi n}} = \sqrt{\frac{5.1774 A \log n}{\pi n}} \qquad (5)$$

The approximation holds, on average, for nodes at least r meters away from the network boundary; nodes within r meters of the boundary have some part of their transmission area outside the network boundary. Since there is zero probability of any nodes existing outside the boundary, the average number of neighbors for these nodes is decreased.

To validate the model and refine the approximation, a simulation is performed using a simple Java program to generate random node configurations consisting of various numbers of nodes and transmission ranges in a 10,000 meter by 10,000 meter network. 25 replications of each configuration are generated with each replication having a unique set of randomly located nodes. Several metrics are calculated including average node degree and the number of nodes in the largest connected set. A fully-connected network with no partitions has a connected set equal to the total number of nodes in the network.

Average node degree is computed by counting the number of neighbors each node has and taking the average across all nodes. Connected sets are measured by choosing a node at random and performing a breadth-first search of the graph, counting unique vertices along the way. If the number of vertices encountered is less than the total number of vertices in the graph, the graph (and the wireless network which it represents) is partitioned. To ensure the largest connected set has been counted, an unvisited vertex from the graph is chosen and the search is continued,

starting the count over from one. The largest connected set encountered is recorded.

Figure 17 shows the size of the largest connected set as a proportion of all nodes in the network versus the transmission range as a proportion of the calculated optimal transmission range. From these results it is determined that at the optimal transmission range (i.e., 100% on the *x* axis), at least 98% of all nodes are connected on average. Using these simulation results, the transmission range of a network that is connected with a 98% probability given a uniform distribution of nodes is presented in Table 2 for each network size.
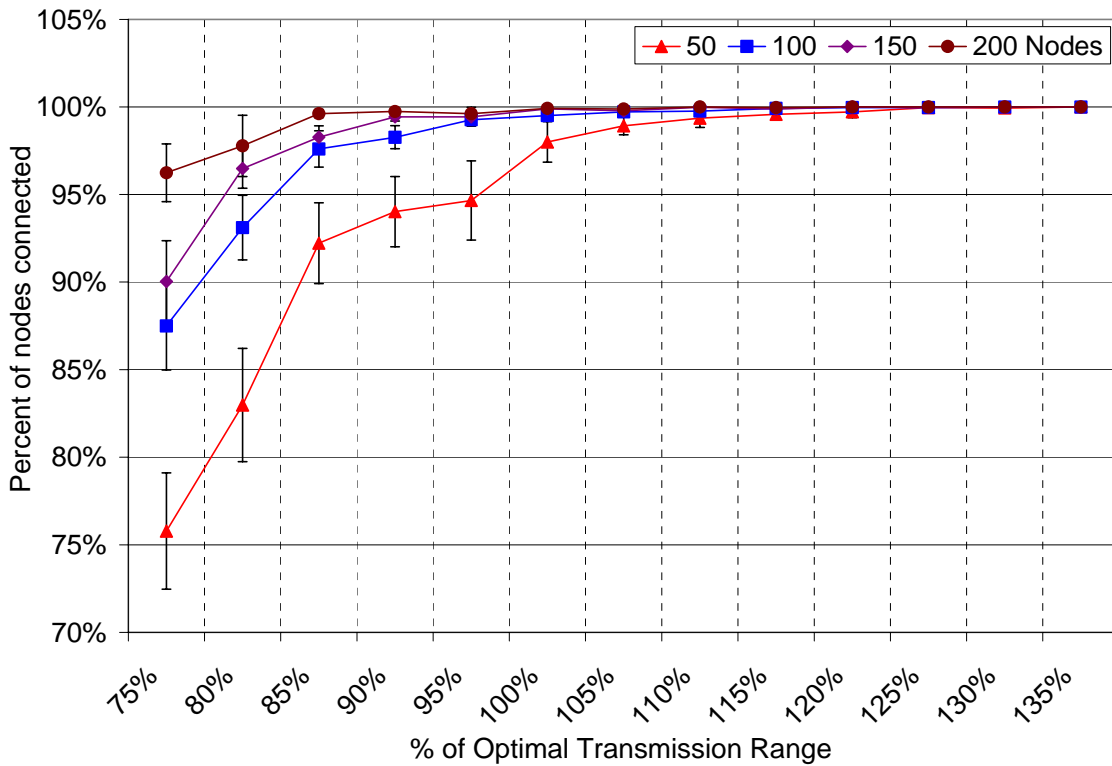
Figure 17. Network connectedness versus proportion of optimal transmission range

Table 2. Optimal transmission range

Nodes	Transmission Range (meters)	Average Connected Set
50	2,366	49.00
100	1,543	97.60
150	1,237	144.73
200	1,033	192.48

o   Levels: Given the 100 square-kilometer network size, transmission range is chosen to be approximately 125%, 175% and 250% of the optimal transmission range for the chosen number of nodes.   Additional configurations using approximately 95% (50, 100 and 150 node scenarios) and 60% (50 node scenarios) of the minimum optimal transmission range are also simulated to compare the impact of the number of nodes given the same transmission range. Simulation execution time for 200-node configurations is over eight hours in some cases; time constraints limited the scope of this study to only three different transmission range values for 200-node networks. Specific transmission range values used for each scenario are presented in Table 3.

Table 3. Transmission range factor levels

Nodes	Transmission Range (m)				
50	1,375	2,180	2,680	3,800	4,930
100		1,375	1,900	2,680	3,800
150		1,375	1,530	2,180	3,100
200			1,375	2,180	3,100

- Swarm size (number of UAVs): Directly impacts the swarm density and in conjunction with transmission range, the necessity and degree of multi-hop routing

    o   Levels: 50, 100, 150, 200

- Workload: Performance is evaluated at different workloads to determine which protocol performs best under similar conditions

  o Levels: Minimal (128 kbps), Low (256 kbps), Medium (512 kbps), High (640 kbps), Overload (1280 kbps)

Table 4 shows the chosen factors and specified levels; (5*5*4*5) = 500 experiments are required to complete a single replicate of the full-factorial experiment. Since the 95% (50, 100 and 150 nodes) and 60% (only 50 nodes) values for transmission range are not used for all swarm sizes, the actual number of experiments required for a single run is (5*(3*4 + 3 + 1)*5) = 400 experiments.

Table 4. Factor levels

Factor	Level				
	1	2	3	4	5
Routing protocol	AODV	OLSR	Greedy	GPSR	None
Transmission range	60%	95%	125%	175%	250%
Swarm size	50	100	150	200	
Workload (kbps)	128	256	512	640	1280

3.8 Evaluation Technique

Since a swarm of autonomous unmanned vehicles is not yet fielded, measurement of an actual system is not feasible. Even if such a system did exist, time and cost would likely make measurement an unlikely technique for evaluation. Furthermore, an analytic model with the fidelity to specify the various factors to be evaluated in this experiment has not been developed.

This leaves simulation of the system as the most reasonable evaluation technique. The system is modeled in OPNET Modeler 12.0 using a customized node model based on the *manet_station_adv* standard node model included with the OPNET Wireless module.

All default values are chosen for the wireless network parameters except for receiver sensitivity (dBm) and transmission power (W) which are altered to achieve the desired transmission range; appropriate values to approximate desired transmission ranges are determined through simulation and are presented in Appendix A. Standard process models for all protocols except GPSR are used; the process model for GPSR is constructed following the description in [KaK00]. Model files are described in Appendices B and C and are available for independent verification, validation, and experiment duplication.

During preliminary simulations the metrics of interest stablized after 400 seconds of simulation time. Each simulation is executed for 1200, with statistics collection beginning at 200 seconds to allow the mobility manager to effectively randomize node placement and for transient network behavior to achieve steady state. 100 values are collected per statistic; each value is the average of the statistics measurement over a 10-second simulation period.

3.9   Experimental Design

A full-factorial experiment is run with 10 repetitions for each configuration, requiring 4,000 simulation runs. Each replication is conducted with a different random number seed, but the same 10 seeds are used for each configuration to ensure uniformity of node mobility across configurations. For comparison, an additional set of 200 simulation runs (4 swarm sizes * 5 workloads * 10 repetitions) is conducted with no routing and a 14 km transmission range which is sufficiently far enough to guarantee connectivity between any two nodes in the network. The variance observed in all target metrics over the 10 repetitions is sufficiently small to not require more than 10 repetitions of each experimental configuration to achieve acceptable confidence interval widths.

3.10 Discussion of Research Metrics and Failure Modes

To determine what causes dropped packets to be discarded, a number of OPNET statistics are collected to identify the specific failure mode. The failure mode statistics discussed in Section 3.5 are derived from OPNET statistics in the following manner:

- Buffer Overflow: Directly measured from the *WLAN Buffer Overflow* statistic, this failure typically occurs during periods of high network load when contention for the medium causes the WLAN MAC to repeatedly backoff or repeat transmissions, or when the node is generating a large amount of traffic.

- Transmission Failure: Taken from the *WLAN Retry Threshold Exceeded* OPNET statistic, this failure occurs when the intended destination node has moved beyond radio range of the transmitter, or as a result of collisions due to the hidden node problem discussed in Section 2.3.2.3. Any routing protocol control packets unsuccessfully transmitted in unicast mode are also recorded by this statistic; as a result, the sum of all dropped packets and packets successfully delivered can exceed the total number of data packets transmitted due to the inclusion of control packets in this metric.

- Protocol Failure: Statistics which are recorded to determine protocol failure vary by protocol and are discussed for each protocol below.

- Routing Failure: Some packets are discarded by the IP processor at an intermediate node, typically due to a routing error causing a loop and exhaustion of the packets time-to-live (TTL). OPNET records these packets under the the *IP Packet Dropped* statistic.

3.10.1 No Routing

With no routing protocol configured on the nodes, packets are simply addressed to the destination node and transmitted. As a result, all dropped packets are due to the destination node being out of range of the transmitter. After seven unsuccessful transmission attempts, such packets are discarded by WLAN and recorded under the *WLAN Retry Threshold Exceeded* statistic.

3.10.2 AODV

While the OPNET implementation of AODV does directly modify the IP common routing table to maintain routes [Opn06b], packets addressed to a destination for which the routing table has no entry are passed to AODV for processing. If AODV fails to discover a route, it is dropped by the protocol and recorded by the AODV Dropped Packets statistic. In addition, the AODV Dropped Packets statistic records the number of packets dropped between statistic measures; in order to facilitate direct comparison among the different routing protocols (which record the equivalent statistic in packets per second), the results are divided by the number of seconds between statistic measures to normalize the data. Furthermore, some AODV control packets are unicast which can inflate the transmission failure metric by including dropped frames which are not data packets.

3.10.3 GPSR

The GPSR model constructed for this study uses the OPNET MANET API to discard packets if a route cannot be found; this causes the packet to be recorded as dropped by IP in addition to the GPSR Dropped Packets statistic. To determine the actual number of packets dropped by IP, the difference between the two statistics is taken for the Routing Failure metric.

3.10.4 OLSR

As the only proactive routing protocol in this study, OLSR maintains a route to every known destination in the IP common routing table. As such, it does not handle data packets directly; packets with no known route are discarded by IP. Since this is actually a failure of the routing protocol, packets dropped by IP are considered to be protocol failures, and the routing failure metric is not used for OLSR.

3.11 Summary

A swarm of autonomous unmanned aerial vehicles is modeled by computer simulation to evaluate methods of routing data throughout the swarm. A full-factorial experiment is performed to evaluate the impact of varying the routing protocol, transmission range, swarm size, and workload on the performance of the routing system. The synergies of spatial multiplexing are hypothesized to overcome the overhead introduced by multiple transmissions necessitated by multi-hop routing.

IV. Results and Analysis

In this chapter the experimental results are presented and analyzed. First, the methods used to verify and validate the simulation models are discussed in Section 4.1. The results of each individual performance metric are presented in Section 4.2 with some statistical analysis. Section 4.3 provides an overall analysis of the results.

4.1 Model Verification and Validation

Simulation accuracy is only as good as the underlying model; simulation models must be validated and verified in order for simulation results to be considered representative of a real system constructed to the same specifications. Model verification establishes the correctness of the model implementation and ensures that the model operates as designed. Model validation ensures the model design is actually representative of the real system.

Several components of the model used in this study were authored and distributed by OPNET Technologies with the OPNET Modeler 12.0 simulation package and are assumed to be correctly verified and validated models. These include: the *manet_station_adv* node model and its component process models, the *mobility_cfg* node model, and the *aodv_rte* and *olsr_rte* routing process models. The *gpsr_rte* process model is a custom-built model and therefore must be fully verified and validated.

4.1.1 GPSR Model Verification

Verification of the GPSR routing process model is conducted by running a series of controlled simulations using three static (non-mobile) configurations of wireless nodes. The nodes in each simulation use the same customized *manet_station_adv_mth* node model as

54

the full experiment, and all simulation parameters not specified here are set to the same values.

The node marked *S* in Figures 18-20 represents a traffic source, generating 100 fixed-length packets per second at a constant 0.01-second inter-arrival time, addressed to the node labeled *D*. All nodes are aligned to a 250-meter grid (there are 250 meters between each adjacent node), and statistics are collected at each node to determine which nodes re-transmit the packets and how many packets are ultimately delivered to *D*. Each layout is simulated twice; once with perimeter routing enabled, and once without. There is a deterministic GPSR routing path for each scenario and configuration. By examining the number of packets delivered to *D*, the hop count of those delivered packets, and the number of packets forwarded by each node in the network, the routing path of the packets in the simulation can be accurately determined and compared to the expected route.
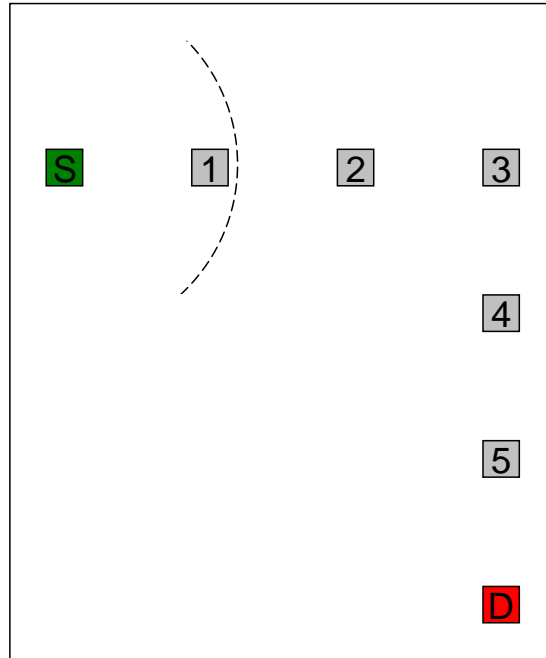
Figure 18. Model verification layout A

Figure 18 shows the layout used to verify basic greedy forwarding correctness. Nodes are set in a 250-meter grid and have 275-meter transmission ranges; the dashed curve represents the extent of node *S*'s transmission range. The expected routing path for both greedy-only and perimeter routing is S-1-2-3-4-5-D (6 hops).

