2018-06

STIGMERGIC CONTROL OF DUAL DIRECTION COMMUNICATION FERRY NODES FOR DENIED COMMUNICATION ENVIRONMENTS

Hietpas, Andrew

Monterey, CA; Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

STIGMERGIC CONTROL OF DUAL DIRECTION COMMUNICATION FERRY NODES FOR DENIED COMMUNICATION ENVIRONMENTS

by

Andrew Hietpas

June 2018

Thesis Advisor: Duane T. Davis
Co-Advisors: John H. Gibson
Robert Hunjet

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			*Form Approved OMB No. 0704-0188*

1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** June 2018	**3. REPORT TYPE AND DATES COVERED** Master's thesis

4. TITLE AND SUBTITLE STIGMERGIC CONTROL OF DUAL DIRECTION COMMUNICATION FERRY NODES FOR DENIED COMMUNICATION ENVIRONMENTS	**5. FUNDING NUMBERS**
6. AUTHOR(S) Andrew Hietpas and Andrew Hietpas	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) I Marine Expeditionary Force, Camp Pendleton, CA 92014	**10. SPONSORING / MONITORING AGENCY REPORT NUMBER**

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release. Distribution is unlimited.	**12b. DISTRIBUTION CODE** A

13. ABSTRACT (maximum 200 words)

For the past decade, the Marine Corps has been able to assume uncontested use of the electromagnetic spectrum, and its command and control systems reflect this assumption. It must prepare for when this is not the case by developing a system designed to operate in contested electromagnetic environments. Previous work by Australian Defence Science and Technology Group researchers on delay-tolerant networking using a UAV swarm as communication ferry nodes provides one solution. They achieve even dispersion through speed adjustments on unidirectional data ferries traversing the polygon formed by ground node locations based on digital pheromone values. This thesis builds upon their work by developing a method to evenly disperse ferry nodes traveling both clockwise and counterclockwise without inter-UAV communication. This reduces the primary disadvantage of delay tolerant networking, the message delivery delay. Unlike the single-direction system, this even dispersion is periodic because the ferry nodes approach one another after reaching even dispersion before reaching even dispersion once again after passing. Due to this, only one ground node in the system should link the two directional pheromones. We found that multiple adjusting ground nodes attempt to pull the system into contradictory equilibrium states. We have verified this concept in simulation and integrated it with the NPS ARSENL program for future testing.

14. SUBJECT TERMS swarming, UAVs, stigmergy, command and control, delay tolerant networking, swarm intelligence, emergence, data ferrying, digital pheromones	**15. NUMBER OF PAGES** 175
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	**18. SECURITY CLASSIFICATION OF THIS PAGE**	**19. SECURITY CLASSIFICATION OF ABSTRACT**	**20. LIMITATION OF ABSTRACT**
Unclassified	Unclassified	Unclassified	UU

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release. Distribution is unlimited.

STIGMERGIC CONTROL OF DUAL DIRECTION COMMUNICATION FERRY NODES FOR DENIED COMMUNICATION ENVIRONMENTS

Andrew Hietpas
Major, United States Marine Corps
BA, Texas A & M University, 2005

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN INFORMATION WARFARE SYSTEMS ENGINEERING
AND
MASTER OF SCIENCE IN INFORMATION TECHNOLOGY MANAGEMENT

from the

NAVAL POSTGRADUATE SCHOOL
June 2018

Approved by: Duane T. Davis
Advisor

John H. Gibson
Co-Advisor

Robert Hunjet, Australian Defence Science and Technology Group
Co-Advisor

Dan C. Boger
Chair, Department of Information Sciences

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

For the past decade, the Marine Corps has been able to assume uncontested use of the electromagnetic spectrum, and its command and control systems reflect this assumption. It must prepare for when this is not the case by developing a system designed to operate in contested electromagnetic environments. Previous work by Australian Defence Science and Technology Group researchers on delay-tolerant networking using a UAV swarm as communication ferry nodes provides one solution. They achieve even dispersion through speed adjustments on unidirectional data ferries traversing the polygon formed by ground node locations based on digital pheromone values. This thesis builds upon their work by developing a method to evenly disperse ferry nodes traveling both clockwise and counterclockwise without inter-UAV communication. This reduces the primary disadvantage of delay tolerant networking, the message delivery delay. Unlike the single-direction system, this even dispersion is periodic because the ferry nodes approach one another after reaching even dispersion before reaching even dispersion once again after passing. Due to this, only one ground node in the system should link the two directional pheromones. We found that multiple adjusting ground nodes attempt to pull the system into contradictory equilibrium states. We have verified this concept in simulation and integrated it with the NPS ARSENL program for future testing.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

ANOVA	Analysis of Variance
ARSENL	Advanced Robotic Systems Engineering Laboratory
DARPA	Defense Advanced Research Projects Agency
DSTG	Defence Science and Technology Group
DTN	Delay-Tolerant Networking
HAIL	Hardware Abstraction and Integration Layer
HRL	Hughes Research Laboratories
IR	Infrared
MASON	Multi-Agent Simulator of Networks
NPS	Naval Postgraduate School
OFFSET	Offensive Swarm-Enabled Tactics
RC	Radio Control
RF	Radio Frequency
ROS	Robot Operating System
SCADS	Self-Organising Communications and Autonomous Delivery Service
SITL	Software-in-the-Loop
UAV	Unmanned Aerial Vehicle
UGS	Unmanned Ground System

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

For someone without a technical undergraduate degree, this thesis required considerable help and advice. Thank you to Dr. Rob Hunjet of the Australian Defence Science and Technology Group for accepting a random, foreign master's student who approached him during one of his visits to Monterey, CA. Through his ideas and prompting, I was able to learn more and take this idea further than I expected.

Thank you to Dr. Duane Davis for taking on a non-Computer Science thesis student. Through his tolerance of my many office calls, which were required for me to get my mind around the ARSENL architecture and his previous work, we were able to merge the many concepts required for this effort. Without his knowledge, tolerance, and willingness to write the HAIL Bridge, I would have been unable to complete this effort.

This thesis came about through John Gibson's willingness to link everyone together. Thank you for all of the work you did to make the international cooperation side of this thesis transparent to me. This allowed me to focus my attention on the technical side, where it was required.

Brad Fraser's previous work provided the launching point for this thesis. Thank you for your willingness to share your code and thoughts throughout this effort. Without your answers to my many questions, no useful results would have come from this thesis.

Tom Stevens' knowledge was essential to incorporating HAIL into this thesis. He not only assisted through many e-mails and phone calls but also was willing to meet in one location, which saved me countless hours when we linked HAIL to ARSENL. He provided the technical knowledge, and I caught the missing commas.

None of this would have been possible without the love and support of my girls at home. My wife, Sherry, is a true inspiration who tolerated the time away, encouraged me at the low points, and loved me throughout. My girls, Anna, Emily, and Rachel, were always there with hugs and laughter to improve morale at every turn. Thank you, girls.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. PROBLEM STATEMENT

The United States military, and the militaries of its partner nations, must plan to operate in a contested physical and radio frequency (RF) environment. Unfortunately, current systems were not designed from the beginning to operate in this environment making communication and shared, situational awareness difficult. The Australian Defence Science and Technology Group (DSTG) has, over the previous years, conducted research addressing this problem through the use of swarming communication ferry nodes whose global behavior is modified with digital pheromones in ground nodes (Fraser & Hunjet, 2016). This promising concept has thus far focused on aerial ferry nodes sequentially servicing ground nodes at the vertices of a polygon by traveling in one direction around its perimeter.

Information is only useful if it arrives at the necessary destination in a timely manner. Towards this end, extending the DSTG concept to use ferry nodes traveling in multiple directions might reduce the message delay times. If the problem of contested communications remains unsolved, the United States military will be unable to leverage all of its capabilities in the dynamic environments of future conflict. To continue addressing this recognized requirement, this thesis will explore the use of stigmergic control of an aerial swarm to realize multi-directional data-ferry swarms. This capability may help the combined force reach the goal of reliable communication in a contested environment by providing a control mechanism for communication ferry nodes without the requirement for end-to-end communication.

B. BACKGROUND

Over the past decade, operations in permissive environments have demonstrated the capabilities of America and her partners when they have effective communication. However, end-to-end communication is not guaranteed on the battlefields of tomorrow. This means that we must either sacrifice capability or develop systems designed to operate in this uncertain environment. Many organizations have sought to leverage the possibilities

brought by swarming unmanned aerial vehicles (UAV). This thesis merges the efforts of two of these organizations to develop a concept that may be of use for future Marine Corps operations. The first of these organizations is the Naval Postgraduate School's (NPS) Advanced Robotic Systems Engineering Laboratory (ARSENL) program. They have steadily built autonomous swarm capability culminating in the simultaneous flight of a swarm of 50 UAVs (Chung et al., 2016). The second organization is the Australian DSTG, which has conducted research in controlling a UAV swarm using digital pheromones to allow communication ferry nodes to operate in a contested RF environment. This thesis builds on this previous work in communication ferrying by providing a mechanism to coordinate bi-directional UAV ferry nodes, sometimes referred to as data mules. Bi-directional data ferrying, as described in the third chapter, can facilitate friendly force communication while reducing message delay inherent in a single-direction system.

C. THESIS OVERVIEW

1. Research Question

Can stigmergic control mechanisms provide evenly distributed dual-direction, ferry UAVs while operating in a radio frequency denied environment?

2. Scope

This thesis focuses on the coordinating algorithm for a dual-direction ferry node swarm and develops a mathematical coupling for dual-direction ferry nodes. This algorithm results in a swarm behavioral change from closely grouped UAVs after launch, as shown in Figure 1, to evenly dispersed UAVs, as shown in Figure 2, without inter-UAV communication. In these figures, the red and green squares represent the ground nodes, the blue squares represent the forward traveling, or prograde, ferry nodes, and the black squares represent the reverse traveling, or retrograde, ferry nodes.

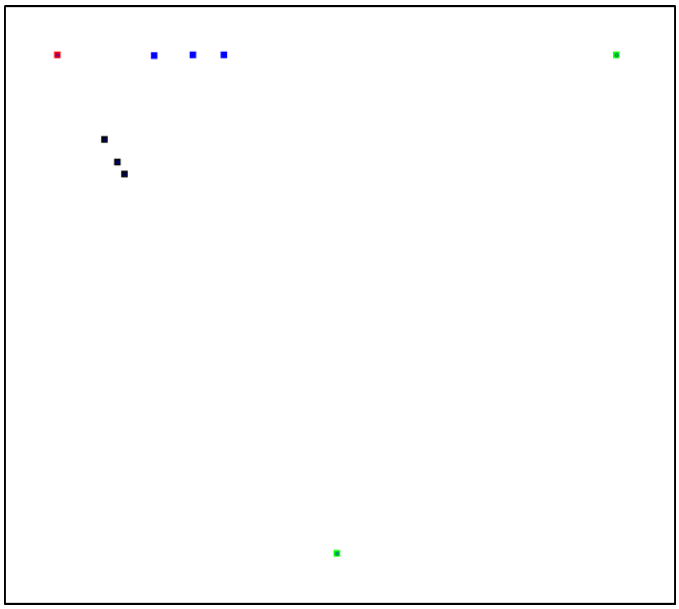


Figure 1. Undispersed ferry nodes immediately after launch

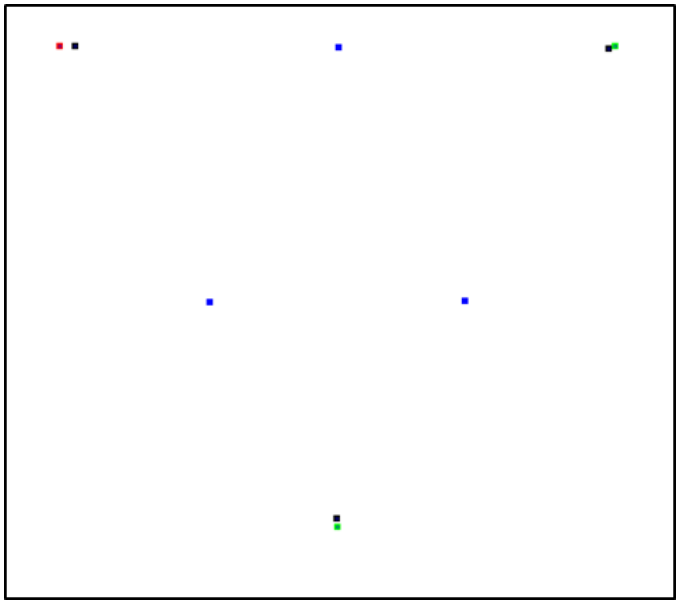


Figure 2. Evenly dispersed ferry nodes after implementing stigmergic control

This coupling is tested through simulation with various ground node layouts ranging from three to ten ground nodes. We use the information from multiple configurations at each layout to determine a universal coupling algorithm. This thesis then discusses the integration of this algorithm into the NPS ARSENL swarm. This integration

is tested using the ARSENL Software-in-the-Loop (SITL) simulation capability with live-fly experiments to verify SITL results scheduled for June 2018.

Since the focus of this thesis is on the control mechanisms for swarming UAVs used as communication ferry nodes, it does not discuss or analyze the various radios that could be used in this effort. It also does not analyze the amount of traffic that could be sent between ground and ferry nodes. Finally, it does not discuss the platform type that would best suit the ferry node role. The platform used for emulation is the NPS ARSENL Zephyr II because of availability and the ability to leverage previous work by NPS (Chung et al., 2016).

3. Thesis Stakeholders

The resulting information from this thesis will benefit two different organizations. The first is the Australian DSTG, which is currently seeking to develop the Self-Organising Communications and Autonomous Delivery Service (SCADS) program (Hunjet et al., 2017). As part of this program, DSTG employs the stigmergic swarming concept discussed in this thesis. This thesis extends and improves on this concept, coordinating dual-direction ferry nodes while also providing additional information about employing these mechanisms on a fixed wing swarm. This will assist in the overall development of the program.

The NPS ARSENL program is the second organization that will benefit from this thesis. Thus far, they have used their swarming technique to fly 50 UAVs simultaneously (Chung et al., 2016), fly UAV swarms against one another in adversarial scenarios (Ochoa, 2016), and employ UAVs in a coordinated search effort (Lau, 2015). The results of this thesis would provide yet another use for their swarming algorithms and techniques. As the ARSENL program gathers more uses for their swarming techniques, they will be better prepared to develop unique UAV solutions for future mission sets.

4. Thesis Main Findings

This thesis developed five key findings. First, we demonstrated even dispersion of dual-direction ferry nodes by applying adjustments based on the residual value of the other

direction's pheromone value without inter-vehicle communication. Second, we determined that a dual-direction system requires different types of ground nodes. Only one of the ground nodes in the system should couple the directional pheromones, and the others only reset the pheromone values. More than one adjusting ground node attempts to pull the system into contradictory equilibrium states. Third, we showed that individual ground node layouts have a unique, optimal combination of parameters, but there is a specific adjustment value which results in evenly distributed ferry nodes for all ground node layouts. Fourth, we determined that the system converged the fastest when the pheromone values were unbounded. Finally, we verified the hypothesis that a dual-direction system would result in overall faster message delivery times than a single-direction system.

D. CHAPTER SUMMARY

This thesis is broken into five chapters. Following this introduction, the second chapter includes a literature review that discusses the concept of communication ferrying and command and control mechanisms for swarming robotics. This same chapter also includes a review of previous work done by DSTG with communication ferrying and UAV swarming by the NPS ARSENL program. The third chapter discusses the developed dual-direction algorithm and its integration it into the ARSENL program. The fourth chapter discusses the results of the simulation conducted on this dual-direction algorithm. It analyzes how the pheromone coupling was successful in ground node layouts ranging from three to ten nodes. Finally, the fifth chapter reviews the primary findings of this thesis before discussing options for future work that will help mature the concept into an operational system.

THIS PAGE INTENTIONALLY LEFT BLANK

II. LITERATURE REVIEW

A. CHAPTER OVERVIEW

This literature review encompasses two broad sections. The first is the technological underpinning of communication ferrying. This section includes a discussion of UAV types, UAV swarms and their qualities, and UAV swarm control mechanisms. The second section includes an overview of the software and equipment used in this thesis to test the dual-direction communication ferry technique. Combined, these sections describe the previous work in UAV swarm operations and communication ferrying this thesis will extend.

B. COMMUNICATION FERRYING

The use of UAV swarms for communication ferrying is based upon two conceptual components. The first is communication ferrying. As defined in Marine Corps Doctrinal Publication 6, "communications are any method or means of conveying information from one person or place to another to improve understanding" (United States Marine Corps [USMC], 1996, p. 94). This understanding provides the commander with the ability to conduct command and control. Command and control is the process through which the commander can understand the situation and communicate actions to different elements (USMC, 1996). The idea of information ferrying has its root in messengers, which were one of the first means used for distant military communication. Historical figures such as Alexander the Great, Hannibal, and Caesar used systems of messengers who could quickly bring information from one location to another when direct communication was not possible ("Military Communication," n.d.). While messengers are still used today, RF technology has provided direct end-to-end communication ("Military Communication," n.d.).

As shown by the Russian electronic warfare capabilities currently in use in Ukraine, the RF medium is contested space (Sukhankin, 2017). Bradley Fraser and Robert Hunjet point out that enemy activity is not the only impediment to effective communication. The geography of the area of operations may also prevent communication. Further, the RF

environment might be jammed by friendly forces (Fraser & Hunjet, 2016). They go on to recommend the use of communication ferries in these situations. In their concept, information is transferred from one location to another by UAV. A group from Raytheon (Usbeck et al., 2014) demonstrated this concept on the ground in 2014. Using a Raspberry Pi and Wave Relay MPU4 tactical radios, they showed that a communication ferry could link "participants in remote operations in which communication infrastructure is not available, inaccessible, or unreliable" (Usbeck et al., 2014, p. 3-5). These characteristics embody the communication infrastructure for most military operations, especially those in new locations. Also in 2014, a group from Brazil demonstrated through simulation the use of data ferrying to connect disjointed parts of a sensor network (Heimfarth & De Araujo, 2014). The concept of data ferrying precedes many of the communication means with which we are more familiar but has been demonstrated in both simulation and real world applications to remain viable using modern technology.

C. DELAY-TOLERANT NETWORKING

Delay-tolerant networking (DTN) is the concept that elements of a network can function without instantaneous communication. Communication ferrying, when combined with DTN, removes the need to maintain continuous connection between nodes of a network (Hunjet et al., 2018). The underlying concept with this approach is that "delay-tolerant traffic does not carry an urgency and only eventual, reliable reception is important" (Henkel & Brown, p. 1). Henkel and Brown use the term delay-tolerant for messages that do not require immediate delivery, but the concept can also be applied to denied communications environments in which the message cannot be sent immediately or directly. Military forces can thus plan for the receipt of information at predictable times, and they can plan operations around this constraint. DTN provides a resilient construct for operations in environments where end-to-end communication cannot be expected.

D. UNMANNED AERIAL VEHICLES

Despite the varying size and complexity, each UAV system requires the same capabilities to operate. They are the air vehicle, the mission planning and control station, the launch and recovery equipment, the payload, the data link, and the ground support

equipment (Fahlstrom, 2012). Depending upon the size of the system, some of these components are consolidated into the same piece of equipment, or an external item serves their purpose. For example, the launch and recovery equipment for a very small UAV could merely be the operator's hand. Nonetheless, all components of the system will be included in this discussion to ensure a thorough analysis. The air vehicle is the component of the system that moves through the air. It is commonly referred to as the UAV. The mission planning and control station is where system information for the UAV and payload are displayed for the operator. For large systems, it can be located in a fixed structure on a different continent than the vehicle, and for small systems, it can consist of a small handheld device (Fahlstrom, 2012). While the exterior of these systems can look different, this component of the system allows the operator to observe the status of, and provide input to, the UAV. For UAV swarms, some functions of the mission planning and control station are handled by the overarching control mechanism. This reduces the workload on the operator, which enables the control of a large number of UAVs. As previously mentioned, the launch and recovery equipment can range from an airfield to a hand, depending upon the size of the UAV (Fahlstrom, 2012). The payload is the primary reason for the UAV system, and it could be a video camera, a weapon, a radio for communication ferry, or other mission-determined capability (Fahlstrom, 2012). The data link consists of both a ground data terminal and an air data terminal. These components create a link between the control station and the UAV to pass guidance information and payload commands to the UAV and sensor data back to the operator (Fahlstrom, 2012). Finally, the ground support equipment provides logistic support to the system ranging from spare parts to testing and maintenance (Fahlstrom, 2012).

1. Unmanned Aerial Vehicle Types

UAVs can be divided into different categories similar to manned platforms: multi-rotor, fixed-wing, single rotor helicopter, fixed-wing hybrid vertical take-off and landing (VTOL), and gliders (Chapman, 2016). Multi-rotor UAVs have the advantage of being inexpensive and easy to use, but they have the disadvantage of minimal time of flight and speed (Chapman, 2016). Fixed-wing UAVs have higher speeds and longer endurance, but they are unable to hover in an area and require more resources to launch and land

(Chapman, 2016). Single-Rotor UAVs can carry more weight than their smaller multi-rotor UAVs, but they require more training for the operators (Chapman, 2016). Fixed-wing hybrid UAVs can both take off vertically and fly like a fixed wing UAV, but they cannot do either as well as standard fixed-wing or single-rotor UAVs (Chapman, 2016). Unmanned gliders do not require fuel and are able to stay aloft for hours, but they are dependent on local weather conditions (Hambling, 2016). Unmanned gliders have recently achieved longer endurance through a soaring algorithm named Autonomous Locator of Thermals (ALOFT) developed by the Naval Research Laboratory that uses the sensors already on the UAV to detect and leverage thermals (Hambling, 2016). The various strengths and weaknesses of different UAV platform types have led to the exploration of heterogeneous groups of UAVs where different platforms' strengths can offset the weaknesses of others. A list of the advantages and disadvantages of each UAV type are shown in Table 1.

Table 1. UAV categories and capabilities. Adapted from Chapman (2016); Hambling (2016).

UAV Type	Advantages	Disadvantages	Typical Uses
Multi-Rotor	-Easy to Use -Inexpensive -Vertical Takeoff -Hovering -Confined Area Operation	-Short Flight Time -Small Payload -Short Range	-Hobbyists -Reconnaissance -Aerial Photography
Fixed-Wing	-Longer flight time -Faster Speed -Longer Range	-Launch and Recovery System -No Hovering -More Training -Expensive	-Reconnaissance -Communication Relay
Single-Rotor	-Longer Flight Time than Multi-Rotor -Faster Speed than Multi-Rotor	-Expensive -More Training	-Resupply -Reconnaissance
Fixed-Wing Hybrid	-Vertical Takeoff -Faster Speed than Rotary Wing UAVs	-Expensive -Slower Speed than Fixed-Wing	-Undetermined
Glider	-Long flight time -Power only for Computer	-Weather dependent -Fixed path	-Undetermined

2. Unmanned Aerial Vehicle Modes of Control

UAV systems have varying modes of control ranging from full teleoperation to full automation. Each type differs from one another based on where information flows and where decisions are made. Full teleoperation control occurs when the ground controller provides all of the inputs as if they were actually on board the aircraft based on the information provided in the control station (Fahlstrom, 2012). Data link latency can lead to significant issues if the data is no longer timely (Fahlstrom, 2012). This concern, coupled with increased onboard processing power, led to the development of autopilot-assisted control. During assisted remote control, the operator still provides input about direction and elevation, that are applied to the UAV while the autopilot provides "… stability and avoiding stalls, spins, and excessive maneuvering loads" (Fahlstrom, 2012, p. 121). The next level of automation is exception control, in which the system controls the UAV over the course of the flight plan and it only notifies the controller if something unexpected, or an exception, occurs (Fahlstrom, 2012). Finally, there is full automation in which the system completes an entire flight plan without outside input (Fahlstrom, 2012). Thus, an unmanned system employing full automation in denied-communication environments must replicate the level of training and experience of a seasoned pilot in software. A key point is that all of these modes of control, except for full automation, require at least one operator for each vehicle. This one-to-one ratio requires significant staffing to supervise the unmanned effort. Of the described control modes, only full automation is appropriate for the denied-communication environment discussed by this thesis because the data link connection cannot be guaranteed.

E. UNMANNED AERIAL VEHICLE SWARMS

Previous work in drone swarms provides insight about possible methods to control a communication ferry swarm. Early on, operators of unmanned systems quickly realized the benefit of multiple systems operating in concert. Traditional UAV control paradigms, however, required individual operators for multi-UAV operations. As discussed previously, this can be problematic. This led to the emergence of swarming as a mechanism for simultaneous control of multiple vehicles. Swarming is defined by Jan Carlo Barca and

Y. Ahmet Sekercioglu in their review of swarm robotics as "a large group of locally interacting individuals with common goals" (Barca & Sekercioglu, 2013, p. 345). Their review provides both an examination of swarming techniques and a detailed list of references. While the size of the swarm can vary based upon the assigned mission, the key part of this definition is that the UAVs are locally interacting. This provides the swarm with the ability to mimic the non-verbal communication that manned aviation relies upon. For this thesis, we will define a swarm as multiple, locally interacting, unmanned systems operating towards a common purpose where the total number of human operators are exceeded by the number of unmanned systems being controlled.

Normal UAV operations have two disadvantages that swarms seek to alleviate. The first, as pointed out by a group from John Hopkins Applied Physics Laboratory, is that groups of unmanned systems will "require as many skilled pilots as there are swarm members" (Bamburger et al., 2006, p. 41). Of note, the use of swarm in this context does not adhere to the definition of swarm robotics set forth in the previous paragraph in that it refers to a multi-UAV operation with individual controllers for each UAV. Ideally, a swarm would allow a single operator to provide the overarching goal for which the swarm would determine and execute the details (Bamberger et al., 2006). This would enable a small team of operators to manage the work of many unmanned systems.

The second disadvantage is that it becomes increasingly difficult to coordinate many vehicles operating in the same area with all of the operators removed from the situation (Parunak, Purcell, & O'Connell, 2002). This problem stems from the data link requirement. With communication delays, operators do not have enough information for safe operations of the unmanned systems in close proximity in dynamic environments. The realization of true swarming will allow locally interacting swarm members to subsume this responsibility and shift decision-making to the location of the information (Parunak et al., 2002).

1. Swarm Qualities

The nature of any particular swarm can be characterized by its components and means of communication. In their swarm review, Barca and Sekercioglu provide three

qualities through which swarms can be compared and evaluated: scalability, robustness, and flexibility (Barca & Sekercioglu, 2013). The Defense Advanced Research Projects Agency (DARPA) recently began a swarming effort called the Offensive Swarm-Enabled Tactics (OFFSET) program in which five qualities were used to evaluate swarming efforts: number, collective complexity, human-swarm interaction, heterogeneity, and agent complexity (Chung, 2017). By combining these two lists, we can develop a consolidated list of six swarm qualities: scalability, robustness, flexibility, human-swarm interaction, heterogeneity, and agent complexity.

First, scalability is a measure of how many units can operate within the system simultaneously, and how easily the system's methodology can be employed in larger swarms (Barca & Sekercioglu, 2013). Scalability aligns reasonably well with DARPA's quality of number (Chung, 2017). Second, robustness is a measure of the system's ability to operate despite failures of system components (Barca & Sekercioglu, 2013). Third, the flexibility of the swarm comes from the possible number and type of formations it can perform (Barca & Sekercioglu, 2013). Depending upon the control type chosen, these formations are sometimes better categorized as behaviors (Fraser & Hunjet, 2016). Flexibility combined with robustness match DARPA's quality of collective complexity (Chung, 2017). Fourth, human-swarm interaction defines how the guidance is passed from the human controller to the swarm (Chung, 2017, 3). Fifth, heterogeneity describes whether the swarm can operate with different types of vehicles or must operate with all air vehicles of the same type (Chung, 2017). Finally, agent complexity is a measure of how many different functions each vehicle can perform (Chung, 2017). With these qualities in mind, swarm efforts can be evaluated while also providing a framework to evaluate what swarm is most appropriate for a given situation.

2. Swarm Control Methods

This review will examine human-swarm interaction through a comparison of UAV control methods. Before beginning, we will discuss a terminology discrepancy between command and control theory and swarm literature that could cause some confusion. Barca and Sekercioglu refer to two broad categories: centralized, where a central planner controls

the swarm's activities, and decentralized, where control arises from the interactions of the elements (Barca & Sekercioglu, 2013). As defined in *On Distributed Communications*, decentralized command and control arises when additional layers are added to centralized control, which results in a hierarchical network (Baran, 1964). Due to this overlap in terms, the use of the term decentralized could be misleading. The confusion is compounded when both terms are used as when Barca and Sekercioglu define decentralized systems as those that "use distributed communication and control mechanisms" (Barca, 2013, p. 347). This use of distributed command and control fits with Baran's description of control methods, which aligns with Barca's and Sekercioglu's description of a decentralized system. As defined by Baran, distributed networks are those where the individual elements communicate directly to one another (Baran, 1964).

This literature review will seek to maintain consistency with command and control theory as described by Paul Baran while merging commonly understood terms from robotics swarming. This review will discuss three broad categories, centralized, hierarchical, and distributed. Centralized matches with both command and control theory and swarm literature. Hierarchical in this review maintains its definition from command and control theory, which also corresponds to hierarchical from swarm theory. Distributed also maintains its definition from command control theory which includes emergent coordination and the use of decentralized in some swarm literature. Decentralized will not be used as a descriptive category in this review.

a. Centralized Control

Centralized control relies on central planners who gather information and then direct individual elements of the swarm (Barca & Sekercioglu, 2013). There are multiple advantages for this method. Centralized swarms are easier to control and demonstrate a large degree of flexibility for the types of formations they can achieve because they are under the direct control of the central planner. For the same reason, it is "easy to predict the behavior of the overall system" (Barca & Sekercioglu, 2013, p. 347). These two qualities make centralized control appealing for situations where the swarm behavior must be tightly regulated. At the same time, centralized control does have some negative

qualities. First, the direct control of each vehicle leads to a "fixed solution" which is difficult to adjust as the environment changes (Legras, Glad, Simonin, & Charpillet, 2008, p. 3). Second, the "central planners" require global knowledge of all assets in the swarm. The required information exchange can lead to significant network traffic, and limits the scalability of centralized swarms. This means that centralized swarms do not scale well without adding sub-swarm, centralized planners (Barca & Sekercioglu, 2013). Third, the effectiveness of a centralized swarm is directly tied to the operation of the central planner (Barca & Sekercioglu, 2013). If this critical system component is lost, the system can no longer function. Finally, centralized systems require a data link connection for communication between the central planners, controllers, and operators. For both these reasons, centralized swarms do not demonstrate a high degree of robustness. Moreover, centralized swarms suffer in RF denied environments because of the difficulty in maintaining direct control of each UAV.

(1) External-Centralized Control

For the purposes of this review, the centralized category will be subdivided into external-centralized and internal-centralized. External-centralized systems have the central planner as a system component external to the UAVs. Researchers at the University of Pennsylvania developed an example of this system. Their goal was to develop a micro-UAV. This effort resulted in a micro-quadcopter with a diameter of 21 cm, a weight of 50 grams without a battery, and a flight time of approximately 11 minutes (Kushleyev, Mellinger, & Kumar, 2013). To reach this size, much of the computing was shifted off the UAVs. The tests were conducted indoors. To track position information, they used a Vicon tracking system (Kushleyev et al., 2013). For the purposes of the micro-UAV system, the position information from this system was stored in shared memory. MATLAB-coded control nodes used this position information and communicated routing information via a radio link to the micro-quadcopters in their respective section (Kushleyev et al., 2013).

The University of Pennsylvania's swarm is an example of an external centralized swarm for two reasons. First, it is centralized because it uses central planners to track the location of all assets and communicate updated flight information to each member. Second,

it is external because these control nodes are not implemented on UAVs within the swarm. Their research highlighted the bandwidth challenges of a centrally controlled swarm and used multiple control nodes to mitigate this problem. This facilitated the rapid calculation of flight data while limiting the number of messages each node had to send. For the purposes of this test, each node controlled up to five UAVs (Kushleyev et al., 2013). Finally, their test demonstrated the advantage of a centralized swarm, as they were able to execute precise, global formations for the swarm. The requirement for reliable communication between searchers and the centralized controllers running on ground stations makes this control paradigm inappropriate for the application envisioned by this work.

(2) Internal-Centralized Swarm

The other subcategory of a centralized swarm is an internal-centralized swarm. Systems using this paradigm implement the control nodes on UAVs within the swarm. Dylan Lau examined an example of this control type in his 2015 thesis exploring coordination algorithms for swarms conducting area searches. He built upon the previous work done by the NPS ARSENL program to "generate dynamic UAV flight paths using onboard processing" (Lau, 2015, p. 38). The initial plan was to use a global search map, however this approach ultimately proved unworkable: as the swarm increased in size, the latency involved with updating this one global map prevented effective operations (Lau, 2015).

The eventual solution to this problem was to break the search area into sub-regions that allowed multiple central planners to control assigned sub-swarms within each sub-region. For the purposes of his experimentation, he labeled these central planners as "master searchers" (Lau, 2015, p. 16). Similar to the University of Pennsylvania experiment, these central planners were responsible for planning the movement of their assigned UAVs. The only change was that these central planners were implemented on swarm UAVs rather than on external control stations. This enabled the master searcher to ensure the effective search of a given area, as shown in Figure 3.

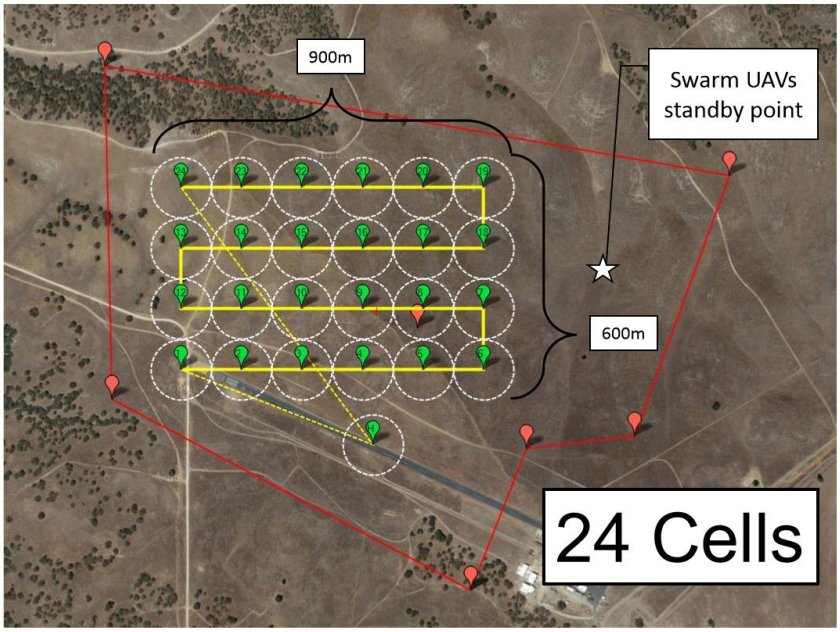


Figure 3. Coordinated search pattern for NPS thesis. Source: Lau (2015).

The use of internal-centralized control enabled the efficient management of the UAVs conducting the search. One advantage of this type of swarm control is that less power is required for control transmissions. In this example, this prevented adjacent swarms from interfering with each other. One potential disadvantage of an internal-centralized control swarm is that the central-controller UAVs may have to carry additional weight. This could be due to additional radios, batteries, or more capable computers. The other disadvantage is that a data link is still required between the UAVs. While individual data link distance requirements are reduced by using UAVs as central controllers, the larger communication requirement remains. Further, this requirement for a reliable data link between searchers and controllers makes this swarm control method unsuitable for the anticipated operating environment this thesis seeks to address.

(3) Hierarchical Swarm

The use of internal-centralized control provides the capability to build a hierarchical swarm. In a hierarchical swarm, the "…elements are controlled by 'squad' level agents, who are in turn controlled by higher-level controllers" (Scharre, 2014, p. 39). In some UAV

literature, the hierarchical model is referred to as multi-star where the external control communicates with one element in each group of UAVs, which subsequently pass messages to their subordinate UAVs (Gupta, Jain & Vaszkun, 2016). This capability mimics the standard military chain of command with leaders at different echelons controlling their subordinate elements. In the search example above, the swarm operator would determine the overall search area. This individual would assign parts of this area to sub-swarms, and the master searchers in each of these sub-swarms would control the movement of their UAVs while they conduct the search. The ideal implementation of the concept would allow the operator to set the overall search area for a master UAV who would then divide it for multiple sub-swarm controllers who would implement the algorithm once again to assign areas to their sub-swarm's searchers.

The use of this control mechanism increases the scalability of the swarm as compared to a solely centralized swarm. It can also increase the robustness of the swarm by allowing branches of the hierarchy to continue operation if others fail. The subordinate elements are able to continue controlling their sub-swarms should one control node fail. This robustness would be increased with the addition of succession of control rules should a node fail. Finally, the human-swarm interaction is improved because the main controller provides overarching guidance, and some of the detailed coordination is passed to the subordinate control nodes. Unfortunately, this approach does not mitigate the requirement for reliable data links between centralized controllers at the lowest level and their assigned UAVs and is therefore unlikely to be applicable in a denied-communication environment.

b. Distributed Control

The intelligence of centralized control systems resides in key nodes, but the intelligence of distributed control systems resides in each agent or their interactions (Sauter et al., 2009). Distributed control occurs when "each UAV independently receives some information and takes an action" (Gaudiano, 2003, p. 5). This allows for the storage of information near where it is needed (Parunak et al., 2002). With their distributed communication and control methods, distributed swarms reduce the possibility of communication delays due to reduced traffic on the data links. They are also inherently

scalable because the path computation is not constrained by the capability of a single planner. Finally, distributed control systems are resilient to the loss of a single node because each UAV in the swarm serves the same purpose if it is a homogenous swarm (Barca & Sekercioglu, 2013). These qualities make distributed swarms appealing for military purposes as they yield scalable and robust systems. A key downside to distributed swarms, however, is that it is difficult to predict the behavior of the overall system since it results from the accumulation of each UAV's decisions (Barca & Sekercioglu, 2013, 347).

There is a disagreement in the literature that requires clarification for the purposes of this review. As mentioned previously, each UAV takes action in a distributed swarm based on its own locally maintained situational awareness. Paolo Gaudiano, Benjamin Shargel, and Eric Bonabeau assert that this decision can be made based on information about their immediate vicinity or information about the entire environment (Gaudiano, Shargel, Bonabeau, & Clough, 2003). In contrast, Jan Carlo Bara and Y. Ahmet Sekercioglu concluded that a distributed system could not make decisions based on complete global knowledge (Barca & Sekercioglu, 2013). For the purposes of this review, a distributed swarm will be one where all flight planning occurs onboard each individual UAV. This planning can be based on either flocking or pheromones. UAVs in flocking-based swarms make decisions based on the observed positions or states of other UAVs, while UAVs in pheromone-based swarms make decisions based on the perceptions of the environment or parts of the system that are not UAVs.

(1) Collective Intelligence

At first thought, it seems unlikely that a large group of UAVs all making separate decisions could serve a greater purpose, but in 1999, Bonabeau published *Swarm Intelligence.* Using biological examples, he demonstrated that "collective intelligence" could emerge from a group of "simple agents" (Bonabeau, 1999, p. 9). They termed this "swarm intelligence." For anyone who has observed a beehive or an ant colony, the idea makes sense. In a May 2001 *Harvard Business Review* article, Bonabeau and Meyer applied the initial idea of swarm intelligence to business practices, and they described how "complex collective behavior can emerge from individuals following simple rules"

(Bonabeau & Meyer, 2001, p. 111). In 2007, Ian Gravagne and Robert Marks developed a case study based on an Icosystem swarming model. Through this, they showed that three basic behaviors could develop a globally useful pattern. In their study, "protector" behavior occurred when each "agent in the swarm picks two others and tries to position itself between them," "refugee" behavior occurred when each "agent in the swarm picks two others and tries to position itself so that the first is directly between itself and the second," and "aggressor" occurred when "every agent in the swarm picks another and chases it" (Gravagne & Marks, 2007, pp. 471-2). These rules demonstrate examples of the first subdivision of distributed swarms, the flocking swarm. In this category, each UAV's position is based on the positions of the other UAVs in the swarm.

In 2015, Robert Hunjet, with the Australian DSTG, modeled these behaviors in Java to demonstrate how they could be applied to military applications (Hunjet, 2015). The aggressor behavior resulted in the convergence of the swarm, as shown in Figure 4. In 4a, each element is randomly placed within the area. The elements begin to congregate as they pick another element to chase, as shown in 4b and 4c. They end up in one location in 4d.

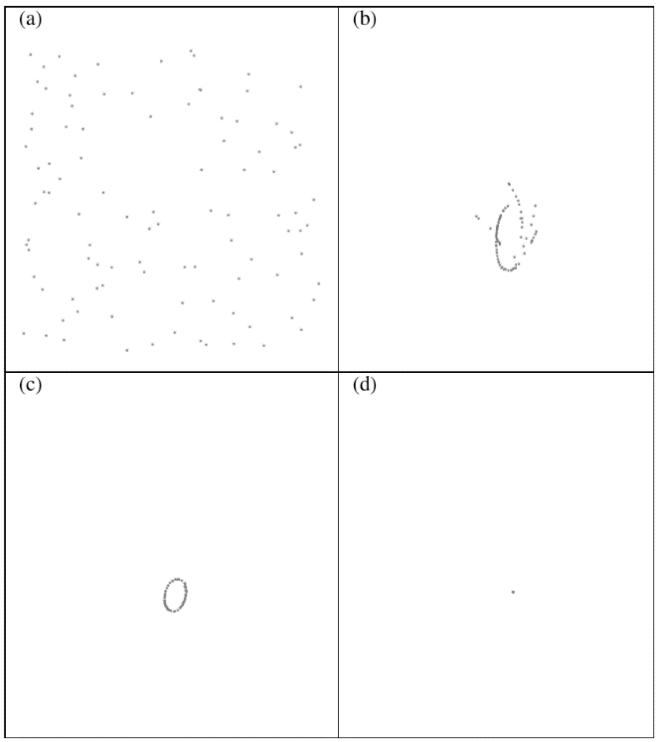


Figure 4. Aggressor behavior swarm convergence. Source: Hunjet (2015).

With the addition of a formally designated leader, Hunjet demonstrated that the aggressor behavior could result in a "follow the leader" global behavior, as shown in Figure 5. As before, the elements begin evenly dispersed in 5a. They move towards each other in 5b and 5c as before, but now with the one leader element, they end up following this leader in 5d.

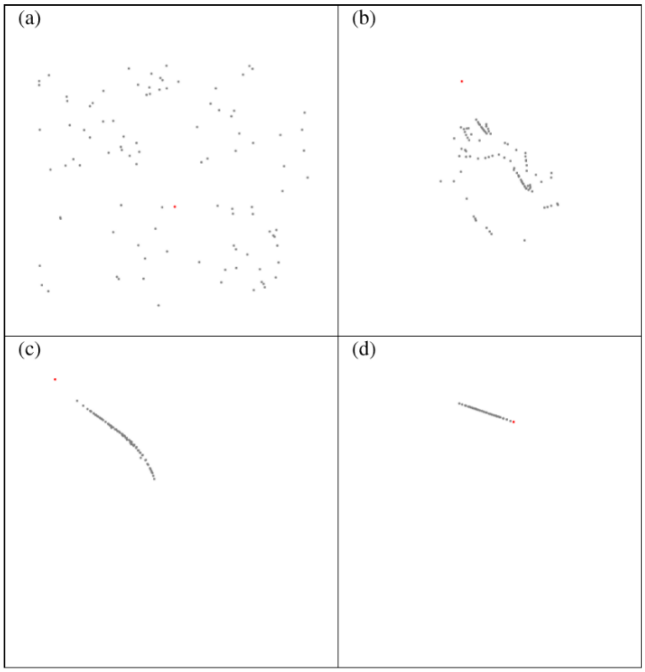


Figure 5. Use of the aggressor behavior to obtain "follow the leader" behavior. Source: Hunjet (2015).

The protector behavior resulted in a convergence similar to the result of the aggressor behavior, as shown in Figure 6. As before the elements begin evenly dispersed in 6a. As each element moves between two others, they begin to move closer, as in 6b and 6c. This results in all elements in one location, as shown in 6d.

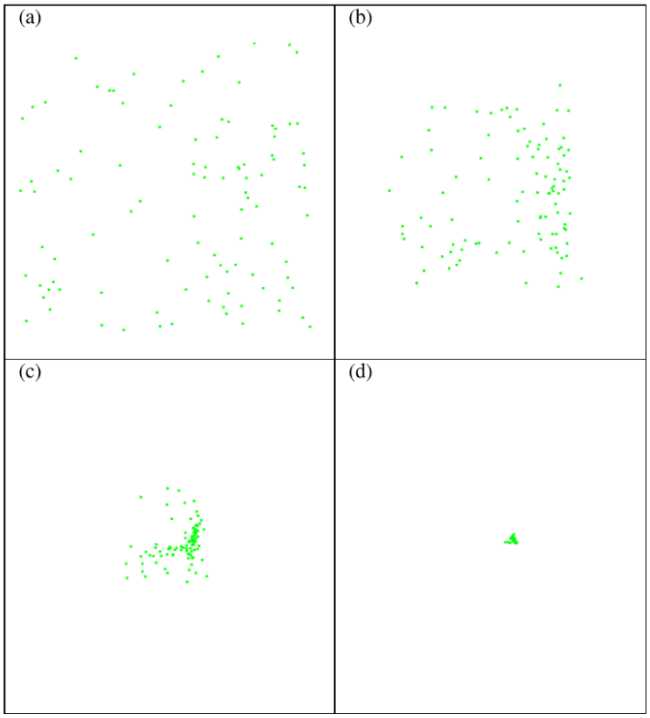


Figure 6. Swarm convergence under the direction of the protector behavior. Source: Hunjet (2015).

Finally, the refugee behavior had the opposite effect, which resulted in the dispersion of the swarm, as shown in Figure 7. As before, they begin evenly dispersed in 7a. The elements begin to disperse as each element picks two others and moves to put one in between it and the other, as shown in 7b and 7c. This results in the dispersion of the elements as shown in 7d.

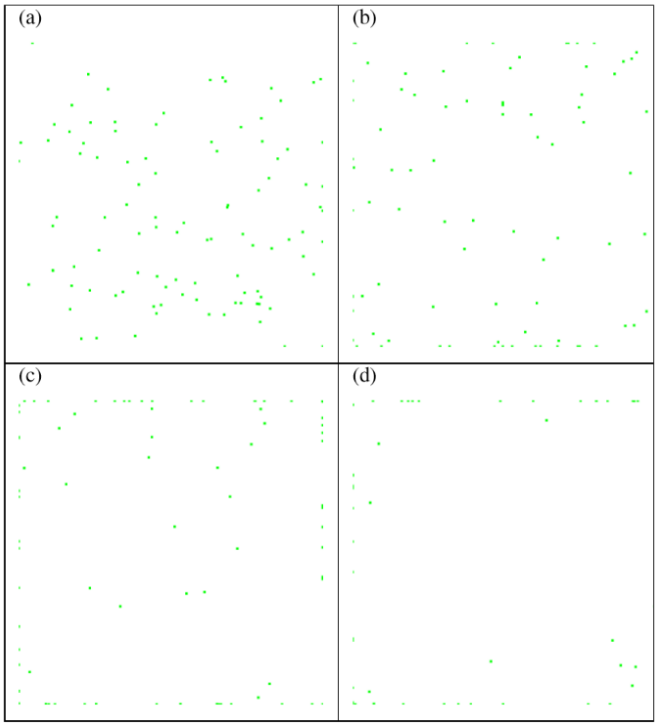


Figure 7. Swarm dispersion resulting from the refugee behavior. Source: Hunjet (2015).

(2) Distributed, Flocking-Based Swarm

As previously mentioned, a flocking-based swarm makes decisions based on the observed location or state of other UAVs. An example of a distributed flocking-based swarm is the NPS ARSENL swarm that demonstrated the simultaneous flight of 50 UAVs during which vehicle-state information was used to control individual vehicle positioning within the formation (Chung et al., 2016). Coordination for the ARSENL swarm relies on a consensus algorithm through which limited state information can be reliably exchanged between swarm vehicles (Davis, Chung, Clement, & Day, 2016). The information from these messages is processed individually by each UAV's autonomy payload to determine the UAV's immediate course of action.

Researchers in the Department of Biological Physics of the Eötvös University, in Budapest, Hungary, developed another outdoor, distributed, flocking-based swarm. Their system uses XBee modules to broadcast state messages. This communication method resulted in a UAV communication range of approximately 50 to 100 meters meaning that

the UAV decisions are based solely on local vice global swarm information (Vásárhelyi et al., 2014). That is, each UAV's flight is based on only those vehicles within communication range (Vásárhelyi et al., 2014). Using this method, they were able to operate a swarm of 10 UAVs in both day and night for up to 20 minutes (Vásárhelyi et al., 2014). While their swarming effort was limited to 10 UAVs, the communication technique shows promise of scaling well to larger swarms without computation or bandwidth issues.

These two examples of distributed flocking swarms demonstrate the characteristics of this method. They provide robust swarms because the loss of one vehicle does not prevent the operation of the overall swarm. This occurs because each UAV is equally capable. Additionally, they are scalable because rather than relying on a centralized planner, the flight planning is moved to each UAV. This capability comes with the disadvantage of requiring more complex UAVs. Additionally, the swarm loses the formation precision demonstrated by centralized control because of the lack of reliable global situational awareness. Finally, these systems do still require inter-vehicle communication to pass position information. Whether their situational awareness is based on global knowledge or local knowledge, individual UAVs must receive this information from other UAVs for now, although improvements to onboard sensors and processing might mitigate or eliminate this requirement in the future. For the time being, however, this type of control is not suited to the environment focused on in this thesis where either distances or electronic warfare prevent the passing of information directly between UAVs.

(3) Distributed, Pheromone-Based Swarm

The second type of distributed swarm, pheromone-based, is also biologically inspired. It is based upon stigmergy, which is a term used "to describe indirect communication through the environment" (Gaudiano, Shargel, Bonabeau, & Clough, 2003, p. 7). This occurs as each element senses and alters its environment (Fraser & Hunjet, 2016). As described by Bonabeau in *Swarm Intelligence*, this term was first used by Pierre Grassé in 1959 to describe the coordination among termites (Bonabeau, 1999, 24). The mechanisms for this coordination took further study to identify. One 1989 study, conducted at the Université Libre de Bruxelles in Brussels, Belgium, identified the use of pheromones

in the Argentine Ant (Goss, Aron, Deneubourg, & Pasteels, 1989). Through their test, the researchers demonstrated that the ants were able to select the shortest path based on their trail of pheromones. This was because the pheromones had a positive feedback process whereby each ant's successive path would strengthen the present pheromone level (Goss et al., 1989).

Numerous researchers have studied and extended the applications of stigmergy that Bonabeau examined in Swarm Intelligence. H. Van Dyke Parunak explored the concept of using digital pheromones to assist with swarming operations. In Making Swarming Happen, he describes three characteristics of pheromones: they aggregate as each individual passes over a given location, they evaporate over time to convey the timeliness of the information, and they propagate to nearby areas (Parunak, 2003). Thus, pheromones are able to communicate strength, timeliness, and direction. With this information, a swarm is able to conduct path planning (Parunak, 2003). Due to the small amount of information conveyed, pheromone transmissions also require minimal bandwidth (Sauter et al., 2005). This makes the use of digital pheromones an appealing choice for swarm control.

The use of digital pheromones for swarm control falls into two categories: internal and external. Internal pheromones are updated on a pheromone map stored by each UAV. Local values are updated as a swarm element encounters something new in the environment, and this updated information is relayed to the other UAVs in the swarm. This is equivalent to flocking-based swarms using global swarm information. Upon receipt of new information, the local pheromone map is updated, and each UAV can conduct path planning based on these aggregated pheromones contained in its onboard map (Sauter et al., 2005).

The John Hopkins University Applied Physics Laboratory, the Army Research Laboratory, and Altarum Institute tested an example of an internal pheromone swarm in 2004 at Aberdeen Proving Ground (Sauter et al., 2005). In this experiment, they demonstrated how a heterogeneous swarm of two UAVs and four unmanned ground vehicles could coordinate among themselves, react to the environment, and accomplish overarching objectives (Bamberger et al., 2006). A key advantage of an internal pheromone swarm is that the controller is able to make changes to the map's pheromones to affect the

overall behavior of the swarm (Walter, Sannier, Reiners, & Oliver, 2006). This capability provides the commander with control similar to centralized swarms if necessary, but the effective implementation of this method relies upon end-to-end connectivity of the swarm.

Thus far, all of the swarming techniques discussed require inter-vehicle communication. Whether they rely on global or local knowledge, these connections update each UAV regarding their surroundings. This connectivity is so essential that even the Barca and Sekercioglu review stated, "connectivity is essential in swarming" (Barca & Sekercioglu, 2013, p. 349). Unfortunately, there is no guarantee of ubiquitous connectivity on today's battlefield. Therefore, for a swarm to be of military use, it must have a method of operating in environments where the UAVs cannot communicate with each other. The final method of distributed swarming, which utilizes indirect communication between swarm members, addresses this requirement.

External pheromones store information in the environment instead of on an internal map. This takes the advantage of a distributed system one-step further by storing all path information directly "where it is generated and … where it is needed" (Parunak et al., 2002, p. 2). Parunak proposed the use of unattended ground sensors (UGS) "placed in the battlespace by air drops or artillery" (Parunak et al., 2002, p. 3). Using a system such as this, however, requires significant resources and preparation of the battlespace. Hughes Research Laboratories (HRL) Laboratories described an implemented external pheromone system in their paper Pheromone Robotics. Instead of digital pheromones, this system used virtual pheromones. In their approach, they used pheromone robots to spread throughout an area to become an "embedded computing grid" (Payton et al., 2001, pp. 319-20). Of note, the robots in the HRL system communicated via infrared (IR) signal, instead of RF signals. The precise line of sight nature of this communication results in robot-to-robot communication, which matches the open routes of the area. The researchers were then able to take advantage of the obstacles' effects on communication instead of attempting to overcome it (Payton et al., 2001).

This concept can be applied to other environments by matching communication means with mobility constraints. To disperse into a suitable grid configuration, robots use a "pheromone intensity" concept that causes a repulsive reaction if nodes are too close and

an attractive reaction if they are too far away (Payton et al., 2001, p. 322). Effectively a hybrid of the refugee and protector behaviors, Robert Hunjet demonstrated how a similar concept could be used for UAV communication nodes to provide complete coverage of an area as shown in Figure 8 (Hunjet, 2015). In 8a, all of the elements are collocated with a preset communication radius. The elements continue to disperse if an element is within their communication radius, as shown in 8b and 8c. This results in uniformly distributed elements throughout the area.

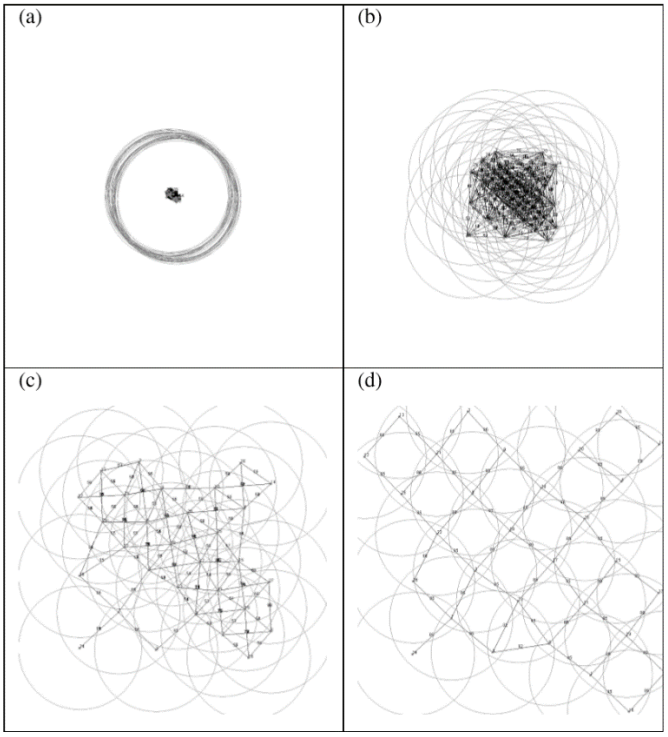


Figure 8. Communication nodes dispersing throughout the area.
Source: Hunjet (2015).

In HRL Laboratory's test, unlike the internally stored global pheromone map mentioned above, the information from each dispersed node implicitly provides routing guidance (Payton et al., 2001). The virtual pheromone consists of a type, a hop-count, and a data field. As each robot receives the pheromone, they decrement the hop-count and then resend it to the adjacent robots. Therefore, the pheromone direction with the highest hop-

count value is evidently the shortest, or preferred, path to the target. This simplifies the communication process by using the pheromone host to pass the nature of the message.

In 2005, a group from the Georgia Institute of Technology used a similar approach to the development of a low cost pheromone robot. The result was the Georgia Tech Network for Autonomous Tasks (GNATs), wherein each robotic node had four IR emitters, four IR receivers, two LEDs, a button, a microcontroller, and a battery (O'Hara et al., 2005). Using network routing concepts, they applied the distributed Bellman-Ford algorithm to develop a "directed graph of shortest hops" wherein they increment along the path rather than decrementing (O'Hara et al., 2005, p. 711). The node nearest the goal sets the pheromone value to one, and each hop increases it by one. This results in the number communicating not only the ideal path, but also its relative distance (O'Hara et al., 2005). Once the network is established, a mobile robot can be routed to the goal by always selecting the node with the lowest value (O'Hara et al., 2005).

c. Hybrid Control

Whether centralized, hierarchical, or distributed, different swarms can be developed for different objectives. As noted by Jan Carlo Barca and Y. Ahmet Sekercioglu, the ideal swarm system may be a combination of multiple control types. They term this a "hybrid" system that can "overcome the difficulties associated with 'pure' centralized or distributed systems" (Barca & Sekercioglu, 2013, p. 347). To summarize the control types reviewed, Table 2 shows each type with advantages and disadvantages. Using this table, the swarm designer can determine the ideal swarm control mechanisms. Given different environments, a swarm may need to operate with different control mechanisms in different locations. Additionally, different control types may be used for different elements of path planning to achieve the desired outcome. Swarms that employ only one control mechanism are tied to capabilities and limitations of that method without any way to mitigate the disadvantages.

On the one hand, the strictly hierarchical swarm "suffer[s] from high latency as the downlink length is longer than inter-UAV distance" (Gupta, Jain & Vaszkun, 2016, p. 1127). This prevents the effective transmission of all control traffic in this manner. On the

other hand, if only the overarching task was transmitted in this manner and execution was left to distributed coordination mechanisms, it becomes a useful technique to achieve higher-level swarm capabilities. The challenge for swarm development then becomes determining which combination of control mechanisms and models to be used for a given mission or task (Scharre, 2014). This concept is similar to the approach used with manned operations. One commander cannot dictate the actions of every subordinate throughout an operation. The commander determines the overarching mission and determines tasks for subordinate elements to accomplish. Once these tasks are communicated, the subordinate elements use cross-coordination, similar to mesh topologies in networking, to adapt to changes during execution. This cross coordination includes both implicit and explicit communication. This concept best equates to that put forward by Gupta, Jain, and Vaszkun as a hierarchical mesh network (2016). To reach their full capability, swarms will need to employ a mix of all control mechanisms on the future battlefield.

Table 2. Swarm control mechanism comparison

Category	Centralized Control		Distributed Control			
Subcategory	External Planner	Internal Planner	UAV Position Based (Flocking)		Environment Based (Pheromone)	
Type			Global	Local	Internal	External
Example	University of Pennsylvania (Kushleyev, 2013)	ARSENL Search (Lau, 2015)	ARSENL (Chung, 2016)	Eötvös University (Vásárhelyi, 2014)	Applied Physics Laboratory (Bamberger, 2006)	GNATs (O'Hara, 2005)
Description	-Movement is controlled by central planners outside of the swarm's UAVs.	-Movement is controlled by central planners who are UAVs in the swarm.	-UAVs make decisions based on other UAV positions. -UAVs have knowledge of all swarm members.	-UAVs make decisions based on other UAV positions. -UAVs have knowledge of only local members.	-Pheromones guide vehicle movement. -Pheromone changes updated on internally stored pheromone map.	-Pheromone information stored in the environment. -Ground nodes to store information.
Advantages	-Swarm's movement is easy to control. -Swarm behavior is easy to predict. -Almost all formations are possible.	-Swarm's movement is easy to control. -Swarm behavior is easy to predict. -Almost all formations are possible.	-Minimize communication delays. -Scale better than centralized options. -Robust since swarm is not tied to central planner.	-Minimize communication delays. -Scale better than global flocking. -Robust since swarm is not tied to central planner.	-UAVs make decisions based on local information. -Scale better than centralized options. -Commander can change the pheromone map.	-UAVs do not need to communicate with each other. -Scale as well as local flocking. -Robust since each UAV serves the same purpose.
Disadvantages	-Fixed solution, slow to adapt. -Global knowledge required for decision-making. -Does not scale well. -Central planners essential.	-Fixed solution, slow to adapt. -Global knowledge required for decision-making. -Does not scale well. -Central planners essential.	-Difficult to predict swarm global behavior. -Difficult to control the swarm. -Global knowledge is required. -UAV needs increased computing power.	-Difficult to predict swarm global behavior. -Difficult to control the swarm. -UAV requires increased computing power.	-UAVs must communicate with each other. -Difficult to predict swarm global behavior. -Difficult to control the swarm.	-Difficult to predict swarm global behavior. -Difficult to control the swarm. -Ground nodes must be dispersed in the environment.

F. MULTI-AGENT SIMULATOR OF NETWORKS (MASON)

DSTG chose George Mason University's Multi-Agent Simulator of Networks (MASON) to simulate the ferry node system. The University's Evolutionary Computation Laboratory and Center originally developed MASON to model social complexity. It provides the following advantages as an agent simulator:

- MASON models are not linked to their display (Luke, 2015). This provides the ability to both observe the effects of algorithm changes and algorithm optimization with bulk runs using one software package.
- MASON is written in Java (Luke, 2015). This provides the ability to run it using various operating systems while leveraging experience with the Java programming language.
- MASON has a "high-quality random number generator" (Luke, 2015, p. 8). This enables a MASON simulation to test a given system with varying inputs.
- MASON simulations are "duplicable" in that simulations will run exactly the same way given the same parameters (Luke, 2015, p. 8). This ensures that the simulation under evaluation will execute the same way each time.

We extended the initial MASON code developed by DSTG's Fraser to evaluate the dual-direction ferry node construct as described in Chapter III.

G. SELF ORGANISING COMMUNICATIONS AND AUTONOMOUS DELIVERY SERVICE (SCADS)

Australia's DSTG developed a program, SCADS, which uses UAVs under hybrid command and control to provide communication ferry service to ground nodes (Hunjet et al., 2017). The goal of the program is to "provide an autonomous data ferrying solution to facilitate effective information exchange through a DTN paradigm when end-to-end communications is unachievable" (Fraser & Hunjet, 2016, p. 1). They use centralized control to communicate the route between the ground nodes to the UAV (Fraser & Hunjet, 2016) in that the ground nodes' initial positions are assumed to be known, and the UAVs

are all launched from the same point. The ground nodes are the elements of the system that generate and receive messages. The system is a hybrid because UAV speed is adjusted based upon the residual value of distributed, external pheromones at each ground node location (Fraser & Hunjet, 2016). Unlike the UGS proposed by Parunak, these pheromones do not cover the entire battlespace, and unlike the GNATs, the pheromone communicates information about other UAVs rather than the environment. At each ground node, the pheromone decays according to

$$p(t+1) = p(t) - D_r, \tag{1}$$

where p(t+1) is the pheromone value at the next time-step, p(t) is the current pheromone value, and D_r is the decay rate (Fraser & Hunjet, 2016).

The residual pheromone value is communicated to the next passing ferry node. This piece of information, when combined with the universally known decay rate, communicates the time the previous ferry passed that ground node. This visiting ferry node resets the pheromone to the maximum value, and it calculates its new speed according to

$$s(t+1) = (s_{\max} - s_{\min})(1.0 - p(t)) + s_{\min}, \tag{2}$$

where s(t+1) is the new ferry speed, s_{max} is the maximum allowable ferry speed, s_{min} is the minimum allowable ferry speed, p(t) is the ground node pheromone level (Fraser, Hunjet, and Coyle, 2017).

Each of these speed adjustments affects the distance between the ferry nodes, and the cumulative effect is that the ferry nodes end up evenly distributed around the perimeter without direct communication between the UAVs. Figure 9 shows the initial close proximity of ferry nodes (blue squares), and Figure 10 shows their even distribution around the perimeter with the accumulated adjustments from the ground nodes (green squares). Once the ferry nodes reach the even dispersion of the converged solution, the ground nodes are visited at regular intervals by the ferry nodes.

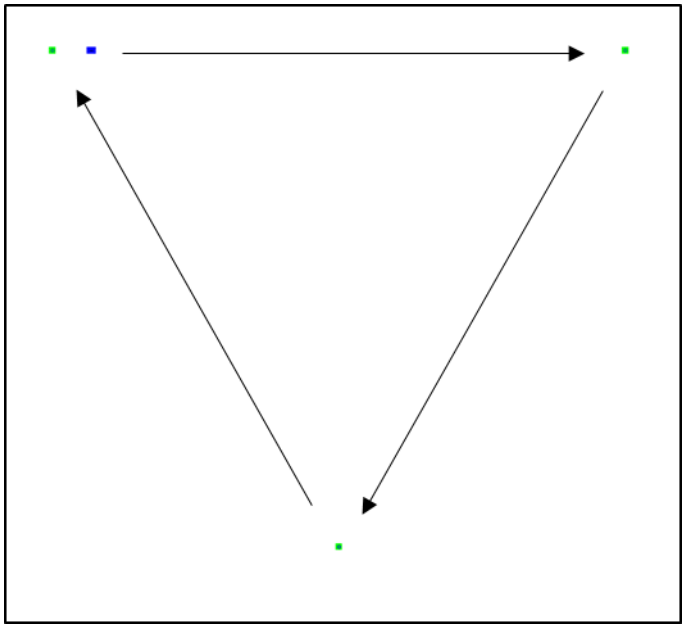


Figure 9. Initial close dispersion of single-direction ferry nodes (Blue Squares)

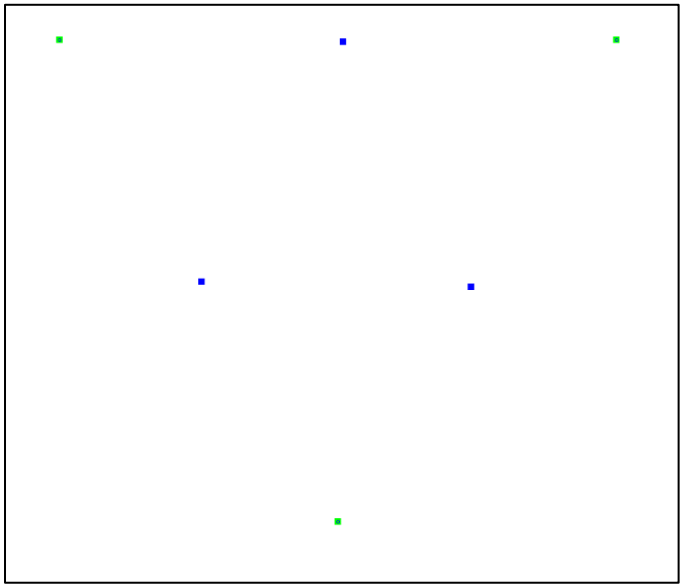


Figure 10. Final even dispersion of dual-direction ferry nodes (Blue Squares)

1. Dual-Direction Ferrying

A system based on single-direction ferry nodes, as shown in Figure 9 and Figure 10 suffers from message latency. Messages intended for ground nodes at the end of the ferry node path must wait for the ferry node to travel the entire distance. A dual-direction

ferry node system, as shown in Figure 11 and Figure 12 would reduce this maximum delivery time by ensuring that each message travels along the shortest part of the perimeter to the intended target. To date, Fraser's and Hunjet's work on dual-direction ferry nodes has only extended to simulating ferry nodes traveling in each direction. However, simply adding the other direction's ferry nodes does not achieve even distribution overall. As shown in Figure 12, the ferry nodes end up evenly distributed only with regard to ferry nodes traversing in the same direction. This alone is an incomplete solution, as a lack of coupling between the two directions does not facilitate even distribution of all ferry nodes or uniform service times at the ground nodes. We argue that linking the pheromone decay associated with both directions of travel can achieve bidirectional even dispersion.

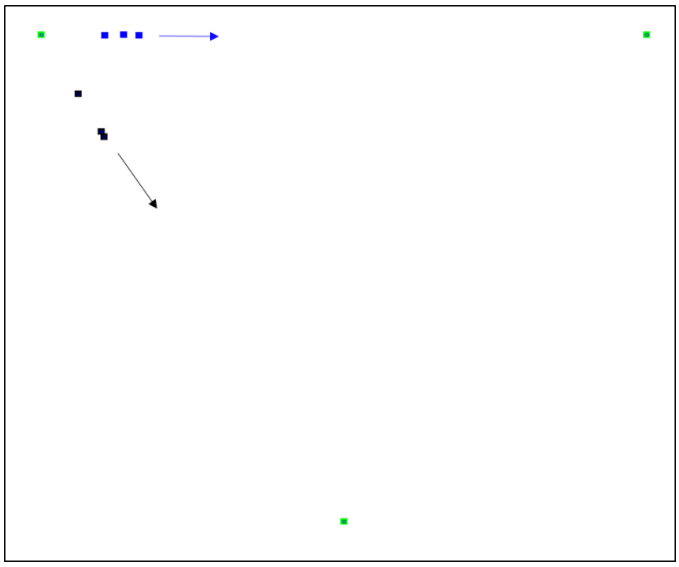


Figure 11. Initial position of unlinked dual-direction ferry nodes

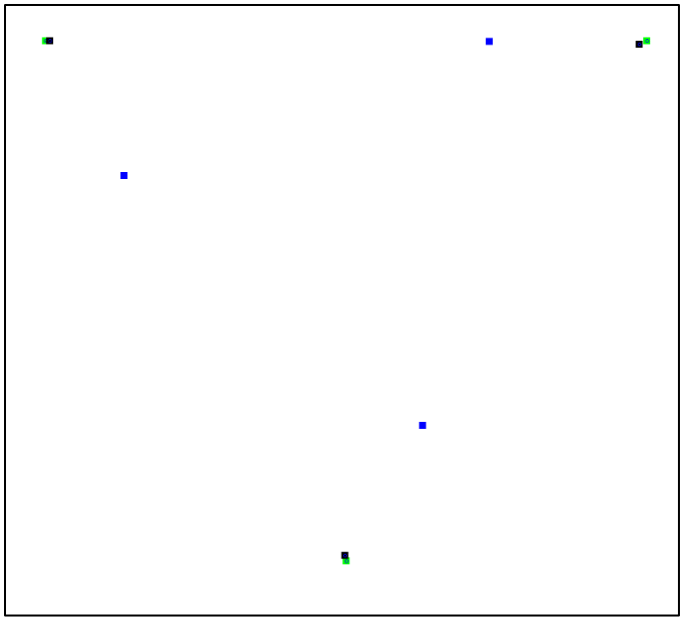


Figure 12. Directional even dispersion of dual-direction ferry nodes without coordination between directions

2. SCADS Live-fly Testing

DSTG has conducted live-fly tests to further the SCADS system. One of these tests occurred at their indoor flight laboratory where they flew quadcopters equipped with Pixhawk flight controllers and ODROID computers running the Robot Operating System (ROS) (Hunjet et al., 2018). These components and architecture are similar to those of the ARSENL system described below. After multiple tests using varying number of ferry nodes traveling in one direction, they found that the visitation time converged in a manner similar to what was observed in MASON simulations (Hunjet et al., 2018).

H. ADVANCED ROBOTIC SYSTEMS ENGINEERING LABORATORY

1. Background

ARSENL seeks to "provide a diverse academic and research venue to foster the holistic, multidisciplinary approach to the design, employment, and future concept development of robotic and unmanned systems" (Chung, 2015). With this goal in mind, they have undertaken multiple projects to develop autonomous capability within a fixed wing swarm. Efforts along multiple paths resulted in the successful field test of 50

simultaneously flying fixed wing UAVs in 2015 (Chung, 2015). Through the employment of such a large swarm, the NPS researchers were able to provide useful information on swarm networking and human-swarm interaction to the knowledge base (Chung et al., 2016). They followed this effort by conducting swarm against swarm challenges to develop additional capabilities (Davis et al., 2016).

2. Zephyr II Platform

The ARSENL swarm uses the Zephyr II UAV, shown in Figure 13. It is an NPS-designed UAV, built upon the Ritewing airframe that uses open-source hardware and software. Overall, each UAV weighs 2.5 kilograms, and can operate for approximately 50 minutes at a time (Davis et al., 2016). The key components of the Zephyr II UAV are listed in Table 3.

Figure 13. ARSENL's Zephyr II UAV. Source: Chung et al. (2016).

Table 3. ARSENL Zephyr II components. Adapted from Chung et al., (2016).

Airframe	Ritewing Zephyr II
Avionics	Pixhawk Autopilot
Autonomy CPU	Hardkernel ODROID U3
Telemetry link	3DR Telemetry Radio (915 MHz)
Payload link	Alfa WiFi USB radio adapter (2.4 GHz)
Front-facing camera	GoPro Hero 3
Power	2 x Thunder Power ProLite 3S 500mAHr
Propulsion	OS 3820-1200W
ESC	Castle Creations EdgeLite 50
Propeller	APC 11x5.5
Servos	2 x Hitec HS5245MG
GPS/Compass	3DR uBlox LEA-6H/compass
Airspeed sensor	MS4525DO/3DR Pitot-Static tube
RC TX/RX	Spektrum DX9 and satellite Rx

3. Robot Operating System (ROS)

Each Zephyr II UAV has an onboard Hardkernel ODROID U3 running the open-source Robot Operating System (ROS) that implements the swarming behaviors (Davis et al., 2016). ROS is a software architecture that uses a publisher-subscriber architecture in which nodes publish messages to topics to which others can subscribe (Quigley et al., 2009). The ROS nodes operating on the ODROID U3 make up the ARSENL "autonomy payload" which is responsible for "higher-level planning and coordination tasks" (Chung et al., 2016, p. 1257). The ARSENL payload operates by receiving updated UAV position information and sending waypoint commands (Day et al., 2015). These waypoint commands are sent through the autopilot bridge to the Pixhawk Autopilot, which is responsible for the "lower-level guidance, navigation, and control tasks" of each UAV (Chung et al., 2016, p. 1257). The "autonomy payload" receives information about other UAVs through the network bridge, which also facilitates UAV-to-UAV and UAV-to-ground communication (Chung et al., 2016). Chung et al. describe the primary ROS nodes in the autonomy payload as:

- /autopilot: This links the "autonomy payload" with the autopilot. It converts position and status information into ROS form and publishes it,

and it converts “autonomy payload” waypoints to the MAVLink messaging standard.

- /network: This allows the UAVs to share their positions with other UAVs and ground stations while also relaying ground station network commands.
- /swarm_tracker: This node combines the shared UAV position information to create a snapshot of the swarm every 100 milliseconds. This snapshot is used to implement the behaviors.
- /swarm_manager: This node facilitates the UAV’s participation in the selected swarm behavior. It switches between behaviors while ensuring that the prerequisites for requested behaviors are met before activation.
- /safety: This node is responsible for safety, and it will deactivate the payload and direct the aircraft to a loiter-position if the UAV enters an unsafe state (Chung et al., 2016).

These nodes, with their connecting topics, are shown in Figure 14.

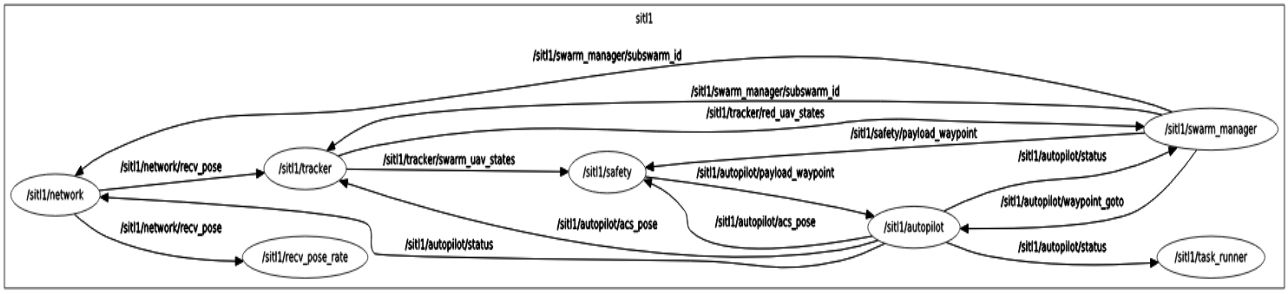


Figure 14. ARSENL ROS Diagram

4. ARSENL Networks

The ARSENL swarm communicates over an 802.11n ad hoc Wi-Fi network using UDP broadcast messages in order to minimize latency. Telemetry messages are sent every one-tenth of a second and autopilot status messages are sent every half-second (Davis et

al., 2016). This network is the primary means for swarm behavior assignment and swarm execution once a behavior is assigned. There are two additional networks used by the ARSENL swarm for safety of flight.

5. Swarm Control

ARSENL achieved the ability to operate 50 UAVs simultaneously by breaking from the standard multi-UAV method of one operator employing a specific number of UAVs. Their testing found that employing these traditional methods prevented expanding the swarm beyond 5-6 UAVs (Chung et al., 2016). With this limitation, each additional group of 5-6 UAVs would require an additional set of support and operator personnel. Instead, they divided the workload into a swarm operator and a swarm monitor with the swarm operator employing the swarm while the swarm monitor supervises the health of each UAV (Chung et al., 2016). ARSENL developed a graphical user interface called Swarm Commander, which provides the ability to monitor the status and location of swarm members while also directing new behaviors (Day et al., 2015).

6. ARSENL Flight Sequence

Sequencing a large number UAVs through launch, operation, and landing led the ARSENL team to develop a mission construct with distinct phases. They use a hybrid control paradigm where swarm behaviors are initiated from a ground terminal. After initiation, individual swarm members implement the behavior through distributed autonomous control. As described by Chung et al., they developed eight operational stages that each UAV passes through during the course of an entire mission. These phases are:

- Pre-Flight: Preparation of UAVs for flight through inspection and software loading;
- Launch: Individual launching of each UAV with either bungee launcher or electric-powered catapult launching system;
- Ingress: Transit of UAV from takeoff location to the flight operating area;

- Swarm Ready: UAV state after ingress when it is ready for initiation and execution of swarm behaviors;
- Landing: Transition from swarming to sequential landing of UAVs from the operating area;
- Recovery and Post-Flight: Recovery, inspection, and maintenance of UAVs in preparation for another mission (Chung et al., 2016, pp. 1255-6).

7. Software-in-the-Loop Simulation

Field-testing of a UAV swarm requires significant planning and support. The NPS ARSENL program developed an emulation program uses a physically based simulation capability, SITL, to test algorithm changes without the need to organize a field experiment (Day et al., 2015). The SITL capability is particularly important since it provides the ability to rapidly cycle through the algorithm refinement process with the actual UAV autonomy and autopilot software without using precious field experimentation resources. Emulations using the ARSENL SITL capability were used to validate the effectiveness of the dual pheromone SCADS concept.

I. HARDWARE ABSTRACTION AND INTEGRATION LAYER (HAIL)

One challenge of UAV vehicle operations is the wide variety of vehicle types and internal components. This can cause issues for system design because each one of these components produces and consumes information in a different manner. Recognizing this, DSTG developed an applications programming interface, called Hardware Abstraction and Integration Layer (HAIL) to enable standardized method calls to access underlying autopilot functionalities. It operates according to the idea that all of these diverse UAV control systems use "the same core classes of information (e.g. position, velocity, navigation)" (Elliot and Stevens, 2016, p. 2). This similarity provides an opportunity to use an intermediate layer to standardize communication. The particulars of a new vehicle's components only have to be added to HAIL instead of fundamentally reworking the control algorithm. The "strength of HAIL is to interface, integrate, and translate between hardware sensors/actuators/effectors and other systems" (Elliot and Stevens, 2016, p. 2). This thesis

integrates HAIL into the ARSENL autonomy payload via a bridging ROS node to provide easy heterogeneous UAV testing in the future.

J. CHAPTER SUMMARY

The recent advancements in swarm robotics promise to provide far-reaching capabilities for military operations. This thesis builds upon work done by both the Australian DSTG, NPS, and other researchers. DSTG developed a unique concept for collocating digital pheromones with operational ground nodes. The decaying value of these pheromones communicates vital information to other ferry nodes to achieve evenly distributed ferry nodes and regular visitation. NPS demonstrated the capability to operate large numbers of fixed-wing UAVs simultaneously. Merging these two capabilities in a system with dual-direction ferry nodes will provide a DTN proof of concept that amphibious forces could employ in the future. This thesis will extend the SCADS concept by developing a dual-direction system capable of mitigating the disadvantage of a message ferry system, delivery delay time. The next chapter will discuss the key to an effective dual-direction system, the coupling between the two directions to enable even distribution.

III. DUAL-DIRECTION LINKED PHEROMONES

A. CHAPTER OVERVIEW

Dual-direction ferry nodes with linked pheromones are explored to assess the potential to reduce delivery time for messages generated at the ground nodes while providing predictable ferry node visitation. This reduced delivery time will assist in mitigating the disadvantage of delayed delivery time associated with DTN. This chapter discusses the mathematical approach used herein to link the two pheromones at each UAV waypoint. It begins with a description of the linking formula, with an overview of the goal, followed by a description of its implementation in MASON. This is followed by a description of the integration of DSTG's HAIL with the ARSENL software. This also includes a discussion of the HAIL Bridge and additional ROS topics that enable communication between ARSENL and HAIL.

B. DUAL PHEROMONE DELIVERY TIME COMPARED TO SINGLE PHEROMONE

The purpose of seeking a dual-direction, data ferrying communication system is to reduce the delivery time for messages as compared to a single-direction system. This reduction occurs because messages travel along the shortest part of the perimeter to their destination. We will use Figure 15 to describe the advantage of a dual-direction system. This figure depicts three ground nodes in an equilateral triangle layout. These ground nodes are represented by the red and green squares. The difference between the differently colored ground nodes will be discussed in Chapter IV, Section B. The ferry nodes are depicted by the black and blue squares. The figure portrays the moment in time when the ferry nodes are evenly dispersed. Three of the ferry nodes are collocated with the ground nodes, and the other three are located halfway between the ground nodes.

Of note, the advantage of the dual-direction system does not require messages to be passed between opposite direction ferry nodes, although this could be added to provide redundant message delivery. This message passing is not required because a ferry node from one direction has already visited the area for which the opposite direction could pass

a message. For example, after the red ground node delivers a message to the collocated forward, prograde, ferry node, the ferry node continues along the green line until it passes the reverse, retrograde, ferry node at the yellow circle. This green line represents the part of the perimeter that this retrograde ferry node could deliver a passed message before arriving over the originating ground node. This is unnecessary because the prograde ferry node would already have serviced any additional ground nodes in this area. The other yellow circles represent the other locations where the ferry nodes pass one another. This is based on the assumption that ground nodes will send the same message to each passing ferry node until confirmation of its delivery.

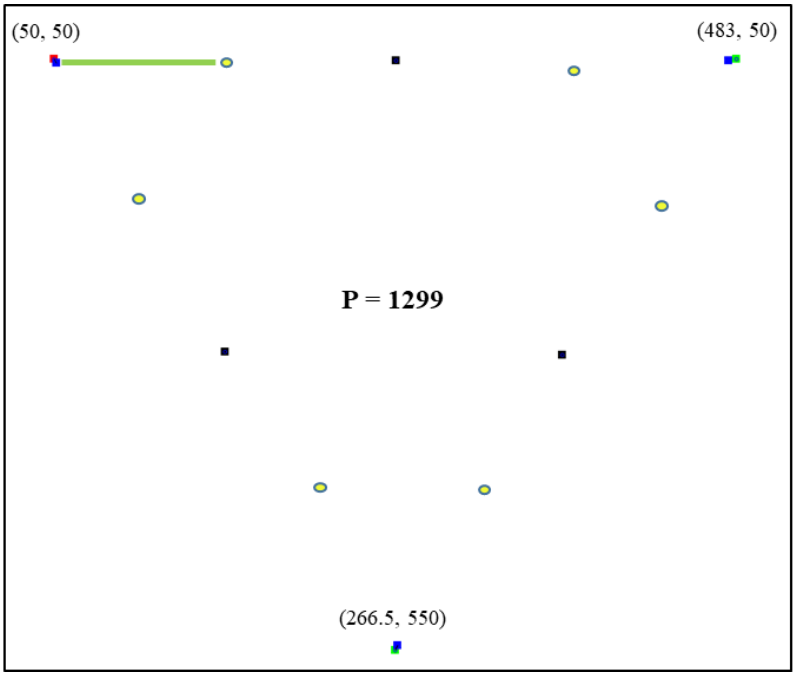


Figure 15. Opposite direction ferry node ferry node passing locations in a dual-direction system

The expected delivery time of a converged system can be calculated based on the perimeter length and the converged speed. The delivery time along the prograde ferry node path is calculated as

$$t_f = \frac{d}{s_c}, \tag{3}$$

where t_f is the prograde delivery time, d is the location along the perimeter from the originating ground node to the receiving ground node in the prograde traveling direction, and s_c is the converged transit speed of the ferry nodes. The delivery time along the retrograde traveling direction is calculated as

$$t_r = \frac{P-d}{s_c} + \frac{P}{2(n)(s_c)}, \tag{4}$$

where t_r is the retrograde delivery time, P is the perimeter length, d is the location along the perimeter in the prograde traveling direction, s_c is the converged speed, and n is the number of ferry nodes traveling in one direction. The second element of this equation accounts for the travel time between message delivery, as shown in Figure 15, and the first retrograde ferry node's passing of the home node. Note that message delivery does not account for either message upload or download times. These are held invariant with respect to the ferry nodes to simplify the model without loss of generality.

Application of these two equations to the previously depicted converged equilateral triangle example where n ferry nodes in each direction are servicing a perimeter length of 1299 are shown in Figure 16. Evidently, the prograde ferry node provides the fastest delivery up to perimeter distance from the originating ground node of approximately 750, but after that, the retrograde ferry node delivers the message the fastest. More generally, setting Equations (3) and (4) equal to one another and solving for d provides the distance at which prograde and retrograde delivery times would be equivalent. The resulting equation is

$$d_e = \frac{P(2n+1)}{4n}, \tag{5}$$

where d_e is the distance of equivalent delivery times. That is, prograde distances in excess of this value will be more quickly delivered by a retrograde node. In the example shown in Figure 15, this equation results in a d_e of 757.75.