The layout presented in Figure 19 is used to demonstrate the model appropriately forwards packets to the *most* greedy neighbor (the node closest to the destination) and that other nodes which overhear these packets appropriately discard them. In this layout, a 500-meter transmission range is used; the dashed line on the left represents the extent of node *S*'s transmission range, and the one on the right shows that nodes 6 and 10 are within node 4's range, but that nodes 9 and 11 are not. The expected routing path, with perimeter routing enabled or disabled, is S-2-4-8-D (4 hops).
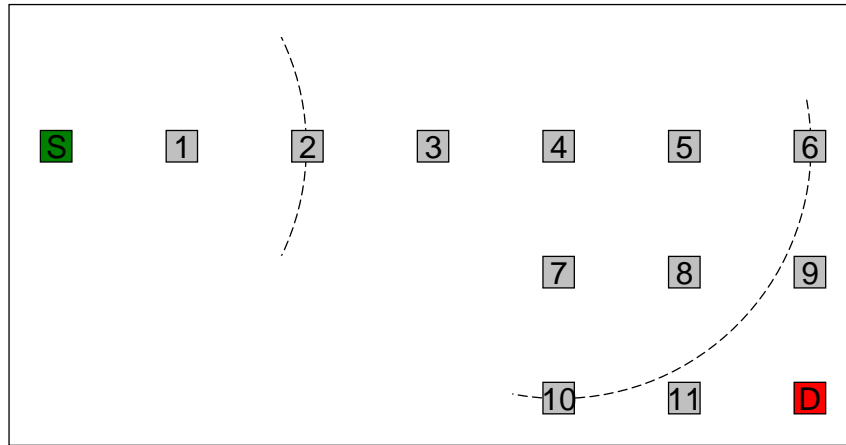
Figure 19. Model verifiation layout B

Figure 20 shows the network layout used to demonstrate appropriate behavior when greedy forwarding fails and perimeter-mode routing is required. Nodes have a 275-meter transmission range, and the dashed curve represents the range of node *S*. With perimeter routing disabled, the expected outcome is for packets to follow S-1-2 and then be discarded

by node 2 as there are no greedy next-hop neighbors. With perimeter routing enabled, however, packets should follow S-1-2-3-4-5-6-7-8-9-10-11-12-13-D (14 hops). Furthermore, the segments S-1-2, 5-6-7, and 9-10-11-12-13-D should be forwarded in greedy mode, whereas segments 2-3-4-5 and 7-8-9 utilize perimeter-mode forwarding.
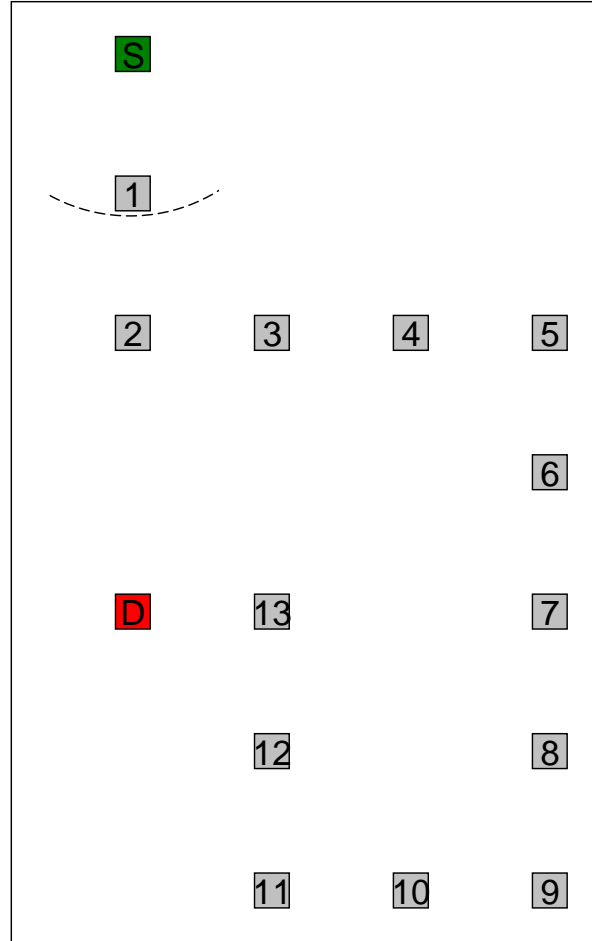
Figure 20. Model verification layout C

In all six verification simulations (layout A through C each with and without perimeter mode enabled), simulated behavior matched the expected outcomes including packet delivery success or failure, packet hop count, routing path, and routing mode. With a

fully and properly verified model, it is assumed that the *gpsr_rte* process model accurately implements the GPSR specification as published in [KaK00].

4.1.2   GPSR Model Validation

Model validation is performed by duplicating the original experiments documented by the authors of GPSR in [KaK00] and comparing the results. A 50-node network is constructed using a 1500-meter by 300-meter network boundary, and WLAN radios are configured for a 250-meter range (-85 dBm packet reception-power threshold and 2.2 mW transmission power). Node mobility is governed by the random waypoint model with a random node velocity chosen uniformly between 0 and 20 meters per second and pause times of 0, 30, 60 and 120 seconds. Traffic consists of 30 constant bit rate traffic flows originated from 22 transmitting nodes. Each traffic flow is generated by transmitting fixed-length 64-byte packets with exponentially distributed inter-arrival times with a 0.25-second mean inter-arrival time, yielding 2048 bits per second, or 2 kilobits per second per flow. Each traffic flow generates all packets addressed to a single destination node chosen randomly at simulation start; 14 of the 22 nodes generate one flow, the other 8 each generate two flows for a total of 30 traffic flows. A summary of experimental factors is presented in Table 5.

Table 5. Model validation experimental factors

Factor	Level					
	1	2	3	4	5	6
Beacon Interval (s)	1.0	1.5	3.0			
Pause Time (s)	0	30	60	120		
Random Number Seed	128	129	130	131	132	133

Each simulation runs for 1,200 seconds with statistics collection beginning at 300 seconds to allow for node mobility to effectively randomize node placement. With an

average node velocity of 10 meters per second, 300 seconds is sufficient time for nodes to travel on average 3000 meters. As the network boundary is 1,500 by 300 meters, random node placement is ensured. Packet delivery success rate and total beacon packets transmitted are measured for comparison to the results in [KaK00]. Each configuration is simulated with six different random number seeds, and the average of each metric across all six simulations is recorded.

Results of the validation simulations are presented in Figure 21 in addition to the results of the same experiments from [KaK00]. The top three lines labeled KARP are the results as published in the GPSR paper; the bottom three lines labeled GPSR are the results of the OPNET model validation simulations.
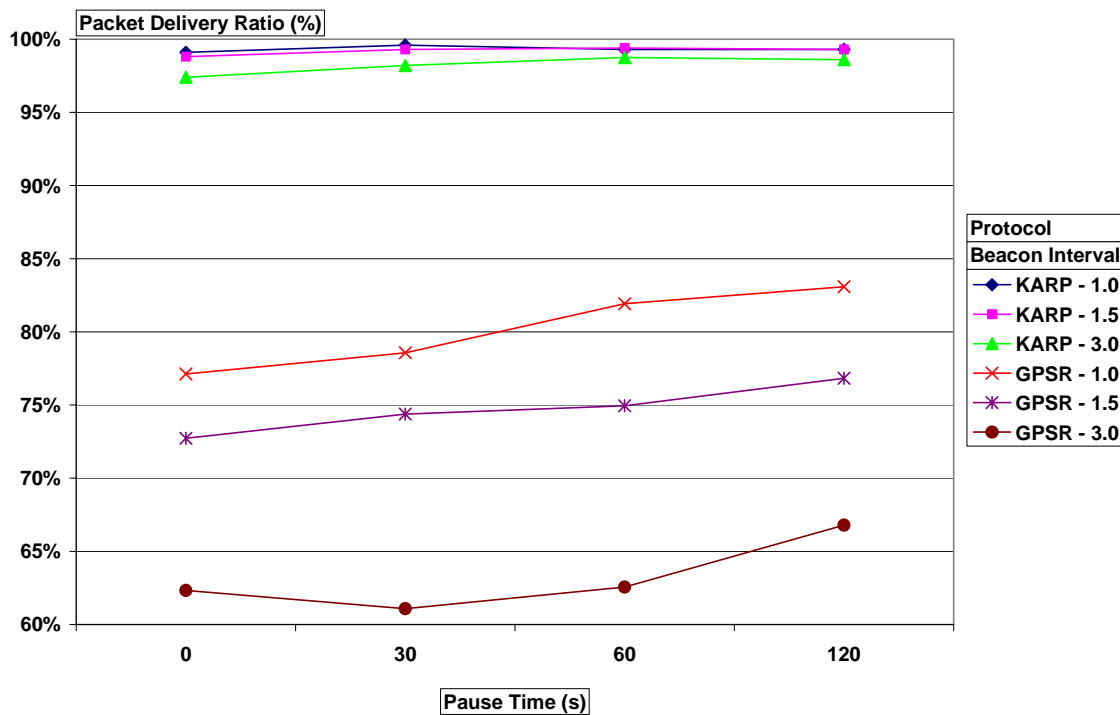
Figure 21. Packet delivery ratio versus pause time

While the absolute results differ between the published GPSR results and those from the validation simulation, the response to the increase in pause time follows the same general profile between the two. In addition, nearly all packet losses in the GSPR model validation simulations occurred at the WLAN level due to exceeding the packet re-transmission threshold; the difference between the two sets of results is attributed to the MAC-layer failure feedback optimization implemented by Karp and Kung that is not implemented in the custom OPNET GPSR model. With a 1.0-second beacon interval, an entry in a given node's neighbor table will persist for 4.5 seconds before expiration if the neighbor is never heard from again. With 10 meters per second average node velocity, nodes travel on average 45 meters before a neighbor table entry expires.

The mobility to transmission range ratio is the average distance a node travels before expiration of a neighbor table entry divided by the transmission range. With a 250-meter transmission range, nodes in the validation simulations have a mobility to transmission range ratio of 0.18, meaning nodes travel, on average, a distance equal to 18% of the transmission range before their neighbor table entry expires. It is reasonable to expect more invalid neighbor table entries for a shorter transmission range than with a larger transmission range. For example, in the UAV scenarios the OPNET GPSR model is built for the minimum transmission range examined is 1,390 meters, which equates to a mobility to transmission range ratio of 8.1%. The maximum transmission range examined is 4,390 meters for a mobility to transmission range ratio is 2.3%.

4.2 Results and Analysis of Performance Metrics

In this section, relevant data from the experiment is presented and analyzed. The three metrics of interest (i.e., packet delivery ratio, hop count and end-to-end delay) are each discussed individually along with a statistical discussion as appropriate.

4.2.1 Analysis of Packet Delivery Ratio

The analysis of variance (ANOVA) uses the general linear model with Packet Delivery Ratio (PDR) as the response and protocol, nodes, transmission range and workload as predictors, including their two and three-way interactions. Since transmission range and number of nodes are covariate, interaction terms with those variables are not considered. The computed model accounts for 96.73% of the variation in PDR and finds that all first, second and third-order terms which do not contain random seed are statistically significant at the 0.05 significance level. Results of the ANOVA are presented in Table 6.

Table 6. ANOVA results for packet delivery ratio

Source	DF	Seq SS	% Variance	Adj SS	Adj MS	F	P
Protocol	**4**	**141.9336**	**41.9%**	**104.2273**	**26.0568**	**7214.81**	**0**
Nodes	3	19.5839	5.8%	0.151	0.0503	13.93	0
Tx Range	**7**	**85.4026**	**25.2%**	**85.4026**	**12.2004**	**3378.13**	**0**
Workload	**4**	**35.5609**	**10.5%**	**24.3409**	**6.0852**	**1684.93**	**0**
Protocol*Nodes	12	14.4968	4.3%	7.3039	0.6087	168.53	0
Protocol*Tx Range	28	3.7816	1.1%	3.7816	0.1351	37.4	0
Protocol*Workload	16	14.9027	4.4%	10.532	0.6582	182.26	0
Nodes*Workload	12	1.4966	0.4%	0.3875	0.0323	8.94	0
Tx Range*Workload	28	2.486	0.7%	2.486	0.0888	24.58	0
Protocol*Nodes*Workload	48	2.3401	0.7%	0.9482	0.0198	5.47	0
Protocol*Tx Range*Workload	112	2.9883	0.9%	2.9883	0.0267	7.39	0
Error	3725	13.4531	4.0%	13.4531	0.0036		

Factors which most contribute to variation in the response are bolded in the table. The general linear model assumes that the error terms (residuals) are independent, normally distributed, and have a zero mean [HiL06]. Figure 22 presents two graphical aids to verify these assumptions. The scatter plot on the left is used to verify independence of the

61

residuals; as no trends or patterns are evident, the residuals are assumed independent. The histogram on the right is used to verify normality and zero mean; the superimposed normal curve shows a reasonably good fit, and it can be seen that the mean is zero.
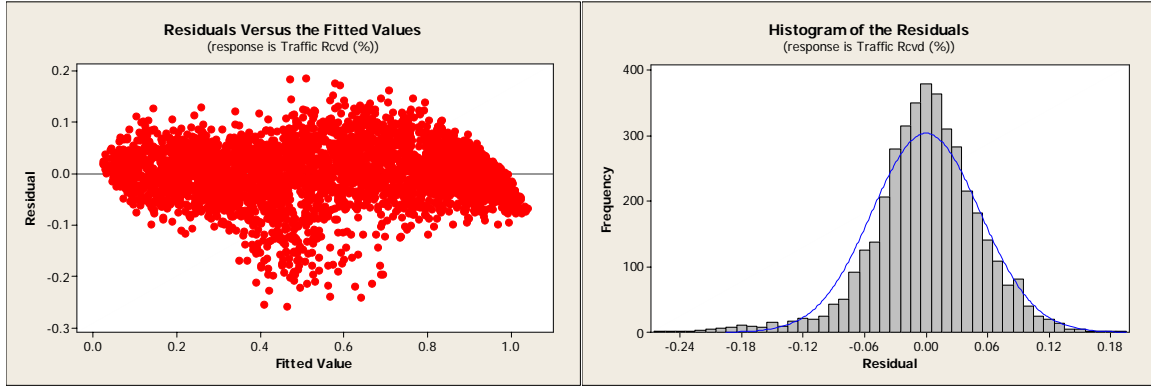
Figure 22. Visual tests to verify ANOVA assumptions for PDR

The ANOVA attributes most of the variance in PDR to the first-order effects of protocol (41.9%), transmission range (25.2%), and traffic workload (10.5%). The first order effect of number of nodes (5.8%), as well as the second-order effects between protocol and workload (4.4%) and protocol and number of nodes (4.3%) also contribute more to the variation in PDR than random error (4.0%). Several plots demonstrating the impact of these factors are presented below with 90% confidence intervals shown.