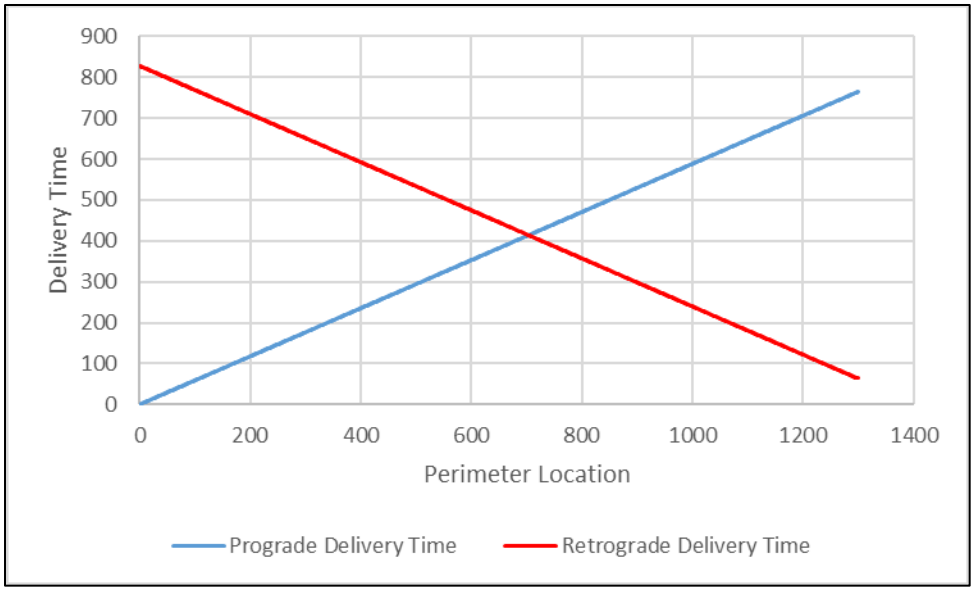


Figure 16. Delivery time comparison between prograde and retrograde ferry nodes around the perimeter

Figure 17 depicts these results on the ground node layout. Of note, the prograde ferry node provides the fastest delivery time for slightly more than half of the perimeter due to the time it takes for the retrograde ferry node to reach the home node.

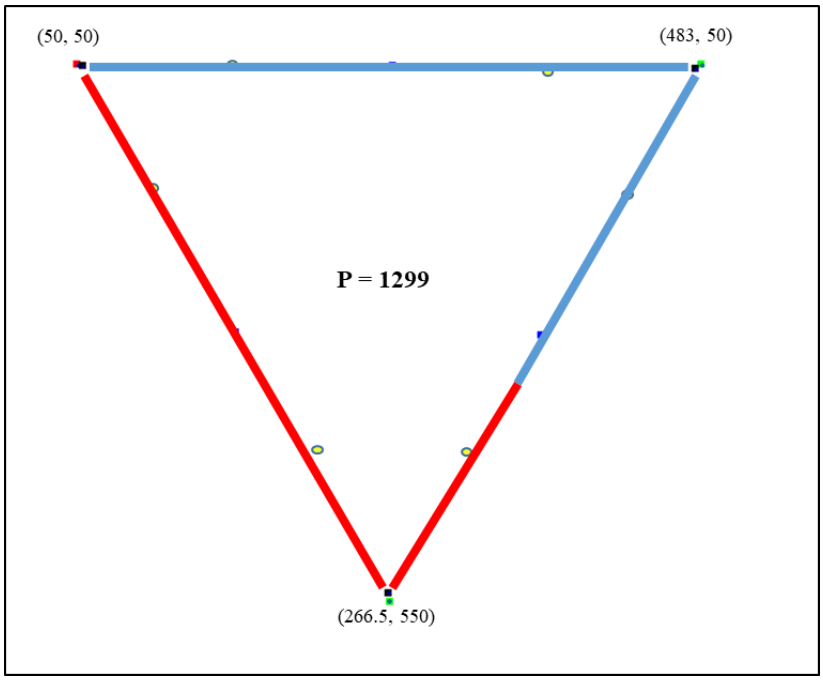


Figure 17. Delivery time comparison between prograde and retrograde nodes depicted on node layout

Of note, because Equation (5) provides maximum prograde distance from the message origin for faster prograde delivery, it can also be used to compute the maximum message delivery time for the system as

$$t_{\max} = \frac{d_e}{s_c} = \frac{P(2n+1)}{s_c(4n)} \tag{6}$$

where t_{max} is the maximum delivery time. This is demonstrably better than the single-direction ferry node system with its maximum delivery time of

$$t_{\max} = \frac{P}{s_c} \tag{7}$$

Figure 18 depicts the difference in maximum delivery times between single and dual-direction systems for perimeter lengths up to 4000 based on Equations (6) and (7).

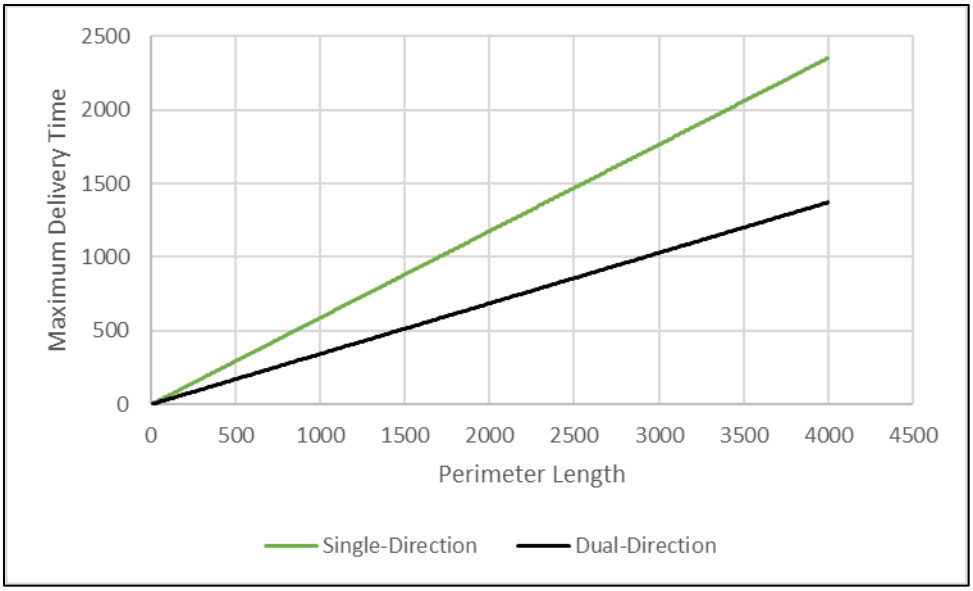


Figure 18. Maximum delivery time between single-direction and dual-direction

Assuming uniform distribution of ground nodes, the average delivery time is approximately half the maximum delivery time that is shown in Figure 18. This is based on half the distance the ferry nodes have to travel to visit the entire perimeter. In the single-direction system, this is equal to half of the total perimeter length. In the dual-direction system, this is equal to one quarter of the perimeter length since the prograde and retrograde ferry nodes each cover half of the perimeter. Finally, assuming the same even distribution between ground nodes, the minimum delivery time between the single-direction and dual-direction is the same. This occurs because the distance from the originating ground node to the receiving ground node is less than *de*.

Intuitively, a dual-direction communication ferry system will provide faster delivery time than a single-direction system in situations where the prograde distance between the originating ground node and the receiving ground node is greater than the equivalent distance as calculated by Equation (5). For situations where the distance between the originating node and the receiving node is less than the equivalent distance, the performance of the single-direction and dual-direction systems are identical. This observation is borne out by the Figure 18 depiction of the relationship between Equations (6) and (7). That is, the difference between the single-direction and dual-direction

maximum delivery time lines captures the additional time required for the single-direction system to deliver messages to ground nodes with a prograde distance from the originating ground node greater than *de*.

C. PHEROMONE COUPLING

These performance guarantees can only be assured if the prograde and retrograde ferry teams are evenly dispersed overall. To achieve this bidirectional even dispersion, the decaying pheromones in each direction must be coupled. Figure 19 depicts exemplar pheromone resets for uncoupled, dual-direction ferry nodes. The blue line depicts the value of the retrograde pheromone at a ground node as a function of time, and the red line depicts the same for the prograde pheromone. The peaks in the graph show that the ferry nodes are not evenly distributed around the perimeter. Ideal pheromone resets from even distribution of all ferry nodes would result the reset location of one direction's ferry nodes occurring halfway between the other direction's reset times.

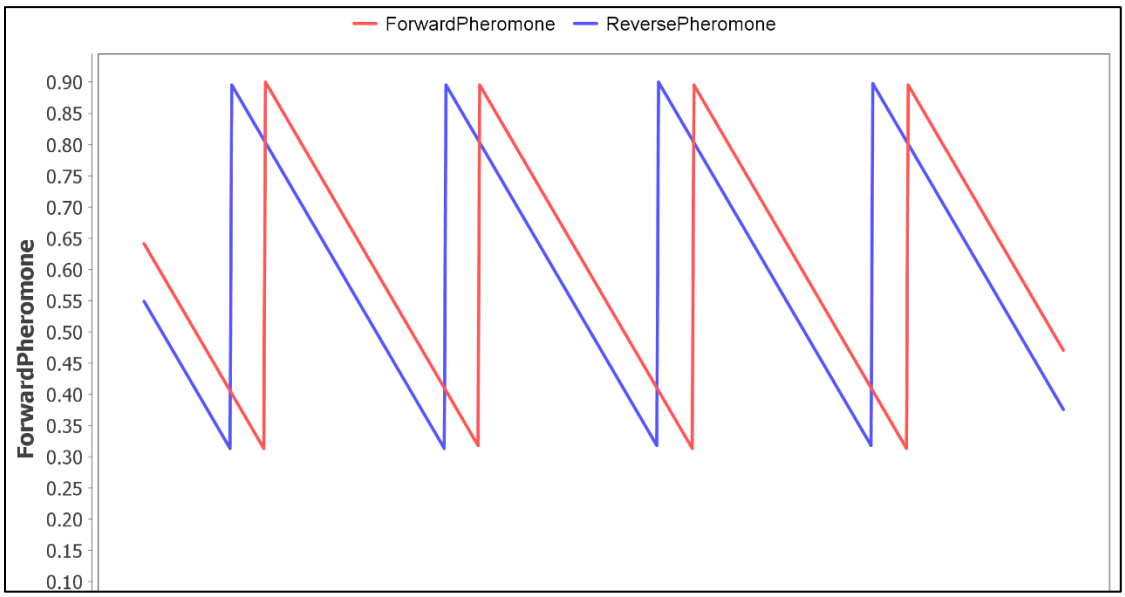


Figure 19. Unlinked dual-direction pheromone graph showing ferry nodes not evenly dispersed

Pheromone value maintenance at each ground node results in two possible locations in the algorithm at which the pheromones can be linked. The first is where the pheromones

decay. Modifications at this algorithmic location involve changing the decay rate based upon which pheromone is larger. This implementation requires that pheromone values decay faster or slower based on the relative value of the other pheromone to achieve an even dispersion. Through extensive simulation, changes at this location proved difficult to control, and they often result in uncontrolled growth or uncontrolled decay of the pheromone. Additionally, since the linear decay of the pheromones at the ground nodes provides the necessary information for all associated ferry nodes to adjust their speed, this adjusted decay rate must be communicated to all dependent ferry nodes at the same time as the pheromone value is passed. For this reason, no efforts to link the prograde and retrograde systems through pheromone decay adjustments proved successful.

The other location of the algorithm to link the prograde and retrograde systems was at the pheromone reset. In Fraser's initial work, the pheromones at each ground node were reset to the maximum value each time a ferry node passed their location. From there, they would linearly decay until the next ferry node passed that location (Fraser, Hunjet, and Coyle, 2017). Linking of the pheromones at reset is accomplished by adjusting the value to which the pheromones are reset rather than the decay rate. System performance, then, remains subject only to pheromone level rather than both pheromone level and decay rate. We identified two possible approaches to changes at this point.

The first applies an adjustment to the opposite direction's pheromone upon reset. For example, if a prograde ferry node passes over a ground node, the prograde pheromone is reset to the standard value of 1.0, and the retrograde pheromone value is updated as well by addition of a fixed value. This addition captures how recent a ferry node from another direction visited that ground node. It was found that through the accumulation of multiple adjustments, the ferry nodes end up offsetting. Within the MASON simulations, multiple values were attempted ranging from 0.3 to 0.7 for this off cycle addition to generate a converged solution for the specific topology described above. Some were successful, but they failed to result in a converged solution once the topology was changed. As discussed in Chapter IV, changes to topology affected the swarm's ability to converge indicating a dependent relationship between topology and algorithm. As the experimental results

indicate, there are many ways to tailor the system for a given layout, but it is much more difficult to find a solution that operates effectively across multiple ground node layouts.

The second approach to pheromone coupling we developed applies the adjustment to the same pheromone that is reset. For example, the prograde pheromone is reset to the standard value of 1.0 as a prograde ferry node passes a ground node. Following reset to the standard maximum value, an additional correction is added or subtracted to this pheromone value to account for the unevenness of the bidirectional distribution. This adjusted reset value is calculated so that the new pheromone decay path will intersect with the desired reset location halfway between the other direction's pheromone resets. If a ferry node passes a ground node before the halfway point, as shown in Figure 20, the adjustment is added to the 1.0, standard reset value. If, however, the ferry node passes after the halfway point, the adjustment is subtracted from the 1.0, standard reset value as shown in Figure 21. The accumulated effects of these bidirectional adjustments result in evenly distributed directional ferry nodes that are equidistant when one ferry node passes over the adjusting ground node.

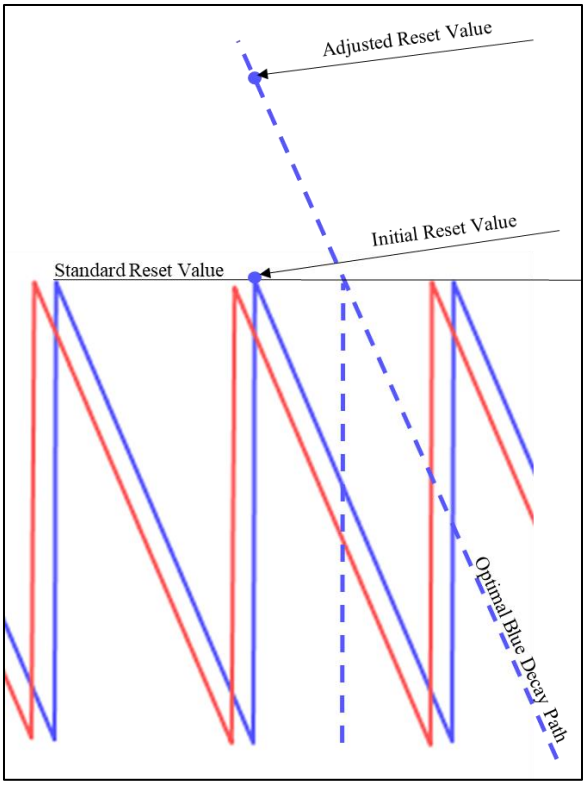


Figure 20. Retrograde reset prior to prograde reset midpoint requiring positive pheromone adjustment

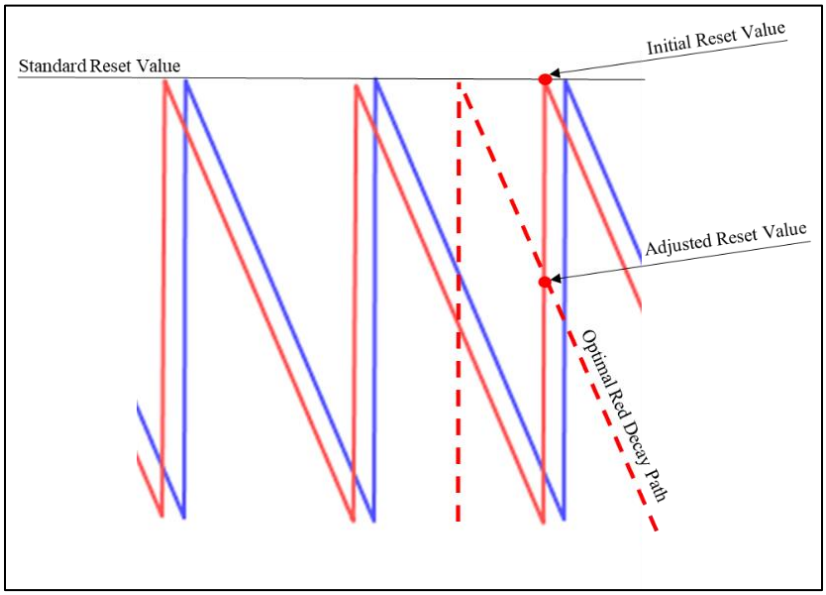


Figure 21. Prograde reset after retrograde reset midpoint requiring negative pheromone adjustment

1. Mathematical Formulation of the Positive Pheromone Adjustment

Figure 22 provides a depiction of sample results of a positive pheromone adjustment applied to the retrograde pheromone. The blue line depicts the retrograde pheromone, and the red line depicts the prograde pheromone with both pheromone values taken at the adjusting ground node. The adjustment is calculated and applied when the retrograde pheromone is reset at time *rt*. Upon retrograde pheromone reset, the next anticipated prograde reset time, *pt3*, is projected based on the previously calculated time between prograde resets, *Δpt*. Because the reset is occurring prior to the midpoint of the previous and next expected prograde pheromone reset, the retrograde pheromone requires a positive adjustment.

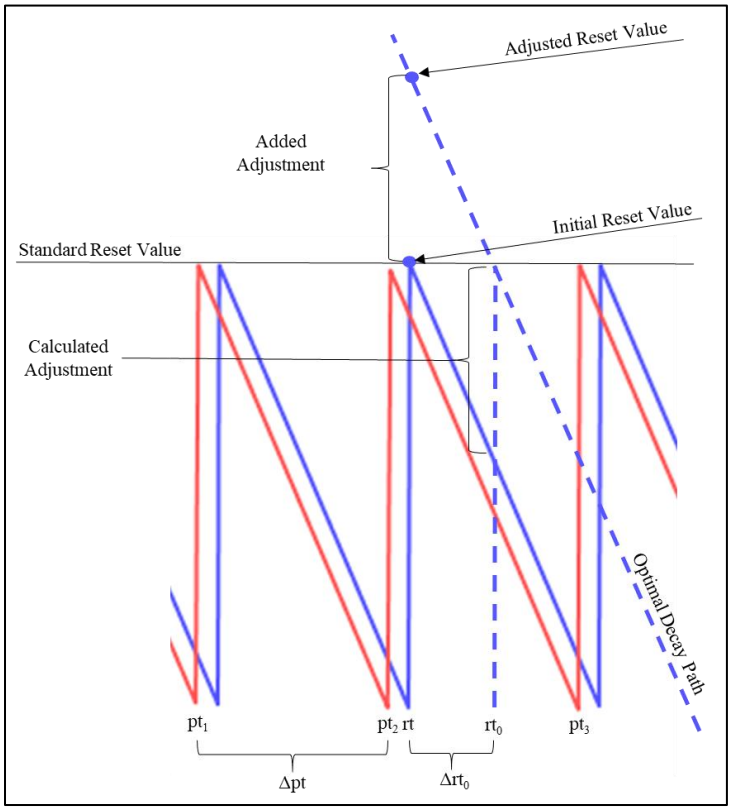


Figure 22. Mathematical formulation of positive adjustment using the retrograde formula for the example

The underlying idea behind this coupling is to adjust one direction's decay path so it intersects with the projected point halfway between the other direction's pheromone resets. To calculate this position, the time between resets must first be calculated. This is done by subtracting the last prograde ferry node reset time from the current one as

$$\Delta pt = pt_2 - pt_1, \tag{8}$$

where Δpt is the time difference between the prograde resets, pt_2 is the later prograde reset time, and pt_1 is the earlier prograde reset time. Following this, the optimal time for retrograde ferry node reset is calculated by adding half of the time between prograde resets to the most recent prograde reset time as

$$rt_0 = pt_2 + \frac{\Delta pt}{2}, \tag{9}$$

where rt_0 is the optimal time for the retrograde ferry node reset, pt_2 is the most recent prograde reset, and Δpt is the difference between the last two prograde resets. The time between the current retrograde pheromone reset and this optimal point is then calculated as

$$\Delta rt_0 = rt_0 - rt, \tag{10}$$

where Δrt_0 is the time between the optimal retrograde reset time, rt_0, and the current retrograde reset time, rt. Since the optimal time is greater than the current time, this results in a positive Δrt_0, which results in a positive adjustment. Finally, the time difference is used to calculate the required adjustment as

$$A = (\Delta rt_0)(D), \tag{11}$$

where A is the adjustment and D is the decay rate. This adjustment is then added to the standard reset value to place the linear decay along the optimal path.

This mathematical formula was implemented in six steps as follows:

1. Reset the pheromone at the standard value of 1.0.

2. Calculate the time between retrograde resets for use when the prograde pheromone is reset. This step is conducted only if there was a previous visitation time.

3. Calculate the retrograde adjustment using information stored from the previous prograde ferry node visitation.

4. Apply the retrograde adjustment to the standard reset value.

5. Test the computed remaining pheromone value to ensure it falls within acceptable limits. If it does not, reset it to the maximum or minimum. These acceptable limits are set by a window value above and below the reset value.

6. Record the reset time for use in adjusting the opposite-direction ferry nodes.

The pseudocode for the above algorithm is shown in Figure 23.

```
pheromoneRetrograde equals 1.0

if LastRetrogradePheromoneResetTime does not equal 0 then
    TimeBetweenRetrogradeResets equals Current Time minus
     LastRetrogradePheromoneResetTime

    RetrogradeAdjustment equals DecayConstant multiplied by
    ((LastProgradePheromoneResetTime plus
    (TimeBetweenProgradeResets divided by 2)) minus current time

pheromoneRetrograde equals pheromoneRetrograde
    plus RetrogradeAdjustment

if pheromoneRetrograde less than (reset minus window) then
    pheromoneRetrograde equals (reset minus window)

if pheromoneRetrograde greater than (reset plus window then
    pheromoneRetrograde equals (reset plus window)

LastRetrogradePheromoneReset equals Current Time
```

Figure 23. Pseudocode for adjusting node coupling of directional pheromones

2. Mathematical Formulation of the Negative Pheromone Adjustment

The negative pheromone adjustment required by the prograde ferry system of the previous example is depicted in Figure 24. As in Figure 22, the blue line depicts the retrograde pheromone reset pattern, and the red line depicts the prograde pheromone reset pattern. Since the prograde pheromone's reset time occurred after the optimal reset time

halfway between the retrograde pheromone resets, it will require a negative adjustment. The adjustment occurs at the current, prograde reset time of *pt* based on the projected next retrograde pheromone reset time, *rt3*.

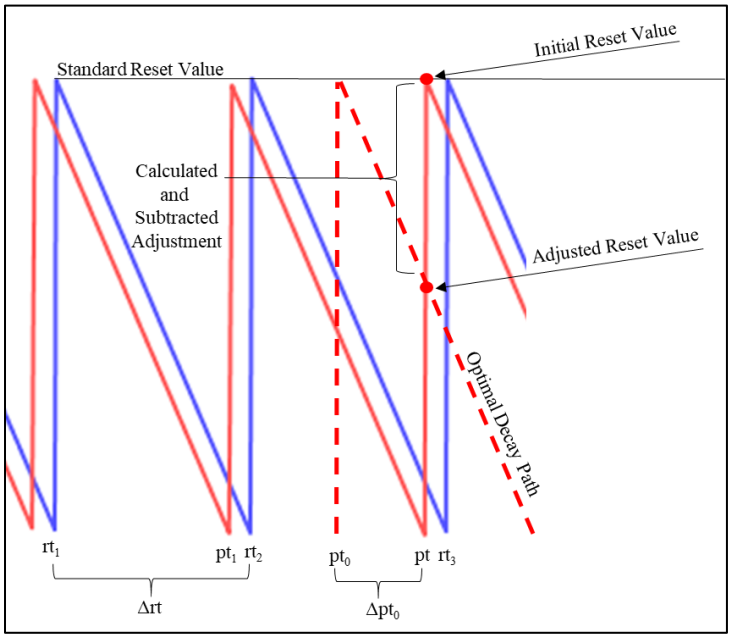


Figure 24. Mathematical formulation of negative adjustment using the prograde formula for the example

The negative adjustment is based on the same concept as the positive adjustment. Since the ferry node arrived at the ground node later than the projected halfway point between the retrograde visits, the newly reset value must be reduced in order to end up on the optimal decay path. The calculation of the adjustment occurs in the same manner as described above in Equations (8) through (11). Since *pt* is greater than pt_0 in the prograde case, the calculated adjustment will be negative. This enables the same algorithm to apply both a negative and positive correction as required based upon whether the ferry node arrives before or after the projected optimal reset location.

3. Effect of Greater Than 1.0 Pheromone on Ferry Node Speed

With the prograde and retrograde pheromones coupled, the adjusted speed of each ferry node after arrival at a ground node is still calculated using Equation (2) from Chapter

II. The only difference is that it is now possible for a pheromone to be greater than 1.0. As shown in Table 4, pheromone values higher than 1.0 result in slower speeds. This is, in fact, the desired result since pheromones with these values indicate too closely spaced ferry nodes.

Table 4. Adjusted ferry node speed based on pheromone value

Pheromone	Adjusted Speed
1.6	0.4
1.4	0.6
1.2	0.8
1	1
0.8	1.2
0.6	1.4
0.4	1.6

D. HAIL/ARSENL INTEGRATION

HAIL provides an abstraction layer to address the fact that UAVs operate using similar categories of information despite the specific UAV model and underlying autopilot. HAIL was integrated for the purposes of this thesis for two reasons. First, it made integration with the SCADS control algorithm easier since it was already designed to operate through HAIL thereby reducing implementation risk. Second, this provides the capability for implementing the dual-direction control algorithm as designed with either another type of UAV or a heterogeneous swarm to support future testing.

A HAIL Bridge was developed to link HAIL with the ARSENL control mechanisms. This was similar to the other bridges mentioned in the ARSENL network description section. It converts UAV state information into formats that HAIL can pass to the SCADS algorithm, and it converts waypoint and speed commands from HAIL to a format that is understandable by ARSENL's on-UAV software.

To effect this change, two nodes were added to the ROS diagram of Figure 14. The first is the /hail_bridge which performs the required format conversion functions. The second is the /ARSENLInterface, which publishes waypoint and speed, commands from HAIL for relay to the HAIL Bridge. These ROS nodes communicate by publishing and

subscribing messages to the topics described in Table 5. Figure 25 depicts these new nodes and topics.

Table 5. HAIL Bridge topic description. Source: Stevens (2016).

Topic	Publisher	Subscriber	Message	Description	Update Frequency
/attitude	/hail_bridge	/ARSENLInterface	MAVAttitude	ground_track, roll, pitch, yaw, etc.	> 1 per second
/acceleration	/hail_bridge	/ARSENLInterface	MAVAcceleration	accel_ground_track, NED accels.	> 1 per second
/command	/ARSENLInterface	/hail_bridge	MAVCommand	commands from HAIL: waypoint, speed, stop, land, takeoff	As needed, max 1 Hz.
/command_ack	/hail_bridge	/ARSENLInterface	MAVCommandAck	Acknowledgement of receipt of command and response code	On receipt of command
/position	/hail_bridge	/ARSENLInterface	MAVGeoPosition	UAV Position / Position Accuracy Information	> 1 per second
/home_position	/hail_bridge	/ARSENLInterface	MAVHomePosition	UAV Home Location: Geo-referenced	On change
/limits	/hail_bridge	/ARSENLInterface	MAVLimits	UAV Capability limits: max speed, range, ceiling, geofence, etc.	On change and once/minute
/power	/hail_bridge	/ARSENLInterface	MAVPower	UAV Battery status	30 seconds
/state	/hail_bridge	/ARSENLInterface	MAVState	UAV state information	On change
/timesync	/hail_bridge	/ARSENLInterface	MAVTime	Time sync message to allow local system time to sync to GPS time	60 seconds
/velocity	/hail_bridge	/ARSENLInterface	MAVVelocity	UAV Velocity: ground speed, air speed	> 1 per second

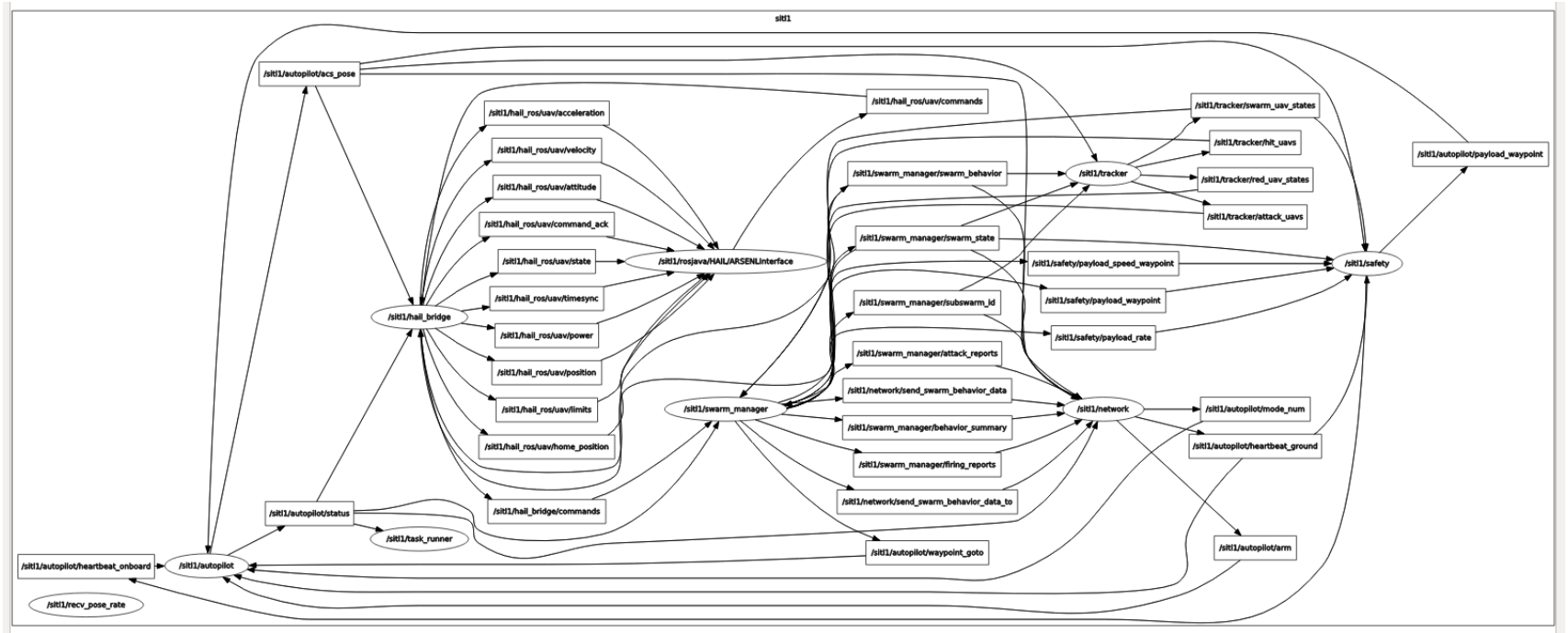


Figure 25. ROS nodes and topics with HAIL Bridge

Figure 26 depicts the ARSENL to HAIL integration architecture. For the purposes of SITL testing, all items in the figure were run on one computer. For the planned live-fly, the UAVs (green rectangles) were individual Zephyr II UAVs, and the ground nodes (blue rectangles) were simulated on a control computer.

After the UAVs reach their holding position, the ARSENL Swarm Manager is used to initiate the HAIL Behavior. Upon initiation, the UAVs begin movement towards their first ground node. Once they arrive at the ground node, the speed adjustment algorithm changes the UAV speed based on the residual pheromone level read from the ground node, and it signals that the ground node pheromone should be reset at the ground node pheromone control algorithm. The pheromone control algorithm is responsible for decaying the pheromone based on the preset decay rate. If the ground node is the adjusting node, it will calculate and apply the necessary adjustment upon pheromone reset.

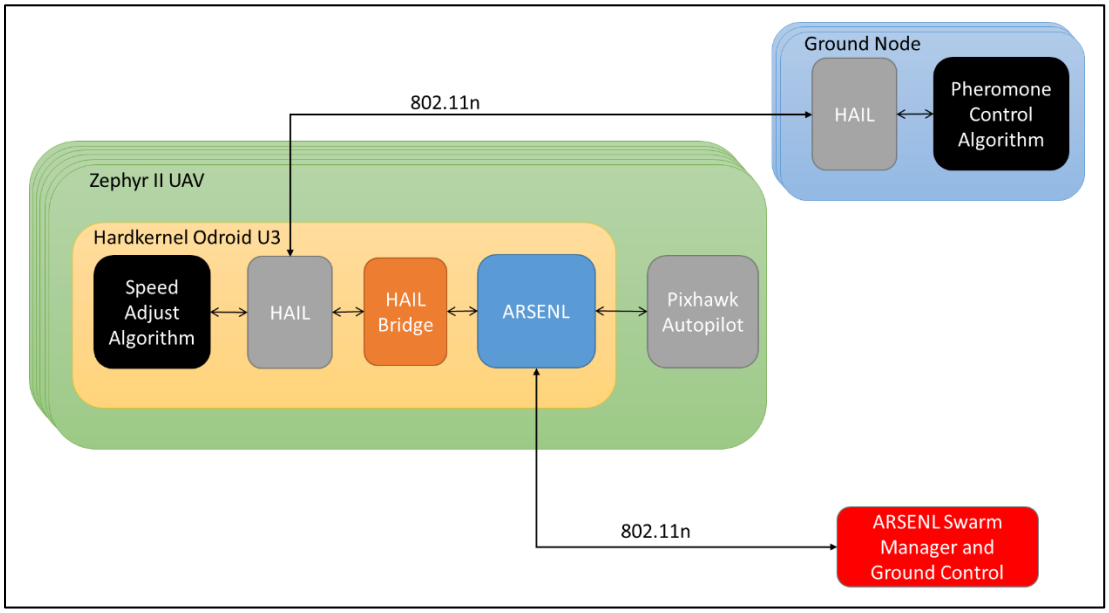


Figure 26. ARSENL/HAIL interface

E. CHAPTER SUMMARY

A system employing dual-direction ferry nodes can be expected to provide faster message delivery than one employing single-direction ferry nodes due to the ability for

messages to travel along the shortest part of the perimeter in a polygon-based topology. With the mathematical relationship linking the two pheromones described in this chapter, the ferry nodes (inclusive of counter-rotating ferry nodes) can be expected to autonomously space themselves evenly along the links connecting the ground nodes thereby allowing for predictable message delivery times. With the addition of the HAIL Bridge, the existing SCADS algorithm was modified to incorporate dual-direction ferry nodes using the ARSENL swarm. The next chapter describes the testing methodology to validate performance assumptions as well as the results of the tests performed.

IV. THESIS RESULTS

A. CHAPTER OVERVIEW

This chapter reviews the results from testing the dual-direction pheromone concept discussed in Chapter III. The mathematical relationship was first tested in simulation using Fraser's original MASON code. This was done to answer three questions. First, would varying the size of the applied adjustment result in faster convergence? Second, would varying the maximum and minimum allowed pheromone values after reset result in faster convergence? Third, would the relationship result in converged visitation time for different ground node layouts? Following this, the algorithm was verified through SITL simulation in preparation for live-fly experimentation.

B. MASON EXPERIMENT DESCRIPTION

As described in Chapter III, we modified the original MASON code developed by Fraser for the purpose of this thesis. To keep the experiments consistent across multiple ground node layouts, we had to standardize the decay rate, the type of ground nodes used, and the collection method of converged time. To represent a severely contested RF environment, the ferry nodes were only able to communicate with the ground nodes if they were collocated.

1. Decay Rate

With varying the length of perimeters during this experiment, the determination of the pheromone decay rate was key. This is the rate that each pheromone is reduced during each time-step. In previous work, Fraser et al. used a decay rate of one divided by the length of a side (Fraser & Hunjet, 2016). Due to our plan to test the algorithm in situations where the sides were not of equal length, we chose to keep the relationship between the perimeter length and decay rate consistent among all of the scenarios. The decay rate was calculated as

$$D_r = \frac{(n)(s_{\min})}{P} \quad , \tag{12}$$

where D_r is the decay rate, n is the number of ferry nodes in one direction, s_{min} is the minimum allowable ferry speed assuming the use of fixed-wing UAVs, and P is the perimeter length. With this decay rate, one direction's evenly distributed ferry nodes traverse the entire perimeter at their minimum speed in the time it takes a pheromone to decay to zero. This reduces the likelihood that the pheromone will decay to zero before the next ferry node visits that ground node. A pheromone, which has decayed to zero, cannot inform the following ferry nodes about the relative distance to another ferry node, and the system cannot make appropriate adjustments.

2. Ground Node Types

The need for different types of ground nodes was the first finding of this thesis. The initial tests of the ground nodes with linked pheromones resulted in evenly distributed ferry nodes, as shown in Figure 27. This even distribution showed that the mathematical coupling described in Chapter III resulted in evenly distributed ferry nodes with a converged visitation time for the ground nodes. Visitation time is defined as the time between successive arrivals of ferry nodes at a given ground node independent of their direction.

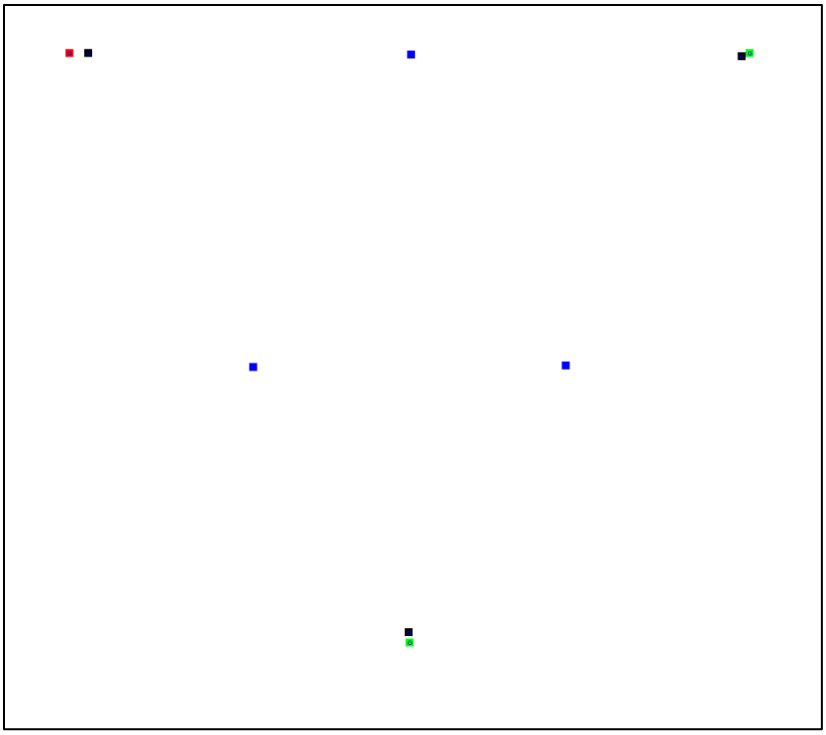


Figure 27. Equilateral triangle layout with all adjusting ground nodes

This converged visitation time failed to occur when the ground node layout did not have equal length sides. The first ground node layout without sides of even lengths tested was the scalene triangle shown in Figure 28. With this ground node layout, the visitation time failed to converge for any of the three ground nodes, as shown in Figures 29, 30, and 31.

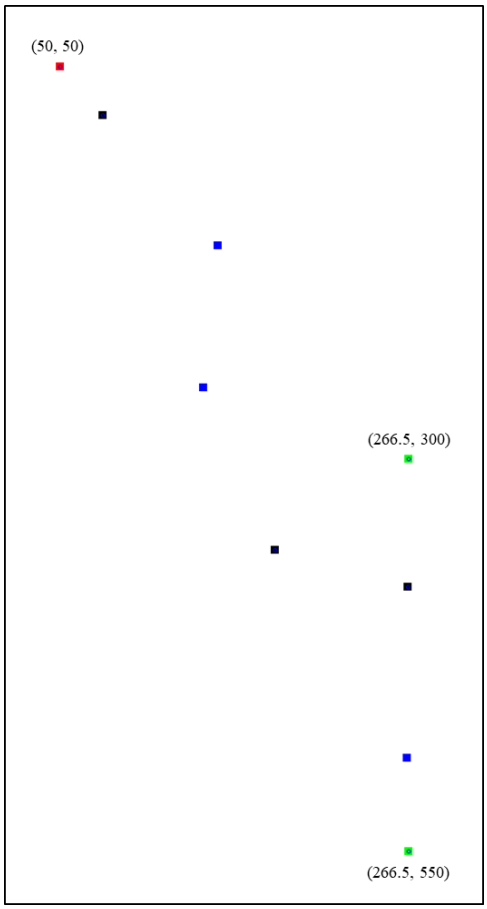


Figure 28. Scalene triangle layout with all adjusting ground nodes

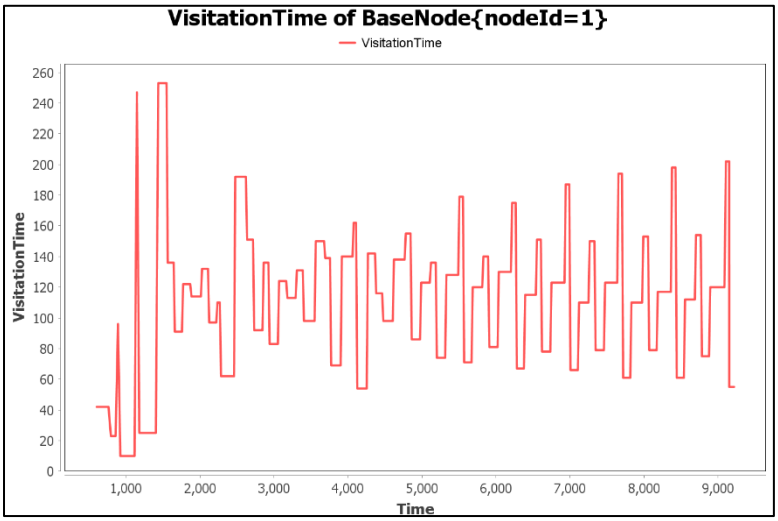


Figure 29. Non-converging visitation time for Ground Node One in scalene triangle

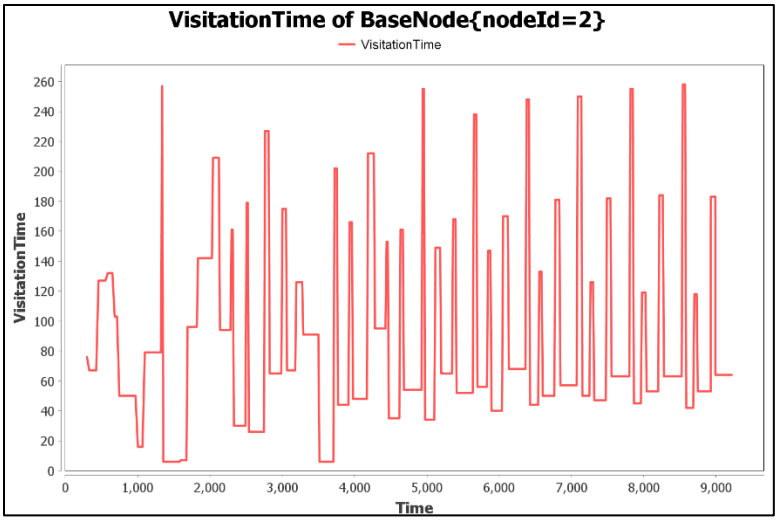


Figure 30. Non-converging visitation time for Ground Node Two in scalene triangle

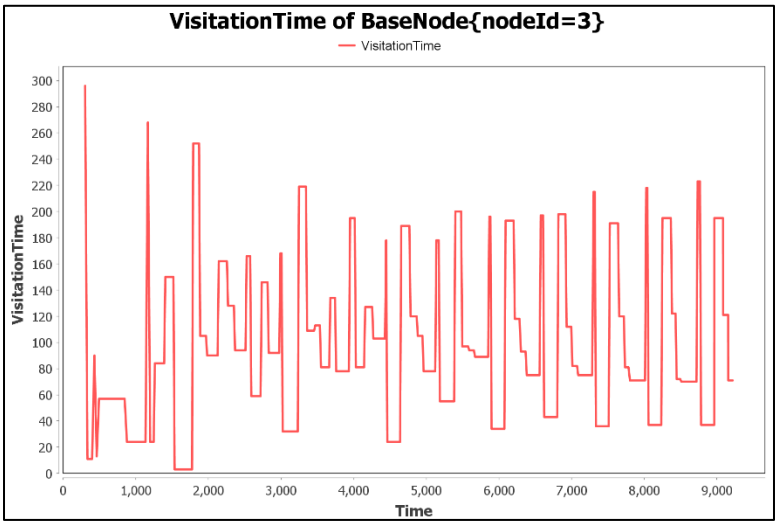


Figure 31. Non-converging visitation time for Ground Node Three in scalene triangle

We determined that this change was due to the different dynamics of a dual-direction communication ferry system as compared to a single-direction ferry system; in a single-direction ferry system, converged visitation time applies equally to all points along the perimeter since the next visitation is solely based on the time of flight for the next ferry node. As shown in Figure 32, each time Ferry Node Number One passes a location on the perimeter, Ferry Node Number Two is always the same distance behind it.