Figure 23 shows packet delivery ratio versus transmission range for GPSR. As expected, packet delivery success rate increases with transmission range, and higher traffic loads experience lower success rates. The sharp dip in panel (a) at 1.375 kilometers is due to the fact that it is only 60% of the optimal transmission range for a 50-node scenario; an examination of the packet failure modes for that scenario reveals that approximately 50% of all packets are WLAN transmission failures, 10% buffer overflows, and 14% are protocol

failures. With the transmission range only at 60% of optimal the network is expected to have more partitions.
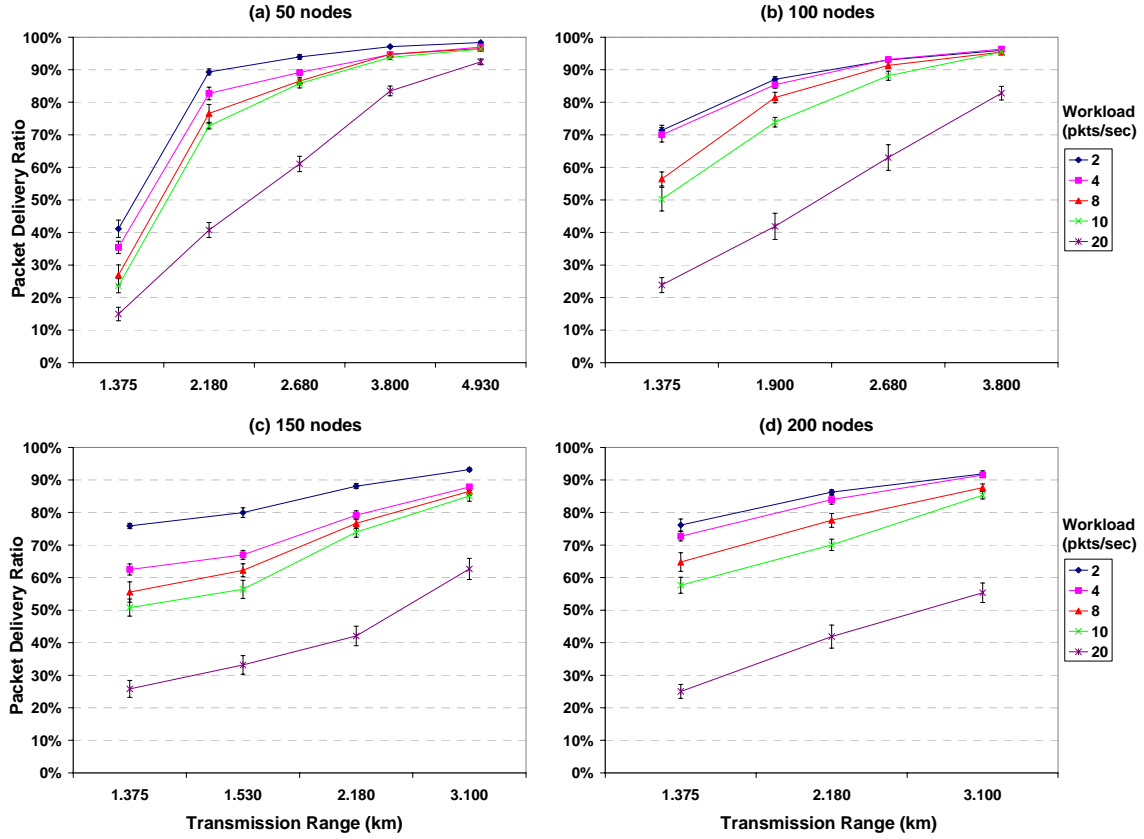
Figure 23. GPSR Packet delivery ratio versus transmission range

In Figure 24, packet delivery ratio is plotted against workload for all five routing protocols using the optimal transmission range for each network size. As traffic load and contention for the medium increases, packet delivery success decreases. While AODV has a higher success rate for lightly-loaded networks, greedy forwarding and GPSR appear to have an advantage as the traffic load increases.
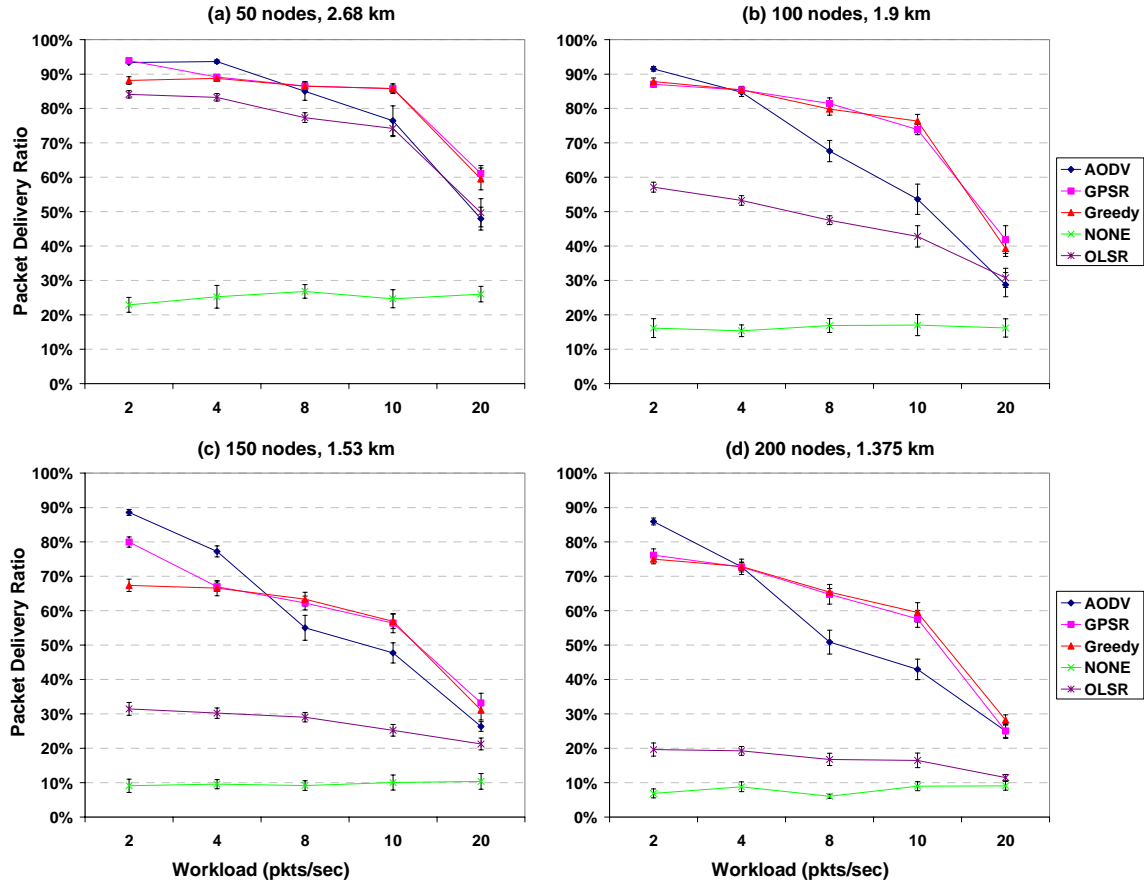
Figure 24. Packet delivery ratio versus workload using optimal transmission range

When forwarding packets in perimeter mode, GPSR sends each packet over a series of short hops even though a more efficient path may exist. In Figure 25, the packet delivery ratio for greedy forwarding and GSPR are plotted against transmission range for four scenarios to determine if the additional traffic load due to the high number of re-transmissions which occur during perimeter forwarding significantly impact the performance of GPSR. Note that the scale on the PDR axis is different in each plot. Even at the most highly-loaded network, with 200 nodes and 20 packets per second traffic load, GPSR performs nearly the same with perimeter mode routing enabled or disabled. An examination of the failure mode data indicates nearly all of the variation between the two is due to

protocol failure; with perimeter mode disabled, packets which find no greedy next-hop neighbor are discarded by the routing protocol.
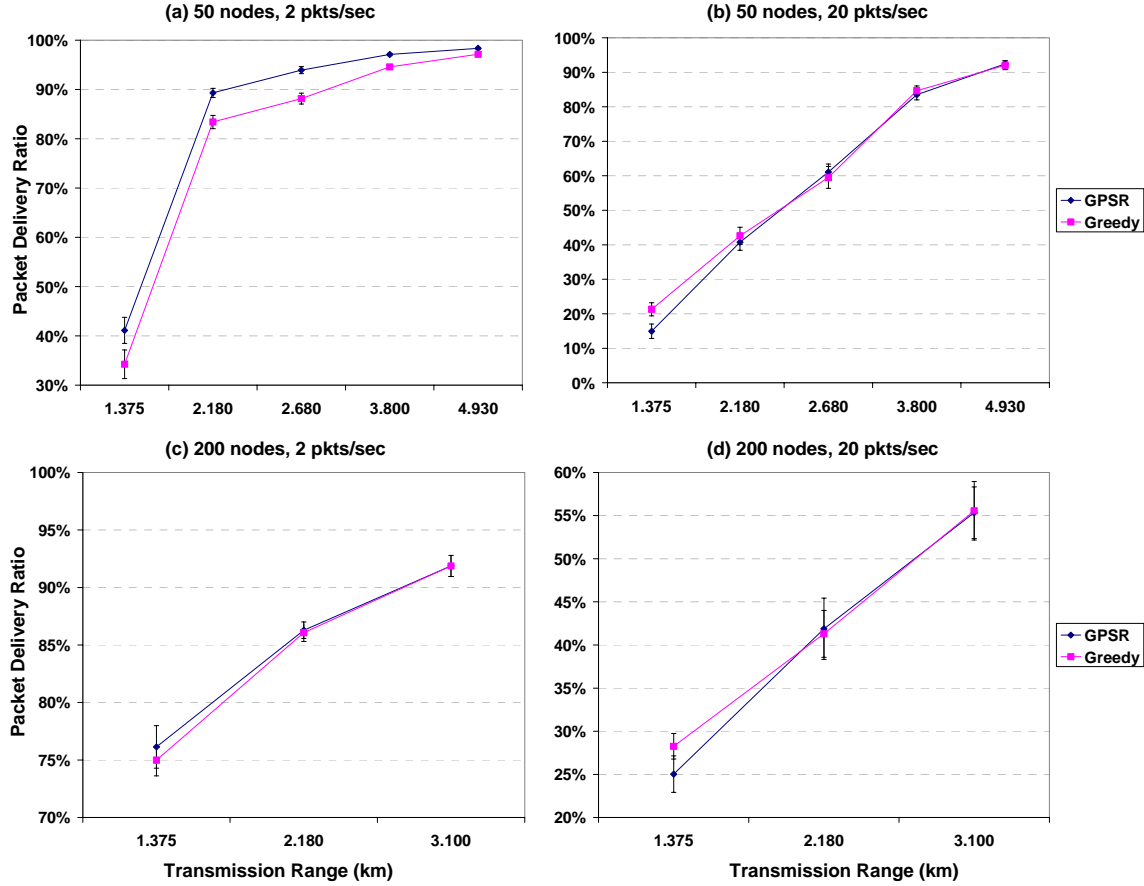
Figure 25. Comparison of GPSR and greedy forwarding

4.2.2 Analysis of Packet Hop Count

Analyzing the average hop count gives an idea of how efficient the routing protocol is; forwarding a packet over too many hops consumes more bandwidth with superfluous transmissions. In addition, transmission delay is typically the dominant factor in end-to-end delay; unnecessary transmissions increase delay.

An ANOVA is also completed for packet hop count with the same predictors used in the PDR ANOVA. The results indicate that all first, second and third-order terms not

containing random seed are statistically significant at the 0.05 significance level, and the resulting model accounts for 95% of the variation in hop count. Terms in bold contribute a higher proportion of the variation in hop count than random error, and italicized terms are not considered statistically significant. The ANOVA table is presented in Table 7.

Table 7. ANOVA results for packet hop count

Source	DF	Seq SS	% Variance	Adj SS	Adj MS	F	P
Protocol	**4**	**2533.946**	**38.4%**	**1490.249**	**372.562**	**3375.27**	**0**
Nodes	3	171.775	2.6%	11.606	3.869	35.05	0
Tx Range	**7**	**2207.98**	**33.5%**	**2207.98**	**315.426**	**2857.64**	**0**
Workload	4	18.915	0.3%	5.561	1.39	12.59	0
Protocol*Nodes	12	202.869	3.1%	232.452	19.371	175.49	0
Protocol*Tx Range	**28**	**910.301**	**13.8%**	**910.301**	**32.511**	**294.54**	**0**
Protocol*Workload	16	29.362	0.4%	6.024	0.376	3.41	0
Nodes*Workload	12	2.541	0.0%	6.953	0.579	5.25	0
Tx Range*Workload	28	40.223	0.6%	40.223	1.437	13.01	0
Protocol*Nodes*Workload	48	13.163	0.2%	29.465	0.614	5.56	0
Protocol*Tx Range*Workload	112	52.945	0.8%	52.945	0.473	4.28	0
Error	3725	411.165	6.2%	411.165	0.11		
Total	3999	6595.185					

The majority of the variation in hop count is attributed to the first order effects of protocol (38.4%), transmission range (33.5%) and their second-order interaction (13.8%); they are also the only terms larger in magnitude than random error (6.2%).