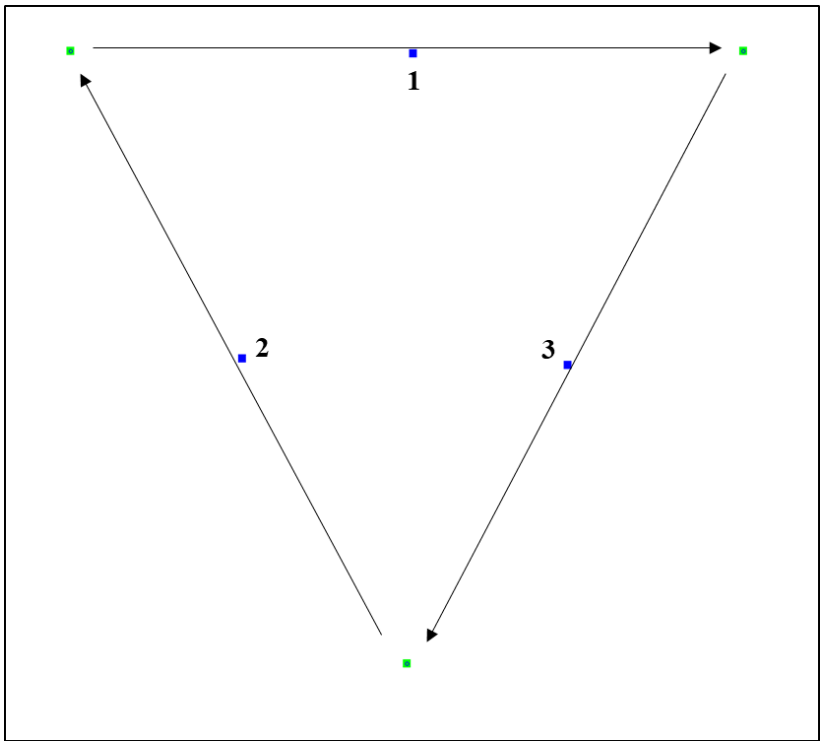


Figure 32. Single-direction ferry node movement

This is not the case for a dual-direction ferry node system. Converged visitation time only occurs at specific points along the perimeter. These are the locations where the evenly dispersed ferry nodes are located when one is at the adjusting node. An example for the equilateral triangle ground node layout is shown in Figure 33. These specific points are where the ferry nodes (depicted with black and blue squares) are located.

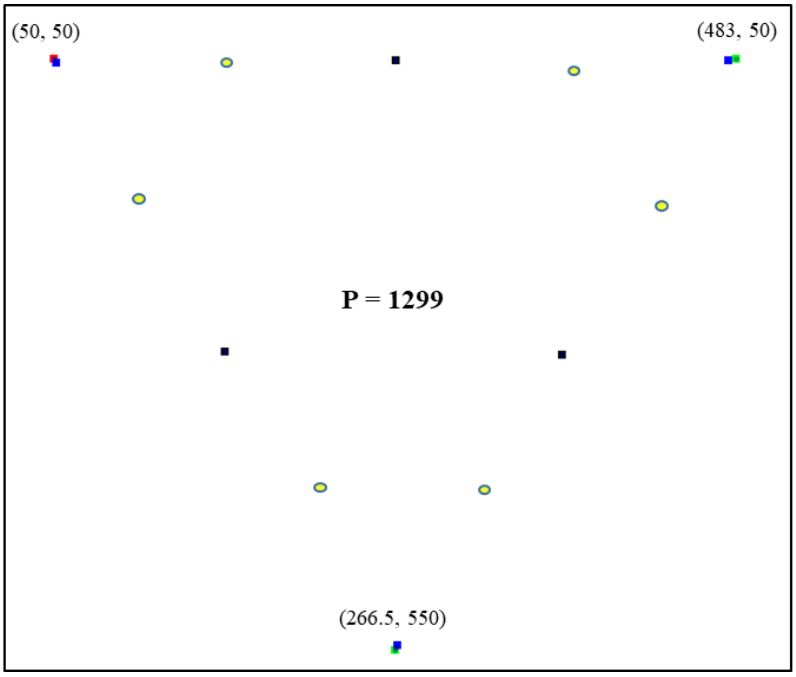


Figure 33. Dual-direction with evenly dispersed ferry nodes at converged visitation time specific points

After this moment, the nodes move towards one another until they pass at the yellow circles as depicted in Figure 34. Passage also occurs at the same distance on the opposite side of the ground nodes as shown in Figure 35.

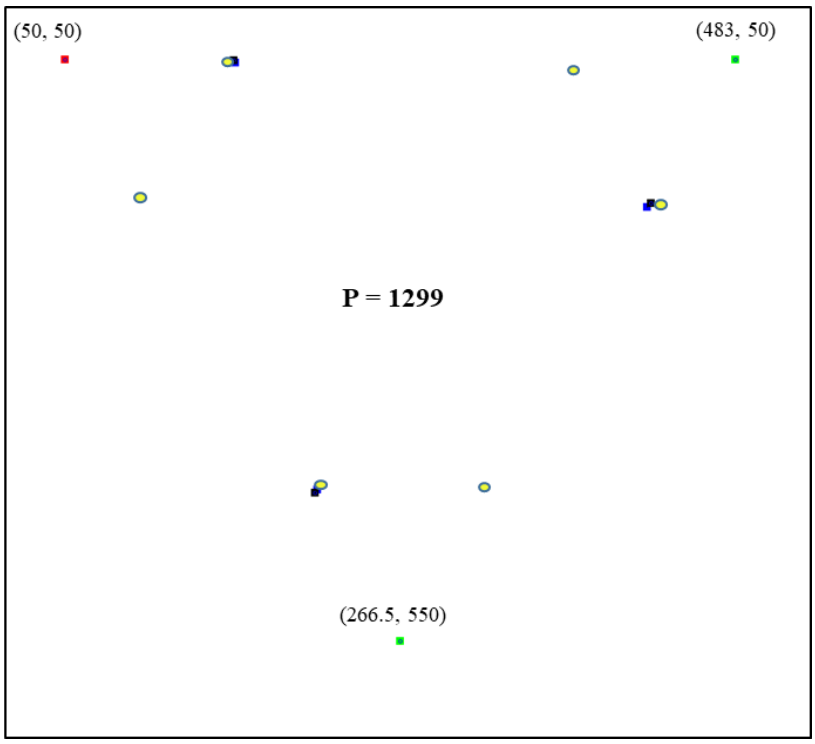


Figure 34. Passing ferry nodes at first passing location

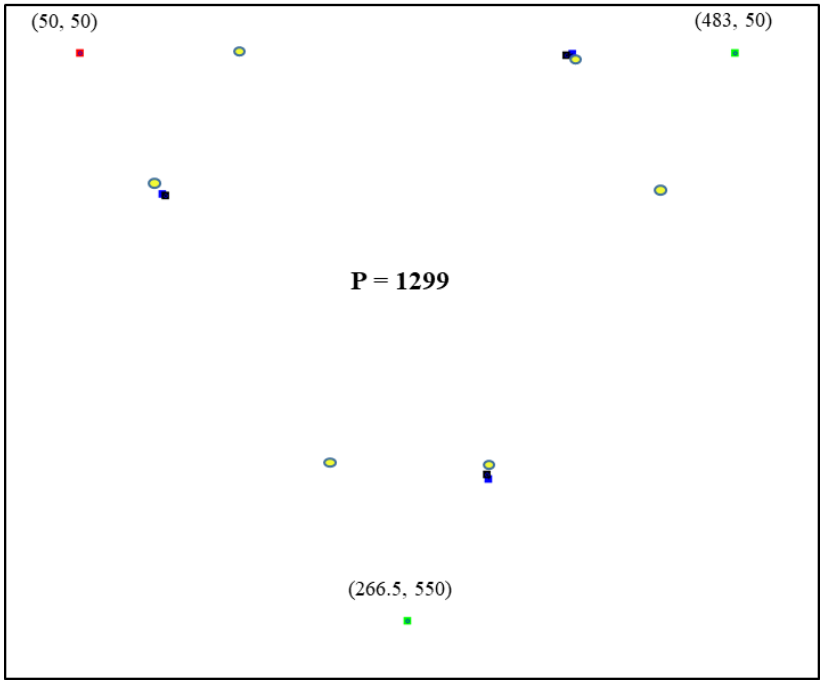


Figure 35. Passing ferry nodes at second passing location

Therefore, any ground nodes not located at one of the equilibrium locations around the perimeter that link the pheromones are inevitably directing the system into an unconverged state as Figures 29, 30, and 31 indicate.

This understanding led us to implement the bi-directional pheromone coupling on only one ground node rather than on all ground nodes. We refer to this type of ground node as an adjusting node because it applies the bi-directional coupling adjustment to the pheromone values. It is identified with a red square in figures depicting MASON simulations. Other ground nodes only reset the directional pheromones to the standard 1.0 value as in Fraser's work described in Chapter II. These ground nodes are referred to as resetting nodes and are shown as green squares in figures depicting MASON simulations. Using this configuration, the simulation was run again for the scalene triangle depicted in Figure 28. This simulation resulted in a converged visitation time for the adjusting node as shown in Figure 36.

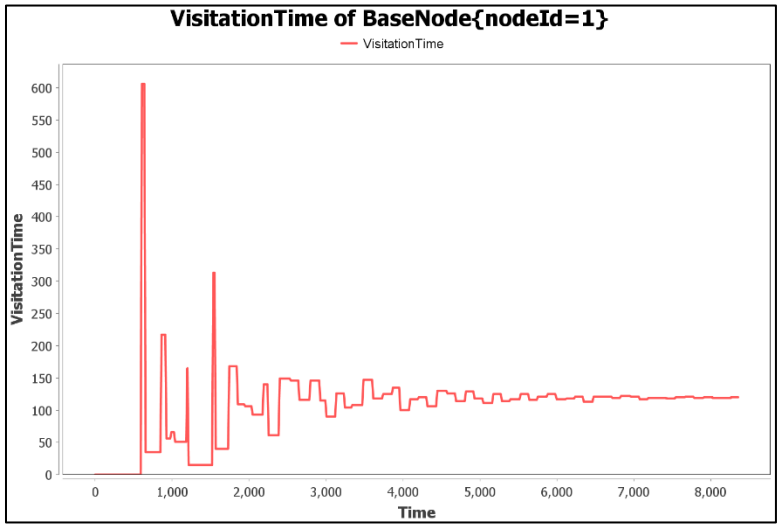


Figure 36. Converged visitation time for adjusting node in scalene triangle ground node layout

The other two nodes ended up with alternating visitation times as shown in Figures 37 and 38. The amplitude of this alternating visitation time appeared to be based on the

ground node's proximity to a position on the perimeter where the ferry nodes pass one another.

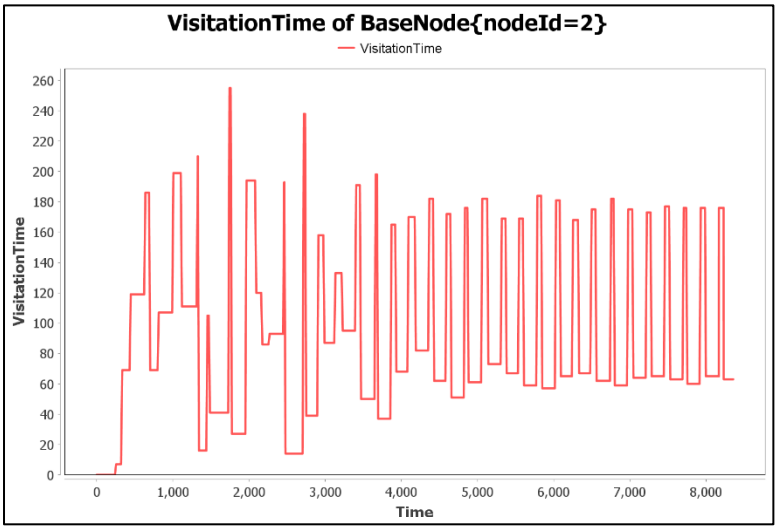


Figure 37. Alternating visitation time for Ground Node Two in scalene triangle

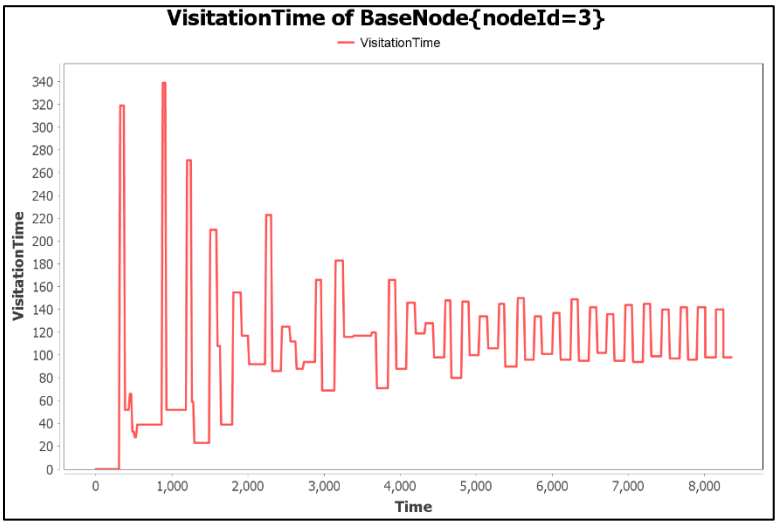


Figure 38. Alternating visitation time for Ground Node Three in scalene triangle

To test this, we returned to the equilateral triangle example and placed a ground node at one of these "passing" locations, as shown in Figure 39 circled in red. As expected, the visitation time for this ground node at the passing location reached a peak visitation

time and was immediately reset at a minimum value as the next ferry node almost immediately passed its location, as shown in Figure 40.

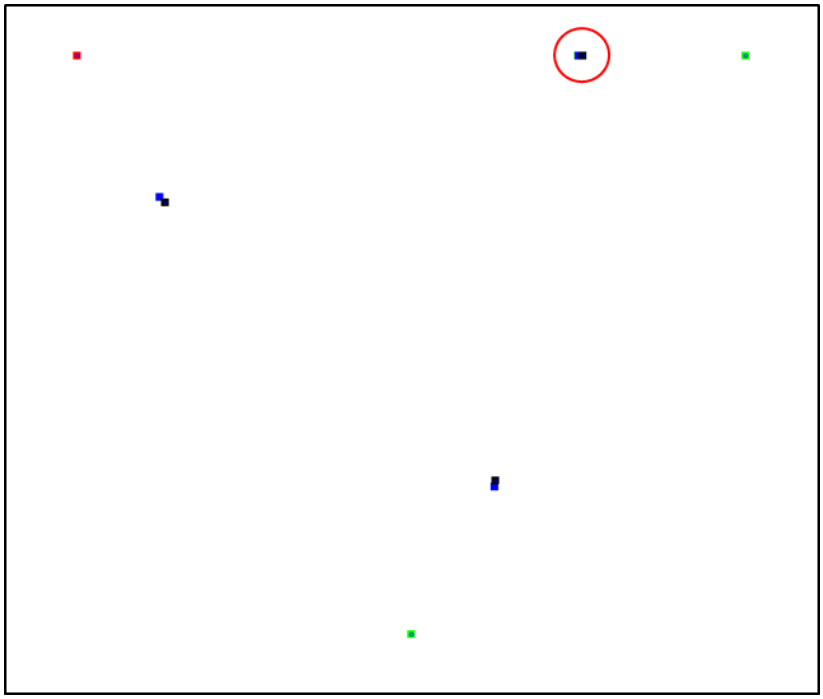


Figure 39. Ground node positioned at ferry node passing location

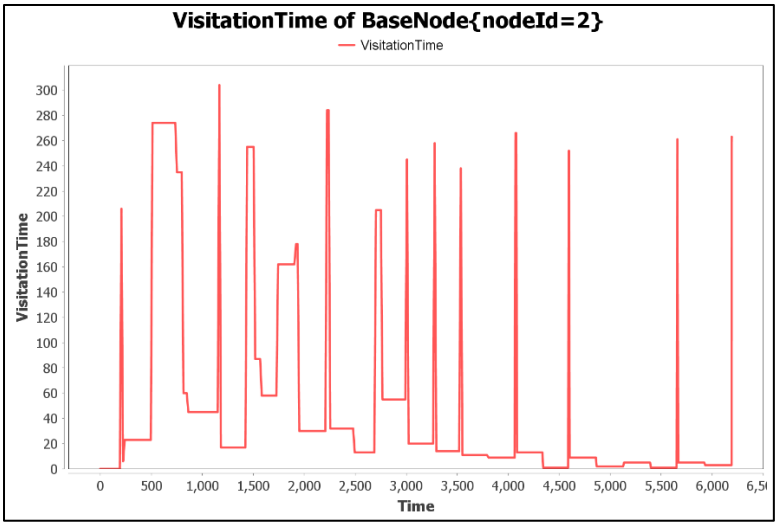


Figure 40. Visitation time of ground node placed at ferry node passing location

From this, we concluded that the visitation time of a dual-direction ferry node system is based upon the perimeter length and the number of directional ferry nodes in the system. The converged visitation time occurs at the points on the perimeter where the ferry nodes end up evenly distributed. These occur at multiples of

$$\frac{1}{2n} \tag{13}$$

of the perimeter, where n is the number of directional ferry nodes.

To verify this, a simulated nine-ground node layout was used to predict another location on the perimeter where the visitation time converged as it does at the adjusting node. The position of the ninth ground node in Figure 41 was placed according to Equation (13) at (200, 376.86). MASON simulation results shown in Figures 42 and 43 demonstrate the expected visitation time convergences at both the adjusting node and the predicted location.

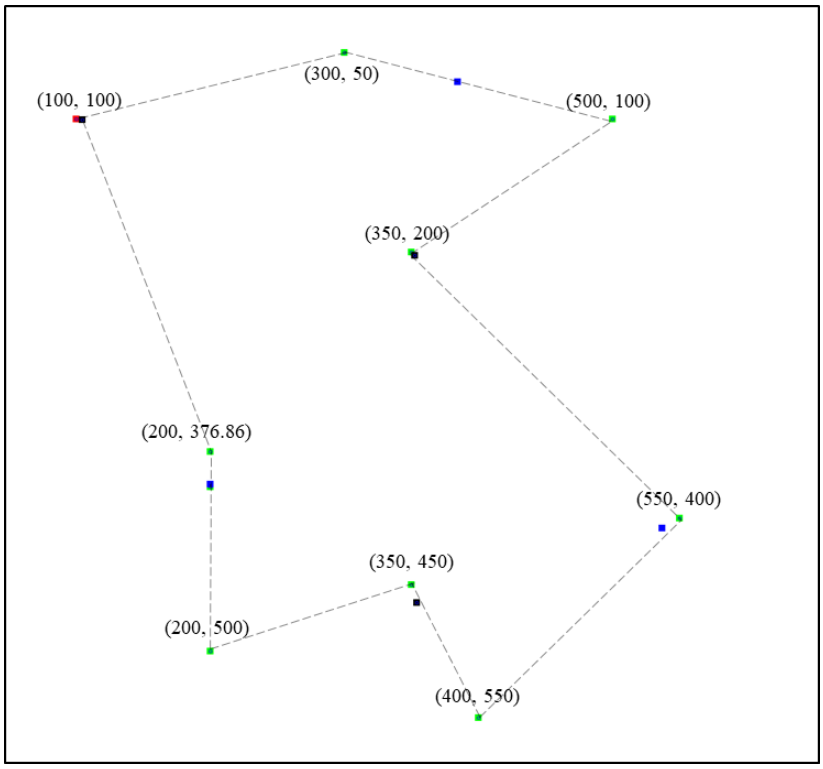


Figure 41. Predicted location where visitation time converges for nine ground node layout

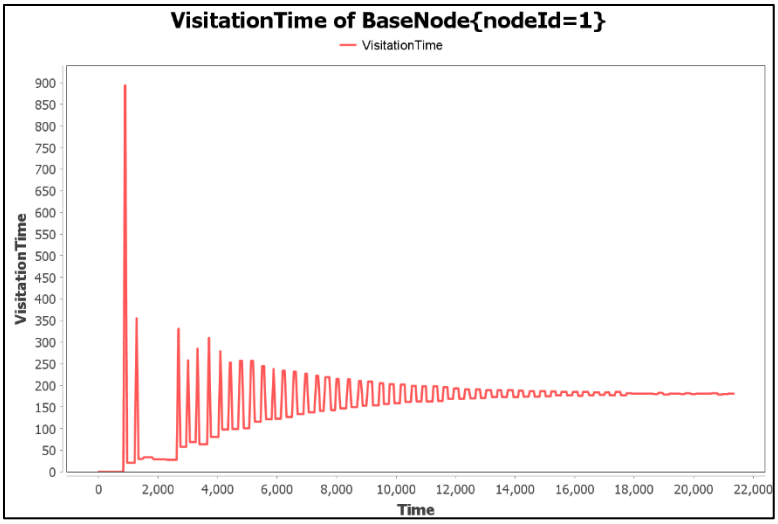


Figure 42. Converged visitation time at the adjusting ground node (Ground Node One)

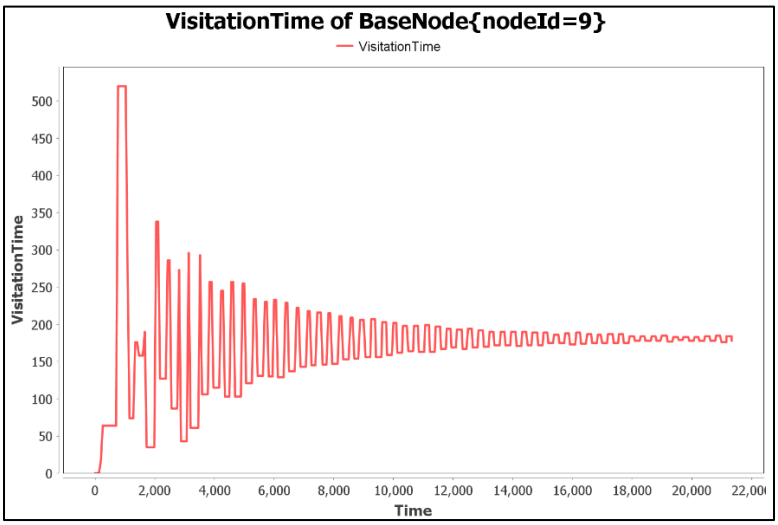


Figure 43. Converged visitation time at the resetting node located according to Equation (13) (Ground Node Nine)

The visitation time at these locations can be estimated as

$$Vt_{conv} = \frac{P}{(s_c)(2n)}, \tag{14}$$

where Vt_{conv} is the converged visitation time, P is the perimeter length, s_c is the converged ferry node speed, and n remains the number of directional ferry nodes. Actual visitation time will vary as the ferry node speed changes at each visitation. Once the system

converges, the visitation time will approach the estimated converged value, but perturbations within the system will still lead to slight variations.

We tested the maximum visitation times in a similar manner. Based on the previous analysis, they occur at locations along the perimeter where the ferry nodes pass one another. These passages occur at multiples of

$$\frac{1}{4n} \tag{15}$$

of the perimeter length which are not also locations covered by Equation (13). To confirm that maximum visitation time would occur at these locations, we used Equation (15) to compute the placement of a tenth ground node along the perimeter at (154.99, 237.47), as shown in Figure 44.

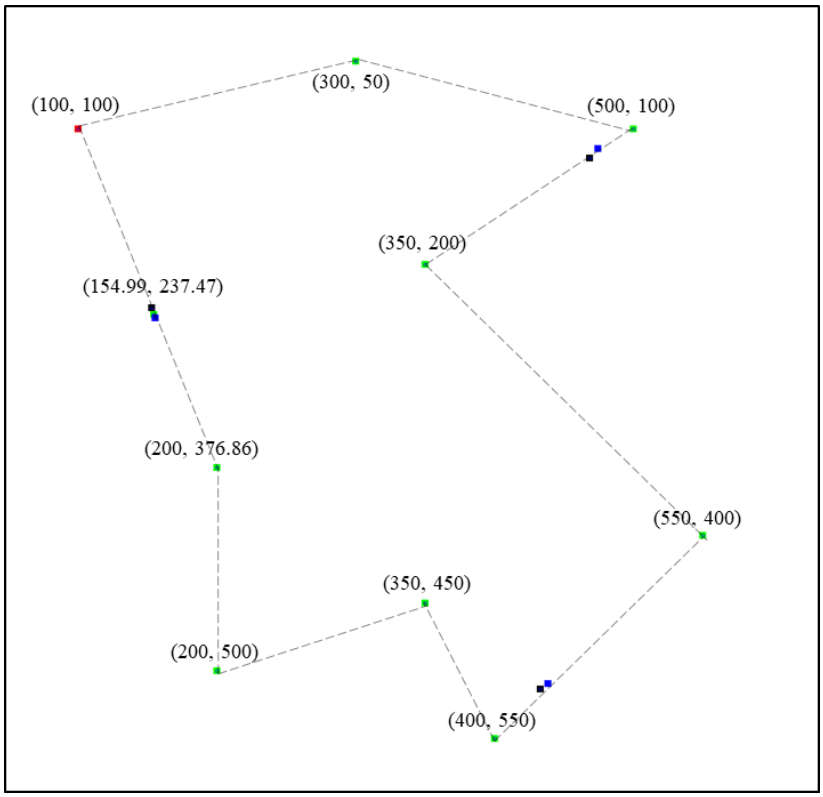


Figure 44. Maximum visitation time confirmation at ground node placed using Equation (15) (Ground Node Ten)

As expected, the MASON simulation resulted in visitation time convergence for Ground Node One, the adjusting node, as shown in Figure 45. It also confirmed that the visitation time reached a maximum at Ground Node Ten, as shown in Figure 46.

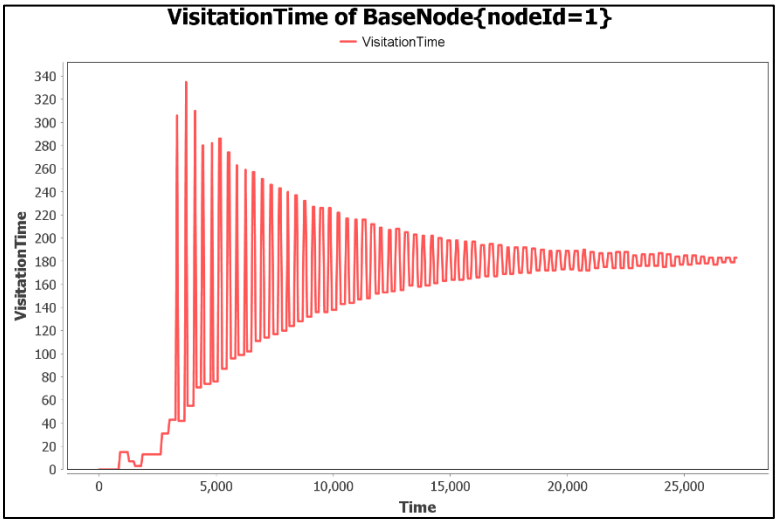


Figure 45. Converged visitation time at Ground Node One

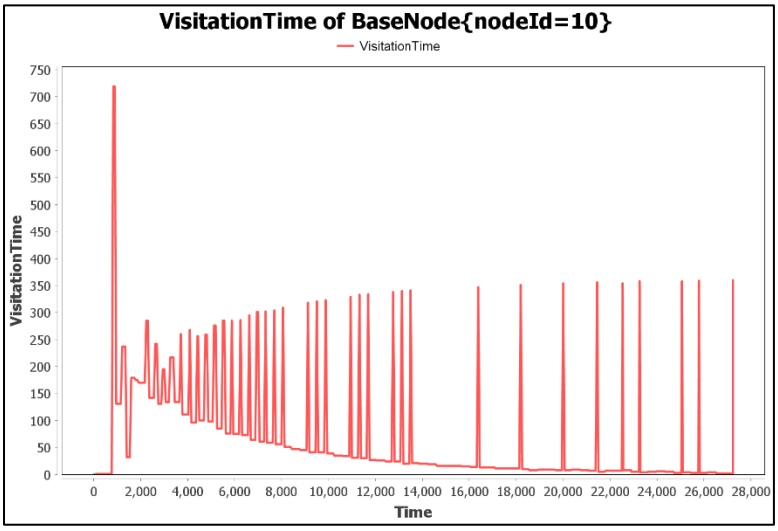


Figure 46. Maximum visitation time at Ground Node Ten placed using Equation (15)

The visitation time at these maximum-time locations can be estimated as

$$Vt_{\max} = \frac{P}{(s_c)(n)}, \tag{16}$$

where Vt_{max} is the maximum visitation time, P is the perimeter length, s_c is the converged speed, and n is the number of directional ferry nodes. As with the converged time, this will vary as ferry node speed changes at each visitation. Nevertheless, Equations (14) and (16) can be used to describe the expected performance for a given system.

3. Converged Time Collection

To facilitate effective comparison of different adjustment sizes and pheromone limits across different ground node layouts, we defined a converged system, as one where the ferry nodes are equidistant along the perimeter at the moment a ferry node is located at the adjusting node. Convergence, as defined, occurred when the visitation times at the primary ground node were approximately equal. To collect this data, we used the visitation time delta function in the original code developed by Fraser (Hunjet et al., 2018). This function measures the change in visitation time as each ferry node passes a ground node. A consistent value of zero time-steps indicates that the ferry nodes are evenly spaced. Due to perturbations in the system, this value never remained strictly at zero so we considered convergence achieved upon observation of three consecutive visitation times below fifteen time-steps. Figure 47 depicts typical ferry node dispersion once this condition is met, and Figure 48 shows the associated visitation time graph.

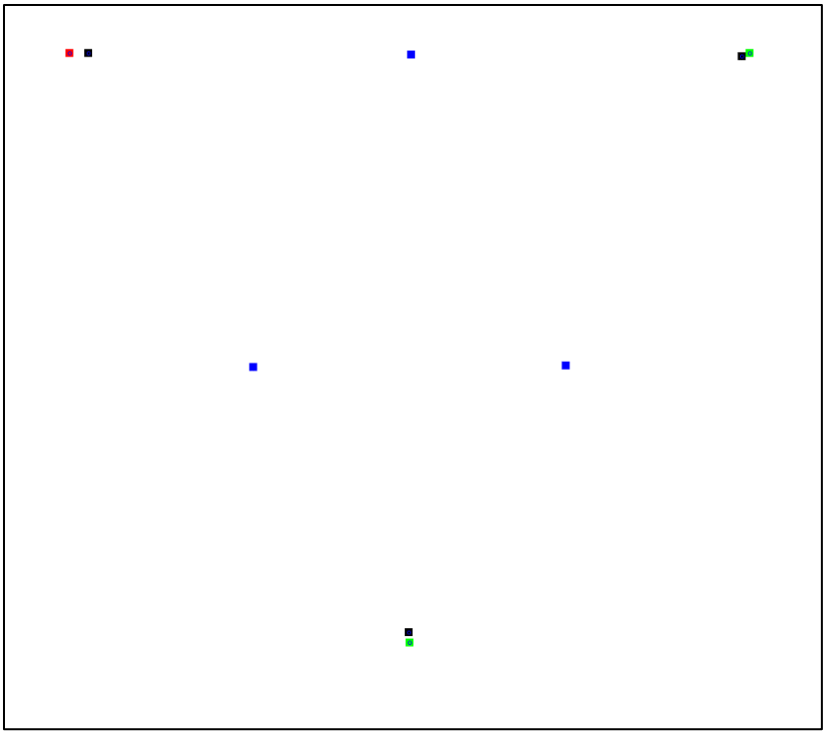


Figure 47. Ferry node positions (black and blue squares) at convergence time collection

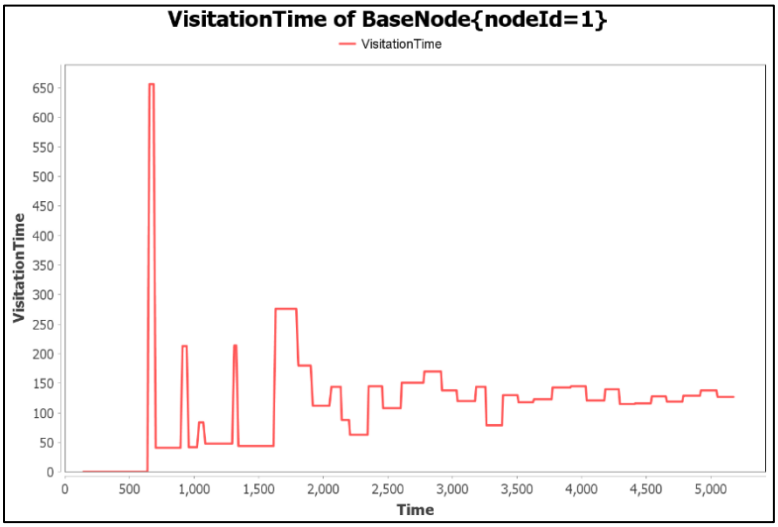


Figure 48. Visitation time graph at convergence time collection

To collect this time for each run, the MASON code was modified to track the number of consecutive visitation times under fifteen time-steps by implementing the pseudo-code in Figure 49.

```
if absolute value VisitationTimeDelta less than 15
        undervalue equals undervalue plus 1
else
        undervalue equals 0

if absolute value VisitationTimeDelta less than 15
and collect is true and undervalue less than 2)
        System Print Current Time
        collect equals false
```

Figure 49. Convergence time collection pseudo-code

C. MASON SIMULATION BULK RUN OVERVIEW

The bulk run functionality of MASON was used to run each combination of parameters 1000 times with a converged time collected for each run. The pheromone coupling presented in this thesis was tested in simulation using these bulk runs for ground node layouts ranging from three to ten ground nodes to determine the optimal parameter combination for inclusion in further testing. Three specific parameters were tested for these ground node layouts: adjustment size, allowable pheromone window, and departure delay. Before reviewing these results, these three parameters, and the thought behind their testing, will be discussed.

D. VARIATION OF THE ADJUSTMENT SIZE PARAMETER

As described in Chapter III, the coupling of the two pheromones seeks to shift the reset times of each direction's ferry nodes by adding or subtracting a calculated adjustment value to the pheromone value after it is reset to 1.0. Since the prograde and retrograde adjustments occur simultaneously, the effects of one adjustment have not fully affected the system by the time the next adjustment is applied. This is shown in Figure 50.

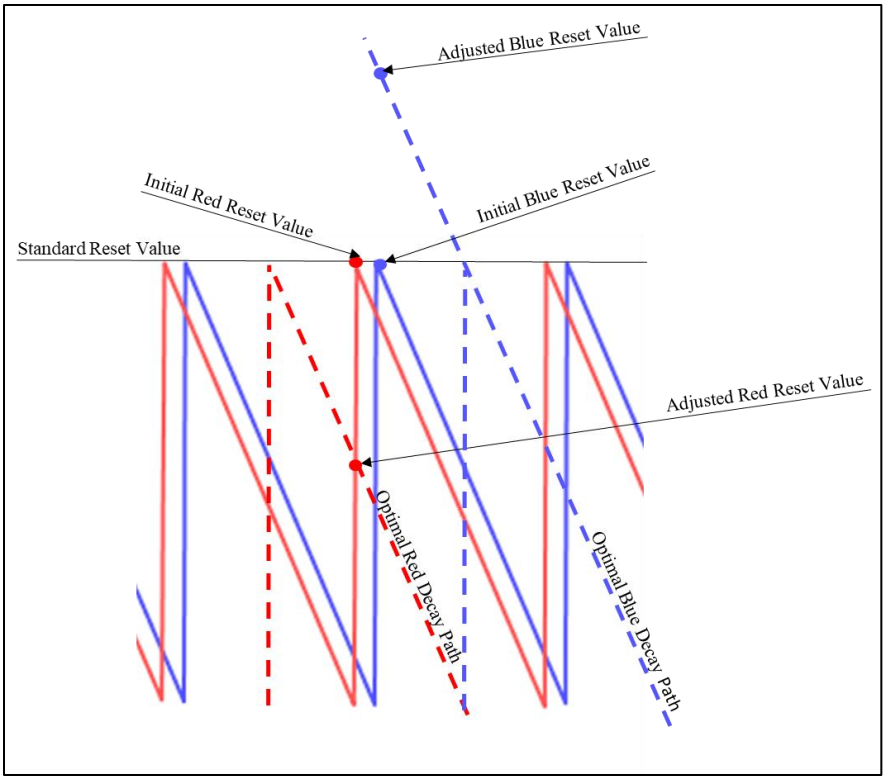


Figure 50. Simultaneous prograde and retrograde pheromone adjustments

To determine if the complete, calculated adjustment resulted in an over-correction or under-correction, adjustment values ranging from 0.8 to 1.6 times the originally calculated adjustment value were tested. The changes to these adjustment sizes are shown in Figure 51. In the figure, A is the calculated adjustment size, A+ represents a magnified adjustment value, and A- represents a reduced adjustment value. This parameter is referred to as “adjustment size” in the results section of this thesis.

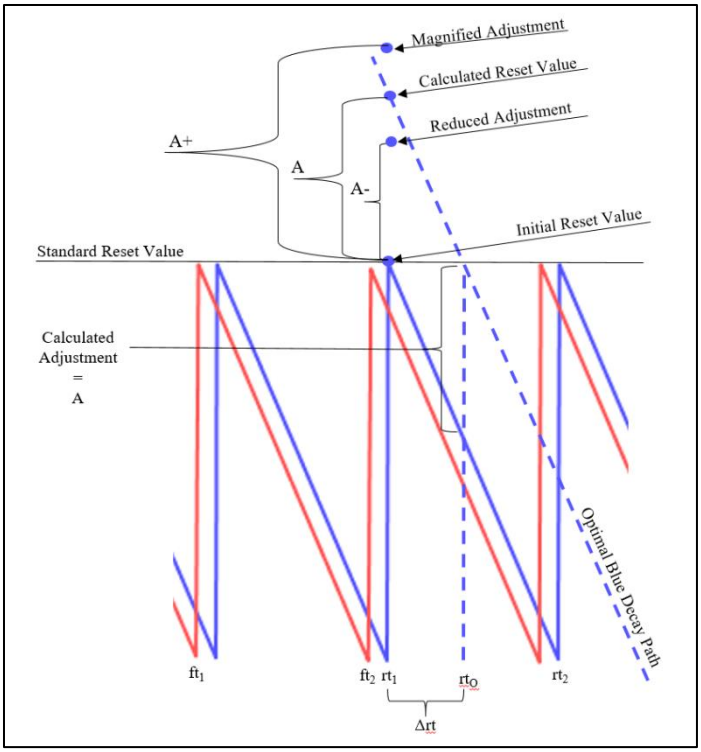


Figure 51. Depiction of magnified or reduced adjustments

E. VARIATION OF THE RESET WINDOW PARAMETER

The second parameter examined in the bulk runs was the size of the reset windows for acceptable pheromone values. These windows provide both an upper and lower bound on the pheromone values. The magnitude of the pheromone adjustment is based upon the difference in ferry node arrival time at the adjusting node and the optimal time, Δpt_0 for prograde nodes and Δrt_0 for retrograde nodes. The largest adjustments occur at the beginning of a simulation because the system is furthest from convergence at startup. In order to minimize overcorrections from these large early adjustments, varying sized windows were tested to determine their effect on the system.

Reset window size was based upon collected pheromone values during a bulk run of 1000 which resulted in the collection of 80,041 pheromone values. Observed pheromone values ranged from 0.131 to 1.481. Windows of 0.3, 0.45, 0.6, 0.75 and 0.9 were used. The 0.3 window was chosen because it would allow pheromone values from 0.7 to 1.3. Based on the observed pheromone values, this window would affect both positive and negative

adjustments. The 0.9 window was chosen because it would allow pheromone values from 0.1 to 1.9. This range would allow the full impact of all adjustments. The others were chosen at even increments between these two values.

The bounds of three of these windows are shown in Figure 52 (the other two are removed for clarity). In this example, the full adjustment would be allowed if the window were set to 0.9 because both the adjusted red reset value and adjusted blue reset value fall within this range. If the window is restricted, these adjusted pheromone values would be adjusted to the corresponding dotted line. This parameter will be referred to as "reset window" in the results section.

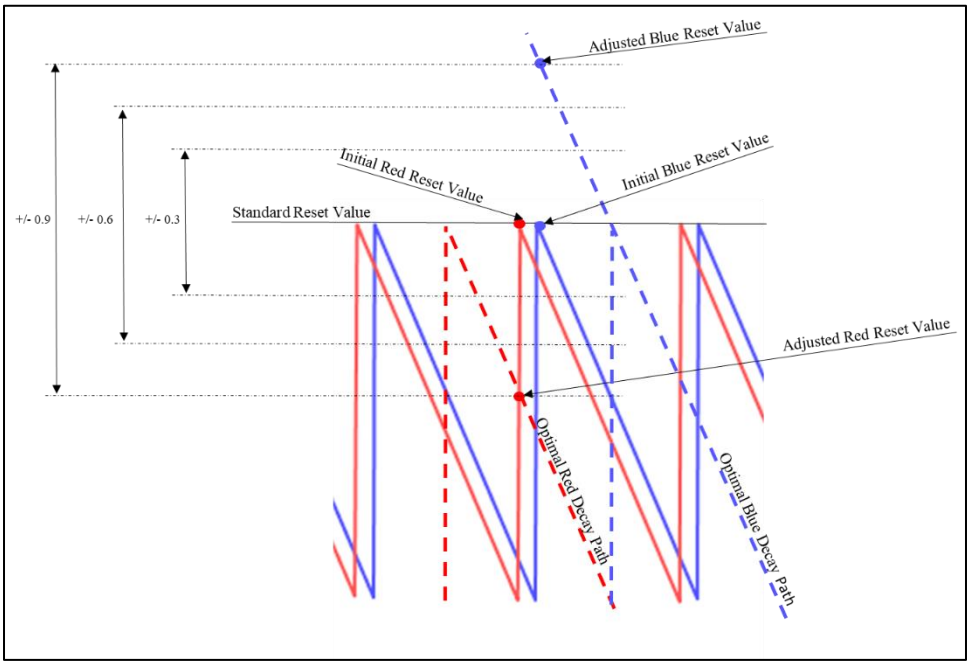


Figure 52. Reset windows bounding pheromone values

F. VARIATION OF THE DELAYED DEPARTURE PARAMETER

During early testing, we noticed that the convergence time appeared to vary based on how closely sequenced the first ferry node from each direction were upon return to the originating ground node. If they returned at approximately the same time, the convergence time appeared higher, and if they were separated in time, the convergence time appeared

lower. To test this, we collected the time between ferry node return and convergence time for 1000 runs with the ferry nodes departing at the same time, 1000 runs with the retrograde ferry nodes delayed by 100 time-steps, and 1000 runs with the retrograde ferry nodes delayed by 200 time-steps. The offset from these delays are shown in Figure 53, departure at the same time; Figure 54, departure with retrograde nodes delayed 100 time-steps; and Figure 55, retrograde nodes delayed 200 time-steps. The plot of the resulting return time separation and the convergence times is shown in Figure 56.