Visual tests to verify the ANOVA assumptions are displayed in Figure 26. While there appear to be some outliers in the right half of the scatter plot, no clear pattern emerges and the residuals are considered independent. The superimposed normal curve shows a very close fit to the histogram of the residuals except for the fact that more than the expected number of data points has a zero residual. The high peak indicates positive kurtosis which could tend to influence the F statistic, causing an inability to reject the null hypothesis (there is no difference in variation between the groups) even though it is incorrect [HiL06]. Since none of the terms have borderline F values, this slight deviation from the normality assumption is accepted.
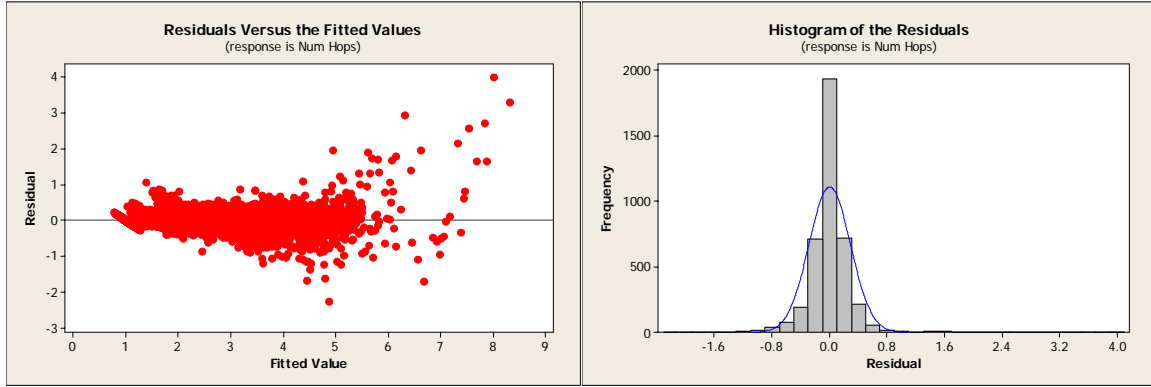
Figure 26. Visual tests to verify ANOVA assumptions for hop count

Figure 27 shows hop count versus transmission range for each evaluated routing protocol. Configurations with no routing are not considered in this section, as all successfully delivered packets are transmitted over only one hop. Results from the ANOVA indicate that most of the impact will be seen between different routing protocols and in response to changes in transmission range. In almost every scenario, AODV has the highest hop count, while Greedy forwarding and GPSR achieve the lowest hop count in nearly every case. Panels (a) and (b) show a spike in GPSR hop count at the lowest transmission range; this is attributed to a higher amount of packets being routed in perimeter mode. The 50 and 100-node scenarios include configurations with a transmission range equal to only 60% (50-node scenario) and 95% (50 and 100-node scenarios) of the optimal transmission range, yielding a sparse network with fewer than the optimal number of neighbors per node.
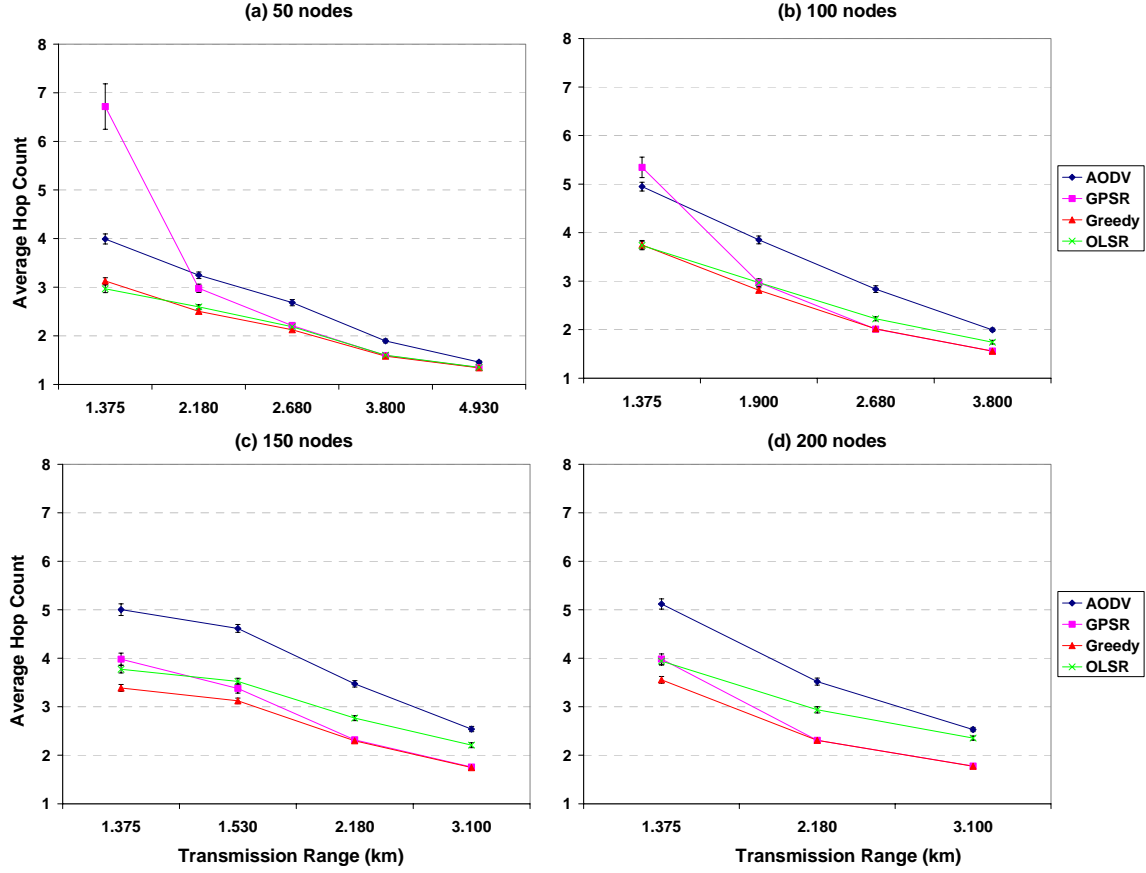
Figure 27. Hop count versus transmission range

As expected from the results of the ANOVA, hop count varies demonstrably with changes in transmission range since more intermediate nodes are required to forward packets from source to destination when a shorter transmission range is used. With longer transmission ranges, transmitting nodes can reach farther across the network with each hop.

Hop count is plotted against workload at the optimal transmission range for each network size in Figure 28. As the ANOVA attributes only 0.3% of the variation in hop count to workload, the relatively flat response is expected, with a somewhat more dramatic difference seen between the different protocols in both Figure 27 and Figure 28.
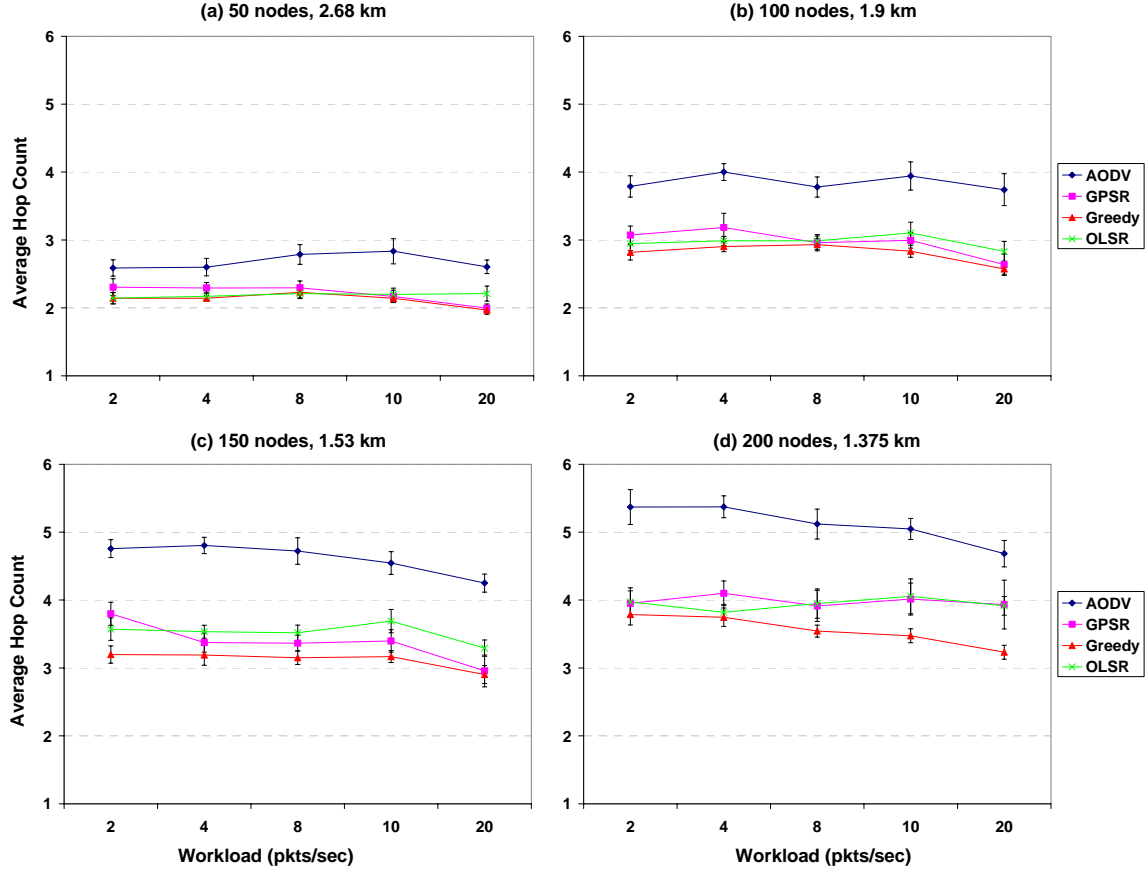
Figure 28. Hop count versus workload at optimal transmission range

4.2.3 Analysis of End-to-end Delay

Results from the ANOVA performed on end-to-end delay are presented in Table 8. All first, second and third-order terms not containing random seed are considered statistically significant at the 0.05 significance level. Factors in bold contribute the most toward variation in delay, while italicized factors are not considered statistically significant. The factors which contribute the most toward delay are workload (36.6%), protocol (9.6%) and the third-order interaction between protocol, workload and number of nodes (8.4%). All other terms contribute less towards delay than random error (8.3%).

Table 8. ANOVA results for end-to-end delay

Source	DF	Seq SS	% Variance	Adj SS	Adj MS	F	P
Protocol	**4**	**639.878**	**9.6%**	**340.687**	**85.172**	**572.26**	**0**
Nodes	3	184.374	2.8%	74.765	24.922	167.45	0
Tx Range	7	213.079	3.2%	213.079	30.44	204.52	0
Workload	**4**	**2434.777**	**36.6%**	**1720.078**	**430.019**	**2889.25**	**0**
Protocol*Nodes	12	434.423	6.5%	309.264	25.772	173.16	0
Protocol*Tx Range	28	396.792	6.0%	396.792	14.171	95.21	0
Protocol*Workload	16	373.87	5.6%	286.776	17.924	120.43	0
Nodes*Workload	12	367.231	5.5%	148.187	12.349	82.97	0
Tx Range*Workload	28	125.398	1.9%	125.398	4.478	30.09	0
Protocol*Nodes*Workload	**48**	**558.676**	**8.4%**	**418.744**	**8.724**	**58.61**	**0**
Protocol*Tx Range*Workload	112	377.708	5.7%	377.708	3.372	22.66	0
Error	3725	554.408	8.3%	554.408	0.149		
Total	3999	6660.614					

Visual tests to confirm the ANOVA assumptions are presented in Figure 29. The distinct line seen on the left side of the scatter plot crosses the points (-1, 1) and (1, -1) and occurs due to the fact that delay is exclusively positive; any fitted value below zero *will* have a positive residual of at least the same magnitude, and all negative residuals must have a positive fitted value at least as large. Positive kurtosis is observed in the histogram of the residuals; with no borderline *F* values, the deviation from normality is accepted.
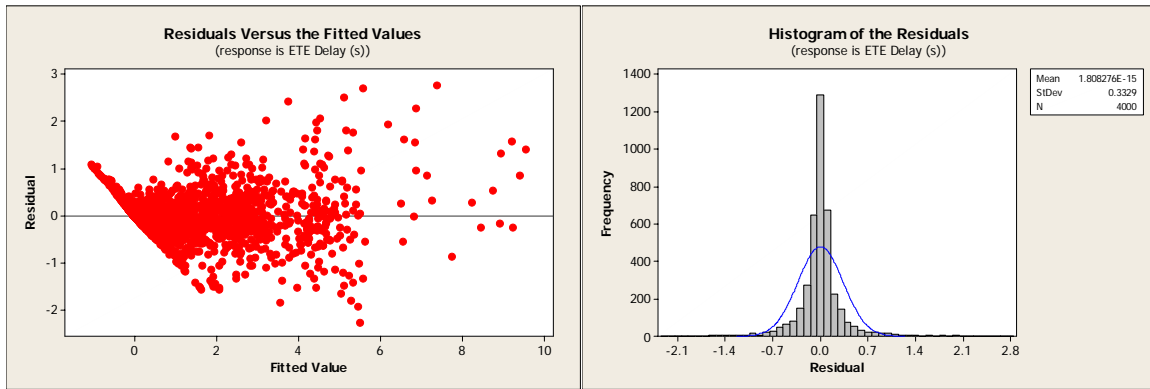
Figure 29. Visual tests to verify ANOVA assumptions for delay

Examination of end-to-end delay provides critical insight to the performance of the routing protocols under evaluation; from the results of the ANOVA traffic workload is expected to have the biggest impact on delay, followed by protocol.
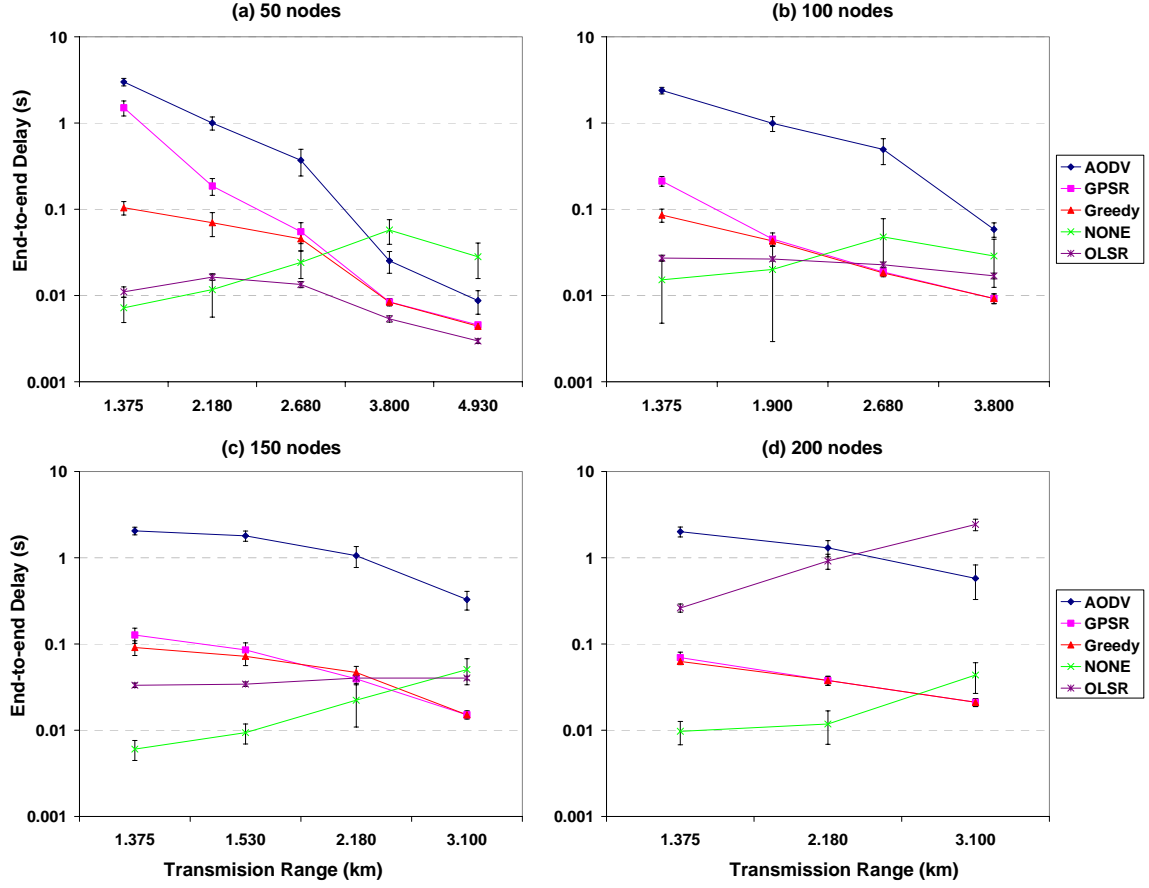
Figure 30. Delay versus transmission range at medium workload (8 pkts/sec)

Figure 30 shows delay versus transmission range for each protocol with a medium (8 packets per second) workload; a log scale is used for end-to-end delay for clarity due to the wide range of values observed. Even with only a moderate workload, AODV transmits packets with a much higher delay than the other protocols. Since the 50-node scenario with 1.375 km transmission range is only 60% of the optimal transmission range for 100 nodes, Greedy forwarding fails more often requiring more packets to use perimeter routing. This causes a marked increase in delay for that configuration under GPSR (seen in panel a), while in other configurations both GPSR and Greedy forwarding provide a delay comparable to that of OLSR, the sole proactive protocol under consideration.