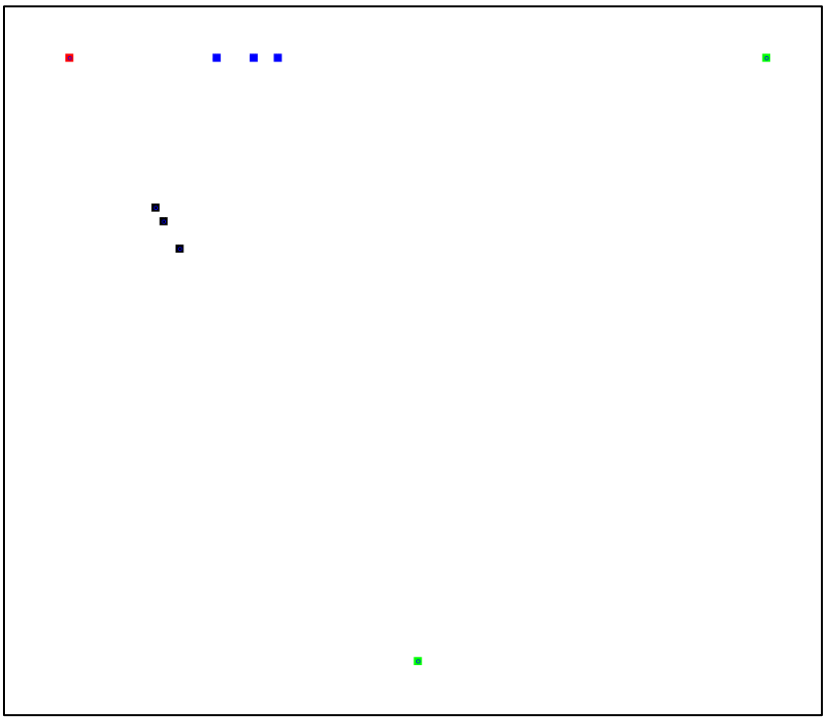


Figure 53. Same departure time for prograde and retrograde nodes

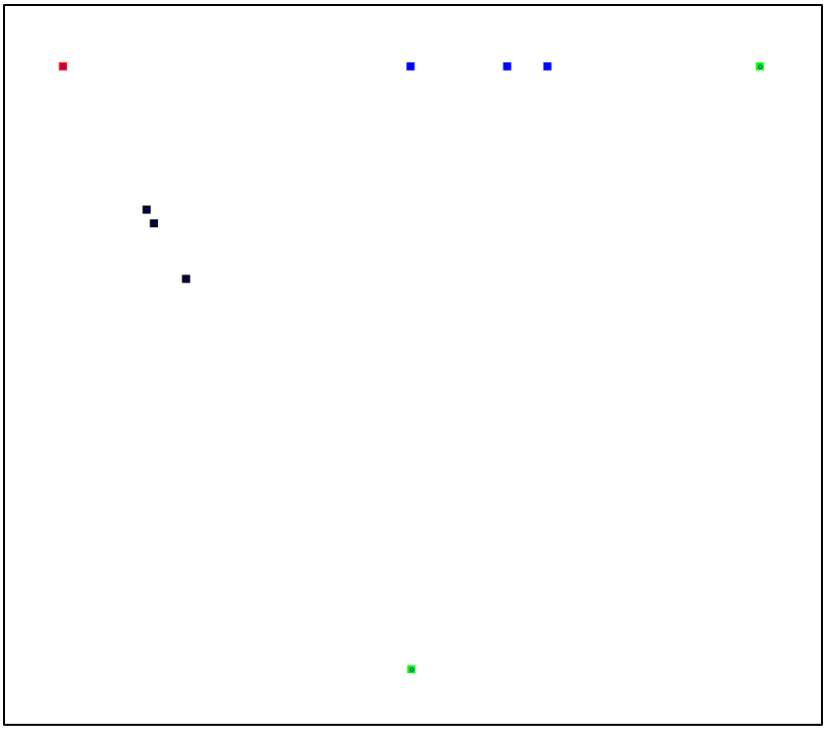


Figure 54. Departure of retrograde nodes delayed by 100 time-steps

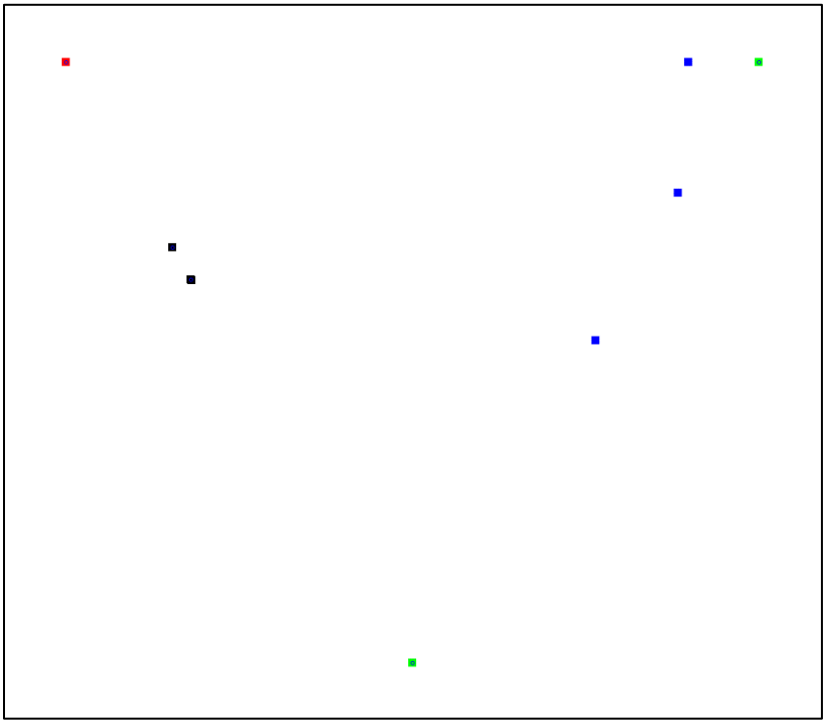


Figure 55. Departure of retrograde nodes delayed by 200 time-steps

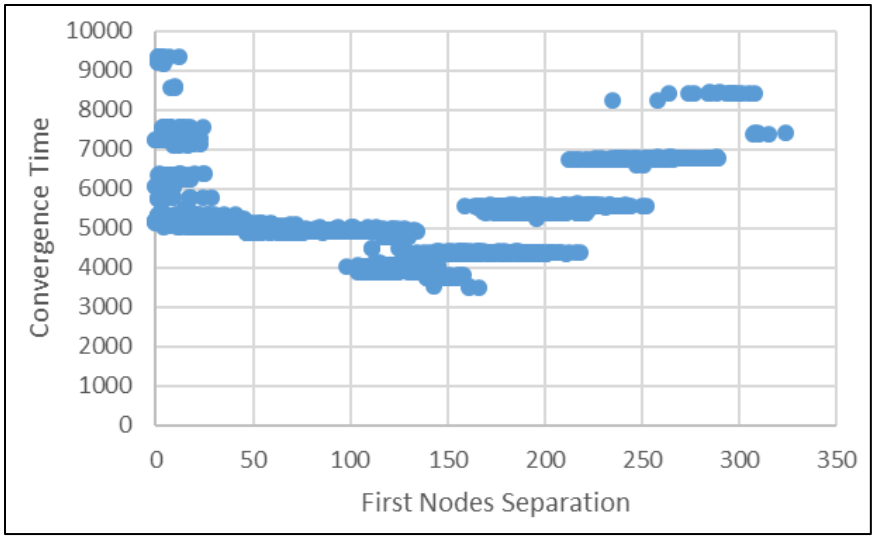


Figure 56. Convergence results from same departure, delayed departure of 100 time-steps, and delayed departure of 200 time-steps

As expected, the convergence times were faster when there was a time difference in the return to the originating ground node between the first ferry node from each direction. Surprisingly, there was an ideal separation distance. If the delay was too large, the convergence times increased as if they had returned at the same time. This idea was incorporated into the testing of three to six ground node layouts in an attempt to minimize the effect of the large returns on the data sets as we determined the optimal parameter combination. As expected, a bulk run with a refined delay time reduced the number of large returns as shown by Figure 57. Time constraints prohibited us from delayed departure testing on the seven to ten ground node layouts. This parameter will be referred to as "departure delay" in the results section.

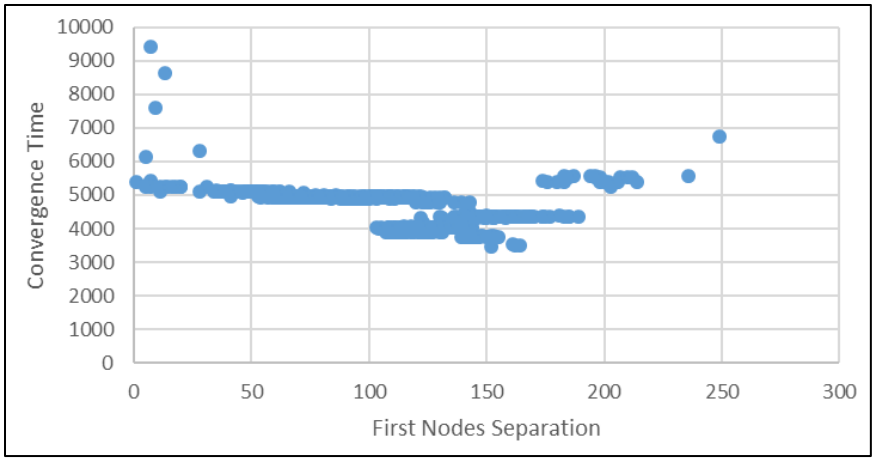


Figure 57. Plot of convergence times with optimal departure delay of 91

G. GROUND NODE LAYOUTS

As previously mentioned, the linked pheromone algorithm was tested on ground node counts ranging from three to ten. Ground node layouts ranged from geometric based to random. The perimeters ranged in length from 1445.58 to 2886.80 units. In each situation, three ferry nodes traveled in each direction. MASON simulation results for each ground node layout are discussed below. Each configuration-specific section below includes a ground node layout, a figure depicting linked pheromones and converged visitation time from one run, the optimal parameter combination, and the delayed departure visitation time results for ground node layouts three, four, five, and six. For each run, the convergence time was measured using the pseudo-code depicted in Figure 49. This resulted in 1000 convergence times for each combination of parameters. Tabular results summarize the 1000 runs for each combination of parameters in the sections below.

The JMP software package, developed by SAS, was used to perform statistical analysis of MASON simulation results for each ground node layout to compare parameter combinations across ground node layouts to find the combination that both applied to all ground node layouts and resulted in the fastest convergence. We first conducted a one-way Analysis of Variance (ANOVA) between the datasets for a given ground node layout to test if at least two means are significantly different. Then, after ANOVA rejected the null hypothesis, we used a Tukey Kramer Honest Significant Difference (HSD) for comparing

all pairwise means with the correction of the Type I error rates for comparing multiple pairs of means. In the following tables, the lowest statistically different average was assigned a value of 1 and the next highest was assigned a value of 2. If an average fell into both categories, it was assigned a value of 1.5. Complete results of this JMP analysis and the statistically different comparisons can be found in Appendix A under the connecting letters report for each ground node layout.

1. Three Equally Spaced Ground Nodes

Figure 58 depicts the equally spaced three ground-node layout with its associated perimeter. Figure 59 is an example pheromone chart from the adjusting node while Figure 60 charts the ferry node visitation time at the adjusting node.

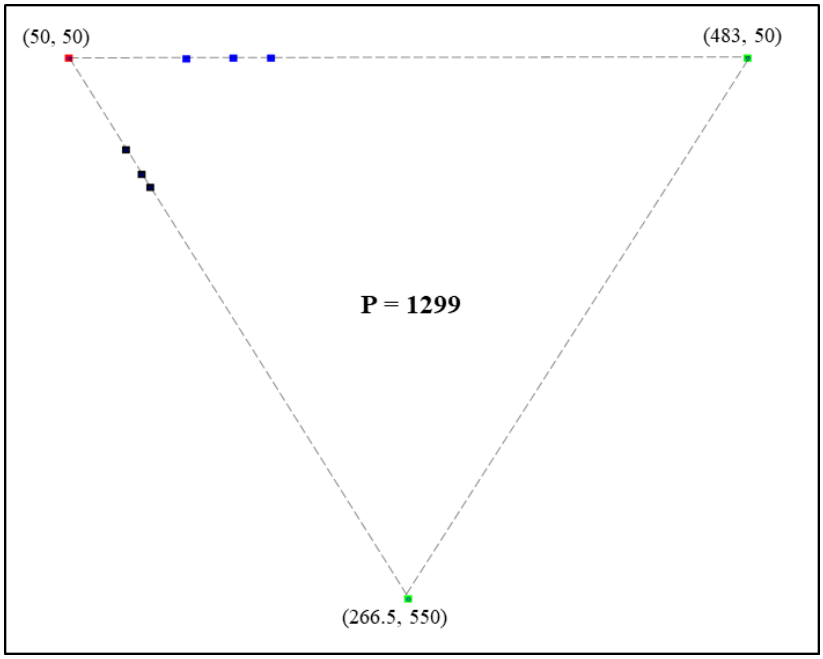


Figure 58. Three equally spaced ground node layout

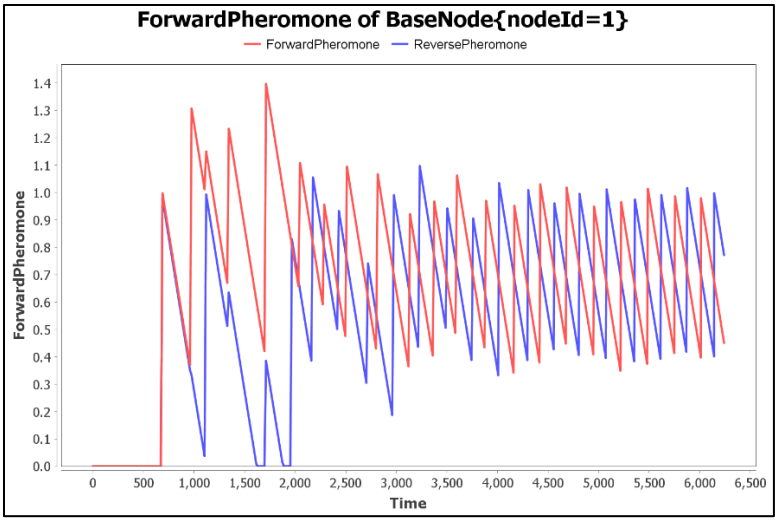


Figure 59. Three equally spaced ground node layout linked pheromone chart

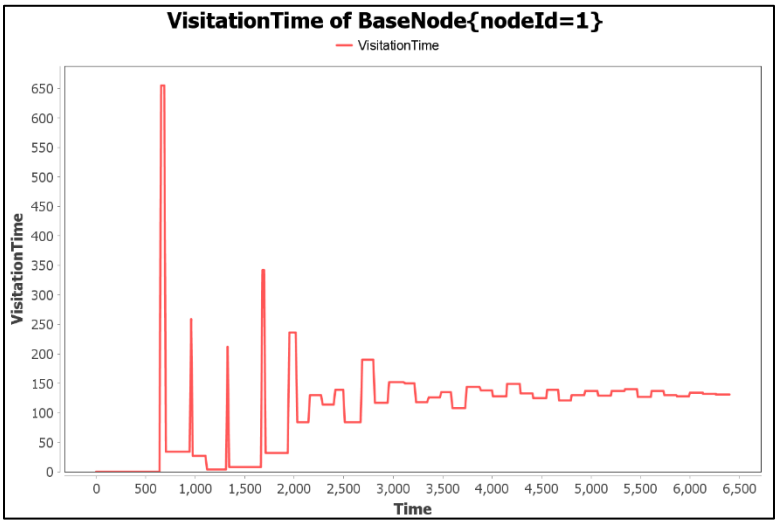


Figure 60. Adjusting node convergence for three equally spaced ground node layout

a. Results

The three ground node layout was the only one that converged the fastest using the smallest reset window value, 0.3, and an adjustment size value of 1.0, for both the same departure time and delayed departure time. These values are highlighted in green in Table 6 and Table 7. We were unable to determine if the small window results were due to the similar polygon side lengths or some other cause. Of note, the equilateral triangle layout converged the fastest when using a 0.9 adjustment size value, highlighted in yellow in

Table 6. While resulting in a faster convergence for the three ground nodes, it failed to always result in convergence for other ground node layouts.

Table 6. Three ground node same departure converged visitation time results

Adjustment Size	0.9	1.0	1.0	1.0	1.0	1.0
Reset Window	0.3	0.3	0.45	0.6	0.75	0.9
Decay Denom	433	433	433	433	433	433
Average	4855.471	5052.456	5305.153	5376.262	5366.525	5391.282
Std Dev	630.557	552.751	707.391	637.235	661.869	683.283
Min	2860	3619	4415	4417	4425	4192
Max	7741	7218	8681	8665	8694	8671
Median	4548	5055	5125	5335	5334	5333
Stat Different	N/A	1	2	2.5	2.5	3

b. Delayed Departure Results

Figure 61 plots the convergence time based upon the separation in return of the first ferry node from each direction back to the originating ground node. These are the results of 1000 departures with the same time, 1000 departures with the retrograde ferry nodes delayed 100 time-steps, and 1000 departures with the retrograde ferry nodes delayed 200 time-steps. Based on these results, we chose a departure delay of 91 time-steps for delayed departure testing. This value was chosen to be halfway in the reduced convergence time section. These results are shown in Table 7.

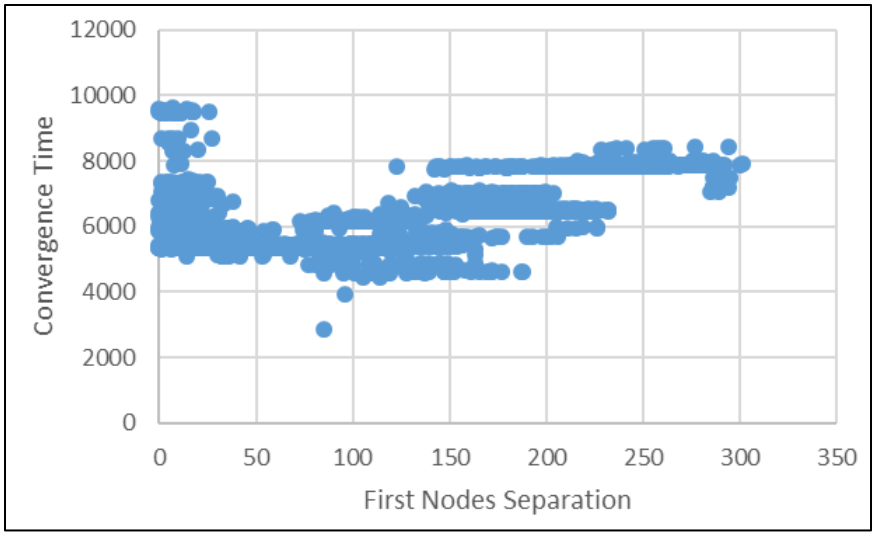


Figure 61. Delayed departure results for three ground nodes

Table 7. Three ground node delayed departure converged visitation time results

Adjustment Size	1.0	1.0	1.0	1.0	1.0
Reset Window	0.3	0.45	0.6	0.75	0.9
Decay Denom	433	433	433	433	433
Departure Delay	91	91	91	91	91
Average	4820.209	5056.988	5083.656	5103.825	5088.239
Std Dev	402.655	419.138	430.405	446.269	426.929
Min	3400	3395	3624	4348	3618
Max	7172	8382	8362	8775	8221
Median	4604.5	5121	5160	5222	5171.5
Stat Different	1	2	2	2	2

2. Four Equally Spaced Ground Nodes

The location of the four ground nodes is shown in Figure 62 with the associated perimeter length. An example pheromone chart from the adjusting node is shown in Figure 63, and the converging visitation time chart from the adjusting node is shown in Figure 64.

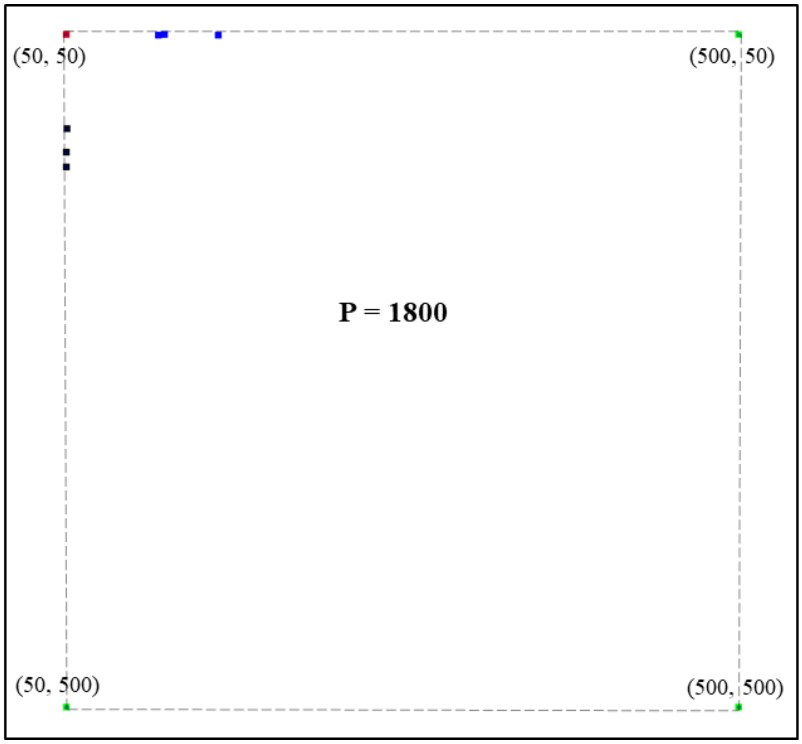


Figure 62. Four equally spaced ground node layout

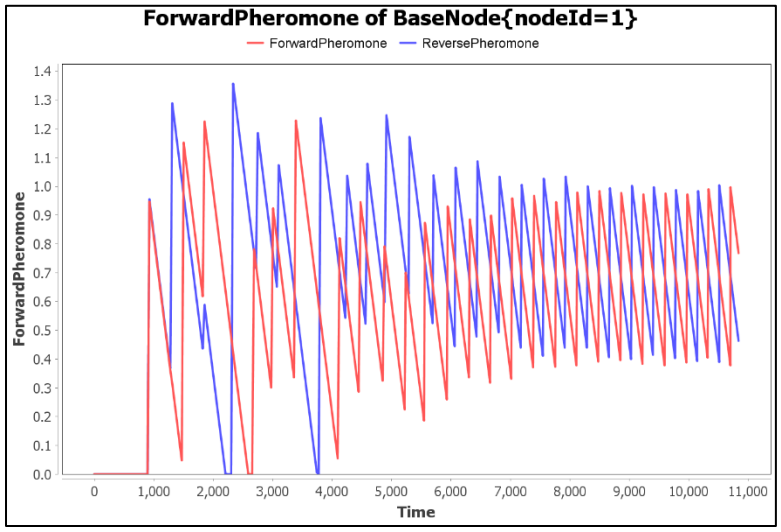


Figure 63. Four equally spaced ground node layout linked pheromone chart

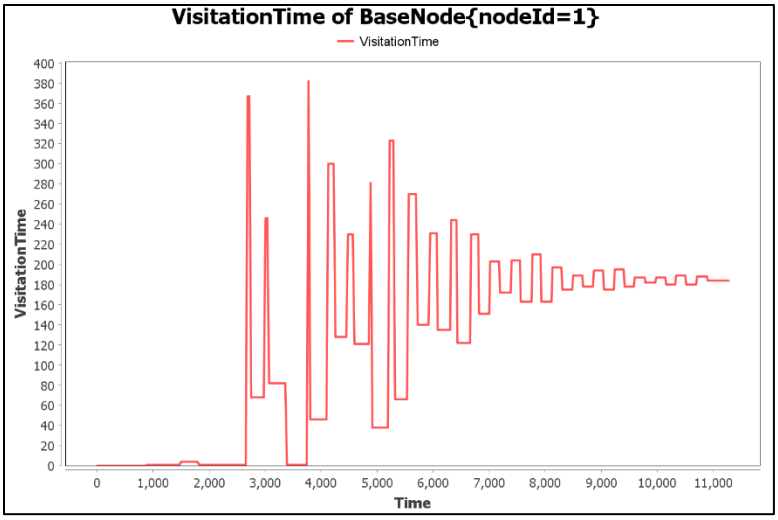


Figure 64. Adjusting node convergence for four equally spaced ground node layout

a. Results

Tables 8 and 9 indicate the visitation time for the four ground nodes converged the fastest using the 0.9 window, highlighted in green, with the same departure time. When the departures were delayed, the 0.75 window, highlighted in green, resulted in the fastest convergence. Even though the average convergence time for the 0.75 window was the smallest for the delayed departure, and 0.9 was the smallest for the same time departure, the difference was not statistically different. Of note, the system could be optimized using an adjustment value of 1.3, highlighted in yellow, but this failed to consistently result in a converged visitation time for other ground node layouts.

Table 8. Four ground node same departure converged visitation time results

Adjustment Size	1.3	1.0	1.0	1.0	1.0	1.0
Reset Window	0.3	0.3	0.45	0.6	0.75	0.9
Decay Denom	600	600	600	600	600	600
Average	6553.196	8011.677	7822.079	7824.768	7825.125	7730.197
Std Dev	754.393	858.144	836.761	821.034	872.095	662.908
Min	6120	6130	5433	6124	6126	5067
Max	12803	14089	14810	14609	14831	12209
Median	6329	7780	7598	7593	7593	7591
Stat Different	N/A	2	1	1	1	1

b. Delayed Departure Results

The convergence time based upon the separation in return of the first ferry node from each direction back to the originating ground node is shown in Figure 65. The number and type of departure delays are the same as the three ground node layout delayed-departure test. Based on these results, we chose a departure delay of 121 time-steps for delayed departure testing. These results are shown in Table 9.

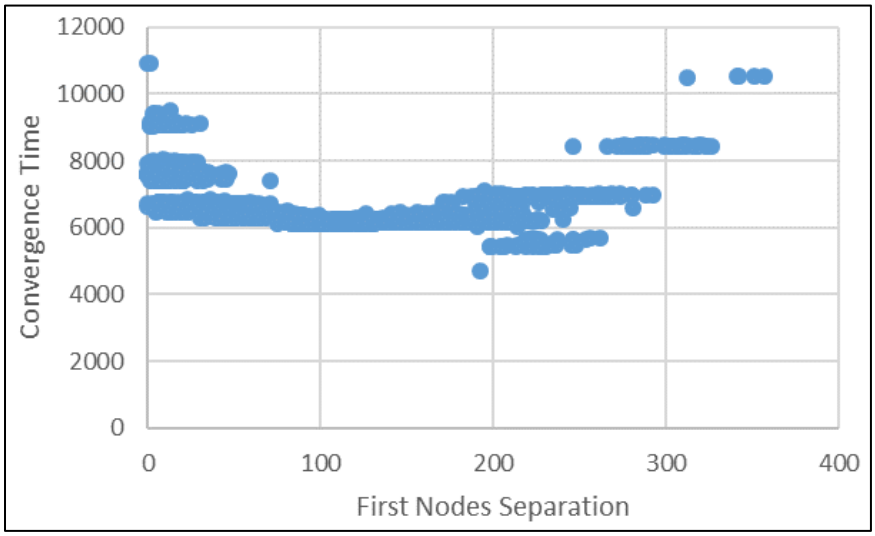


Figure 65. Delayed departure results for four ground nodes

Table 9. Four ground node delayed departure converged visitation time results

Adjustment Size	1.0	1.0	1.0	1.0	1.0
Reset Window	0.3	0.45	0.6	0.75	0.9
Decay Denom	600	600	600	600	600
Departure Delay	121	121	121	121	121
Average	6870.273	6739.172	6714.015	6691.266	6746.908
Std Dev	780.774	816.880	767.014	759.643	775.523
Min	5422	4367	4382	5422	5419
Max	9925	14184	9911	10843	10491
Median	6371	6332.5	6326.5	6319.5	6336
Stat Different	2	1	1	1	1

3. Five Ground Nodes

As with the previous ground node layout discussions, Figures 66 through 68 depict the five ground node layout and its perimeter, an example pheromone chart from the adjusting node, and the visitation time at the adjusting node respectively.

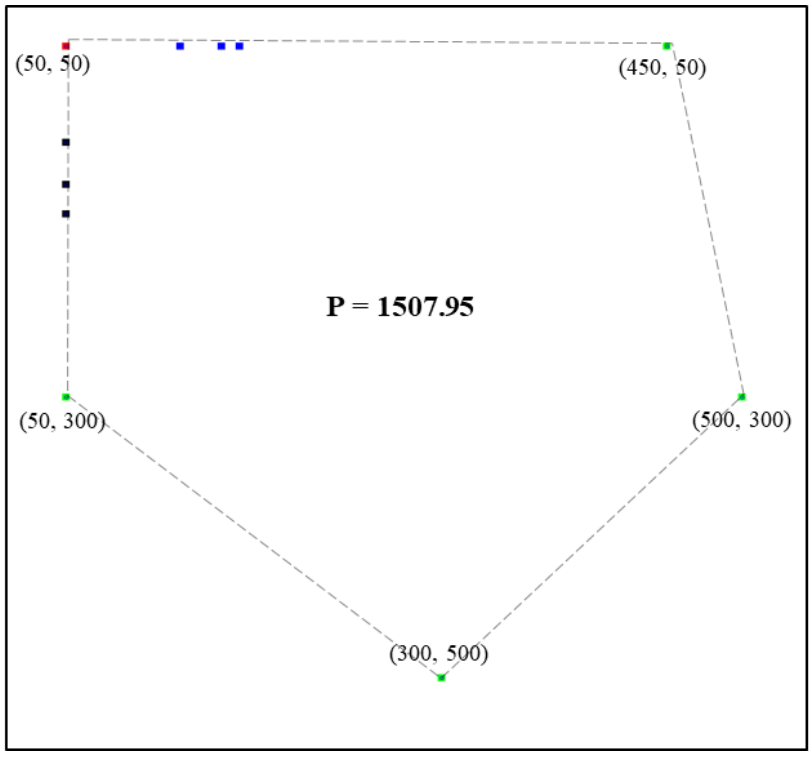


Figure 66. Five ground node layout

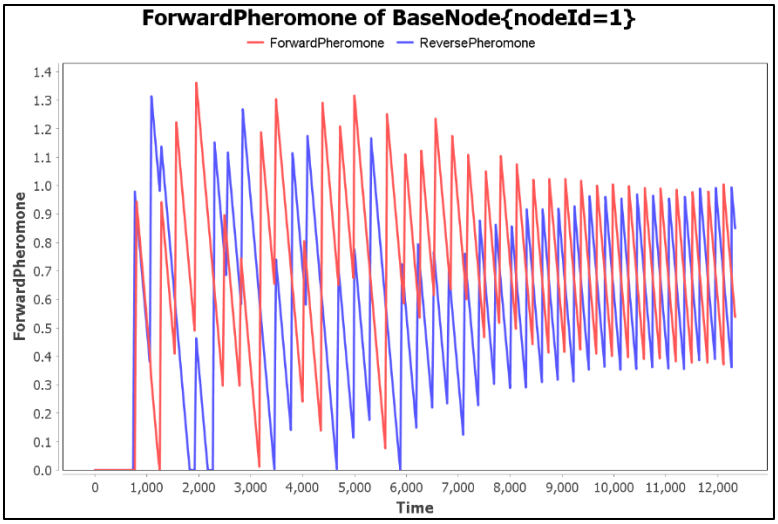


Figure 67. Five ground node layout linked pheromone chart

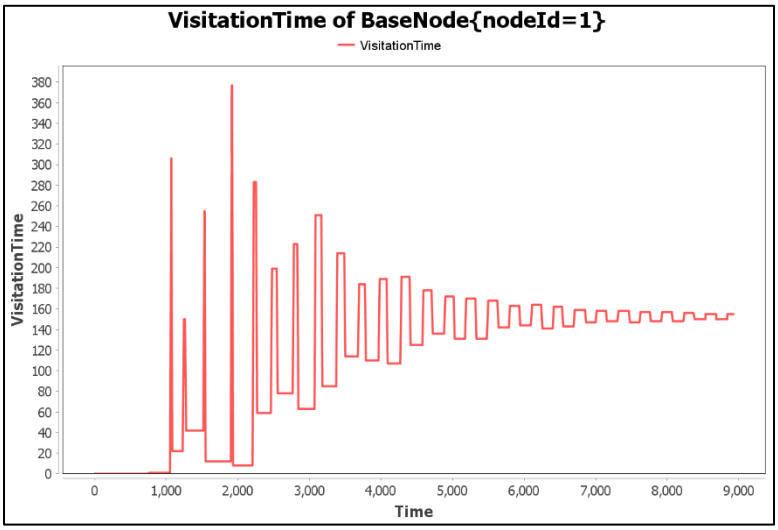


Figure 68. Adjusting node convergence for five ground node layout

a. Results

As shown in Table 10, the 0.6 reset window had the lowest average converged visitation time (highlighted in green) when the ferry nodes departed at the same time, but the differences for all window sizes were not statistically different. As shown in Table 11, the 0.75 window had the lowest convergence time (highlighted in green) when the retrograde ferry nodes were delayed, but the changes between 0.45, 0.6, 0.75, and 0.9 were not statistically different. Due to this, the optimal window for the five-ground node layout

could be either 0.45, 0.6, 0.75, or 0.9. Of note, the system could be optimized for this five-ground node layout by using a 1.5 adjustment value, highlighted in yellow, but this failed to always result in a converged visitation time for other ground node layouts.

Table 10. Five ground node same departure converged visitation time results

Adjustment Size	1.5	1.0	1.0	1.0	1.0	1.0
Reset Window	0.45	0.3	0.45	0.6	0.75	0.9
Decay Denom	481	481	481	481	481	481
Average	5385.986	7685.109	7623.117	7535.68	7646.298	7536.208
Std Dev	1193.797	1087.099	1319.979	1279.988	1324.618	1235.525
Min	3608	4046	3615	4033	3612	4067
Max	9074	11819	11685	11975	14390	11677
Median	5095.5	7489	7372.5	7344.5	7349.5	7338.5
Stat Different	N/A	1	1	1	1	1

b. Delayed Departure Results

Figure 69 plots the convergence time based upon the separation in return of the first ferry node from each direction back to the originating ground node. As before, these are the results from 3000 runs with departures ranging from the same time to the retrograde ferry nodes delayed 200 time-steps. Based on these results, we chose a departure delay of 82 time-steps for delayed departure testing. These results are shown in Table 11.

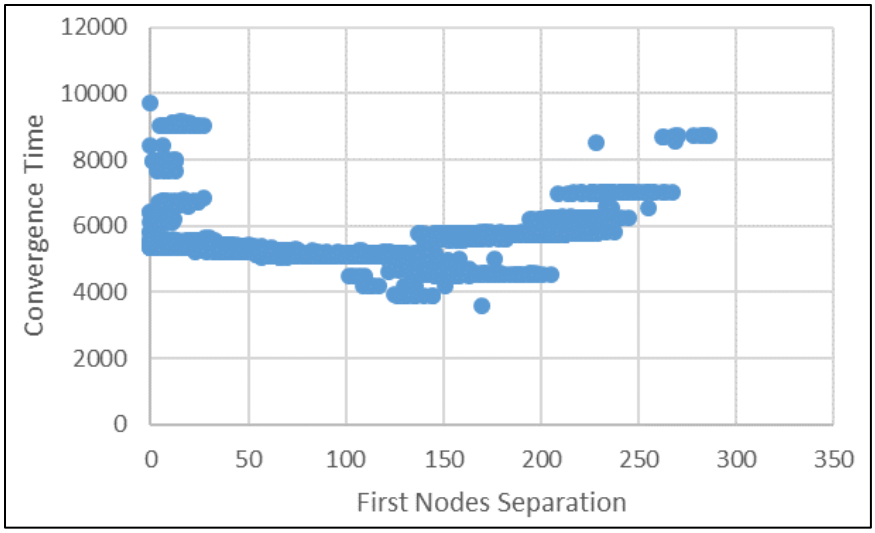


Figure 69. Delayed departure results for five ground nodes

Table 11. Five ground node delayed departure converged visitation time results

Adjustment Size	1.0	1.0	1.0	1.0	1.0
Reset Window	0.3	0.45	0.6	0.75	0.9
Decay Denom	481	481	481	481	481
Departure Delay	82	82	82	82	82
Average	6357.142	6176.595	6217.324	6162.216	6190.709
Std Dev	884.790	910.386	863.648	843.243	861.023
Min	3591	3613	4070	3879	3890
Max	11235	11436	11432	11577	11451
Median	6304	6294	6297	6296	6296
Stat Different	2	1	1	1	1

4. Six Ground Nodes

Figure 70 depicts the six-ground node layout with its associated perimeter. Figure 71 is an example pheromone chart from the adjusting node. Figure 72 charts the ferry node visitation time at the adjusting node.

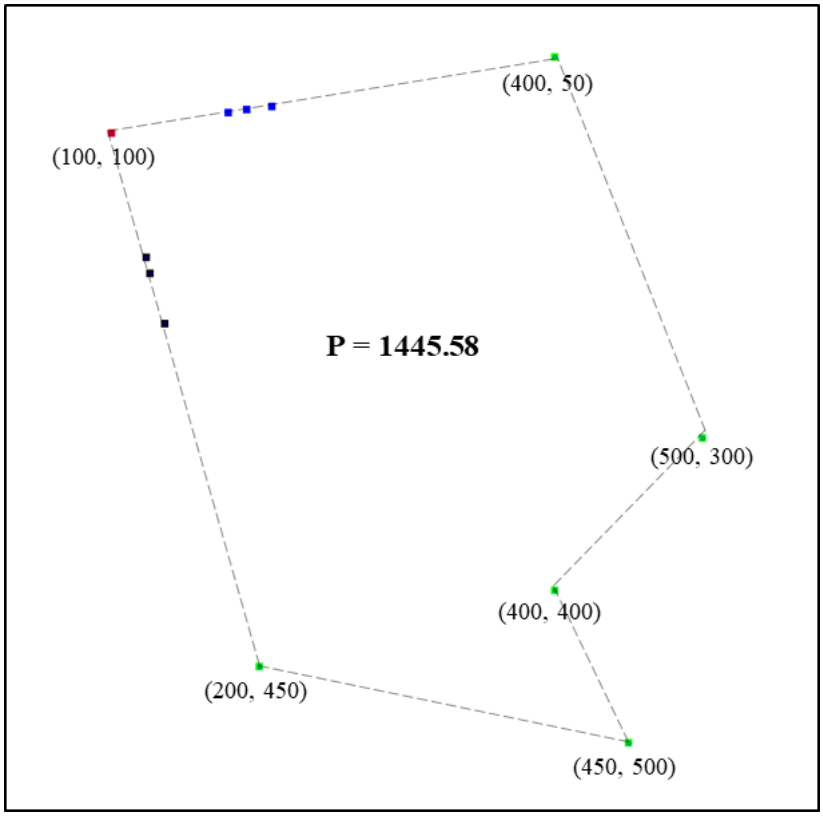


Figure 70. Six ground node layout

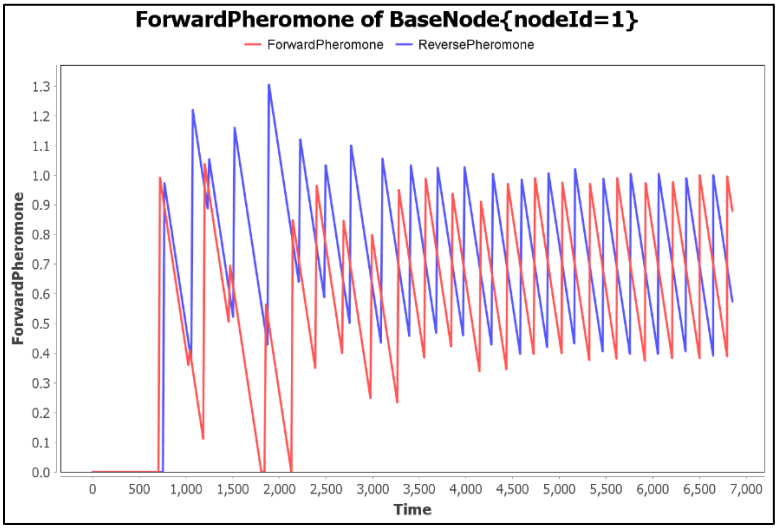


Figure 71. Six ground node layout linked pheromone chart

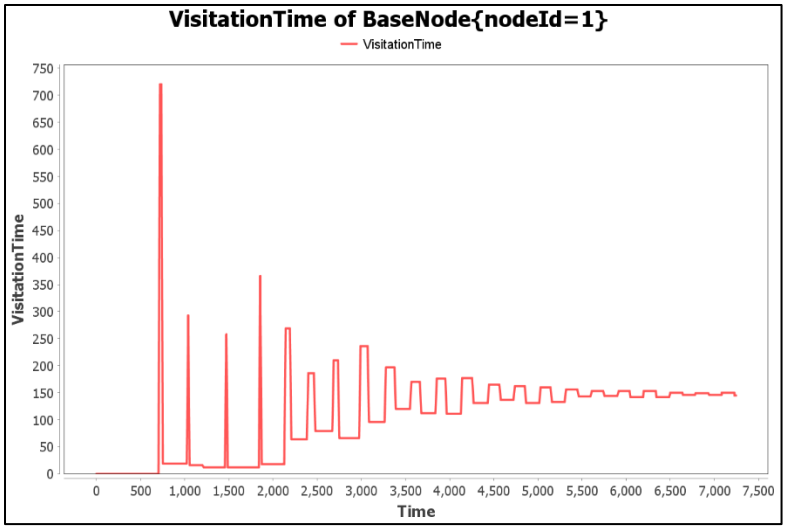


Figure 72. Adjusting node convergence for six ground node layout

a. Results

As shown in Tables 12 and 13, the reset window of 0.9 resulted in the fastest average convergence time (highlighted in green) for the six-node layout for both same departure time and delayed departure time experiments. Of note, an adjustment size of 1.4 (highlighted in yellow), resulted in the fastest convergence for this ground node layout, but it failed to achieve convergence for all ground node layouts.

Table 12. Six ground node same departure converged visitation time results

Adjustment Size	1.4	1.0	1.0	1.0	1.0	1.0
Reset Window	0.45	0.3	0.45	0.6	0.75	0.9
Decay Denom	482	482	482	482	482	482
Average	5012.01	6747.656	6546.896	6462.313	6520.861	6448.793
Std Dev	435.370	790.706	910.091	852.196	909.514	858.939
Min	3733	4909	4020	4025	4897	4900
Max	7140	10663	10666	9938	10505	10503
Median	4895	6526	6215	6196	6201.5	6194
Stat Different	N/A	2	1	1	1	1

b. Delayed Departure Results

The convergence time based upon the separation in return of the first ferry node from each direction back to the originating ground node is shown in Figure 73. The number and type of departure delays are the same as the previous ground node layouts. Based on these results, we chose a departure delay of 94 time-steps for delayed departure testing. These results are shown in Table 13.

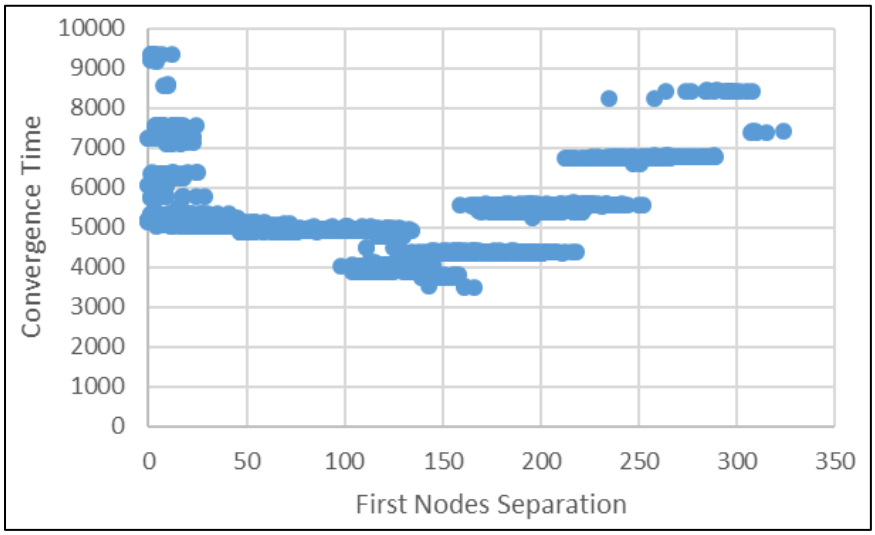


Figure 73. Delayed departure results for six ground nodes

Table 13. Six ground node delayed departure converged visitation time results

Adjustment Size	1.0	1.0	1.0	1.0	1.0
Reset Window	0.3	0.45	0.6	0.75	0.9
Decay Denom	482	482	482	482	482
Departure Delay	94	94	94	94	94
Average	5405.855	5304.87	5323.191	5330.586	5263.47
Std Dev	741.068	748.786	749.048	795.259	757.243
Min	3466	3467	3462	3468	3464
Max	8939	9974	9986	9997	9979
Median	5158	5070.5	5071	5072.5	5064.5
Stat Different	2	1	1.5	1.5	1

5. Seven Ground Nodes

The location of the seven ground nodes is shown in Figure 74 with the associated perimeter length. An example pheromone chart from the adjusting node is shown in Figure 75, and the converging visitation time chart from the adjusting node is shown in Figure 76.

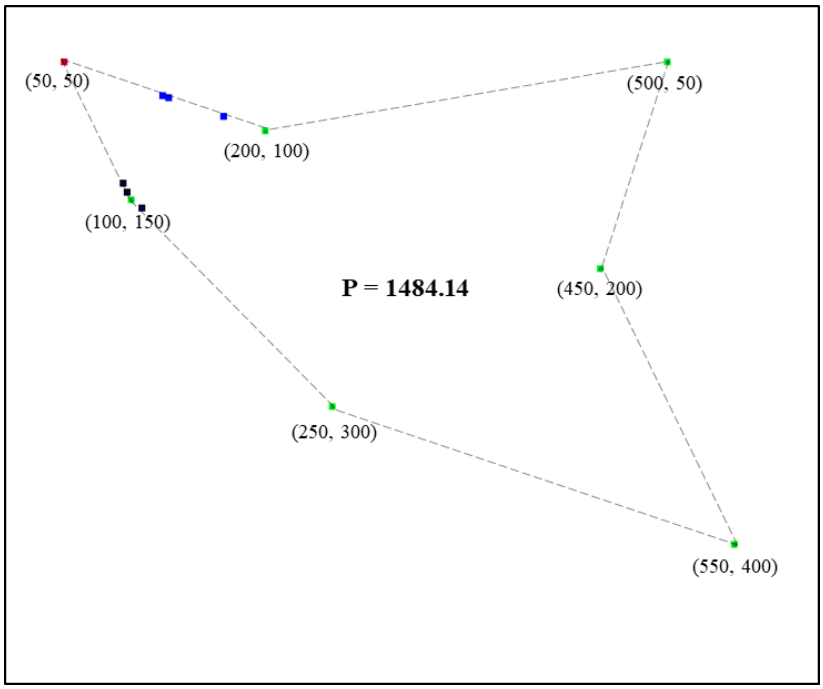


Figure 74. Seven ground node layout

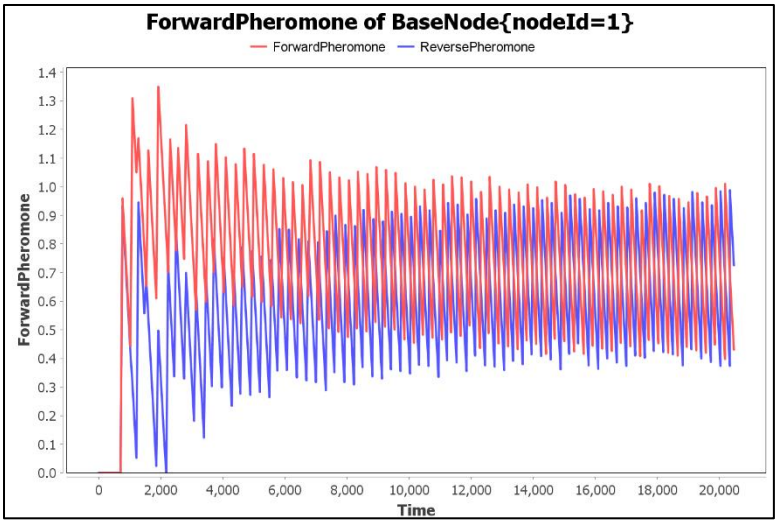


Figure 75. Seven ground node layout linked pheromone chart

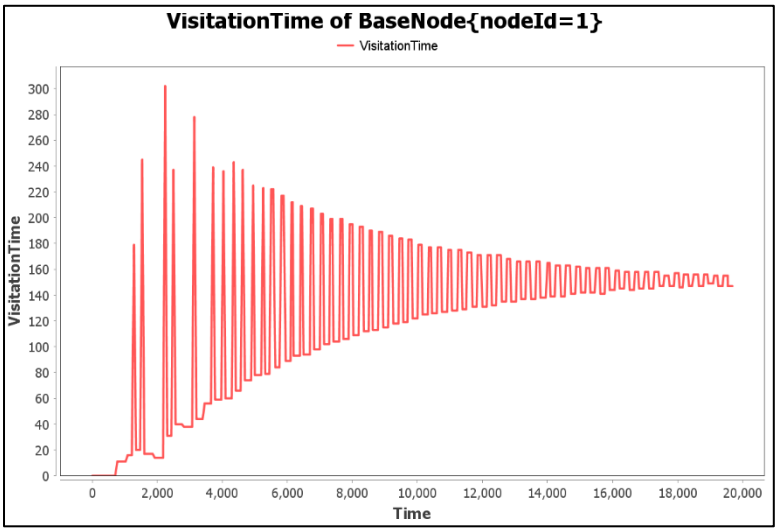


Figure 76. Adjusting node convergence for seven ground node layout

As shown in Table 14, a reset window of 0.6 resulted in the lowest average convergence time (highlighted in green) for the seven-ground node layout, but the difference between this result and those from the 0.75 and 0.9 reset window results were not statistically different.

Table 14. Seven ground node same departure converged visitation time results

Adjustment Size	1.0	1.0	1.0	1.0	1.0
Reset Window	0.3	0.45	0.6	0.75	0.9
Decay Denom	495	495	495	495	495
Average	18772.02	18718.99	18525.26	18538.49	18542.82
Std Dev	1072.628	1183.556	1029.510	1062.919	1098.969
Min	15648	15355	15346	15316	15916
Max	23835	25653	24737	24894	24739
Median	18640	18637	18433	18490.5	18369
Stat Different	2	2	1	1	1

6. Eight Ground Nodes

As with the previous ground node layout discussions, the Figures 77 through 79 depict the eight ground node layout and its perimeter, an example pheromone chart from the adjusting node, and the visitation time at the adjusting node respectively.

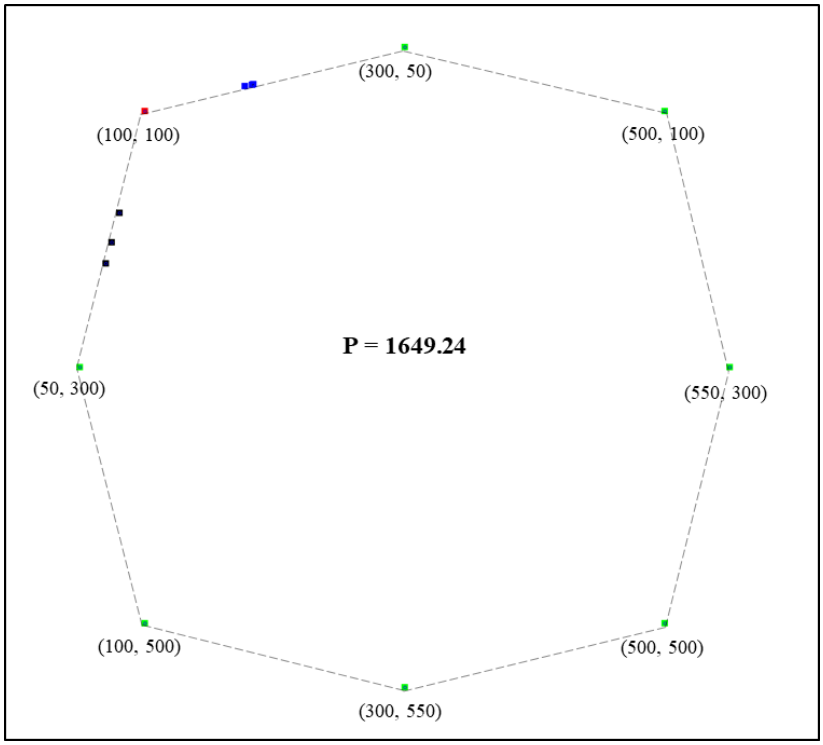


Figure 77. Eight equally spaced ground node layout

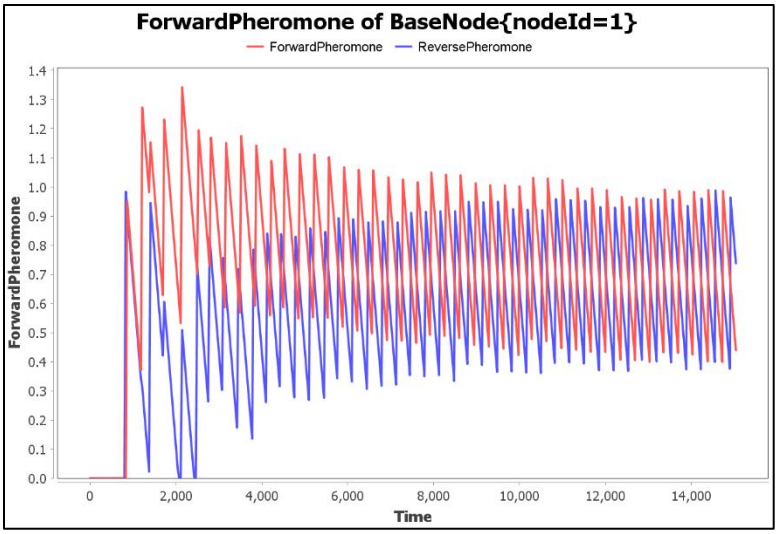


Figure 78. Eight equally spaced ground node layout linked pheromone chart

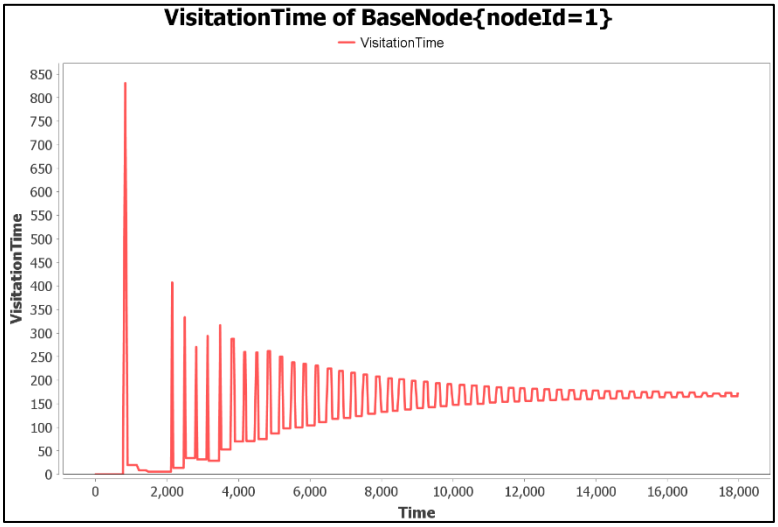


Figure 79. Adjusting node convergence for eight equally spaced ground node layout

The results for the eight-ground node layout were similar to those of the seven-ground node layout. As shown in Table 15, the variations in the data between each reset window size were not statistically different, except for the 0.3 reset window.

Table 15. Eight ground node same departure converged visitation time results

Adjustment Size	1.0	1.0	1.0	1.0	1.0
Reset Window	0.3	0.45	0.6	0.75	0.9
Decay Denom	550	550	550	550	550
Average	14910.26	14777.53	14714.1	14731.15	14777.95
Std Dev	752.266	704.318	830.057	797.938	960.027
Min	12556	12889	12214	12718	11884
Max	20667	19999	21012	20684	20679
Median	14904	14727	14573.5	14578	14577.5
Stat Different	2	1	1	1	1

7. Nine Ground Nodes

Figure 80 depicts the nine-ground node layout with its associated perimeter. Figure 81 is an example pheromone chart from the adjusting node. Figure 82 charts the ferry node visitation time at the adjusting node.

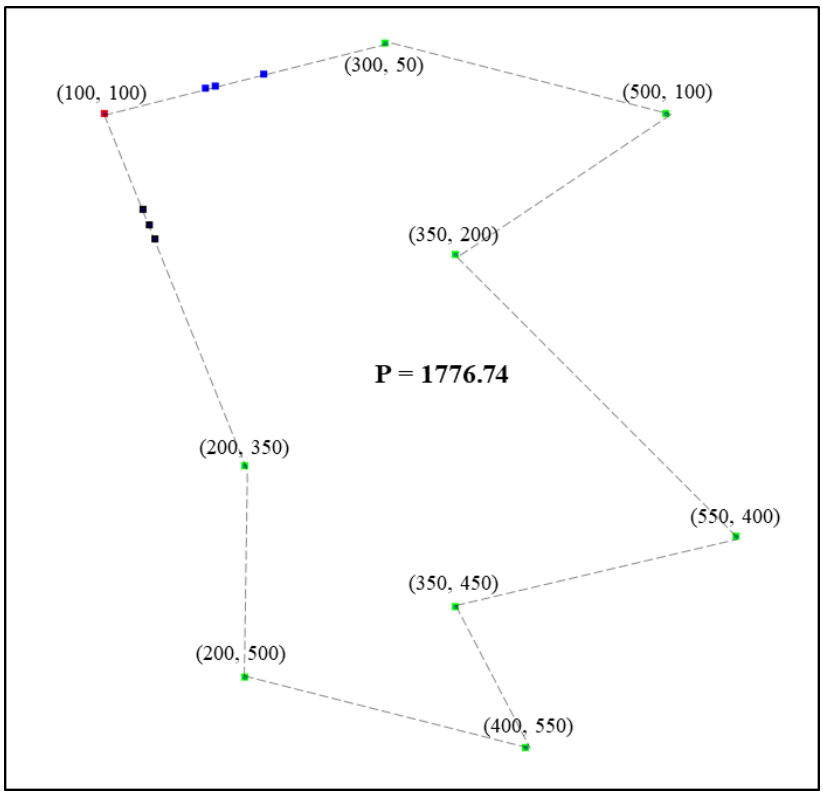


Figure 80. Nine ground node layout

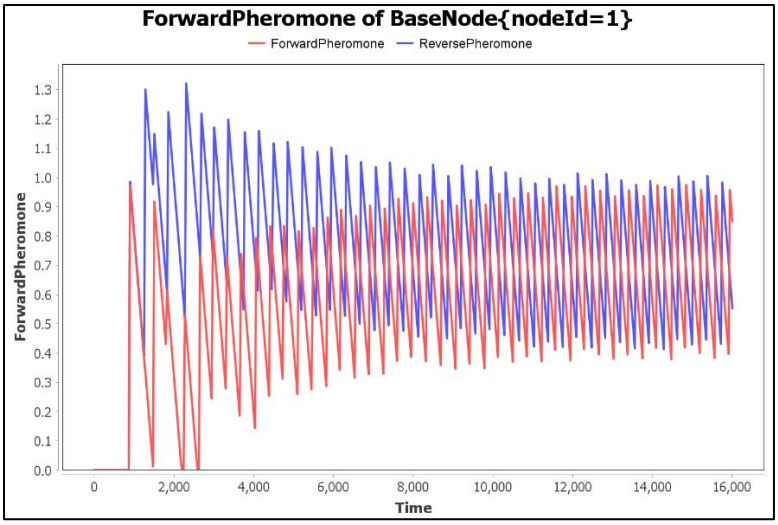


Figure 81. Nine ground node layout linked pheromone chart

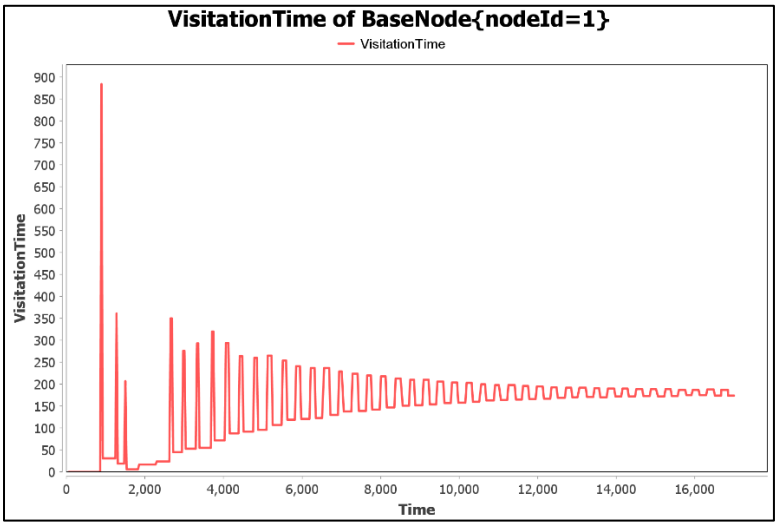


Figure 82. Adjusting node convergence for nine ground node layout

Table 16 indicates the 0.45, 0.6, 0.75, and 0.9 reset windows resulted in similar convergence times that were not statistically different.

Table 16. Nine ground node same departure converged visitation time results

Adjustment Size	1.0	1.0	1.0	1.0	1.0
Reset Window	0.3	0.45	0.6	0.75	0.9
Decay Denom	592	592	592	592	592
Average	16123.12	15925.27	15840.3	15806.98	15808.39
Std Dev	994.489	1021.524	997.277	956.169	955.040
Min	12880	12500	12491	13043	13226
Max	20819	21913	20095	20101	19355
Median	16089.5	15899	15743.5	15741.5	15731.5
Stat Different	2	1	1	1	1

8. Ten Ground Nodes

The location of the ten ground nodes is shown in Figure 83 with the associated perimeter length. An example pheromone chart from the adjusting node is shown in Figure 84, and the converging visitation time chart from the adjusting node is shown in Figure 85.

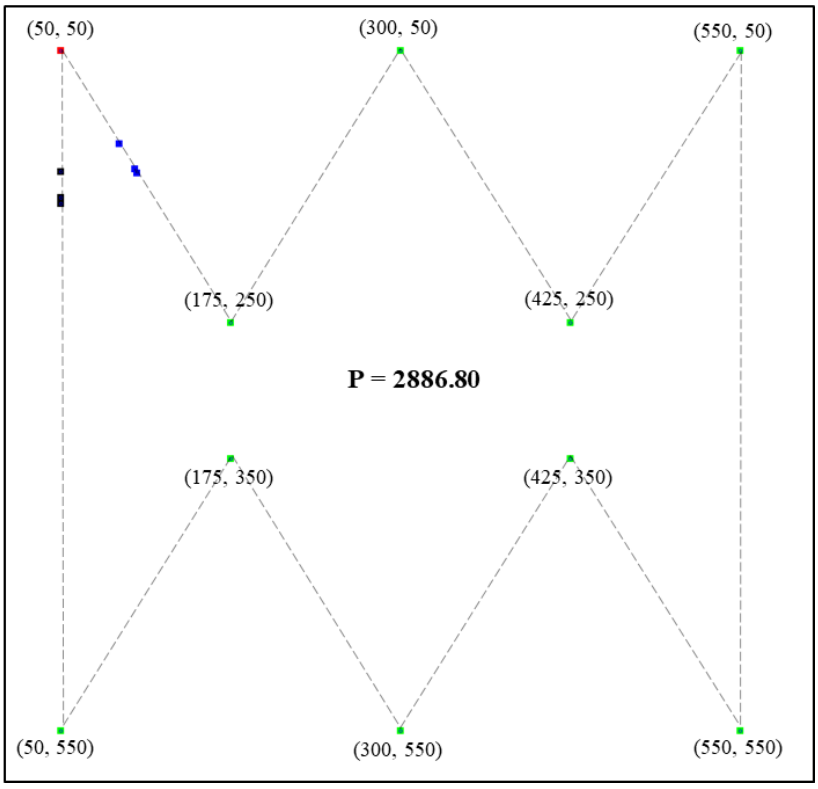


Figure 83. Ten ground node layout

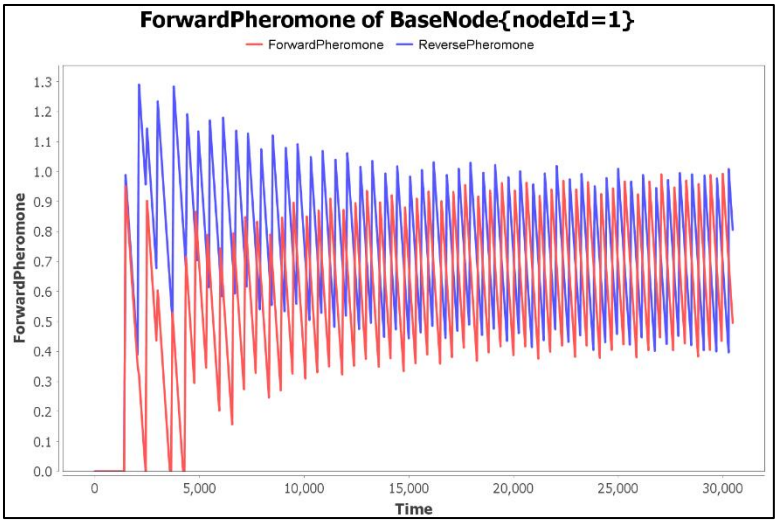


Figure 84. Ten ground node layout linked pheromone chart

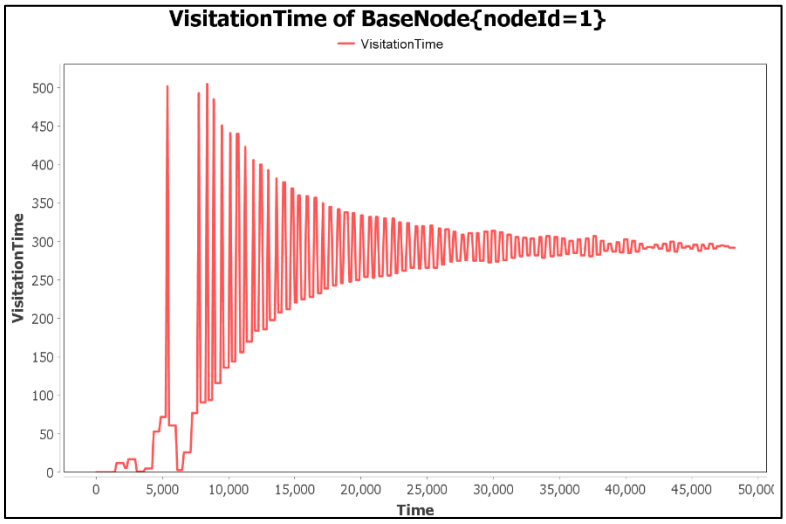


Figure 85. Adjusting node convergence for ten ground node layout

Table 17 indicates that the 0.75 reset window resulted in the fastest convergence for the ten ground node layout.

Table 17. Ten ground node same departure converged visitation time results

Adjustment Size	1.0	1.0	1.0	1.0	1.0
Reset Window	0.3	0.45	0.6	0.75	0.9
Decay Denom	962	962	962	962	962
Average	35099.79	34822.28	34649.39	34592.66	34660.72
Std Dev	3564.356	3902.366	3701.097	3639.681	3670.368
Min	23846	26753	24438	26514	22110
Max	48924	55120	53486	48651	53762
Median	34842.5	34098	34087.5	33992.5	34263.5
Stat Different	2	1.5	1.5	1	1.5

H. UNIVERSAL ADJUSTMENT SIZE AND RESET WINDOW

During testing, we found that each ground node layout had a unique adjustment size that resulted in the optimal convergence time, but these tailored solutions were not universally applicable to arbitrary ground node layouts. The unmodified adjustment size, or 1.0, was the only one that resulted in a converged visitation time for all 1000 iterations for each ground node layout. Therefore, an adjustment size of 1.0 was used for further testing.

The five tested reset window sizes had varying results across the different ground node layouts as well. The only consistent result was that the smallest reset window size, 0.3, failed to yield the fastest convergence time for ground node layouts ranging from four to ten. The three-ground node layout was unique in that the smallest window converged the fastest. Larger reset windows resulted in a statistically different improvement for the more complex ground node layouts. Across the ground node layouts, the 0.75 and 0.9 reset windows most often resulted in the fastest convergence (although the differences between these and other values were often statistically insignificant). A reset window size of 0.9 was used for further testing.

I. MASON TESTING OF DUAL PHEROMONE DELIVERY TIME COMPARED TO SINGLE PHEROMONE

Faster message delivery in the form of reduced maximum and average delivery times is the advantage of a dual-direction communication ferry system over a single-direction system. Chapter III discussed this advantage by analyzing the movement of a

message around the equilateral triangle. Fraser's MASON code was used to experimentally verify this message delivery advantage for ground node layouts from three to ten. To facilitate direct comparison between different ground node layouts, two assumptions were made. The first was that all messages were of the same size and transmission time is instantaneous. The second was that all messages are destined for all other ground nodes. These assumptions will not apply to real-world systems, but in this case, they abstract the performance observation from the underlying system-specific communications requirements.

As simulated, ground nodes generate messages, which are passed to ferry nodes as they pass the sending ground node's location. Messages are delivered by the ferry nodes as they pass the receiving ground nodes. After a message was delivered to all of the other ground nodes, the delivery time was recorded. The measured delivery time is the time from when a transmitting ground node first transmits the data packet (to a ferry node) to the time when all other ground nodes have received that data packet. The total number of collected dissemination times for each ground node layout and associated average delivery times are listed in Table 18. As the table indicates, we tested each ground node layout with three single-direction ferry nodes, six single-direction ferry nodes, and three dual-direction ferry nodes traveling in each direction. Across all ground node types, the dual-direction system was faster than both three single-direction and six single-direction ferry nodes. These faster times are highlighted in green in the table. Distribution graphs for the message delivery times for each ground node layout are shown in Appendix B.

Table 18. Message delivery comparison

Ground Nodes	Direction	Directional Ferry Nodes	Message Count	Average (Time-Steps)	Standard Deviation	Min (Time-Steps)	Max (Time-Steps)
3	Single	3	31418	398.10	76.56	250	561
3	Single	6	16661	392.75	46.73	276	516
3	Dual	3	15958	335.20	42.11	228	535
4	Single	3	42194	737.66	106.36	526	950
4	Single	6	22334	763.51	65.11	592	902
4	Dual	3	19502	691.14	93.03	549	975
5	Single	3	52638	701.99	92.19	496	909
5	Single	6	27163	742.64	62.91	571	911
5	Dual	3	36097	661.36	83.05	480	907
6	Single	3	61877	737.47	125.12	455	1039
6	Single	6	32345	781.18	120.32	523	1138
6	Dual	3	45083	700.83	118.02	466	1055
7	Single	3	72733	792.38	104.75	564	989
7	Single	6	37675	858.40	87.12	631	1193
7	Dual	3	44692	765.19	96.44	570	1074
8	Single	3	82886	904.59	94.10	737	1047
8	Single	6	43504	1001.20	59.92	839	1177
8	Dual	3	41442	876.77	80.54	723	1126
9	Single	3	91876	1024.89	96.35	763	1297
9	Single	6	48106	1104.14	84.82	864	1401
9	Dual	3	48765	964.13	93.00	749	1981
10	Single	3	109813	1700.59	186.56	1268	2084
10	Single	6	55171	1835.69	142.84	1443	2315
10	Dual	3	56406	1591.92	148.09	1254	2060

J. SOFTWARE-IN-THE-LOOP VERIFICATION

Following ground node layout experimentation, the integrated HAIL–ARSENL implementation was updated to use the calculated adjustment size value of 1.0 and reset window of 0.9 that proved effective in MASON simulations. In preparation for live-fly testing, we verified our integrated architecture and on-UAV implementation in the ARSENL SITL simulation environment.

SITL simulations are typically run on a single computer. For this work, this means that all ferry nodes and ground nodes run as instances on the same physical computer. Individual nodes, however, are logically separated and communicate with one another exclusively via an onboard network that realistically simulates live-fly conditions. Computational limitations did effectively limit the number of simulated nodes to two ferry nodes and three ground nodes.

Using two ferry nodes and three ground nodes, we verified the functionality of the algorithm. Figure 86 depicts the ground nodes and the ferry nodes orbiting at a standby location before initiation of the algorithm. The two ferry nodes are depicted as blue chevrons labeled ‘1’ and ‘2’. The adjusting ground node is depicted as a red square. For the purposes of the SITL testing it was labeled Ground Node 101. The resetting ground nodes are depicted as green squares.

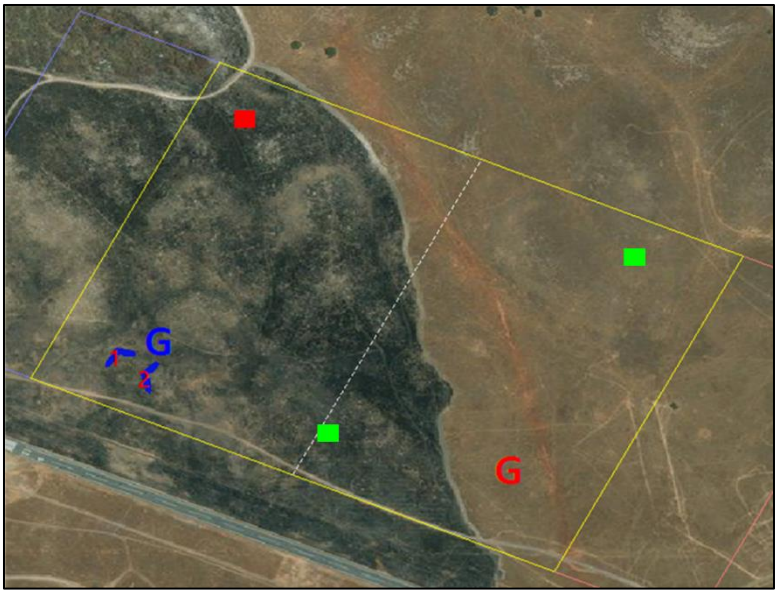


Figure 86. Initial location of two ferry nodes in SITL

After initiating the algorithm, we first verified that the adjusting ground node was both resetting the pheromone and correctly applying the adjustment value. The reduced value of the second pheromone reset depicted in Figure 87 indicates that the adjusting node was functioning correctly.

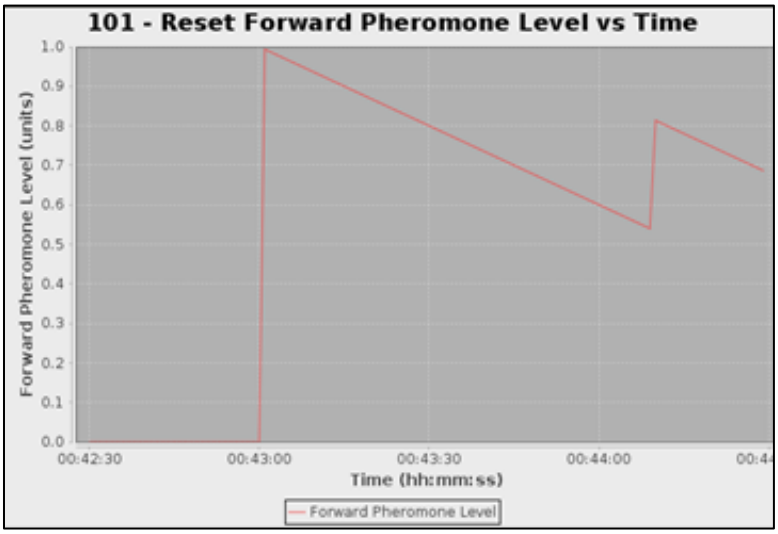


Figure 87. Prograde pheromone value at the adjusting node

These adjustments resulted in the convergence of the visitation time at the adjusting node as shown in Figure 88. This convergence was also shown in the even distribution of ferry nodes as shown in Figure 89.

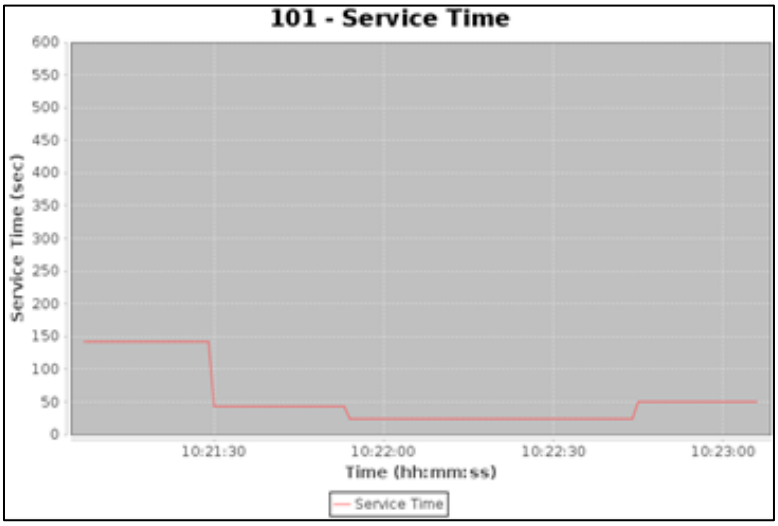


Figure 88. Convergence of visitation time at adjusting node

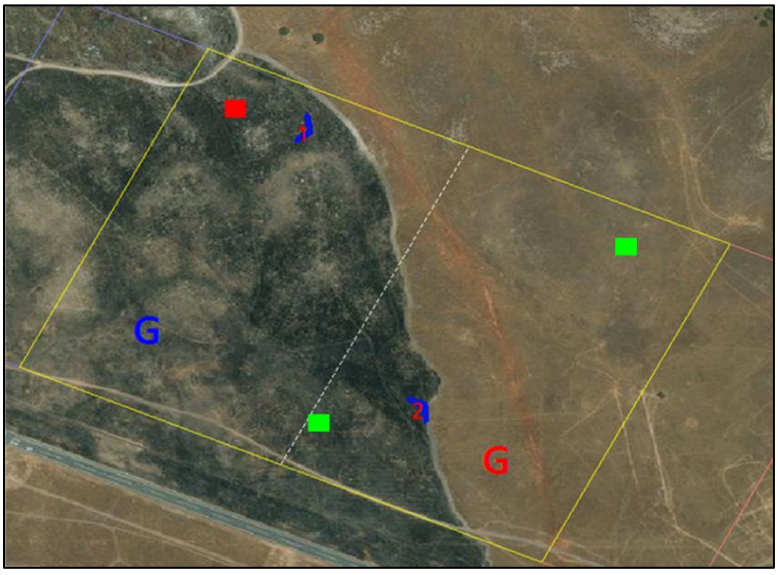


Figure 89. Evenly dispersed ferry nodes in SITL

The results from the SITL experimentation aligned with those of the MASON simulation testing, validating our MASON simulation setup. Despite the reduced number

of ferry nodes that prevented more direct comparisons, these experiments do verify the correctness of the coupling algorithm implementation on the ARSENL Zephyr platforms in preparation for live-fly testing. Further, it demonstrates the efficacy of the algorithm's implementation on an actual aerial swarm platform and the appropriate function of the HAIL Bridge. Live-fly experimentation at Camp Roberts, California is scheduled for June 2018 to provide further verification of the on-UAV ARSENL implementation.

K. CHAPTER SUMMARY

Leveraging Fraser's previous work, we were able to use MASON simulations to validate the usefulness of the pheromone coupling relationship described in Chapter III. Through these simulations, we demonstrated that pheromone adjustments applied when the pheromone is reset result in a converged visitation time at the adjusting node for ground node layouts ranging from three to ten. The size of the adjustment, and the reset window of allowable pheromone values, provided mechanisms to tailor the algorithm for testing and optimization. Using the bulk run functionality, we were able to test 1000 iterations of each combination of these parameters. Each ground node layout had a unique combination, which resulted in the fastest convergence time, but only the full adjustment size (1.0) worked for all tested ground node layouts. These results indicate that the described mathematical coupling could enable an even dispersion of bidirectional ferry nodes for various network topologies in a denied communication environment. This even dispersion will provide uniform visitation with reliable message delivery and receipt at the adjusting ground node.

V. CONCLUSION

A. SUMMARY

This thesis focused on expanding the previous work by Australia's Defence Science and Technology Group to develop a dual-direction delay tolerant data-ferry system using an autonomous aerial swarm. In this work, we focused on the mathematical relationship between the two directional pheromones and implementation of the capability on the NPS ARSENL swarm. This chapter reviews the main findings of our work before discussing possible future work. While this thesis provides a possible method to implement a dual-direction delay tolerant ferry node system, further work is required for the concept to be made suitable for fielding in an operational system. The future work section in this chapter discusses some of the next steps, which might eventually result in a fielded capability for the warfighter.

B. MAIN FINDINGS

1. Adjusting Pheromone Values at Reset Creates Even Distribution

The key finding of this thesis is that stigmergic control mechanisms can provide evenly distributed dual-direction ferry UAVs while operating in a RF denied environment. This was achieved through the intentional adjustment of the pheromone values as they are reset. All other findings build upon this concept. Two possibilities were identified for incorporation of the prograde-retrograde pheromone link. The first is where the pheromones decay, and the second is where the pheromones reset. Attempts to link the prograde and retrograde systems at the decay location were difficult to control, and this method ultimately detracts from the information implicitly conveyed by the steady pheromone decay rate. Changes at the time of reset proved easier to predict and adjust to effect the global behavior of the system without negatively influencing the pheromone-based information.

2. Different Types of Ground Nodes

The dual-direction ferry node system operated under different dynamics than the single-direction ferry node system. When a single ferry node system converges, the distance between the ferry nodes remains the same around the perimeter. All ground nodes, then, regardless of location, will influence the ferry nodes to the same equilibrium state. This is not the case for a dual-direction ferry node system, where even distribution only occurs at specific locations around the perimeter. Influence from ground nodes not at one of these locations will actually disrupt the system's convergence because the equilibrium positions to which these nodes attempt to influence the system will conflict with other nodes' influences.

The implementation of two different types of ground nodes for the dual-direction ferry node system, adjusting nodes and resetting nodes, was required to ensure convergence. Adjusting nodes are physically located at the preferred equal-distribution points and are used to link the two pheromones. For our testing, we only used one adjusting node; however, it is easily shown that one or more of these nodes placed at equilibrium points will result in system convergence. The second type of ground node was the resetting ground node. These ground nodes only reset the directional pheromone to the maximum value without regard to the status of the opposite-direction system's pheromone. In this, they function the same way as Fraser's single-direction system's ground nodes. It was noted that any ground node not located at an equal-distribution point must be a resetting node for the bi-directional system to converge. For our testing, all ground nodes with the exception of a single adjusting node were resetting nodes. The use of these two types of ground nodes led to a converged visitation time for the adjusting node in ground node layouts ranging from three to ten ground nodes. The visitation time for the resetting nodes varied predictably based upon their location around the perimeter up to twice the converged visitation time at the adjusting nodes.

3. Universal Adjustment Value

During our testing, we found that each ground node layout had a specific adjustment size that resulted in the fastest convergence time and that optimal adjustment

sizes were unique to that particular ground node layout. That is, when optimal adjustment sizes for one layout were used on other ground node layouts, they did not result in the fastest convergence time or they failed to reliably converge. The possible improvements from these types of adjustments will be discussed in the future work section.

After recognizing that there was no universally optimal adjustment value for arbitrary ground node layouts, we attempted to find one adjustment value that resulted in converged ferry nodes for all ground node layouts. For this, we experimentally found that the raw calculated adjustment value, without amplification or reduction, could be used to achieve a converged ferry node system for all tested ground node layouts.

4. Largest Reset Window Led to Fastest Convergence

The window for allowable pheromone values was the other parameter that we tested. The largest pheromone adjustment values occurred when the ferry nodes were least dispersed (most likely immediately after initiation). Since large adjustments shortly after system initiation have the potential to over control the system, saturation limits, referred to here as reset windows, to the adjustment value were tested. We experimented with reset window sizes from the following set of values: 0.3, 0.45, 0.6, 0.75, and 0.9. The optimal reset window size varied between ground node layouts, but for most ground node layouts, the 0.9 window had the lowest average convergence time or yielded convergence times that did not statistically differ from the experimentally identified optimum reset window. Of note, this window size allows the pheromone to increase or decrease by the full size of the adjustment.

5. Advantage of Dual-Direction System

Finally, we demonstrated that a dual-direction system results in lower maximum and average delivery times through both analysis and simulation than a single-direction system of the same size. In Chapter III, it was mathematically derived that a message destined for one other ground node travels along the shortest length of the perimeter to its destination. In Chapter IV, we experimentally demonstrated that a message destined for all other ground nodes would be delivered faster using a dual-direction system.

C. FUTURE WORK AND RECOMMENDATIONS

1. Incrementing Instead of Decrementing Pheromone Values

Choosing the decay rate is a key part to making the algorithm for this thesis work correctly. If it is too small, the pheromones decay too slowly, which delays the swarm's convergence. On the other hand, if the rate is too large, the pheromone can decay to zero before the next ferry node visits, and information is thus not properly conveyed to follow on ferry nodes. The swarm will fail to converge if this happens too often during a given run.