Given the packet delivery ratios seen in Figure 24, GPSR appears to have a distinct advantage over OLSR with a comparable end-to-end delay and higher successful packet delivery ratio. In the 200-node scenario shown in panel (d), OLSR becomes overloaded and end-to-end delay jumps above even AODV at the longest transmission range. This is attributed to the large amount of routing control packets required by OLSR to maintain routes to every destination at all times. Even with multi-point relays, the large number of neighbors each node can communicate with creates a large amount of protocol overhead.

Figure 31. Delay versus workload at optimal transmission range

End-to-end delay is plotted against workload using the optimal transmission range for each scenario in Figure 31, using a log scale for delay. With no routing control packets
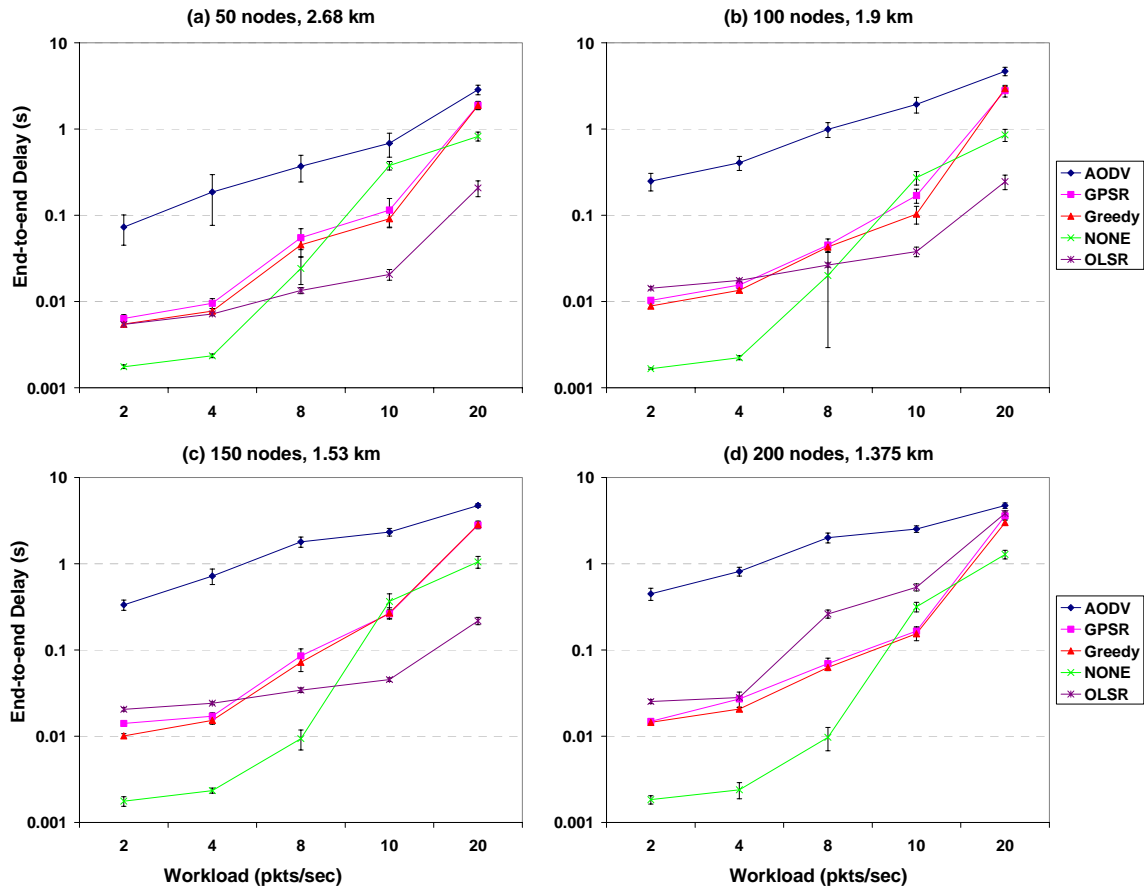
or re-transmissions, the increase in delay for the NONE configurations at higher workloads can only be attributed to contention for the wireless channel due to high traffic load. The dramatic response in delay for all protocols with increasing workload is expected from the results of the ANOVA.

Greedy forwarding and GPSR encounter increased delay under the highest (overload) workload, though still achieve a lower delay than AODV. The proactive nature of OLSR keeps delay fairly constant under increasing load until it appears to become overloaded in the 200-node configuration, as seen previously in Figure 30. AODV introduces the highest delay throughout all configurations and workloads.

4.3   Analysis of Transmission Failures

An analysis of the mode of packet transmission failure is performed to assist in modifying the system to improve performance. Examining packet failure mode data reveals that about half of the discarded packets are WLAN transmission failures, while the rest are evenly distributed between buffer overflows and protocol failures for OLSR and AODV. Failure modes with GPSR are almost exclusively WLAN transmission failures except in the 20 pkts/sec case where 40% of the discarded packets are buffer overflows. Average failure mode values over all network sizes and transmission ranges are shown in Figure 32 for the 2, 8, and 20 pkts/sec cases; routing failure is excluded for clarity because less than 1% of all packet failures were in that category.

The sharp increase in buffer overflows at a 20 pkts/sec workload indicates that AODV, Greedy forwarding and GPSR are still able to process packets but the wireless channel has become too congested to transmit the data as fast as necessary. OLSR experiences fewer buffer overflows because it is discarding more packets at the routing stage

and is not passing as much data to the MAC. The upward trend in transmission failures as workload increases for AODV, GPSR and Greedy forwarding indicates that packets are experiencing transmission failure as a result of channel contention and collision rather than nodes moving out of range from the intended next-hop neighbor.
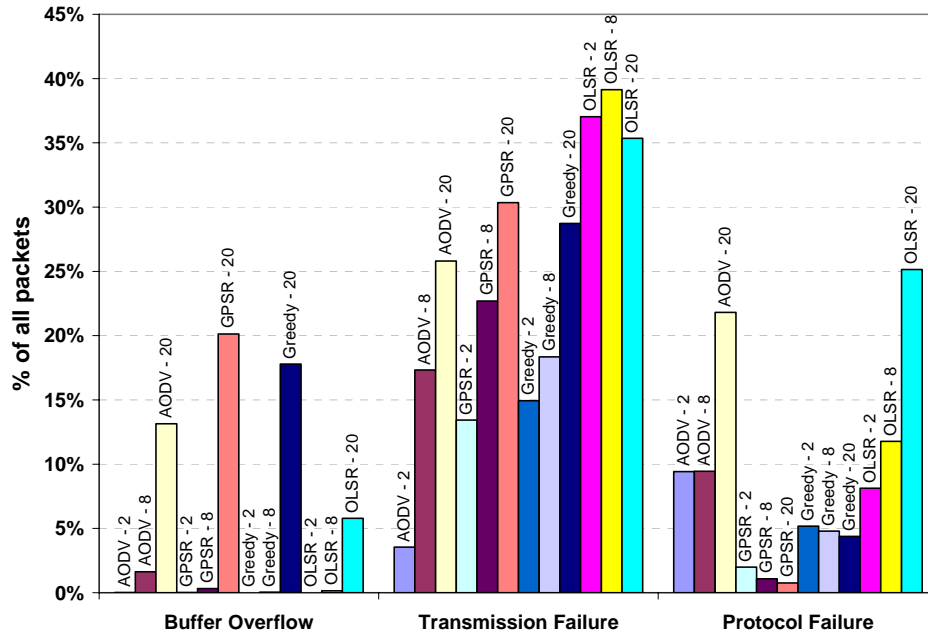
Figure 32. Packet failure mode for all network sizes and transmission ranges

A comparison of GPSR and Greedy forwarding is shown in Figure 33 for each network size. The difference for each failure mode between GPSR and Greedy is largest in the 50-node network and decreases with increasing nodes until there is virtually no difference in the 200-node network. An examination of packet delivery rate in Figure 25 indicates that there is no statistically significant difference in PDR for GPSR and Greedy forwarding in networks larger than 50 nodes except at the shortest transmission range for each network. In fact, they are the only simulations in which perimeter mode routing was used for more than a minimal number of packets. From this result, we can conclude that

there is no benefit to enabling perimeter mode in networks larger than 50 nodes with sufficient transmission range because any packets unable to be delivered using greedy forwarding probably could not be delivered using perimeter mode routing anyway. Enabling perimeter mode routing in these cases only serves to add traffic to the channel which most likely does not actually accomplish additional packet delivery.
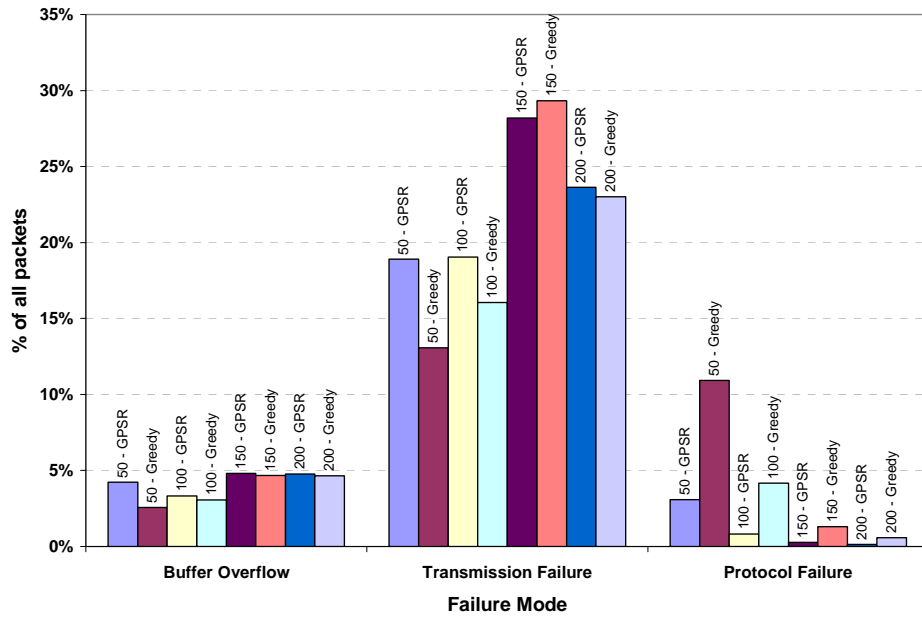
Figure 33. Packet failure mode for GPSR and Greedy for different network sizes

In fact, over all simulation configurations less than two percent of all data packet transmissions were perimeter-mode packets. A whole class of routing protocols has been developed to solve the problem of delivering packets when greedy forwarding fails; we'll instead focus on increasing the number of packets greedy forwarding can successfully deliver. By eliminating perimeter-mode routing and the many inefficient hops it requires, fewer overall packet transmissions reduce contention for the channel, which should require fewer WLAN backoff delays and improve average end-to-end delay.
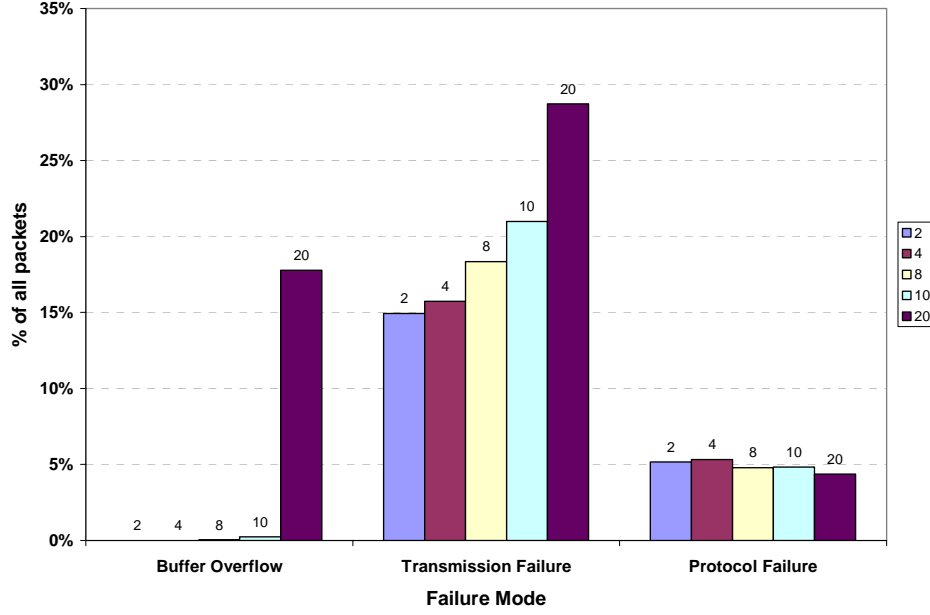
Figure 34. Failure mode for greedy forwarding at different workloads

Figure 34 shows packet failure for greedy forwarding at varying workloads. The sharp increase in buffer overflows at 20 pkts/sec can be mitigated by spreading packet generation over a larger number of nodes; rather than have 8 nodes each generating 20 pkts/sec, 16 nodes each generating 10 pkts/sec or 32 nodes each generating 5 pkts/sec would present the same aggregate traffic workload to the network, while spreading the load over more nodes reduces the possibility of overflowing the WLAN transmit buffer at any particular node.

There is a very slight decrease in the amount of protocol failures as traffic load increases. While further investigation is necessary to find a definitive cause, it may be due to more accurate neighbor location data as a result of more frequent data packet transmissions. Each node along an active routing path is transmitting 20 data pkts/sec, each with embedded location data, and all nodes within transmission range of those active nodes are

able to precisely update their neighbor tables with those nodes' locations at that frequency. This increased accuracy can allow nodes to make more accurate routing decisions.

Transmission failures can occur for one of two main reasons: either the intended destination node has moved beyond the transmission range of the sender, or the transmission has experienced a collision at the receiver and could not be correctly received. While the Distributed Coordination Function employed by the 802.11 MAC is intended to minimize packet collisions, they can still happen due to the hidden terminal problem. Since there is no reason to expect an increase in traffic workload would impact the rate at which nodes move beyond range of each other, the increase in transmission failures as traffic workload goes up is attributed to packet collisions. Some constant amount of those transmission failures, likely close to the 15% experienced in the 2 pkts/sec simulation, are due to nodes moving beyond transmission range. Spreading packet generation to a larger number of nodes is expected to have a similar effect on transmission failures (due to packet collisions) as is expected for buffer overflow failures. Efforts to improve packet delivery for greedy forwarding should be focused on reducing transmission failures.

Transmission failures due to nodes moving beyond transmission range of the sender occur because the sending node assumes all nodes in its neighbor table are reachable. When a neighbor moves beyond range of the sending node, its entry will remain in the neighbor table until it expires, which is 4.5 seconds in this experiment. Even if the neighbor is also transmitting data packets at 20 pkts/sec, since it has moved beyond range they will not be received by the sending node. The MAC-layer feedback optimization presented in Section 2.3.3.6 is designed to allow the node to update its neighbor table by deleting the entries of any node which has not acknowledged a data packet. Furthermore, since data packets from

each packet generating node are addressed to the same destination node for the duration of the simulation, there are likely many packets queued up in the WLAN transmit buffer addressed to the same next-hop neighbor which has moved beyond range, especially since the first failed transmission will be attempted 7 times by the WLAN MAC before being discarded. Also suggested by the authors of GPSR, traversing the WLAN transmit buffer and removing any packet addressed to the failed neighbor (and returning them to the routing module for reprocessing) is also expected to provide a significant decrease in transmission failures [KaK00].

4.4   Overall Analysis

Statistical analysis of the data collected via simulation finds that choice of routing protocol provides a statistically significant change in packet delivery ratio, packet hop count, and end-to-end delay. The analysis also indicates that routing protocol is the most significant predictor of PDR and hop count, while workload is the most significant predictor for delay due to contention for the wireless channel.

Since perimeter-mode forwarding requires many more packet hops that the true shortest path, a comparison of GPSR with and without perimeter-mode routing enabled (Figure 25) is conducted to determine if perimeter-mode routing is beneficial. This reveals that perimeter forwarding provides only a marginal increase in packet delivery success with a 0.45 increase in average hop count, and increases average end-to-end delay by 35.6%.

As expected, configurations with no routing protocol configured have the lowest packet delivery ratio until transmission range approached the length of the network boundary. Packets that are successfully delivered, however, are done so in only on hop and

78

with minimal delay, as expected. With no routing enabled and a 14-kilometer transmission range, 100% of generated packets are successfully delivered.

When configured to use the optimal transmission range for each scenario, GPSR (72.7%) and AODV (69.8%) have comparable successful delivery rates, while OLSR (47.4%) drops significantly as the number of nodes in the network increases. Looking at end-to-end delay, however, GPSR's end-to-end delay is 53% smaller than AODV in all configurations using the optimal transmission range.

4.5 Summary

This chapter presented and analyzed the results from the experiments conducted on the routing protocols in a UAV swarm. Simulation verification and validation methods were discussed and simulation metrics were described. The various ways packet delivery could fail were presented, and the results of each performance metric were analyzed statistically. An overall analysis and discussion of the results was also presented.

V. Conclusions and Discussion

A summary of the conclusions drawn from the data is presented in Section 5.1, and the significance of these finding is discussed in Section 5.2. Several recommendations on areas to continue this research effort are presented in Section 5.3. The chapter is summarized in Section 5.4.

5.1 Research Conclusions

The results of over 4,000 computer simulations supports the hypothesis that a geographic routing protocol, specifically GPSR, is an efficient and effective routing protocol for a swarm of UAVs. Furthermore, when considering successful packet delivery ratio and end-to-end delay, GPSR outperforms AODV with an equivalent packet delivery ratio but a 53% shorter end-to-end delay. GPSR also outperforms OLSR with a comparable end-to-end delay but with a 25% higher packet delivery ratio.

GPSR does not perform as well, however, as the no-routing scenario (with a 14.525 km transmission range). It was hypothesized that spatial multiplexing could overcome the redundant transmissions required for multiple-hop routing, and that a higher total throughput could be achieved; this is not the case. The baseline no-routing network is able to deliver 100% of the offered packets, even at a 1,280 kbps workload, while GPSR achieved only 25% (200-node configuration) to 60% (50-node configuration) packet delivery ratio at optimal transmission range. However, it is not always feasible to have radios capable of transmitting with enough power to reach across the entire network. In the absence of this option, GPSR is the protocol of choice among the three evaluated here for use in a swarm of UAVs.

5.2   Significance of Research

While communication relay nodes would certainly be necessary to provide connectivity into and out of the UAV swarm, it may not be feasible in all scenarios for every UAV to have a wireless radio with enough power to directly transmit to any other node in the network. It is shown that such a swarm of inexpensive UAVs is feasible using only low-power radios and employing a mobile ad hoc routing protocol such as GPSR to relay data throughout the swarm when the swarm is tolerant to some data loss.

5.3   Recommendations for Further Research

The simulations used for this study relied on a single random mobility model—the random waypoint mobility model. The random waypoint mobility model is, however, not indicative of the mobility pattern for all UAV swarms. It is recommended that other mobility patterns be evaluated, including a pattern which models the mobility expected by the UAV search swarm described in [MMP06].

Transmission ranges for each node configuration (50, 100, 150 and 200-node networks) were based on a multiple of the calculated optimal transmission range. As the level for each configuration was different, the transmission range and number of nodes factors were covariate and statistical analysis proved difficult. It is recommended that a common set of transmission ranges be constructed and simulated on all four node configurations to permit more accurate statistical analysis. The common set of transmission ranges should include a range that approximates the optimal transmission range of each network size.

Described as having "a profound effect" on the results in [KaK00], the MAC-layer feedback and interface queue traversal optimizations discussed in Appendix C could increase GPSR's packet delivery ratio.

This research did not model any layers above the network layer; the impact of ad hoc routing protocols on the TCP transport-layer protocol should evaluated to determine how retransmission of dropped packets by the transport layer 1) impacts overall packet delivery ratio, and 2) affects network congestion and throughput.

Another natural progression would be to examine the impact of varying different physical-layer parameters. This research assumed constant transmit power across all nodes for the duration of each simulation; a potential modification is to transmit packets with varying power depending on the distance to the intended destination node, or the number of neighbors a node has. It may be possible to decrease channel contention by transmitting at a lower power level when possible while maintaining packet delivery success by increasing power when necessary.

5.4   Summary

In this chapter, overall conclusions drawn from the results of the research effort were presented with some discussion on their significance. In addition, recommendations for future research were presented.

Appendix A–Approximate Transmission Range Determination

While the experimental design calls for varying the transmission range of the wireless radios, the OPNET software does not allow for the direct specification of a radio transmission range. Instead, for each packet the simulation package completes a 14-stage radio transceiver pipeline computation for every potential receiver to determine if each receiver has accurately received the packet or not [Opn06b]. Figure 35 shows an overview of the different pipeline stages.
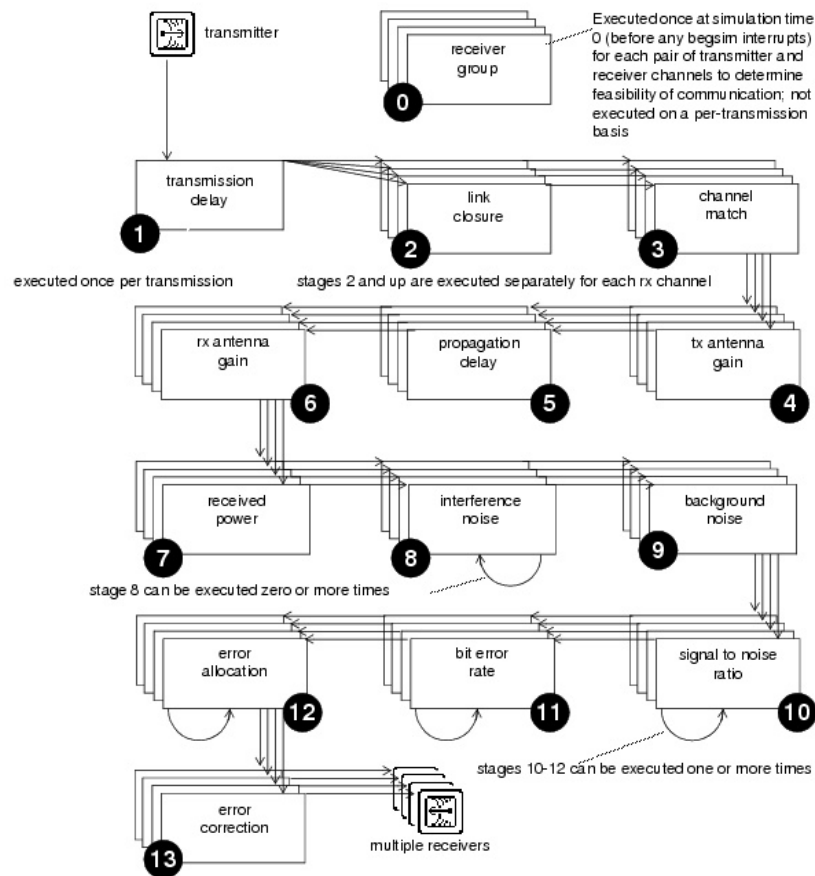

Figure 35. OPNET Radio Transceiver Pipeline Stages [Opn06b]

The simulated transmission range can be affected by modifying parameters in the antenna gain stages and received power stage. The parameters that can be modified are

transmit power and packet-reception power threshold, which is the lower limit of the amount of power the receiver must sense in order to accurately receive the packet. [Opn06a] states that received power is computed by

$$P_{rx} = P_{tx} \times G_{tx} \times \left(\frac{\lambda^2}{16\pi^2 r^2} \right) \times G_{rx} \qquad (6)$$

where

$$P_{rx} = power\ received\ (watts)$$
$$P_{tx} = power\ transmitted\ (watts)$$
$$G_{tx} = transmit\ antenna\ \ absolute\ gain$$
$$G_{rx} = transmit\ antenna\ \ absolute\ gain$$
$$\lambda = wavelength\ (m)$$
$$r = distance\ (m)$$

Since transmit and receive antennas are zero-gain (dB), both gain terms become 1. Signal wavelength is computed using

$$\lambda = \frac{c}{f} \qquad (7)$$

where c is the speed of light and f is the 2.462 GHz center frequency of the IEEE 802.11b PHY specification [IEE03b]. Rearranging (6) to solve for distance and substituting in (7) yields

$$r = \frac{c}{f\,4\pi} \sqrt{\frac{P_{tx}}{P_{rx}}} \qquad (8)$$

which can be used to estimate maximum transmission range by substituting in the transmission power and minimum packet-reception power threshold values specified in the simulation.

A simple simulation is conducted to validate (8). Two wireless nodes using the *wlan_station_adv* standard node model are placed at the exact same location in a blank OPNET project space. One is a stationary traffic generating node, broadcasting 100-byte packets through the entire duration of the simulation at uniform and constant 0.01-second intervals (100 packets per second). The transmit power parameter is promoted and set at simulation run time, and all other parameters are left at their default values. The other node generates no traffic, but moves at a constant velocity of 10 meters per second in a constant direction, so that at any given simulation time t, the two nodes are $10t$ meters apart. The packet reception-power threshold parameter is promoted and set at simulation run time, and all other parameters are left at their default values. Packet reception-power threshold is specified in the simulator in dBm (decibel milliwatts); the conversion to watts for use with (8) is $watts = 0.001 \times (0.1 \times dBm)^{10}$.

A series of transmit power values is computed using (8) which approximate an even distribution of transmission ranges between 0 and 20 kilometers for -95 and -90 dBm packet reception-power threshold values. The Traffic Received (pkts/sec) statistic is recorded for the mobile node, and statistics are collected every 2.5 seconds (or 25 meters) for the duration of the 2,500-second simulation. For each transmit power/packet reception-power threshold pair, the shortest distance at which at least 95 packets per second (95% of transmitted packets) are correctly received by the mobile node is recorded as the maximum transmission range for that configuration.

Table 9. Range Test Results

P_{tx} (mW)	P_{rx} = -90 dBm		P_{rx} = -95 dBm	
	Predicted Range (km)	Actual Range (km)	Predicted Range (km)	Actual Range (km)
1	0.31	0.275		
2	0.43	0.4		
3	0.53	0.5		
4	0.61	0.6		
5	0.69	0.675		
6	0.75	0.725		
7	0.81	0.8		
8	0.87	0.85		
9	0.92	0.9		
10	0.97	0.95		
15	1.19	1.175		
20	1.37	1.375		
25	1.53	1.53		
30	1.68	1.68		
35	1.81	1.83		
40	1.94	1.95		
45	2.06	2.08		
50	2.17	2.18		
60	2.37	2.40		
70	2.56	2.58		
80	2.74	2.78	4.87	3.73
90	2.91	2.93	5.17	3.95
100	3.06	3.10	5.45	4.175
125	3.43	3.48	6.09	4.675
150	3.75	3.80	6.67	5.125
175	4.05	4.10	7.21	5.55
200	4.33	4.40	7.71	5.925
225	4.60	4.65	8.17	6.275
250	4.85	4.93	8.62	6.625
300	5.31	5.38	9.44	7.25
350	5.73	5.83	10.19	7.825
400	6.13	6.23	10.90	8.15
450	6.50	6.60	11.56	8.675
500			12.18	9.375
600			13.35	10.275
700			14.42	10.975
800			15.41	11.85
900			16.35	12.325
1000			17.23	13.275
1100			18.07	13.9
1200			18.88	14.525