Use of increasing pheromones rather than decreasing ones as implemented for this research would prevent situations where the pheromones do not communicate information and might mitigate the effects of suboptimal implementation decisions. Implementation would require at least two steps. First, the reset value would have to be rethought. The initial thought is that it would change to 0.0, but this would prevent dual-direction adjustments unless negative values were allowed. Regardless of whether or not negative values are allowed, a reset value of 1.0 could be retained or a different positive value could be chosen. Second, the speed calculation would have to be adjusted to accept an increasing pheromone value.

2. Test Algorithm for Moving Ground Nodes

All testing done involved stationary ground nodes. Future testing could seek to apply this algorithm to moving ground nodes. The results from this testing would verify that the concept is useful under more realistic scenarios. Based on the results from this thesis, the calculated adjustment value and maximum reset window may evenly disperse the ferry nodes, but a converged system is unlikely due to the constant movement of the ground nodes. Testing of moving ground nodes would determine whether further modifications are required.

3. Variations in Decay or Increase Rate

The decay rate for experiments in support of this work was set at a constant rate proportional to the perimeter length. This allowed comparisons between the different

ground node layouts. A larger decay rate results in faster ferry speeds, and these faster speeds provide faster convergence. Future testing could determine a better decay rate (or increase rate if the pheromone mechanism is changed per the first recommendation). Efforts along this line of research could also explore potential of dynamic rates decay during each simulation, wherein decay rates change based on system performance or by overt operator action. This could better optimize the decay rate for the given situation or enable system responsiveness to changing operational contexts. While more challenging, such dynamic pheromone rate values might be possible with synchronized clocks and communicated decay rates.

4. Evaluate Pheromone Adjustment Size Tailoring for Given Ground and Ferry Node Layouts

During our adjustment size testing, we found that a specific adjustment size resulted in the fastest convergence for each ground node layout. These optimal adjustment sizes provided a marked improvement over the standard adjustment value, but different ground node layouts did not share this same optimal adjustment size. Further testing might be useful in identifying an analytic means of deriving the optimal value from the number of ferry nodes, the length of the perimeter, and the number of ground nodes.

5. Identify Adjusting Node Transition Mechanisms

We found that the dual-direction ferry node system could only operate using two different types of ground nodes: adjusting nodes and resetting nodes. The use of different types of nodes, however, comes at a cost to the robustness of the system. If this one adjusting node fails to function, for instance, the prograde and retrograde pheromones will be uncoupled, and the bi-directional system will not converge. Development of a more capable ground node that can play the role of either adjusting or resetting node and detect when a role change is required will make this system resilient to node failure. One option would be to assign a priority to the ground nodes based on the priority of the ground units that they support. If the ferry nodes make a certain number of laps around the perimeter of the system without hearing from an adjusting node, then another high-priority node can be

notified and then assume the adjusting role until the primary node is contacted again. Other mechanisms to address this single point of failure issue are worth exploration as well.

6. Live-Fly Testing to Validate Algorithms in a Real-World Environment

Future live-fly testing will allow for further development of the system. Through the integration of ARSENL and HAIL, and the development of the coupling algorithm, we have provided the foundation for future live-fly testing to refine the operational capabilities of the resulting system. Live-fly testing using three ferry nodes in each direction and simulated ground nodes is planned for June 2018. Future live-fly tests can extend this by implementing actual ground nodes and varying numbers of ferry nodes.

D. CONCLUSION

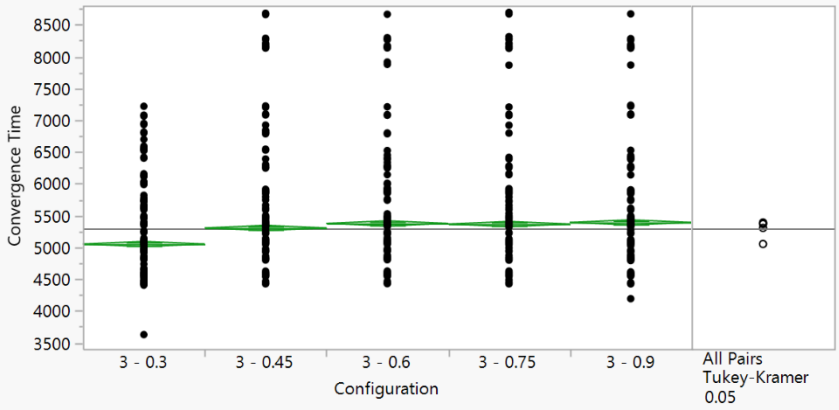
The complexity of the operational environment in which military forces can expect to operate will only increase in the future, and with this, availability of the electromagnetic spectrum can no longer be assumed. Use of DTN relying on ferry nodes implemented by swarming UAVs can maintain communication in these contested environments. In order to achieve the desired global behavior, however, the implementation cannot be reliant upon inter-ferry node communication. The work presented in this thesis extends previous work by DSTG demonstrating the use of stigmergy to effect emergent behaviors among the ferry nodes to provide the desired functionality. We extend their work by developing a method to achieve predictable results from a dual-direction ferry node system. This approach addresses the one primary disadvantage of data ferrying, message delivery delay. This work has demonstrated that the stigmergic approach to control ferry nodes traveling in both directions achieves significant reductions in both maximum and average message delivery times.

APPENDIX A. JMP CONVERGENCE TIME RESULTS

A. THREE GROUND NODE LAYOUT

1. Same Departure Time

Oneway Analysis of Convergence Time By Configuration



Oneway Anova

Summary of Fit

Rsquare	0.03639
Adj Rsquare	0.035618
Root Mean Square Error	650.6844
Mean of Response	5298.336
Observations (or Sum Wgts)	5000

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio	Prob > F
Configuration	4	79864606	19966152	47.1578	<.0001*
Error	4995	2114834239	423390.24		
C. Total	4999	2194698845			

Means for Oneway Anova

Level	Number	Mean	Std Error	Lower 95%	Upper 95%
3 - 0.3	1000	5052.46	20.576	5012.1	5092.8
3 - 0.45	1000	5305.15	20.576	5264.8	5345.5
3 - 0.6	1000	5376.26	20.576	5335.9	5416.6
3 - 0.75	1000	5366.53	20.576	5326.2	5406.9
3 - 0.9	1000	5391.28	20.576	5350.9	5431.6

Std Error uses a pooled estimate of error variance

Means Comparisons

Comparisons for all pairs using Tukey-Kramer HSD

Confidence Quantile

q*	Alpha
2.72883	0.05

HSD Threshold Matrix

Abs(Dif)-HSD

	3 - 0.9	3 - 0.6	3 - 0.75	3 - 0.45	3 - 0.3
3 - 0.9	-79.41	-64.39	-54.65	6.72	259.42
3 - 0.6	-64.39	-79.41	-69.67	-8.30	244.40
3 - 0.75	-54.65	-69.67	-79.41	-18.04	234.66
3 - 0.45	6.72	-8.30	-18.04	-79.41	173.29
3 - 0.3	259.42	244.40	234.66	173.29	-79.41

Positive values show pairs of means that are significantly different.

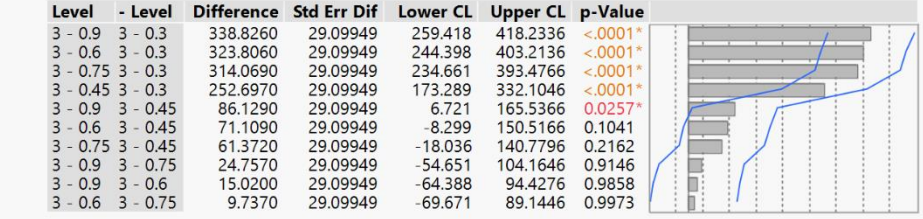
Connecting Letters Report

Level				Mean
3 - 0.9	A			5391.2820
3 - 0.6	A	B		5376.2620
3 - 0.75	A	B		5366.5250
3 - 0.45		B		5305.1530
3 - 0.3			C	5052.4560

Levels not connected by same letter are significantly different.

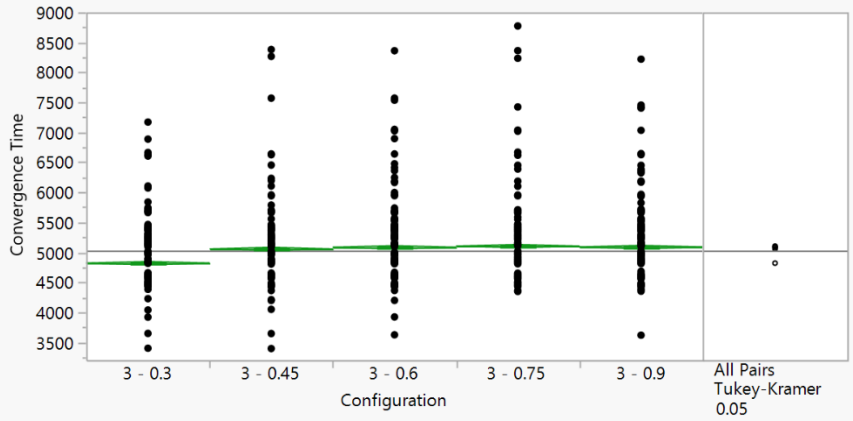
Ordered Differences Report

Level	- Level	Difference	Std Err Dif	Lower CL	Upper CL	p-Value
3 - 0.9	3 - 0.3	338.8260	29.09949	259.418	418.2336	<.0001*
3 - 0.6	3 - 0.3	323.8060	29.09949	244.398	403.2136	<.0001*
3 - 0.75	3 - 0.3	314.0690	29.09949	234.661	393.4766	<.0001*
3 - 0.45	3 - 0.3	252.6970	29.09949	173.289	332.1046	<.0001*
3 - 0.9	3 - 0.45	86.1290	29.09949	6.721	165.5366	0.0257*
3 - 0.6	3 - 0.45	71.1090	29.09949	-8.299	150.5166	0.1041
3 - 0.75	3 - 0.45	61.3720	29.09949	-18.036	140.7796	0.2162
3 - 0.9	3 - 0.75	24.7570	29.09949	-54.651	104.1646	0.9146
3 - 0.9	3 - 0.6	15.0200	29.09949	-64.388	94.4276	0.9858
3 - 0.6	3 - 0.75	9.7370	29.09949	-69.671	89.1446	0.9973



2. Delayed Departure Time

Oneway Analysis of Convergence Time By Configuration



Oneway Anova

Summary of Fit

Rsquare	0.05881
Adj Rsquare	0.058056
Root Mean Square Error	425.3187
Mean of Response	5030.583
Observations (or Sum Wgts)	5000

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio	Prob > F
Configuration	4	56459792	14114948	78.0280	<.0001*
Error	4995	903575663	180896.03		
C. Total	4999	960035455			

Means for Oneway Anova

Level	Number	Mean	Std Error	Lower 95%	Upper 95%
3 - 0.3	1000	4820.21	13.450	4793.8	4846.6
3 - 0.45	1000	5056.99	13.450	5030.6	5083.4
3 - 0.6	1000	5083.66	13.450	5057.3	5110.0
3 - 0.75	1000	5103.83	13.450	5077.5	5130.2
3 - 0.9	1000	5088.24	13.450	5061.9	5114.6

Std Error uses a pooled estimate of error variance

Means Comparisons

Comparisons for all pairs using Tukey-Kramer HSD

Confidence Quantile

q*	Alpha
2.72883	0.05

HSD Threshold Matrix

Abs(Dif)-HSD

	3 - 0.75	3 - 0.9	3 - 0.6	3 - 0.45	3 - 0.3
3 - 0.75	-51.90	-36.32	-31.74	-5.07	231.71
3 - 0.9	-36.32	-51.90	-47.32	-20.65	216.13
3 - 0.6	-31.74	-47.32	-51.90	-25.24	211.54
3 - 0.45	-5.07	-20.65	-25.24	-51.90	184.87
3 - 0.3	231.71	216.13	211.54	184.87	-51.90

Positive values show pairs of means that are significantly different.

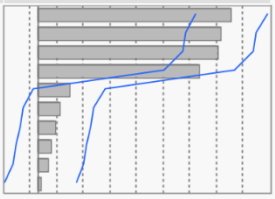
Connecting Letters Report

Level		Mean
3 - 0.75	A	5103.8250
3 - 0.9	A	5088.2390
3 - 0.6	A	5083.6560
3 - 0.45	A	5056.9880
3 - 0.3	B	4820.2090

Levels not connected by same letter are significantly different.

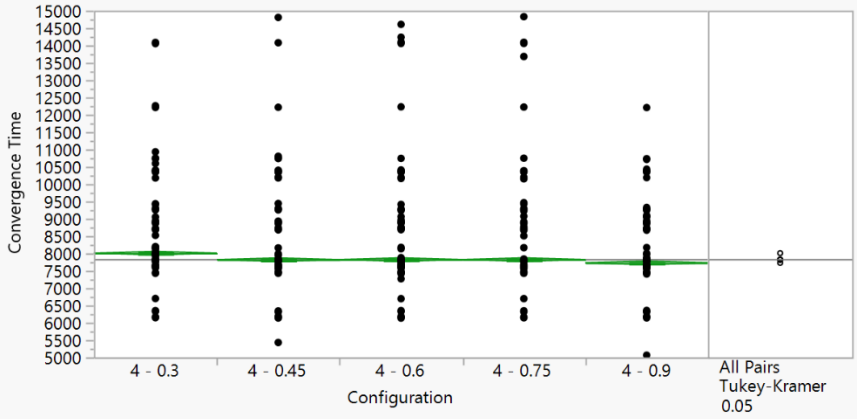
Ordered Differences Report

Level	- Level	Difference	Std Err Dif	Lower CL	Upper CL	p-Value
3 - 0.75	3 - 0.3	283.6160	19.02083	231.711	335.5206	<.0001*
3 - 0.9	3 - 0.3	268.0300	19.02083	216.125	319.9346	<.0001*
3 - 0.6	3 - 0.3	263.4470	19.02083	211.542	315.3516	<.0001*
3 - 0.45	3 - 0.3	236.7790	19.02083	184.874	288.6836	<.0001*
3 - 0.75	3 - 0.45	46.8370	19.02083	-5.068	98.7416	0.0995
3 - 0.9	3 - 0.45	31.2510	19.02083	-20.654	83.1556	0.4698
3 - 0.6	3 - 0.45	26.6680	19.02083	-25.237	78.5726	0.6264
3 - 0.75	3 - 0.6	20.1690	19.02083	-31.736	72.0736	0.8268
3 - 0.75	3 - 0.9	15.5860	19.02083	-36.319	67.4906	0.9247
3 - 0.9	3 - 0.6	4.5830	19.02083	-47.322	56.4876	0.9993



B. FOUR GROUND NODE LAYOUT

1. Same Departure Time



Oneway Anova

Summary of Fit

Rsquare	0.012618
Adj Rsquare	0.011827
Root Mean Square Error	813.7167
Mean of Response	7842.769
Observations (or Sum Wgts)	5000

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio	Prob > F
Configuration	4	42265790.5	10566448	15.9582	<.0001*
Error	4995	3307363485	662134.83		
C. Total	4999	3349629276			

Means for Oneway Anova

Level	Number	Mean	Std Error	Lower 95%	Upper 95%
4 - 0.3	1000	8011.68	25.732	7961.2	8062.1
4 - 0.45	1000	7822.08	25.732	7771.6	7872.5
4 - 0.6	1000	7824.77	25.732	7774.3	7875.2
4 - 0.75	1000	7825.13	25.732	7774.7	7875.6
4 - 0.9	1000	7730.20	25.732	7679.8	7780.6

Std Error uses a pooled estimate of error variance

Means Comparisons

Comparisons for all pairs using Tukey-Kramer HSD

Confidence Quantile

q*	Alpha
2.72883	0.05

HSD Threshold Matrix

Abs(Dif)-HSD

	4 - 0.3	4 - 0.75	4 - 0.6	4 - 0.45	4 - 0.9
4 - 0.3	-99.30	87.25	87.61	90.29	182.18
4 - 0.75	87.25	-99.30	-98.95	-96.26	-4.38
4 - 0.6	87.61	-98.95	-99.30	-96.61	-4.73
4 - 0.45	90.29	-96.26	-96.61	-99.30	-7.42
4 - 0.9	182.18	-4.38	-4.73	-7.42	-99.30

Positive values show pairs of means that are significantly different.

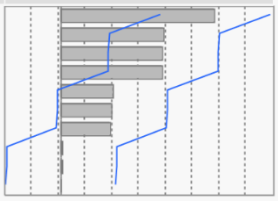
Connecting Letters Report

Level		Mean
4 - 0.3	A	8011.6770
4 - 0.75	B	7825.1250
4 - 0.6	B	7824.7680
4 - 0.45	B	7822.0790
4 - 0.9	B	7730.1970

Levels not connected by same letter are significantly different.

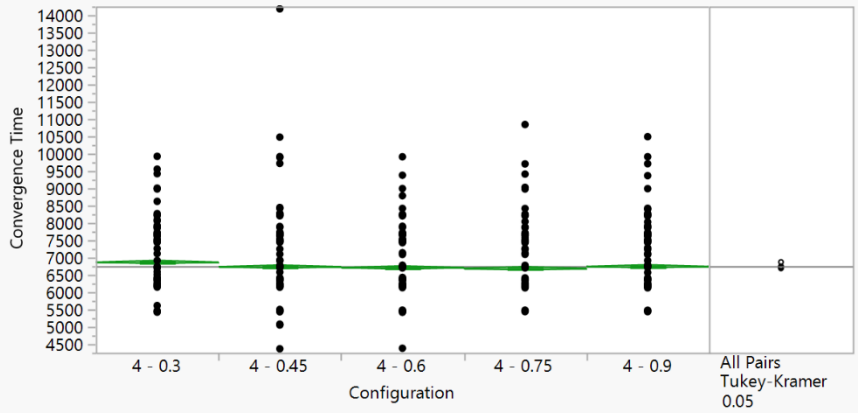
Ordered Differences Report

Level	- Level	Difference	Std Err Dif	Lower CL	Upper CL	p-Value
4 - 0.3	4 - 0.9	281.4800	36.39052	182.176	380.7836	<.0001*
4 - 0.3	4 - 0.45	189.5980	36.39052	90.294	288.9016	<.0001*
4 - 0.3	4 - 0.6	186.9090	36.39052	87.605	286.2126	<.0001*
4 - 0.3	4 - 0.75	186.5520	36.39052	87.248	285.8556	<.0001*
4 - 0.75	4 - 0.9	94.9280	36.39052	-4.376	194.2316	0.0689
4 - 0.6	4 - 0.9	94.5710	36.39052	-4.733	193.8746	0.0707
4 - 0.45	4 - 0.9	91.8820	36.39052	-7.422	191.1856	0.0853
4 - 0.75	4 - 0.45	3.0460	36.39052	-96.258	102.3496	1.0000
4 - 0.6	4 - 0.45	2.6890	36.39052	-96.615	101.9926	1.0000
4 - 0.75	4 - 0.6	0.3570	36.39052	-98.947	99.6606	1.0000



2. Delayed Departure Time

Oneway Analysis of Convergence Time By Configuration



Oneway Anova

Summary of Fit

Rsquare	0.00631
Adj Rsquare	0.005515
Root Mean Square Error	780.2185
Mean of Response	6752.327
Observations (or Sum Wgts)	5000

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio	Prob > F
Configuration	4	19309933.6	4827483	7.9303	<.0001*
Error	4995	3040660724	608741		
C. Total	4999	3059970658			

Means for Oneway Anova

Level	Number	Mean	Std Error	Lower 95%	Upper 95%
4 - 0.3	1000	6870.27	24.673	6821.9	6918.6
4 - 0.45	1000	6739.17	24.673	6690.8	6787.5
4 - 0.6	1000	6714.02	24.673	6665.6	6762.4
4 - 0.75	1000	6691.27	24.673	6642.9	6739.6
4 - 0.9	1000	6746.91	24.673	6698.5	6795.3

Std Error uses a pooled estimate of error variance

Means Comparisons

Comparisons for all pairs using Tukey-Kramer HSD

Confidence Quantile

q*	Alpha
2.72883	0.05

HSD Threshold Matrix

Abs(Dif)-HSD

	4 - 0.3	4 - 0.9	4 - 0.45	4 - 0.6	4 - 0.75
4 - 0.3	-95.216	28.149	35.885	61.042	83.791
4 - 0.9	28.149	-95.216	-87.480	-62.323	-39.574
4 - 0.45	35.885	-87.480	-95.216	-70.059	-47.310
4 - 0.6	61.042	-62.323	-70.059	-95.216	-72.467
4 - 0.75	83.791	-39.574	-47.310	-72.467	-95.216

Positive values show pairs of means that are significantly different.

Connecting Letters Report

Level		Mean
4 - 0.3	A	6870.2730
4 - 0.9	B	6746.9080
4 - 0.45	B	6739.1720
4 - 0.6	B	6714.0150
4 - 0.75	B	6691.2660

Levels not connected by same letter are significantly different.

Ordered Differences Report

Level	- Level	Difference	Std Err Dif	Lower CL	Upper CL	p-Value
4 - 0.3	4 - 0.75	179.0070	34.89243	83.7914	274.2226	<.0001*
4 - 0.3	4 - 0.6	156.2580	34.89243	61.0424	251.4736	<.0001*
4 - 0.3	4 - 0.45	131.1010	34.89243	35.8854	226.3166	0.0016*
4 - 0.3	4 - 0.9	123.3650	34.89243	28.1494	218.5806	0.0038*
4 - 0.9	4 - 0.75	55.6420	34.89243	-39.5736	150.8576	0.5008
4 - 0.45	4 - 0.75	47.9060	34.89243	-47.3096	143.1216	0.6451
4 - 0.9	4 - 0.6	32.8930	34.89243	-62.3226	128.1086	0.8802
4 - 0.45	4 - 0.6	25.1570	34.89243	-70.0586	120.3726	0.9517
4 - 0.6	4 - 0.75	22.7490	34.89243	-72.4666	117.9646	0.9663
4 - 0.9	4 - 0.45	7.7360	34.89243	-87.4796	102.9516	0.9995

C. FIVE GROUND NODE LAYOUT

1. Same Departure Time

Oneway Analysis of Convergence Time By Configuration

Oneway Anova

Summary of Fit

Rsquare	0.00229
Adj Rsquare	0.001491
Root Mean Square Error	1252.488
Mean of Response	7605.282
Observations (or Sum Wgts)	5000

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio	Prob > F
Configuration	4	17988405.3	4497101	2.8667	0.0219*
Error	4995	7835789856	1568727		
C. Total	4999	7853778261			

Means for Oneway Anova

Level	Number	Mean	Std Error	Lower 95%	Upper 95%
5 - 0.3	1000	7685.11	39.607	7607.5	7762.8
5 - 0.45	1000	7623.12	39.607	7545.5	7700.8
5 - 0.6	1000	7535.68	39.607	7458.0	7613.3
5 - 0.75	1000	7646.30	39.607	7568.7	7723.9
5 - 0.9	1000	7536.21	39.607	7458.6	7613.9

Std Error uses a pooled estimate of error variance

Means Comparisons

Comparisons for all pairs using Tukey-Kramer HSD

Confidence Quantile

q*	Alpha
2.72883	0.05

HSD Threshold Matrix

Abs(Dif)-HSD

	5 - 0.3	5 - 0.75	5 - 0.45	5 - 0.9	5 - 0.6
5 - 0.3	-152.85	-114.04	-90.86	-3.95	-3.42
5 - 0.75	-114.04	-152.85	-129.67	-42.76	-42.23
5 - 0.45	-90.86	-129.67	-152.85	-65.94	-65.41
5 - 0.9	-3.95	-42.76	-65.94	-152.85	-152.32
5 - 0.6	-3.42	-42.23	-65.41	-152.32	-152.85

Positive values show pairs of means that are significantly different.

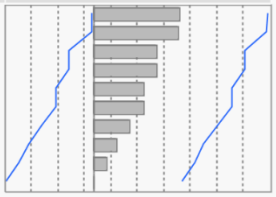
Connecting Letters Report

Level		Mean
5 - 0.3	A	7685.1090
5 - 0.75	A	7646.2980
5 - 0.45	A	7623.1170
5 - 0.9	A	7536.2080
5 - 0.6	A	7535.6800

Levels not connected by same letter are significantly different.

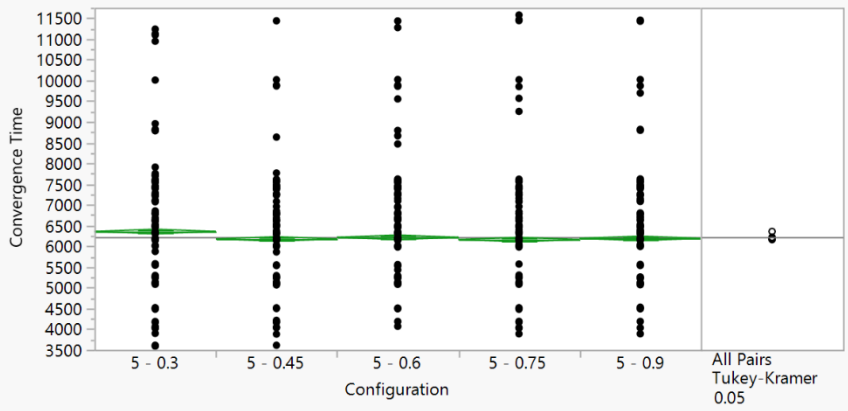
Ordered Differences Report

Level	- Level	Difference	Std Err Dif	Lower CL	Upper CL	p-Value
5 - 0.3	5 - 0.6	149.4290	56.01298	-3.421	302.2790	0.0590
5 - 0.3	5 - 0.9	148.9010	56.01298	-3.949	301.7510	0.0605
5 - 0.75	5 - 0.6	110.6180	56.01298	-42.232	263.4680	0.2785
5 - 0.75	5 - 0.9	110.0900	56.01298	-42.760	262.9400	0.2832
5 - 0.45	5 - 0.6	87.4370	56.01298	-65.413	240.2870	0.5227
5 - 0.45	5 - 0.9	86.9090	56.01298	-65.941	239.7590	0.5288
5 - 0.3	5 - 0.45	61.9920	56.01298	-90.858	214.8420	0.8031
5 - 0.3	5 - 0.75	38.8110	56.01298	-114.039	191.6610	0.9581
5 - 0.75	5 - 0.45	23.1810	56.01298	-129.669	176.0310	0.9939
5 - 0.9	5 - 0.6	0.5280	56.01298	-152.322	153.3780	1.0000



2. Delayed Departure Time

Oneway Analysis of Convergence Time By Configuration



Oneway Anova

Summary of Fit

Rsquare	0.006498
Adj Rsquare	0.005702
Root Mean Square Error	872.9218
Mean of Response	6220.797
Observations (or Sum Wgts)	5000

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio	Prob > F
Configuration	4	24892858.9	6223215	8.1670	<.0001*
Error	4995	3806152677	761993		
C. Total	4999	3831045536			

Means for Oneway Anova

Level	Number	Mean	Std Error	Lower 95%	Upper 95%
5 - 0.3	1000	6357.14	27.604	6303.0	6411.3
5 - 0.45	1000	6176.60	27.604	6122.5	6230.7
5 - 0.6	1000	6217.32	27.604	6163.2	6271.4
5 - 0.75	1000	6162.22	27.604	6108.1	6216.3
5 - 0.9	1000	6190.71	27.604	6136.6	6244.8

Std Error uses a pooled estimate of error variance

Means Comparisons

Comparisons for all pairs using Tukey-Kramer HSD

Confidence Quantile

q*	Alpha
2.72883	0.05

HSD Threshold Matrix

Abs(Dif)-HSD

	5 - 0.3	5 - 0.6	5 - 0.9	5 - 0.45	5 - 0.75
5 - 0.3	-106.53	33.29	59.90	74.02	88.40
5 - 0.6	33.29	-106.53	-79.91	-65.80	-51.42
5 - 0.9	59.90	-79.91	-106.53	-92.41	-78.04
5 - 0.45	74.02	-65.80	-92.41	-106.53	-92.15
5 - 0.75	88.40	-51.42	-78.04	-92.15	-106.53

Positive values show pairs of means that are significantly different.

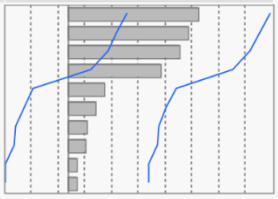
Connecting Letters Report

Level			Mean
5 - 0.3	A		6357.1420
5 - 0.6		B	6217.3240
5 - 0.9		B	6190.7090
5 - 0.45		B	6176.5950
5 - 0.75		B	6162.2160

Levels not connected by same letter are significantly different.

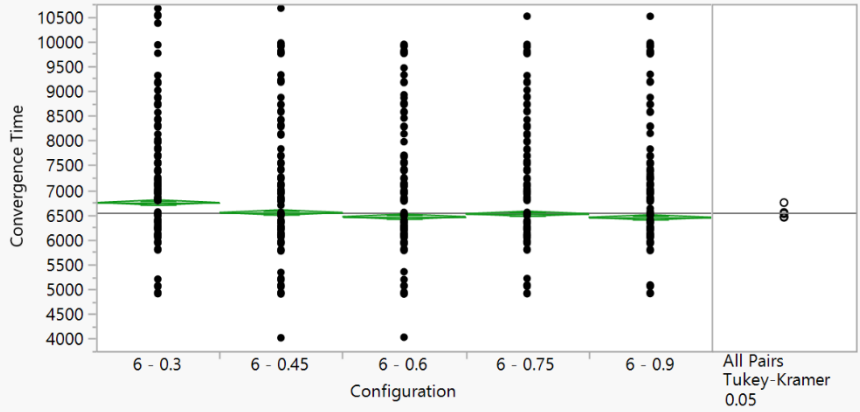
Ordered Differences Report

Level	- Level	Difference	Std Err Dif	Lower CL	Upper CL	p-Value
5 - 0.3	5 - 0.75	194.9260	39.03825	88.3972	301.4548	<.0001*
5 - 0.3	5 - 0.45	180.5470	39.03825	74.0182	287.0758	<.0001*
5 - 0.3	5 - 0.9	166.4330	39.03825	59.9042	272.9618	0.0002*
5 - 0.3	5 - 0.6	139.8180	39.03825	33.2892	246.3468	0.0032*
5 - 0.6	5 - 0.75	55.1080	39.03825	-51.4208	161.6368	0.6202
5 - 0.6	5 - 0.45	40.7290	39.03825	-65.7998	147.2578	0.8352
5 - 0.9	5 - 0.75	28.4930	39.03825	-78.0358	135.0218	0.9496
5 - 0.6	5 - 0.9	26.6150	39.03825	-79.9138	133.1438	0.9604
5 - 0.45	5 - 0.75	14.3790	39.03825	-92.1498	120.9078	0.9961
5 - 0.9	5 - 0.45	14.1140	39.03825	-92.4148	120.6428	0.9964



D. SIX GROUND NODE LAYOUT

1. Same Departure Time



Oneway Anova

Summary of Fit

Rsquare	0.015202
Adj Rsquare	0.014413
Root Mean Square Error	865.4145
Mean of Response	6545.304
Observations (or Sum Wgts)	5000

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio	Prob > F
Configuration	4	57748205.8	14437051	19.2766	<.0001*
Error	4995	3740966574	748942.26		
C. Total	4999	3798714780			

Means for Oneway Anova

Level	Number	Mean	Std Error	Lower 95%	Upper 95%
6 - 0.3	1000	6747.66	27.367	6694.0	6801.3
6 - 0.45	1000	6546.90	27.367	6493.2	6600.5
6 - 0.6	1000	6462.31	27.367	6408.7	6516.0
6 - 0.75	1000	6520.86	27.367	6467.2	6574.5
6 - 0.9	1000	6448.79	27.367	6395.1	6502.4

Std Error uses a pooled estimate of error variance

Means Comparisons

Comparisons for all pairs using Tukey-Kramer HSD

Confidence Quantile

q*	Alpha
2.72883	0.05

HSD Threshold Matrix

Abs(Dif)-HSD

	6 - 0.3	6 - 0.45	6 - 0.75	6 - 0.6	6 - 0.9
6 - 0.3	-105.61	95.15	121.18	179.73	193.25
6 - 0.45	95.15	-105.61	-79.58	-21.03	-7.51
6 - 0.75	121.18	-79.58	-105.61	-47.06	-33.54
6 - 0.6	179.73	-21.03	-47.06	-105.61	-92.09
6 - 0.9	193.25	-7.51	-33.54	-92.09	-105.61

Positive values show pairs of means that are significantly different.

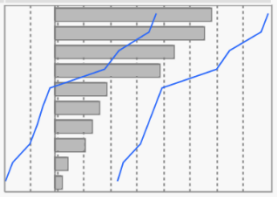
Connecting Letters Report

Level		Mean
6 - 0.3	A	6747.6560
6 - 0.45	B	6546.8960
6 - 0.75	B	6520.8610
6 - 0.6	B	6462.3130
6 - 0.9	B	6448.7930

Levels not connected by same letter are significantly different.

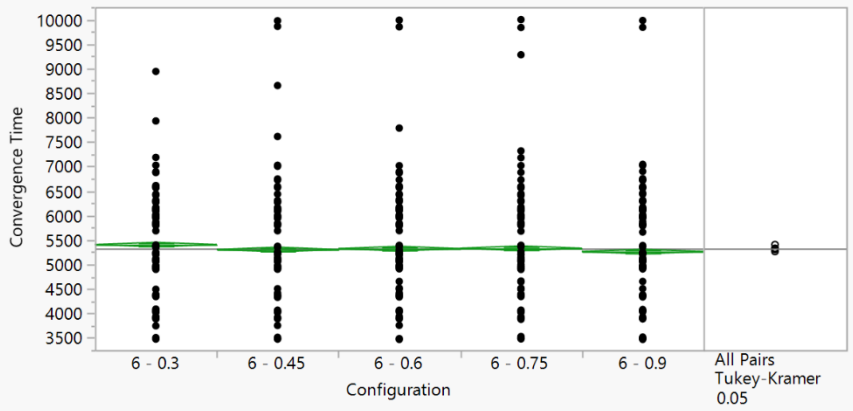
Ordered Differences Report

Level	- Level	Difference	Std Err Dif	Lower CL	Upper CL	p-Value
6 - 0.3	6 - 0.9	298.8630	38.70251	193.250	404.4756	<.0001*
6 - 0.3	6 - 0.6	285.3430	38.70251	179.730	390.9556	<.0001*
6 - 0.3	6 - 0.75	226.7950	38.70251	121.182	332.4076	<.0001*
6 - 0.3	6 - 0.45	200.7600	38.70251	95.147	306.3726	<.0001*
6 - 0.45	6 - 0.9	98.1030	38.70251	-7.510	203.7156	0.0832
6 - 0.45	6 - 0.6	84.5830	38.70251	-21.030	190.1956	0.1853
6 - 0.75	6 - 0.9	72.0680	38.70251	-33.545	177.6806	0.3381
6 - 0.75	6 - 0.6	58.5480	38.70251	-47.065	164.1606	0.5541
6 - 0.45	6 - 0.75	26.0350	38.70251	-79.578	131.6476	0.9623
6 - 0.6	6 - 0.9	13.5200	38.70251	-92.093	119.1326	0.9968



2. Delayed Departure Time

Oneway Analysis of Convergence Time By Configuration



Oneway Anova

Summary of Fit

Rsquare	0.003731
Adj Rsquare	0.002933
Root Mean Square Error	758.5236
Mean of Response	5325.594
Observations (or Sum Wgts)	5000

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio	Prob > F
Configuration	4	10761398.1	2690350	4.6760	0.0009*
Error	4995	2873913163	575358		
C. Total	4999	2884674561			

Means for Oneway Anova

Level	Number	Mean	Std Error	Lower 95%	Upper 95%
6 - 0.3	1000	5405.86	23.987	5358.8	5452.9
6 - 0.45	1000	5304.87	23.987	5257.8	5351.9
6 - 0.6	1000	5323.19	23.987	5276.2	5370.2
6 - 0.75	1000	5330.59	23.987	5283.6	5377.6
6 - 0.9	1000	5263.47	23.987	5216.4	5310.5

Std Error uses a pooled estimate of error variance

Means Comparisons

Comparisons for all pairs using Tukey-Kramer HSD

Confidence Quantile

q*	Alpha
2.72883	0.05

HSD Threshold Matrix

Abs(Dif)-HSD

	6 - 0.3	6 - 0.75	6 - 0.6	6 - 0.45	6 - 0.9
6 - 0.3	-92.568	-17.299	-9.904	8.417	49.817
6 - 0.75	-17.299	-92.568	-85.173	-66.852	-25.452
6 - 0.6	-9.904	-85.173	-92.568	-74.247	-32.847
6 - 0.45	8.417	-66.852	-74.247	-92.568	-51.168
6 - 0.9	49.817	-25.452	-32.847	-51.168	-92.568

Positive values show pairs of means that are significantly different.

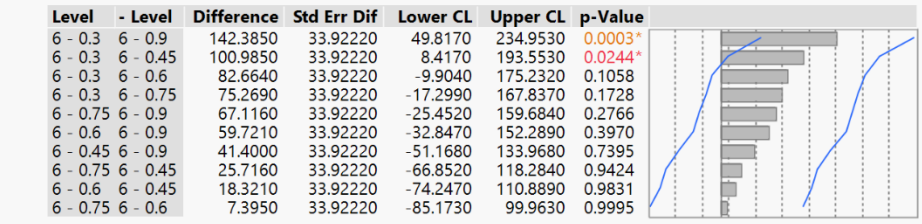
Connecting Letters Report

Level			Mean
6 - 0.3	A		5405.8550
6 - 0.75	A	B	5330.5860
6 - 0.6	A	B	5323.1910
6 - 0.45		B	5304.8700
6 - 0.9		B	5263.4700

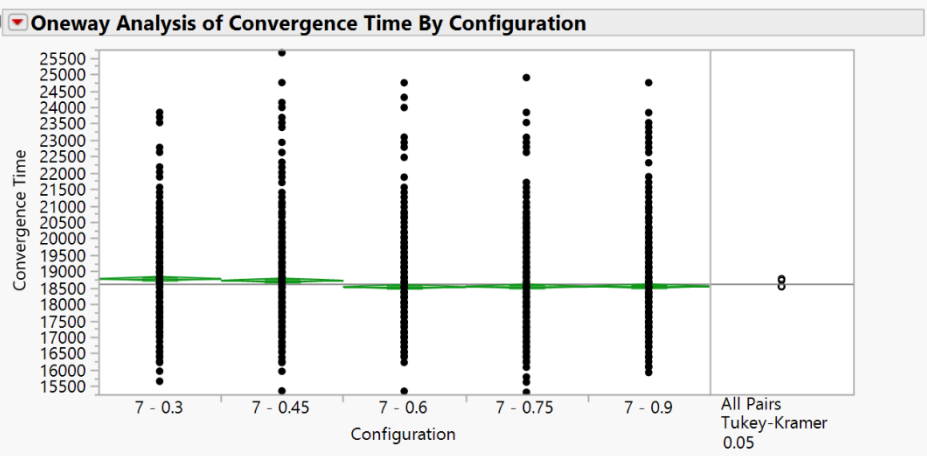
Levels not connected by same letter are significantly different.

Ordered Differences Report

Level	- Level	Difference	Std Err Dif	Lower CL	Upper CL	p-Value
6 - 0.3	6 - 0.9	142.3850	33.92220	49.8170	234.9530	0.0003*
6 - 0.3	6 - 0.45	100.9850	33.92220	8.4170	193.5530	0.0244*
6 - 0.3	6 - 0.6	82.6640	33.92220	-9.9040	175.2320	0.1058
6 - 0.3	6 - 0.75	75.2690	33.92220	-17.2990	167.8370	0.1728
6 - 0.75	6 - 0.9	67.1160	33.92220	-25.4520	159.6840	0.2766
6 - 0.6	6 - 0.9	59.7210	33.92220	-32.8470	152.2890	0.3970
6 - 0.45	6 - 0.9	41.4000	33.92220	-51.1680	133.9680	0.7395
6 - 0.75	6 - 0.45	25.7160	33.92220	-66.8520	118.2840	0.9424
6 - 0.6	6 - 0.45	18.3210	33.92220	-74.2470	110.8890	0.9831
6 - 0.75	6 - 0.6	7.3950	33.92220	-85.1730	99.9630	0.9995



E. SEVEN GROUND NODE LAYOUT



Oneway Anova

Summary of Fit

Rsquare	0.009085
Adj Rsquare	0.008291
Root Mean Square Error	1090.757
Mean of Response	18619.52
Observations (or Sum Wgts)	5000

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio	Prob > F
Configuration	4	54483598.7	13620900	11.4485	<.0001*
Error	4995	5942806222	1189751		
C. Total	4999	5997289821			

Means for Oneway Anova

Level	Number	Mean	Std Error	Lower 95%	Upper 95%
7 - 0.3	1000	18772.0	34.493	18704	18840
7 - 0.45	1000	18719.0	34.493	18651	18787
7 - 0.6	1000	18525.3	34.493	18458	18593
7 - 0.75	1000	18538.5	34.493	18471	18606
7 - 0.9	1000	18542.8	34.493	18475	18610

Std Error uses a pooled estimate of error variance

Means Comparisons

Comparisons for all pairs using Tukey-Kramer HSD

Confidence Quantile

q*	Alpha
2.72883	0.05

HSD Threshold Matrix

Abs(Dif)-HSD

	7 - 0.3	7 - 0.45	7 - 0.9	7 - 0.75	7 - 0.6
7 - 0.3	-133.11	-80.08	96.09	100.42	113.64
7 - 0.45	-80.08	-133.11	43.05	47.39	60.61
7 - 0.9	96.09	43.05	-133.11	-128.78	-115.55
7 - 0.75	100.42	47.39	-128.78	-133.11	-119.89
7 - 0.6	113.64	60.61	-115.55	-119.89	-133.11

Positive values show pairs of means that are significantly different.

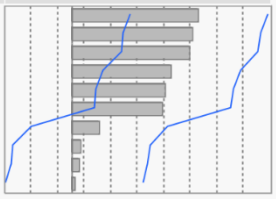
Connecting Letters Report

Level		Mean
7 - 0.3	A	18772.021
7 - 0.45	A	18718.989
7 - 0.9	B	18542.822
7 - 0.75	B	18538.488
7 - 0.6	B	18525.264

Levels not connected by same letter are significantly different.