As seen in the results of the simulation in Table 9 and Figure 36, while the observed maximum transmission range during simulation with a -90 dBm packet reception-power threshold closely mirrored the predicted value, the predictions are approximately 30 percent

over for the -95 dBm simulations. This is attributed to the lower signal-to-noise ratio encountered by receiving a signal with much lower signal strength, corresponding to a higher bit-error rate and a higher proportion of packets discarded by later pipeline stages due to bit errors. The results in Table 9 are then used to specify transmission range in other simulations.
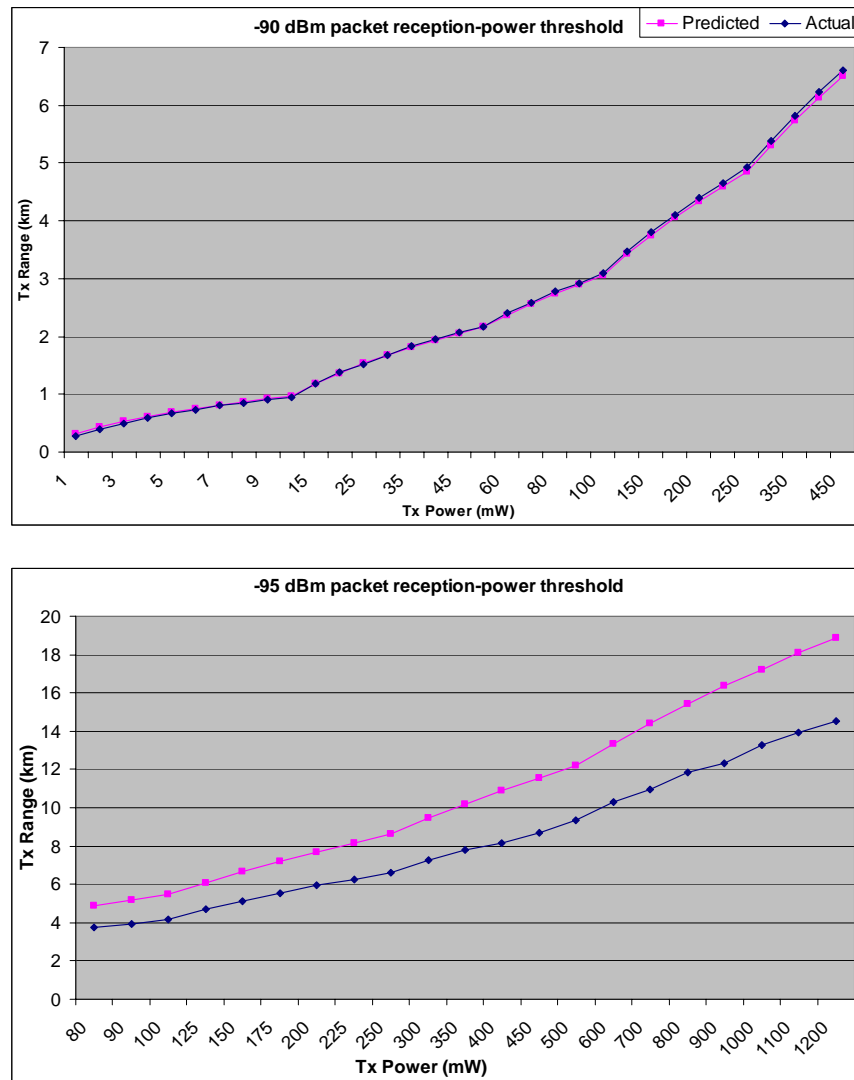

Figure 36. Transmission Range versus Transmit Power

Appendix B–Modifications to OPNET Standard Libraries

In order to provide an accurate direct comparison between the custom GPSR routing process and the built-in OPNET routing protocols, the GPSR model is designed to interface with the *manet_station_adv* node model in exactly the same manner. An excellent guide to interfacing a custom MANET protocol with OPNET's IP implementation is presented in [Opn06b]. An overview of the modifications is presented in this section.

B.1 Header File Modifications

Two header files are modified as described below. The customized header files must be stored in the /OPNET/12.0.A/models/std/include folder as the compiler cannot resolve multiple header files with the same name.

- *ip_higher_layer_proto_reg_sup.h*: added an entry (`IpC_Protocol_Gpsr = 202`) to the `IpT_Rte_Protocol` enumerated data type on line 49

- *ip_rte_v4.h*: added an entry (`IpC_Rte_Gpsr`) to the `IpT_Rte_Protocol` enumerated data type on line 281 and added the `IPC_RTE_PROTO_GPSR (1<<12)` macro definition on line 372

B.2 External Source Modifications

A single external source file is modified to support handling checks for GPSR packets. The *ip_rte_support.ex.c* file can be stored in any folder; when OPNET adds the new folder to the model directories, the custom file will be used rather than the original in the /models/std/ip folder. In each of the functions described below, the OPNET model checks packets for the standard MANET routing protocols included in OPNET. In each of the following files, an additional check for GPSR is added:

- Lines 1164 & 1858 – function `ip_rte_pkt_arrival`: handles the arrival of a packet at IP

- Lines 5806 & 5856 – function `ip_rte_datagram_dest_get`: extracts the destination address from an IP datagram

- Line 6036 – function `ip_rte_pkt_is_routing_pkt_ext`: determines if the packet is a routing control packet

- Lines 7246 & 7425 – function `ip_rte_mcast_datagram_dest_get`: extracts the destination from a multicast IP datagram

B.3 Process Model Modifications

Like the modified external source file, modified process model files can be stored in any folder. The following process models are modified as described:

- *manet_mgr.pr.m*

  o Defined a GPSR macro in header block line 8

  o Modified the `manet_mgr_routing_protocol_determine` function at function block lines 83-88 to recognize the selection of GSPR as an attribute

  o Modified the `manet_mgr_routing_process_create` function at function block lines 160-178 to launch the custom *gpsr_rte* process model

  o Added *GPSR Parameters* to the model attributes to support setting the *Beacon Interval* and *Neighbor Timeout Multiplier* parameters

89

- *wlan_dispatch.pr.m*

  o Added *Promiscuous Mode* and *Modified IFS Values* attributes to the *Wireless LAN Parameters* compound model attribute for use by *wlan_mac.pr.m*

- *wlan_mac.pr.m*

  o Modified the `wlan_mac_sv_init` function at function block lines 244 and 249 to use modified slot time (50 μs), SIFS (28 μs) and DIFS (128 μs) values for longer transmission ranges when *Modified IFS Values* is set to TRUE

  o Modified the `wlan_physical_layer_data_arrival` function at function block line 4052 for promiscuous use of the wireless interface when the *Promiscuous Mode* attribute is set to TRUE

- *ip_dispatch.pr.m*

  o Modified the `ip_dispatch_number_of_hops_update` function at function block line 7006 to recognize hop counts up to 255 rather than the standard 32 for compatibility with GPSR packets

Appendix C–Implementation Details

C.1 GPSR

With the components of the *manet_station_adv* node model appropriately modified to support a custom GPSR routing protocol, a customized version is created to promote GPSR statistics to the node level. The routing logic for the GPSR protocol is stored in the *gpsr_rte.pr.m* process model. All packets passed to IP from higher layers are delivered to the GPSR process for processing, and all packets that arrive at IP from the WLAN MAC which are not addressed to the current node are also passed to GPSR.
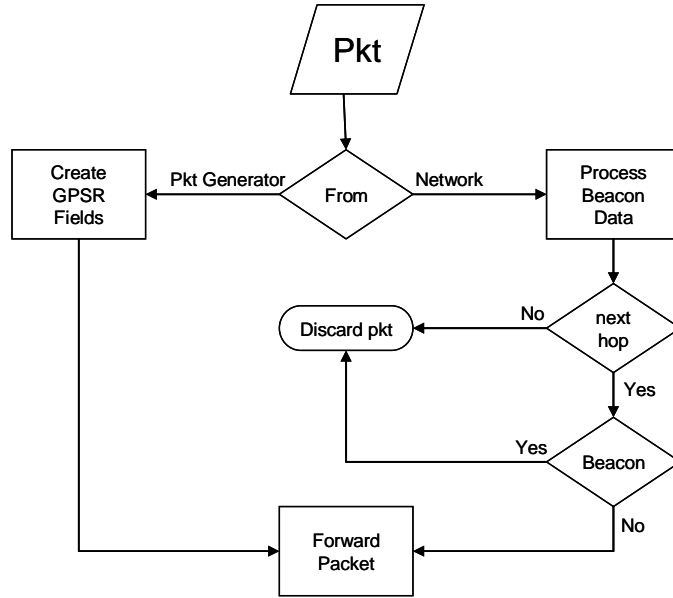
Figure 37. GPSR packet arrival process

Figure 37 shows the initial processing of a packet delivered to GPSR. If the packet was generated at the current node, GPSR fields are created and stored in the IP header options field and the packet forwarding process takes over. Packets arriving from the network are processed for neighbor location information. Overheard data packets addressed

to other nodes and beacon packets are discarded. Data packets which identify the current

node as the next hop are delivered to the packet forwarding process.
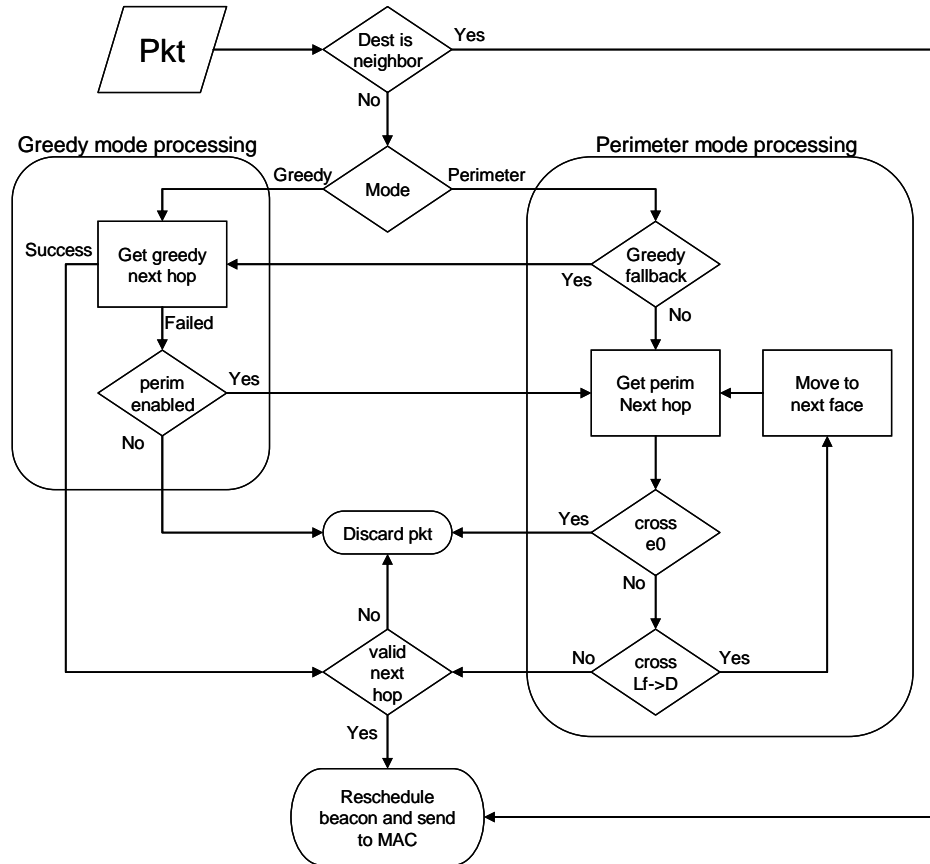
Figure 38. GPSR packet forwarding process

The GPSR packet forwarding process is outlined in Figure 38. Before any

processing is done, the neighbor table is checked to determine if the packet's destination is a

known neighbor. If so, the destination address is recorded in the packet's next hop field and

it is sent to the MAC for transmission. All other packets require further processing

depending on the mode in which the packet was delivered to the current node. Packets are

examined after perimeter and greedy mode processing for a valid next-hop address; if for

any reason the next-hop address is not a valid IP address, the packet is discarded.

- Greedy mode processing

    1. The neighbor table is searched for a greedy next hop neighbor. If successful, the packet is sent to the MAC for transmission.

    2. If no greedy next hop neighbor is found and perimeter mode is disabled, the packet is discarded.

    3. If perimeter mode is enabled, the packet is placed in perimeter mode and processing continues perimeter mode step 2.

- Perimeter mode processing

    1. The location of the current node is compared to the Lp field in the packet header (location the packet entered perimeter mode). If the current node is closer, the packet is returned to greedy mode and processing begins at greedy mode step 1.

    2. The neighbor table is planarized and a perimeter-mode next hop node is computed.

    3. If the edge between the current node and the next hop location crosses e0, a routing loop has been encountered and the packet is discarded.

    4. If the edge between the current node and the next hop location crosses the line between Lf (the point at which the packet entered the current face), the packet is marked for routing around the next face and returned to perimeter mode step 2.

    5. Otherwise the packet is sent to the MAC for transmission.

Beacon packets are scheduled using a self interrupt. Each time a data packet is transmitted, the next beacon interrupt is rescheduled; this ensures excess beacon packets are not transmitted since neighbors will use the overheard data packets as beacons.

All GPSR-specific header information is stored in the *options* field of the IP header; this ensures all GPSR-specific data is maintained with every packet segment if packets are fragmented by IP. For efficiency reasons and in order to fit all data required by GPSR into the space available in the *options* field, some transformations and creative data storage must take place. GPSR header information is presented in Table 10.

Table 10. GPSR header format

0		31
last_hop (IP address)		
source_loc (point)		
dest_loc (point)		
e0_a	e0_b	
Lp (point)		
Lf (point)		
mode (int8)		

Fields of type point consist of two unsigned 16-bit integers that represent the X and Y coordinates of the represented location within the network boundary. The network space is divided into a 2^{16} by 2^{16} grid, and locations are stored as the coordinates in that grid. Absolute coordinates supplied by OPNET (for this study, between 0 and 10,000) are converted to GPSR coordinates (between 0 and 65,536) for use by the GPSR process; this affords approximately 15-centimeter precision in storing node locations which is an acceptable level of precision given transmission ranges measured in kilometers.

The *e0* fields, normally the IP address of the two nodes on either end of the first edge of a tour around a face during perimeter routing, are stored using only the 16 least

significant bits of the IP address. The 16 most significant bits are copied from the *last_hop* field when needed.

The *mode* field is used to specify whether the packet is a beacon, perimeter mode or greedy mode packet. Beacon packets use only the *source_loc* and *mode* fields, using 40 bits of header. Greedy packets add *last_hop,* and *dest_loc* fields for a total of 104 bits. Perimeter-mode packets use all fields and require 200 bits of header.

The GPSR specification published in [KaK00] discusses several protocol optimizations which were implemented by the authors. Two of these optimizations are not implemented in the OPNET GPSR model: support for MAC-layer failure feedback and interface queue traversal.

Support for MAC-layer feedback involves notifying the GPSR process when WLAN MAC has been unable to successfully transmit a frame, indicating that the intended destination node has left radio range. GPSR then uses that information to expire that node's entry in the neighbor table before reaching the expiration time and prevents GPSR from attempting further transmissions to that node.

Interface queue traversal is dependent on the MAC-layer feedback optimization and involves removing all packets from the WLAN packet buffer which are addressed to a node which has encountered a transmission failure, passing all of the packets back to GPSR for reprocessing.

While the authors describe that these optimizations had "a profound effect" on their results, their implementation in OPNET is nontrivial and violates the separation principal of the OSI model. Furthermore, the other routing protocols under evaluation do not benefit

from such optimizations in their OPNET implementations, therefore utilizing them for GPSR only would prevent a fair comparison between protocols.

C.2 OPNET Mobility Manager

The OPNET simulator has a built-in mobility management model called the *random_mobility_cfg* node model [Opn06e]. This node is placed in the network and is used to conand manage random node mobility of UAV nodes. The mobility manager is configured to follow the random waypoint mobility model, choosing waypoints between (0, 0) and (10,000, 10,000) and a fixed node velocity of 25 meters per second with fixed 0-second pauses time between waypoints. Mobility begins after one second of simulation to allow all nodes to initialize, and continues through the end of simulation.

Appendix D–OPNET Distributed Simulation Execution

OPNET Modeler 12.0 adds a new feature which allows the distribution of simulations over multiple workstations [Opn06c]. This capability even permits a single multi-core or multi-processor workstation to simultaneously execute multiple simulations. All simulation parameters and results are passed over the network and are available on the host workstation upon simulation completion, just as if all simulations were conducted locally.

The host workstation must be configured as follows to support distributed simulations:

- The following steps require administrative permissions
    - Share the folder containing all project-related model files
    - Allow TCP & UDP Port 7007 and OPNET Modeler 12.0 access in the Windows Firewall

- The following steps can be completed by any user
    - Set the *des.distributed_mode* preference to TRUE
    - Specify in the *des.distributed_server_host_info* preference the list of workstations that simulations can be distributed to, following the format "workstation_name:port_number:num_of_simulations" where port_number is the port the DES server will be listening on (default is 7007) and number_of_simulations is the number of simultaneous simulations the workstation can support, typically up to the number of processor cores within the workstation

In order to properly coneach workstation to act as a Distributed Event Simulation (DES) server, the following tasks are performed:

- As administrator

  o Install OPNET 12.0 Modeler, Models and Documentation

  o Give Domain Users full access rights to /Program Files/OPNET/12.0.A/models

  o Allow TCP & UDP Port 7007 and OPNET Modeler 12.0 access in the Windows Firewall

- As any user

  o Map a drive to the shared folder on the host workstation

  o Add the mapped drive to the OPNET model directories preference (include subdirectories)

  o Copy the modified header files to the models/std/include/ folder on the workstation, overwriting the existing header files

  o Run op_des_server.exe (also on the host workstation)

When a simulation set consisting of multiple simulation runs is executed on the host workstation, it will distribute one simulation to each processor specified in the *des.distributed_server_host_info* preference, starting at the top. It is advised not to run a simulation on every core in the host workstation to allow some processor availability to be dedicated to management of the shared folder and collating and storing statistics from the DES server workstations.

Bibliography

[AuM05]   C. J. Augeri and B. E. Mullins, "Harvesting Information in Distributed Sensor Networks," Unpublished report, Air Force Institute of Technology, July 6, 2005.

[Bel58]   R. Bellman, "On a Routing Problem," in *Quarterly of Applied Mathematics*, 16(1), pp.87-90, 1958.

[BJM04]   J. Broch, D. B. Johnson, and D. Maltz, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks," IETF Internet Draft, draft-ietf-manet-dsr-01.txt, July 2004 (work in progress).

[Bla02]   H. Blackshear, Jr., "Developing a Conceptual Unmanned Aerial Vehicle Communications Mobile Ad Hoc Network Simulation Model," vol. AD-A404703, 2002.

[BMJ98]   J. Broch, D. A. Maltz, D. B. Johnson, Y. Hu and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols," in *Proceedings of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking,* pp. 85-97, 1998.

[Bre97]   P. Brenner, "A Technical Tutorial on the IEEE 802.11 Protocol," Breezecom Wireless Communications Publications, http://www.sss-mag.com/pdf/802_11tut.pdf, 1997.

[CBD02]   T. Camp, J. Boleng and V. Davies, "A survey of mobility models for ad hoc network research," *Wireless Communications and Mobile Computing*, vol. 2, pp. 483-502, 2002.

[ClJ03]   T. Clausen and P. Jacquet, "Optimized Link State Routing Protocol (OLSR)," RFC 3626, IETF Network Working Group, 2003.

[DoD01]   Department of Defense (DoD), *Department of Defense Dictionary of Military and Associated Terms,* Joint Publication 1-02, Washington DC: HQ DoD, April 2001 (as amended through 14 April 2006).

[GaS69]   K. Gabriel and R. Sokal, "A new statistical approach to geographic variation analysis," in *Systematic Zoology,* vol. 18, pp. 249-278, 1969.

[GGL01]   D. L. Gu, M. Gerla, H. Ly, K. Xu, J. Kong and X. Hong, "Design of multilevel heterogeneous ad hoc wireless networks with UAVs," in *Proceedings of SPIE Wireless and Mobile Communications Conference,* pp. 327-338, 2001.

[GPL00]   D. L. Gu, G. Pei, H. Ly, M. Gerla, B. Zhang and X. Hong, "UAV aided intelligent routing for ad hoc wireless network in single-area theater," in *IEEE Wireless Communications and Networking Conference,* pp. 1220-1225 vol.3, 2000.

[GuK00]   P. Gupta and P. R. Kumar, "The capacity of wireless networks," in *IEEE Transactions on Information Theory*, vol. 46, pp. 388-404, 2000.

[HiL06]   T. Hill and P. Lewicki. *Statistics: Methods and Applications.* Tulsa, OK: StatSoft, 2006.

[IEE03a]   IEEE Computer Society LAN MAN Standards Committee, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," IEEE Std. 802.11-1999 (R2003), 2003.

[IEE03b]   IEEE Computer Society LAN MAN Standards Committee, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Higher-Speed Physical Layer Extension in the 2.4 GHz Band," IEEE Std. 802.11b-1999 (R2003), 2003.

[ISO94]   ISO/IEC 7498-1, Information Technology—Open Systems Interconnection—Basic Reference Model: The Basic Model, 1994.

[JMC01]   P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, L. Viennot, "Optimized link state routing protocol for ad hoc networks," in *Technology for the 21st Century*, pp. 62- 68, 2001.

[JoM96]   D. B. Johnson and D. A. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks," *Mobile Computing,* T. Imielinski and H. Korth, Eds., Kluwer, pp. 153-181, 1996.

[JPS01]   R. Jain, A. Puri and R. Sengupta, "Geographical routing using partial information for wireless ad hoc networks," *IEEE Personal Communications,* vol. 8, pp. 48-57, 2001.

[KaK00]   B. Karp and H. T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in *MobiCom '00: Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, pp. 243-254, 2000.

[KuR05]   J. Kurose and K. Ross. *Computer Networking: A Top-Down Approach Featuring the Internet* (3rd Edition). Boston: Pearson Education, 2005.

[LAN03]   C. A. Lua, K. Altenburg and K. E. Nygard, "Synchronized multi-point attack by autonomous reactive vehicles with simple local communication," in *Proceedings of the 2003 IEEE Swarm Intelligence Symposium,* pp. 95-102, 2003.

[LGS04]   J. Li, L. Gewali, H. Selvaraj and V. Muthukumar, "Hybrid greedy/face routing for ad hoc sensor network," in *Euromicro Symposium of Digital System Design*, pp. 574-578, 2004.

[LMN02]   A. Laouti, P. Muhlethaler, A. Najid and E. Plakoo, "Simulation Results of the OLSR Routing Protocol for Wireless Network," Med-Hoc-Net, 2002.

[Mis99]   P. Misra (1999), "Routing Protocols for Ad Hoc Mobile Wireless Networks," Retrieved June 15, 2006 from http://www.cse.wustl.edu/~jain/cis788-99/ftp/adhoc_routing/.

[MMP06]   K. M. Morris, B. E. Mullins, D. J. Pack, G. W. P. York, and R. O. Baldwin, "Impact of Limited Communications on a Cooperative Search Algorithm for Multiple UAVs," *IEEE International Conference on Networking, Sensing and Control* (submitted), 2006.

[MuG96]   S. Murthy and J. J. Garcia-Luna-Aceves, "An efficient routing protocol for wireless networks," in *ACM Mobile Networks and Applications*, vol. 1, pp. 183-197, 1996.

[MWH01]   M. Mauve, A. Widmer and H. Hartenstein, "A survey on position-based routing in mobile ad hoc networks," in *IEEE Network,* vol. 15, pp. 30-39, 2001.

[Opn06a]   OPNET Documentation Team, "Modeling Wireless Networks," *Wireless Module User Guide,* OPNET Modeler 12.0.A Product Documentation, 2006.

[Opn06b]   OPNET Documentation Team, "Radio Transceiver Pipeline," *Wireless Module User Guide,* OPNET Modeler 12.0.A Product Documentation, 2006.

[Opn06c]   OPNET Documentation Team, "Discrete Event Simulation," *OPNET Editors Reference,* OPNET Modeler 12.0.A Product Documentation, 2006.

[Opn06d]   OPNET Documentation Team, "Understanding MANET Model Internals and Interfaces," *Defesnse Solutions*, OPNETWork 2006 Proceedings, Session 1941, 2006.

[Opn06e]   OPNET Documentation Team, "Modeling Node and Subnetwork Movement," *Wireless Module User Guide*, OPNET Modeler 12.0.A Product Documentation, 2006.

[PeB94]   C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," in *SIGCOMM '94: Proceedings of the Conference on Communications Architectures, Protocols and Applications*, pp. 234-244, 1994.

[PeR99]   C. E. Perkins and E. M. Royer, "Ad hoc on-demand distance vector routing," in *Proceedings of the Second IEEE Workshop on Mobile Computing Systems and Applications*, pp. 90-100, 1999.

[RoT99]   E. M. Royer and Chai-Keong Toh, "A review of current routing protocols for ad hoc mobile wireless networks," *IEEE Personal Communications*, vol. 6, pp. 46-55, 1999.

[RRP03]   A. Rao, S. Ratnasamy, C. Papadimitriou, S. Shenker and I. Stoica, "Geographic routing without location information," in *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking,* pp. 96-108, 2003.

[Ton06]   A. Tønneson (2004), "Mobile Ad-Hoc Networks," Retrieved 21 January, 2006 from http://www.olsr.org/docs/wos3-olsr.pdf.

[Tou80]   G. Toussaint, "The relative neighborhood graph of a finite planar set," in *Pattern Recognition* vol. 12, pp. 261-268, 1980.

[USA05]   United States Air Force, *The U.S. Air Force Remotely Piloted Aircraft and Unmanned Aerial Vehicle Strategic Vision*, 2005.

[Wik06]   Wikipedia (2006), "OSI model," Retrieved May 3, 2006 from http://en.wikipedia.org/wiki/Osi_model.

[Wik07a]   Wikipedia (2007), "Exposed terminal problem," Retrieved January 28, 2007 from http://en.wikipedia.org/wiki/Exposed_terminal_problem.

[Wik07b]   Wikipedia (2007), "Synergy," Retrieved January 29, 2007 from http://en.wikipedia.org/wiki/Synergy.

[XuK04]   F. Xue and P. R. Kumar, "The number of neighbors needed for connectivity of wireless networks," in *ACM Wireless Networks*, vol. 10, no. 2, pp. 169-181, 2004.

[YGE01]   Y. Yu, R. Govindan and D. Estrin, "Geographical and energy aware routing: A recursive data dissemination protocol for wireless sensor networks," UCLA, Los Angeles, CA, Tech. Rep. UCLA/CSD-TR-01-0023, 08/14/2001, 2001.

[YPH06]   G. W. P. York, D. J. Pack, and J. Harder, "Comparison of Cooperative Search Algorithms for Mobile RF Targets using Multiple Unmanned Aerial Vehicles." Invited chapter in *Cooperative Control and Optimization,* University of Florida Press, Gainesville, 2006.

Vita

    Captain Matthew T. Hyland was born on Long Island, New York in 1979 and graduated from Connetquot High School in 1997. He entered undergraduate studies at Cornell University in Ithaca, New York where he graduated with a Bachelor of Science degree in Computer Science in December 2001 when he was also commissioned as a Second Lieutenant through Detachment 520 of the Air Force Reserve Officer Training Corps.

    His first assignment was at Dover Air Force Base, Delaware as a Deputy Flight Commander in the 436th Communications Squadron. In April 2005, he was selected to be an Executive Officer for the 436th Airlift Wing Commander. In August 2006, he entered the Graduate School of Engineering and Management, Air Force Institute of Technology where he was inducted as a member of the Tau Beta Pi engineering honor society. Upon graduation, he will complete a three-month internship with the Defense Advanced Projects Research Agency in Arlington, Virginia, after which he will be assigned to the 70th Intelligence Support Squadron at Fort George Meade, Maryland.

1. REPORT DATE (DD-MM-YYYY)	2. REPORT TYPE	3. DATES COVERED (From – To)
22-03-2007	Master's Thesis	March 2006 – March 2007

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Performance Evaluation of Ad Hoc Routing Protocols in a Swarm of Autonomous Unmanned Aerial Vehicles	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)	5d. PROJECT NUMBER
Hyland, Matthew T., Captain, USAF	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)	8. PERFORMING ORGANIZATION REPORT NUMBER
Air Force Institute of Technology Graduate School of Engineering and Management (AFIT/EN) 2950 Hobson Way WPAFB OH 45433-7765	AFIT/GCS/ENG/07-07

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This thesis investigates the performance of three mobile ad hoc routing protocols in the context of a swarm of autonomous unmanned aerial vehicles (UAVs). It is proposed that a wireless network of nodes having an average of $5.1774 \log n$ neighbors, where n is the total number of nodes in the network, has a high probability of having no partitions. By decreasing transmission range while ensuring network connectivity, and implementing multi-hop routing between nodes, spatial multiplexing is exploited whereby multiple pairs of nodes simultaneously transmit on the same channel.

The proposal is evaluated using the Greedy Perimeter Stateless Routing (GPSR), Optimized Link State Routing (OLSR), and Ad hoc On-demand Distance Vector (AODV) routing protocols in the context of a swarm of UAVs using the OPNET network simulation tool. The first-known implementation of GPSR in OPNET is constructed, and routing performance is observed when routing protocol, number of nodes, transmission range, and traffic workload are varied. Performance is evaluated based on proportion of packets successfully delivered, average packet hop count, and average end-to-end delay of packets received.

Results indicate that the routing protocol choice has a significant impact on routing performance. While GPSR successfully delivers 50% more packets than OLSR, and experiences a 53% smaller end-to-end delay than AODV when routing packets in a swarm of UAVs, increasing transmission range and using direct transmission to destination nodes with no routing results in a level of performance not achieved using any of the routing protocols evaluated.

15. SUBJECT TERMS
Computer Networks, Communication protocols, Wireless communications, Performance analysis

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Barry E. Mullins, Ph. D. (ENG)
			UU	118	19b. TELEPHONE NUMBER (Include area code)
U	U	U			(937) 255-3636, ext 4916 (barry.mullins@afit.edu)