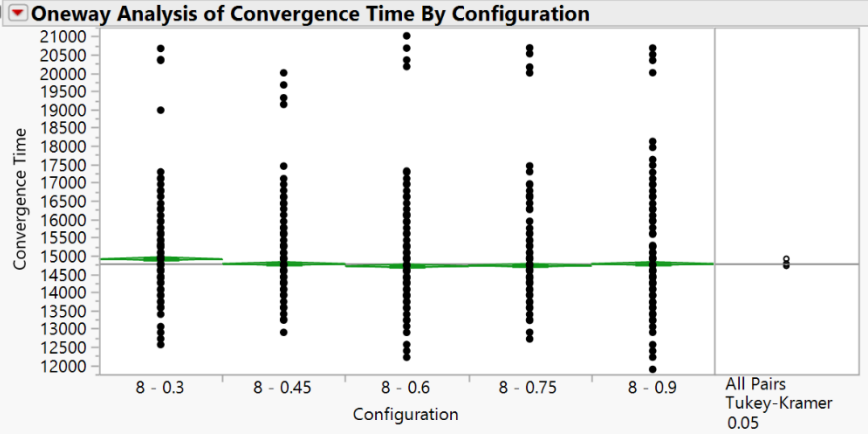
Ordered Differences Report

Level	- Level	Difference	Std Err Dif	Lower CL	Upper CL	p-Value
7 - 0.3	7 - 0.6	246.7570	48.78014	113.644	379.8698	<.0001*
7 - 0.3	7 - 0.75	233.5330	48.78014	100.420	366.6458	<.0001*
7 - 0.3	7 - 0.9	229.1990	48.78014	96.086	362.3118	<.0001*
7 - 0.45	7 - 0.6	193.7250	48.78014	60.612	326.8378	0.0007*
7 - 0.45	7 - 0.75	180.5010	48.78014	47.388	313.6138	0.0020*
7 - 0.45	7 - 0.9	176.1670	48.78014	43.054	309.2798	0.0028*
7 - 0.3	7 - 0.45	53.0320	48.78014	-80.081	186.1448	0.8133
7 - 0.9	7 - 0.6	17.5580	48.78014	-115.555	150.6708	0.9964
7 - 0.75	7 - 0.6	13.2240	48.78014	-119.889	146.3368	0.9988
7 - 0.9	7 - 0.75	4.3340	48.78014	-128.779	137.4468	1.0000



F. EIGHT GROUND NODE LAYOUT

Oneway Analysis of Convergence Time By Configuration



Oneway Anova

Summary of Fit

Rsquare	0.007113
Adj Rsquare	0.006318
Root Mean Square Error	813.5502
Mean of Response	14782.2
Observations (or Sum Wgts)	5000

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio	Prob > F
Configuration	4	23684033.4	5921008	8.9460	<.0001*
Error	4995	3306010013	661864		
C. Total	4999	3329694046			

Means for Oneway Anova

Level	Number	Mean	Std Error	Lower 95%	Upper 95%
8 - 0.3	1000	14910.3	25.727	14860	14961
8 - 0.45	1000	14777.5	25.727	14727	14828
8 - 0.6	1000	14714.1	25.727	14664	14765
8 - 0.75	1000	14731.1	25.727	14681	14782
8 - 0.9	1000	14778.0	25.727	14728	14828

Std Error uses a pooled estimate of error variance

Means Comparisons

Comparisons for all pairs using Tukey-Kramer HSD

Confidence Quantile

q*	Alpha
2.72883	0.05

HSD Threshold Matrix

Abs(Dif)-HSD

	8 - 0.3	8 - 0.9	8 - 0.45	8 - 0.75	8 - 0.6
8 - 0.3	-99.283	33.028	33.453	79.829	96.883
8 - 0.9	33.028	-99.283	-98.858	-52.482	-35.428
8 - 0.45	33.453	-98.858	-99.283	-52.907	-35.853
8 - 0.75	79.829	-52.482	-52.907	-99.283	-82.229
8 - 0.6	96.883	-35.428	-35.853	-82.229	-99.283

Positive values show pairs of means that are significantly different.

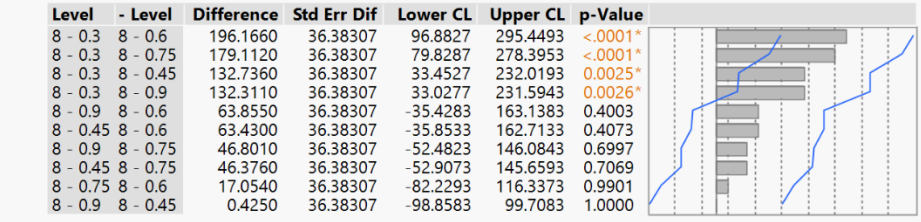
Connecting Letters Report

Level		Mean
8 - 0.3	A	14910.261
8 - 0.9	B	14777.950
8 - 0.45	B	14777.525
8 - 0.75	B	14731.149
8 - 0.6	B	14714.095

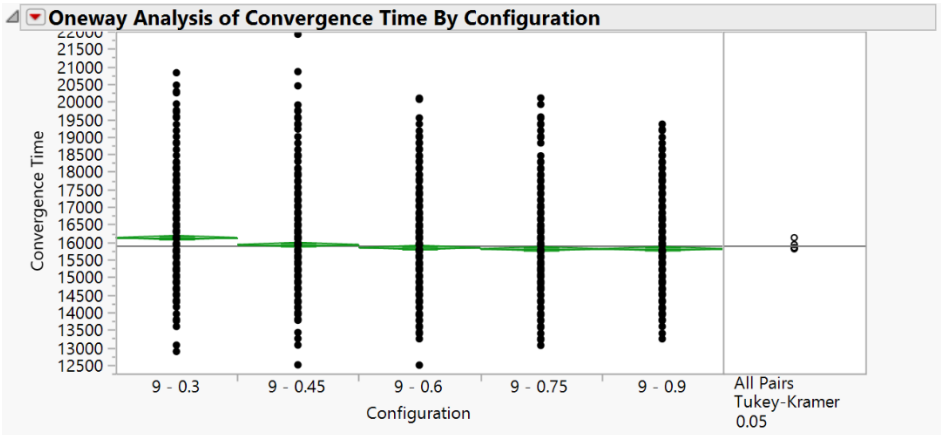
Levels not connected by same letter are significantly different.

Ordered Differences Report

Level	- Level	Difference	Std Err Dif	Lower CL	Upper CL	p-Value
8 - 0.3	8 - 0.6	196.1660	36.38307	96.8827	295.4493	<.0001*
8 - 0.3	8 - 0.75	179.1120	36.38307	79.8287	278.3953	<.0001*
8 - 0.3	8 - 0.45	132.7360	36.38307	33.4527	232.0193	0.0025*
8 - 0.3	8 - 0.9	132.3110	36.38307	33.0277	231.5943	0.0026*
8 - 0.9	8 - 0.6	63.8550	36.38307	-35.4283	163.1383	0.4003
8 - 0.45	8 - 0.6	63.4300	36.38307	-35.8533	162.7133	0.4073
8 - 0.9	8 - 0.75	46.8010	36.38307	-52.4823	146.0843	0.6997
8 - 0.45	8 - 0.75	46.3760	36.38307	-52.9073	145.6593	0.7069
8 - 0.75	8 - 0.6	17.0540	36.38307	-82.2293	116.3373	0.9901
8 - 0.9	8 - 0.45	0.4250	36.38307	-98.8583	99.7083	1.0000



G. NINE GROUND NODE LAYOUT



Oneway Anova

Summary of Fit

Rsquare	0.014438
Adj Rsquare	0.013649
Root Mean Square Error	985.235
Mean of Response	15900.81
Observations (or Sum Wgts)	5000

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio	Prob > F
Configuration	4	71028361.8	17757090	18.2933	<.0001*
Error	4995	4848586622	970688.01		
C. Total	4999	4919614984			

Means for Oneway Anova

Level	Number	Mean	Std Error	Lower 95%	Upper 95%
9 - 0.3	1000	16123.1	31.156	16062	16184
9 - 0.45	1000	15925.3	31.156	15864	15986
9 - 0.6	1000	15840.3	31.156	15779	15901
9 - 0.75	1000	15807.0	31.156	15746	15868
9 - 0.9	1000	15808.4	31.156	15747	15869

Std Error uses a pooled estimate of error variance

Means Comparisons

Comparisons for all pairs using Tukey-Kramer HSD

Confidence Quantile

q*	Alpha
2.72883	0.05

HSD Threshold Matrix

Abs(Dif)-HSD

	9 - 0.3	9 - 0.45	9 - 0.6	9 - 0.9	9 - 0.75
9 - 0.3	-120.24	77.62	162.59	194.50	195.91
9 - 0.45	77.62	-120.24	-35.27	-3.36	-1.95
9 - 0.6	162.59	-35.27	-120.24	-88.32	-86.91
9 - 0.9	194.50	-3.36	-88.32	-120.24	-118.83
9 - 0.75	195.91	-1.95	-86.91	-118.83	-120.24

Positive values show pairs of means that are significantly different.

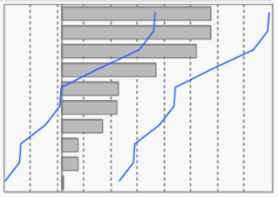
Connecting Letters Report

Level		Mean
9 - 0.3	A	16123.123
9 - 0.45	B	15925.269
9 - 0.6	B	15840.301
9 - 0.9	B	15808.389
9 - 0.75	B	15806.980

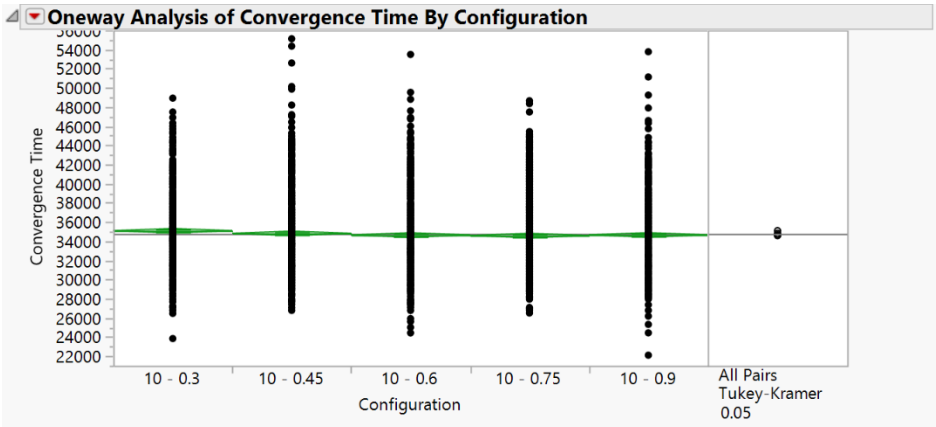
Levels not connected by same letter are significantly different.

Ordered Differences Report

Level	- Level	Difference	Std Err Dif	Lower CL	Upper CL	p-Value
9 - 0.3	9 - 0.75	316.1430	44.06105	195.908	436.3782	<.0001*
9 - 0.3	9 - 0.9	314.7340	44.06105	194.499	434.9692	<.0001*
9 - 0.3	9 - 0.6	282.8220	44.06105	162.587	403.0572	<.0001*
9 - 0.3	9 - 0.45	197.8540	44.06105	77.619	318.0892	<.0001*
9 - 0.45	9 - 0.75	118.2890	44.06105	-1.946	238.5242	0.0564
9 - 0.45	9 - 0.9	116.8800	44.06105	-3.355	237.1152	0.0614
9 - 0.45	9 - 0.6	84.9680	44.06105	-35.267	205.2032	0.3023
9 - 0.6	9 - 0.75	33.3210	44.06105	-86.914	153.5562	0.9429
9 - 0.6	9 - 0.9	31.9120	44.06105	-88.323	152.1472	0.9510
9 - 0.9	9 - 0.75	1.4090	44.06105	-118.826	121.6442	1.0000



H. TEN GROUND NODE LAYOUT



Oneway Anova

Summary of Fit

Rsquare	0.002473
Adj Rsquare	0.001675
Root Mean Square Error	3697.299
Mean of Response	34764.97
Observations (or Sum Wgts)	5000

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio	Prob > F
Configuration	4	169306156	42326539	3.0963	0.0148*
Error	4995	6.8282e+10	13670019		
C. Total	4999	6.8451e+10			

Means for Oneway Anova

Level	Number	Mean	Std Error	Lower 95%	Upper 95%
10 - 0.3	1000	35099.8	116.92	34871	35329
10 - 0.45	1000	34822.3	116.92	34593	35051
10 - 0.6	1000	34649.4	116.92	34420	34879
10 - 0.75	1000	34592.7	116.92	34363	34822
10 - 0.9	1000	34660.7	116.92	34432	34890

Std Error uses a pooled estimate of error variance

Means Comparisons

Comparisons for all pairs using Tukey-Kramer HSD

Confidence Quantile

q*	Alpha
2.72883	0.05

HSD Threshold Matrix

Abs(Dif)-HSD

	10 - 0.3	10 - 0.45	10 - 0.9	10 - 0.6	10 - 0.75
10 - 0.3	-451.21	-173.70	-12.14	-0.80	55.92
10 - 0.45	-173.70	-451.21	-289.64	-278.31	-221.59
10 - 0.9	-12.14	-289.64	-451.21	-439.88	-383.15
10 - 0.6	-0.80	-278.31	-439.88	-451.21	-394.48
10 - 0.75	55.92	-221.59	-383.15	-394.48	-451.21

Positive values show pairs of means that are significantly different.

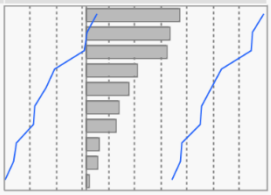
Connecting Letters Report

Level			Mean
10 - 0.3	A		35099.789
10 - 0.45	A	B	34822.281
10 - 0.9	A	B	34660.718
10 - 0.6	A	B	34649.386
10 - 0.75		B	34592.663

Levels not connected by same letter are significantly different.

Ordered Differences Report

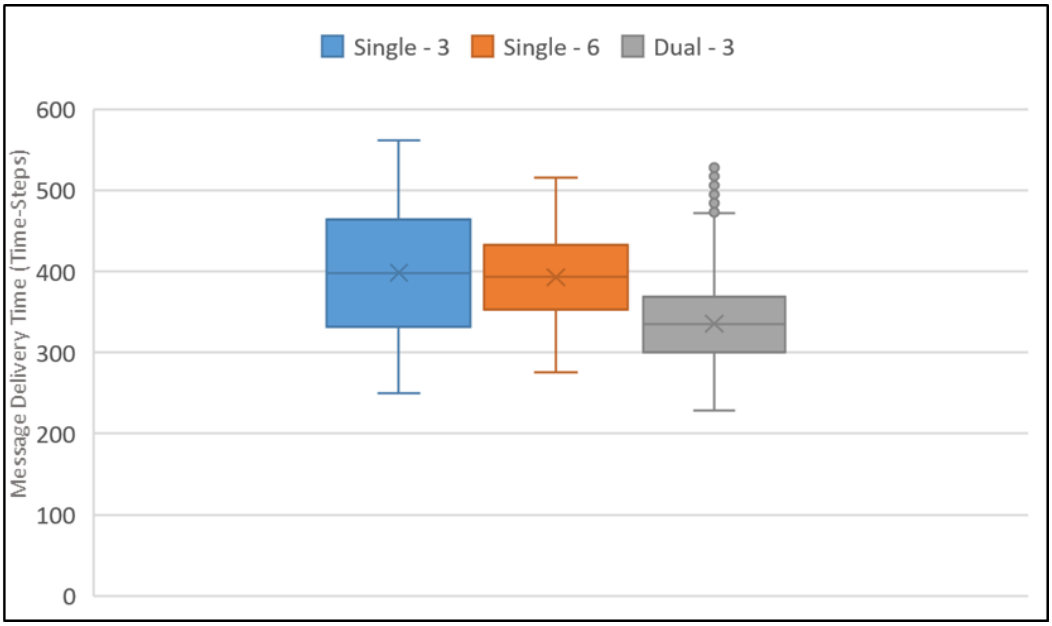
Level	- Level	Difference	Std Err Dif	Lower CL	Upper CL	p-Value
10 - 0.3	10 - 0.75	507.1260	165.3482	55.919	958.3335	0.0185*
10 - 0.3	10 - 0.6	450.4030	165.3482	-0.804	901.6105	0.0507
10 - 0.3	10 - 0.9	439.0710	165.3482	-12.136	890.2785	0.0609
10 - 0.3	10 - 0.45	277.5080	165.3482	-173.699	728.7155	0.4475
10 - 0.45	10 - 0.75	229.6180	165.3482	-221.589	680.8255	0.6350
10 - 0.45	10 - 0.6	172.8950	165.3482	-278.312	624.1025	0.8340
10 - 0.45	10 - 0.9	161.5630	165.3482	-289.644	612.7705	0.8656
10 - 0.9	10 - 0.75	68.0550	165.3482	-383.152	519.2625	0.9940
10 - 0.6	10 - 0.75	56.7230	165.3482	-394.484	507.9305	0.9970
10 - 0.9	10 - 0.6	11.3320	165.3482	-439.875	462.5395	1.0000



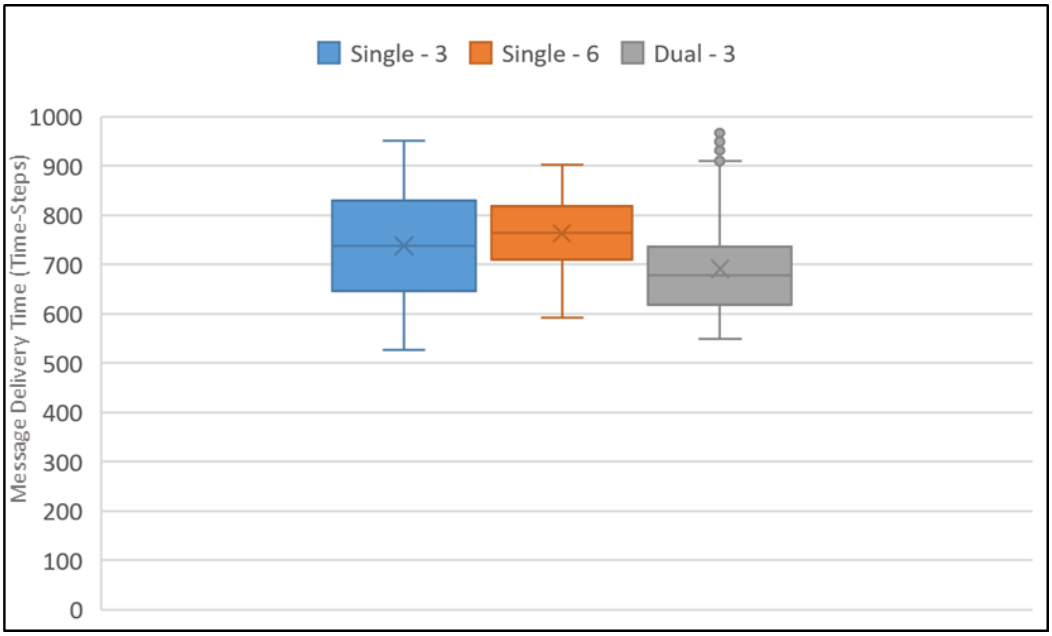
THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. MESSAGE DELIVERY TIME CHARTS

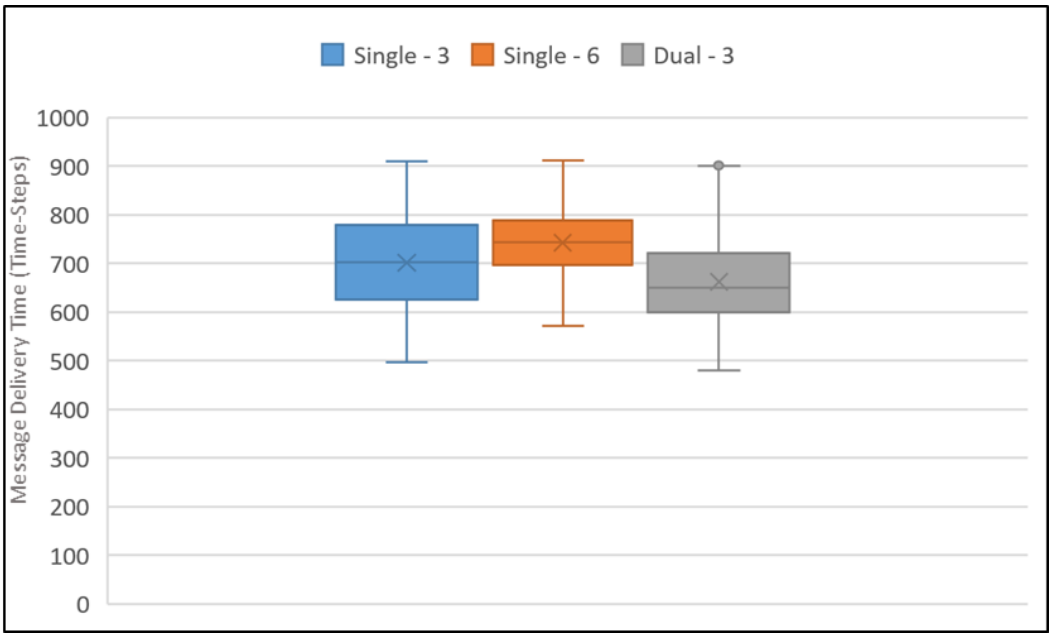
A. THREE GROUND NODE LAYOUT



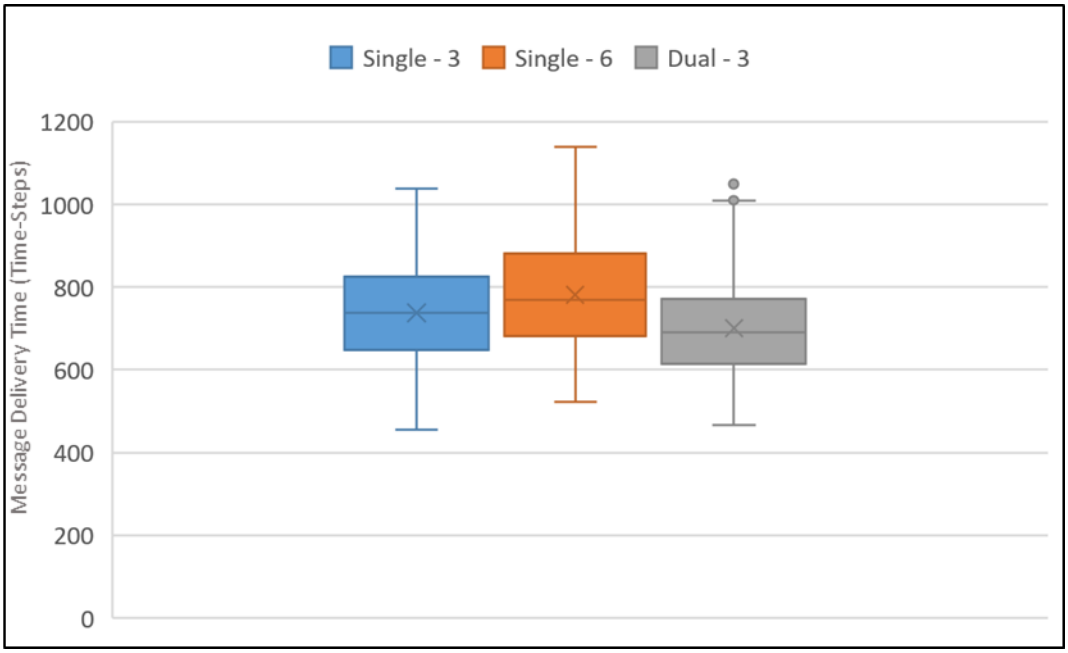
B. FOUR GROUND NODE LAYOUT



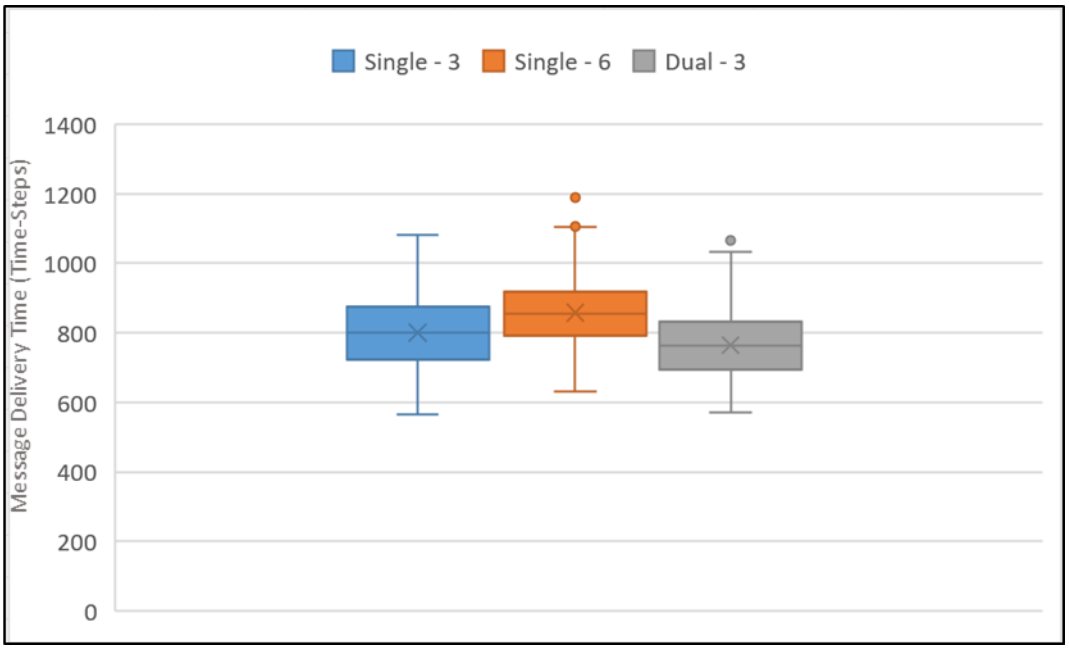
C. FIVE GROUND NODE LAYOUT



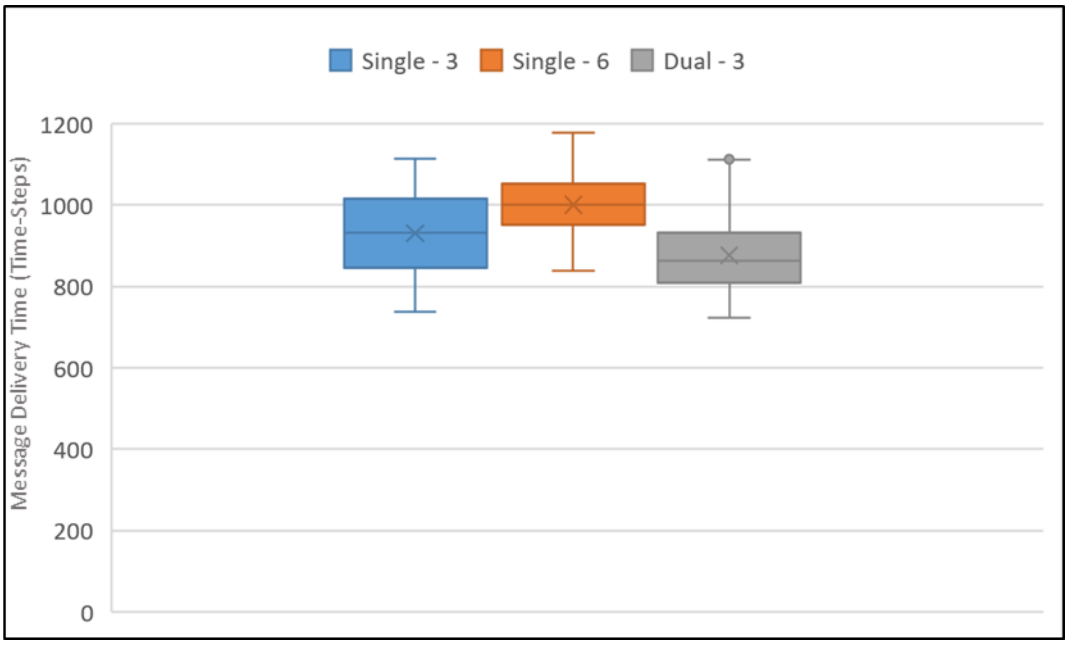
D. SIX GROUND NODE LAYOUT



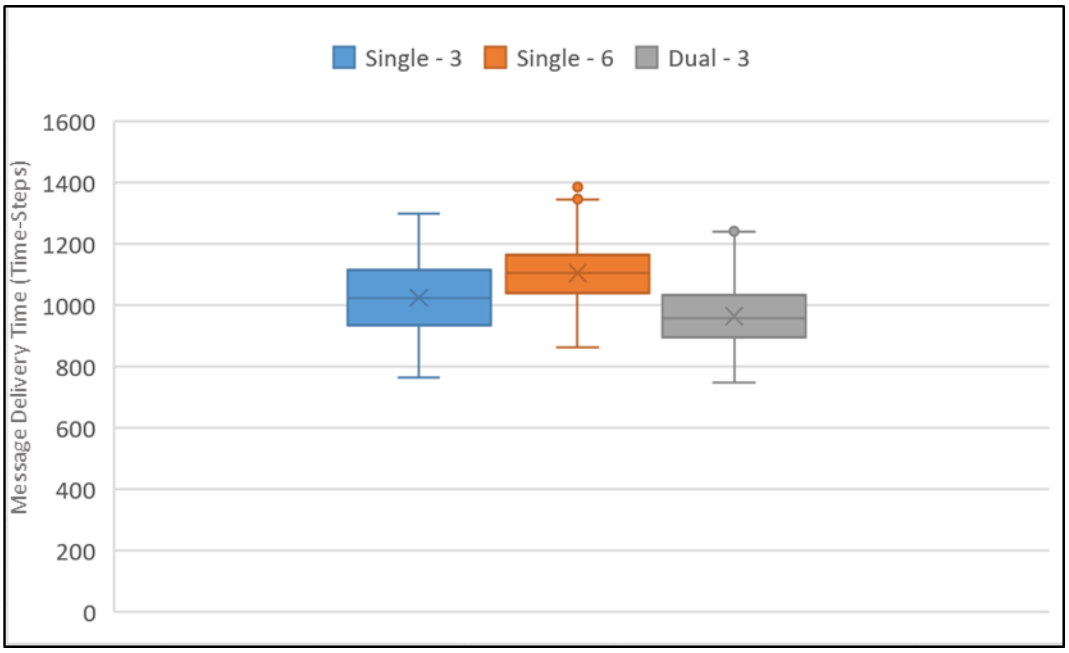
E. SEVEN GROUND NODE LAYOUT



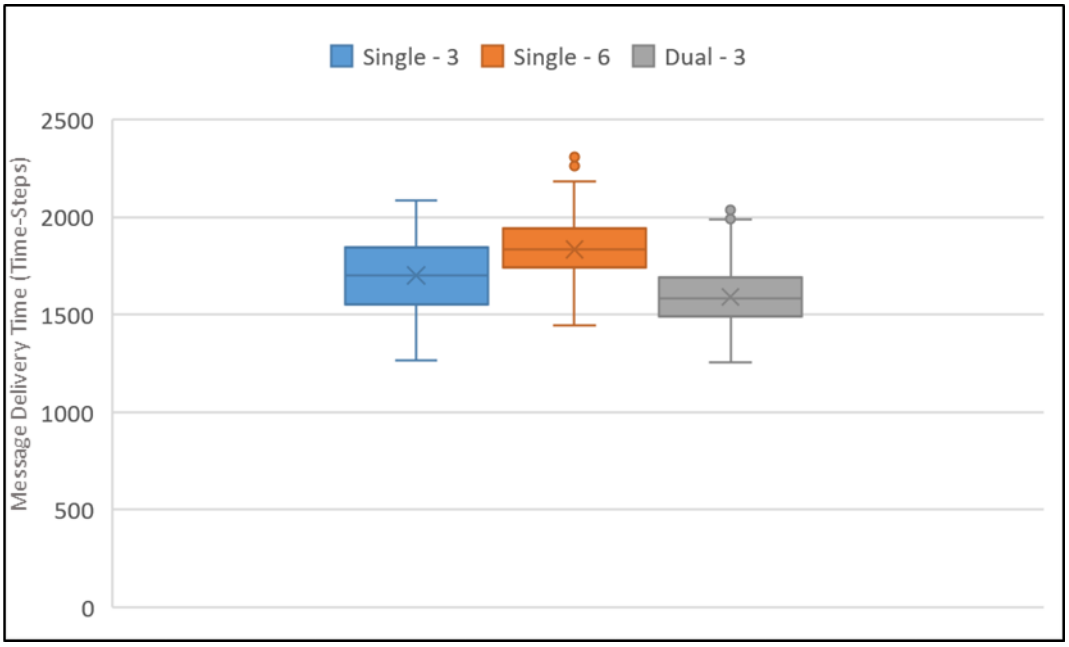
F. EIGHT GROUND NODE LAYOUT



G. NINE GROUND NODE LAYOUT



H. TEN GROUND NODE LAYOUT



LIST OF REFERENCES

Bamberger, R. J., Watson, D. P., Scheidt, D. H., & Moore, K. L. (2006). Flight demonstrations of unmanned aerial vehicle swarming concepts. *John Hopkins APL Technical Digest*, *27*(1).

Baran, P. (August 1964). *On distributed communications: Introducation to distributed communications networks.* The RAND Corporation. Retrieved from https://www.rand.org/content/dam/rand/pubs/research_memoranda/2006/RM3420.pdf

Barca, J. C. & Sekercioglu, Y. A. (2013). Swarm robotics reviewed. *Robotica*, *31*(3), 345-359. doi:10.1017/S026357471200032X.

Bonabeau, E. (1999). In Dorigo M., Theraulaz G. (Eds.), *Swarm Intelligence from Natural to Artificial Systems*. New York: Oxford University Press.

Bonabeau, E. & Meyer, C. (2001). Swarm Intelligence: A whole new way to think about business. *Harvard Business Review*, May 2001, 106-114.

Bürkle, A., Segor, F., & Kollmann, M. (2011). Towards autonomous micro UAV swarms. *Journal of Intelligent & Robotic Systems*, *61*(1), 339-353. doi:10.1007/s10846-010-9492-x.

Chapman, A. (2016). Types of drones: multi-rotor vs fixed-wing vs single rotor vs hybrid VTOL. Retrieved from https://www.auav.com.au/articles/drone-types/

Chung, T. H. (n.d.) ARSENL Retrieved May 6, 2018 from https://wiki.nps.edu/display/~thchung/ARSENL.

Chung, T. H., Clement, M. R., Day, M. A., Jones, K. D., Davis, D., & Jones, M. (2016). Live-fly, large-scale field experimentation for large numbers of fixed-wing UAVs. Paper presented at the *2016 IEEE International Conference on Robotics and Automation*, 1255-1262. doi:10.1109/ICRA.2016.7487257.

Chung, T.H., OFFensive swarm enabled tactics (OFFSET). (February 16, 2017). FedBizOpps, Retrieved from https://search.proquest.com/docview/1868569344

Davis, D. T., Chung, T. H., Clement, M. R., & Day, M. A. (2016). Consensus-based data sharing for large-scale aerial swarm coordination in lossy communications environments. Paper presented at the *2016 International Conference on Intelligent Robots and Systems*. 3801-3808. doi:10.1109/IROS.2016.7759559.

Davis, D. T., Chung, T. H., Clement, M. R., & Day, M. A. (2016). Multi-swarm infrastructure for swarm versus swarm experimentation. *Distributed Autonomous Robotic Systems*. 649-663. doi:10.1007/978-3-319-73008-0_45.

Day, M. A., Clement, M. R., Russo, J. D., Davis, D., & Chung, T. H. (2015). Multi-UAV software systems and simulation architecture. Paper presented at the *2015 International Conference on Unmanned Aircraft Systems*. 426-435. doi:10.1109/ICUAS.2015.7152319.

Department of Defense. (2013). Unmanned systems integrated roadmap FY2013-2038. Retrieved from https://publicintelligence.net/dod-unmanned-systems-2013/

Ducatelle, F., Di Caro, G., & Gambardella, L. Cooperative stigmergic navigation in a heterogeneous robotic swarm. Retrieved from http://www.swarmanoid.org/upload/pdf/DucDicGam10b.pdf

Elliot, M. & Stevens, T. (2016). Dynamic range extension using HARLEQUIN and HAIL. *MILCOM 2016 - 2016 IEEE Military Communications Conference (MILCOM)*, Baltimore, MD, 2016, pp. 835-841. doi: 10.1109/MILCOM.2016.7795433.

Military Communication. (2007) In *Encyclopaedia Britannica*. Retrieved July 12, 2017, from https://www.britannica.com/technology/military-communication

Fahlstrom, P. G. (2012). In Gleason T. J. (Ed.), *Introduction to UAV Systems* (4th ed.). Hoboken, NJ: John Wiley & Sons.

Foster, J. (2014). Swarming Unmanned Aerial Vehicles (UAVs): Extending Marine Aviation Ground Task Force Communications Using UAVs (Master's thesis). Retrieved from http://calhoun.nps.edu/handle/10945/44564.

Fraser, B., & Hunjet, R. (2016). Data ferrying in tactical networks using swarm intelligence and stigmergic coordination. *2016 26th International Telecommunication Networks and Applications Conference (ITNAC)*, 1-6. doi:10.1109/ATNAC.2016.7878772.

Fraser, B., Hunjet, R., & Coyle, A. (2017). A swarm intelligent approach to data ferrying in sparse disconnected networks. doi:10.1080/17445760.2017.1370094

Fuhrmann, M., & Horowitz, M. C. (2017). Droning on: explaining the proliferation of unmanned aerial vehicles. *International Organization*, *71*(2), 397. doi:10.1017/S0020818317000121.

Gaudiano, P., Shargel, B., Bonabeau, E., & Clough, B. T. (2003). Swarm intelligence: A new C2 paradigm with an application to control swarms of UAVs. Retrieved from http://www.dtic.mil/docs/citations/ADA418490

Goss, S., Aron, S., Deneubourg, J. L., & Pasteels, J. M. (1989). Self-organized shortcuts in the argentine ant. *Naturwissenschaften*, *76*(12), 579-581. doi:10.1007/BF00462870.

Gravagne, I. A., & Marks, R. J. (2007). Emergent behaviors of protector, refugee, and aggressor swarms. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 37*(2), 471-476. doi:10.1109/TSMCB.2006.883427.

Hambling, D. (2016). Gliding algorithm lets drones surf the winds for hours. Retrieved from http://www.popsci.com/new-software-lets-drones-surf-winds-for-hours

Harper, J. (2017). Special operations command creating new drone technology initiative. Retrieved from https://about.bgov.com/blog/special-operations-command-creating-new-drone-technology-initiative/

Heimfarth, T. & De Araujo, J. P. (2014). Using unmanned aerial vehicle to connect disjoint segments of wireless sensor network. Paper presented at the *2014 IEEE 28th International Conference on Advanced Information Networking and Applications*. 907-914. doi:10.1109/AINA.2014.110.

Henkel, D., Brown, T. X., (2006). On controlled node mobility in delay-tolerant networks of unmanned aerial vehicles. Retrieved from http://ecee.colorado.edu/~timxb/timxb/pubs/06UAVMobControl.pdf

Hunjet, R. (2015). Autonomy and self-organisation for tactical communications and range extension. doi:10.1109/MilCIS.2015.7348941.

Hunjet, R., Stevens, T., Elliot, M., Fraser, B., and George, P. (2017) Survivable communications and autonomous delivery service: A generic swarming framework enabling communications in contested environments. *MILCOM 2017 - 2017 IEEE Military Communications Conference (MILCOM)*, Baltimore, MD, 2017, pp. 788-793. doi: 10.1109/MILCOM.2017.8170775.

Hunjet, R., Fraser, B., Stevens, T., Hodges, L., Mayen, K., Barca, J. C., Cochrane, M., Cannizzaro, R., & Palmer, J. L. (2018). Data ferrying with swarming UAS in tactical defence networks. *2018 IEEE International Conference on Robotics and Automation.*

Icosystem. (n.d.) Retrieved July 2, 2017, from www.icosystem.com

JMP. (n.d.) Retrieved May 6, 2018 from https://www.jmp.com/en_us/about.html

Kushleyev, A., Mellinger, D., & Kumar, V. (2013). Towards a swarm of agile micro quadrotors. *Autonomous Robots, 35*(4), 287-300. doi:10.1007/s10514-013-9349-9.

Lau, D. (2015). Investigation of Coordination Algorithms for Swarm Robotics Conducting Area Search (Master's thesis). Retrieved from http://calhoun.nps.edu/handle/10945/47293

Legras, F., Glad, A., Simonin, O., & Charpillet, F. (2008). Authority sharing in a swarm of UAVs: Simulation and experiments with operators. *Simulation, Modeling, and Programming for Autonomous Robots* (pp. 293-304). Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/978-3-540-89076-8_29.

Luke, S. (2015). Multiagent simulation and the MASON library. George Mason University. Retrieved from https://cs.gmu.edu/~eclab/projects/mason/manual.pdf

Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Wheeler, R, & Ng, A. (2009). ROS: an open-source robot operating system. *ICRA Workshop on Open Source Software.*

Ochoa, Victoria. (2016). ARSENL team advances team vs. team swarm research. Retrieved from https://my.nps.edu/-/arsenl-team-advances-team-vs-team-swarm-research

O'Hara, K. J., Bigio, V. L., Dodson, E. R., Irani, A. J., Walker, D. B., & Balch, T. R. (2005). Physical path planning using the GNATs. Paper presented at the *2005 International Conference on Robotics and Automation.* 709-714. doi:10.1109/ROBOT.2005.1570201.

Parunak, H. V., Posdamer, J., Sauter, J. A., & Brueckner, S. (2001). Adaptive control of distributed agents through pheromone techniques and interactive visualization. Retrieved from http://www.dtic.mil/docs/citations/ADA398009

Parunak, H. V. (2003). Making swarming happen. Retrieved from http://abcresearch.org/abcresearch.org/papers/

Parunak, H. V., Purcell, M., O'Connell, R. (2002). Digital pheromones for autonomous coordination of swarming UAVs. Retrieved from http://abcresearch.org/abcresearch.org/papers/

Parunak, H. V., Brueckner, S., & Sauter, J. (2002). Digital pheromone mechanisms for coordination of unmanned vehicles. Retrieved from http://abcresearch.org/abcresearch.org/papers/

Payton, D., Daily, M., Estowski, R., Howard, M., & Lee, C. (2001). Pheromone robotics. *Autonomous Robots, 11*(3), 319-324. doi:1012411712038.

Sauter, J. A., Matthews, R., Parunak, H. V., & Brueckner, S. A. (2005). Demonstration of digital pheromone swarming control of multiple unmanned air vehicles. Retrieved from http://abcresearch.org/abcresearch.org/papers/

Sauter, J., Matthew, R. S., Robinson, J. S., Moody, J., & Riddle, S. P. (2009). Swarming unmanned air and ground systems for surveillance and base protection. Retrieved from https://www.researchgate.net/publication/242072697_Swarming_Unmanned_Air_and_Ground_Systems_for_Surveillance_and_Base_Protection

Sayler, K. (2015). A world of proliferated drones: a technology primer. Retrieved from http://drones.cnas.org/wp-content/uploads/2016/03/CNAS-World-of-Drones_052115.pdf

Scharre, P. (2014). Robotics on the battlefield, part II: the coming swarm. *Center for a New American Security*. Retrieved from https://www.cnas.org/publications/reports/robotics-on-the-battlefield-part-ii-the-coming-swarm

Stevens, T. (2017). HAIL-ROS interface design, Version 3.1. Defense Science and Technology Group.

Sukhankin, S. (2017). Russian electronic warfare in Ukraine: between real and imaginable. Retrieved from http://www.realcleardefense.com/articles/2017/05/26/russian_electronic_warfare_in_ukraine_111460.html

Usbeck, K., Gillen, M., Loyall, J., Gronosky, A., Sterling, J., Kohler, R., Newkirk, R., Canestrare, D. (2014). Data ferrying to the tactical edge: a field experiment in exchanging mission plans and intelligence in austere environments. doi:10.1109/MILCOM.2014.218.

United States Marine Corps. (1996). *Command and Control* (*MCDP 6*). Washington, D.C.

Vasarhelyi, G., Viragh, C., Somorjai, G., Tarcai, N., Szörenyi, T., Nepusz, T., & Vicsek, T. (2014). Outdoor flocking and formation flight with autonomous aerial robots. Paper presented at the *International Conference on Intelligent Robots and Systems.* 3866-3873. doi:10.1109/IROS.2014.6943105.

Walter, B., Sannier, A., Reiners, D., & Oliver, J. (2006). UAV swarm control: calculating digital pheromone fields with the GPU. *The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology*, *3*(3), 167-176. doi:10.1177/154851290600300304.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California