Design and Evaluation of UAV Swarm Command and Control Strategies

Alexander G. Madey[1,2], Gregory R. Madey[2]
[1]Trinity School at Greenlawn, [2]University of Notre Dame
South Bend, IN USA
alexmadey@gmail.com, gmadey@nd.edu

Keywords: UAVs, Emergent Behavior, Command and Control, Swarming, NetLogo

Abstract

We present an approach to developing unmanned aerial vehicle (UAV) swarming behaviors and command and control (C2) strategies to govern them. In recent years, the military has become increasingly interested in the development and applications of UAVs. Recent attention has shifted toward designing UAVs which are not only unmanned, but also autonomous or self-controlled. One possible method is to utilize a large number of small, autonomous UAVs which form a cohesive group or "swarm" to accomplish complex missions as a whole. Swarms offer numerous benefits over single UAVs which include higher coverage, redundancy in numbers and reduced long-range bandwidth requirements. A major challenge to engineering a swarm is not only designing the swarming behavior, but finding an effective way to control the behavior so that the swarm can be directed to complete the desired mission. In this paper, we used the agent-based modeling toolkit NetLogo to create two different mission types. We then created UAV swarming behaviors and ways in which those behaviors can be altered to accomplish each mission. Despite the fact that these models are still preliminary, and lacking in full realism, this work has demonstrated the potential usefulness of agent-based modeling in the engineering of UAV swarms.

1. INTRODUCTION

In recent years, the military has become increasingly interested in expanding the capabilities of unmanned aerial vehicles (UAVs) [Miller 2006, Phillips 2008]. UAV technology has already solved an array of problems faced by manned aerial vehicles (MAVs). Since UAV pilots operate from the ground, pilot fatigue is less of an issue in long, "dull" missions. More importantly, pilots no longer must risk their lives in "dirty" or "dangerous" missions [Office of the Secretary of Defense 2002]. There are already a wide range of UAVs currently being used or developed by the military. These UAVs can range from tiny room-to-room reconnaissance UAVs to large UAVs used for high altitude surveillance and striking [Milam 2004]. However, despite the many advances in UAV technology, problems still arise. Even though they are remotely controlled, UAVs still require one or more operators on the ground. With demand for UAVs rapidly growing, the military could eventually outstrip their recruitment and training capacities, resulting in a shortage of UAV pilots. Another problem is the rapidly growing long-range bandwidth requirements needed for communication between UAVs and ground control, which can often be thousands of miles apart.

One possible solution to these problems is UAV swarms. UAV swarms would be made from a large number of small UAVs, each having certain behaviors which cause them to form into a swarm. This swarm would then be able to perform complex missions as a whole. The first challenge in engineering a swarm is designing the low-level behaviors which cause the UAVs to form a swarm. Swarming behaviors have already been demonstrated by simulations such as Reynold's BOIDS, which simulates the flocking of birds [Reynolds 1987]. More recent research has focused specifically on UAV swarms used for military purposes [Bugajski 2010, Ha 2010]. Another challenge to designing a functional swarm is finding ways in which the UAVs' behavior can be altered to give operators better control over the swarm as a whole [Hexmoor et al. 2005].

This paper explores an approach to designing UAV swarming behaviors and a way to command and control these behaviors in the completion of two distinct mission types. Using 2-dimensional NetLogo simulations, two possible applications of UAV swarms are explored. The first involves the detection and tracking of a contaminant plume, which could be the result of a chemical spill, a dirty bomb, or a chemical or biological weapon. The second focuses on the patrolling of a defined body of water for detection and tracking of vessels of interest which could be, for example, smugglers or enemy vessels.

In the remainder of this paper, we first provide background information on current UAV technology and swarming concepts. Next, we discuss the detailed methodology used to develop and implement the NetLogo simulations and present preliminary results. Finally, we provide a summary of results, conclusions, and suggestions for future research.

2. BACKGROUND

This paper extends work started in a project investigating the application of Dynamic Data Driven Applications Systems (DDDAS), a new simulation paradigm, to the problem of mission planning, and command and control of UAV swarms [Darema 2005, Madey et al. 2012, McCune 2013, Purta 2013a, Wei 2013].

2.1. UAV Swarm Challenges

The challenges to designing an effective UAV swarm are numerous. Many of these challenges lie in the hardware and sensors of UAVs; challenges which must be kept in mind when designing software. One issue, as mentioned above, is bandwidth requirements. The behavior of individual UAVs should preferably be as autonomous as possible, as frequent communication with command and control would use too much bandwidth. Frequent long-range communication could also drain large amounts of energy, as long-range radios can require significant power. For communication between the individual UAVs, short range relay radios can be used, which require far less power. Any software designed to control swarms needs to take these limitations into account.

Ideally, a simulation must take into account as realistically as possible the flight and sensor constraints of the UAVs. Our simulation does not take into account all of the physics and constraints involved with flying UAVs, but we do strive for a certain level of realism as an initial first step in the simulation of UAV swarms.

2.2. Swarming Concepts

In the development of swarms, it is critically important to have a strong grasp of the concepts behind swarming. As mentioned above, swarms are made up of multiple autonomous agents which form together into a cohesive unit. As a whole, the swarm exhibits “emergent behavior” which any single agent would not exhibit but which occur because of interactions between the agents. These emergent behaviors can often be difficult to predict by examining any single agent in the swarm. Swarming can be seen throughout nature, in organisms such as birds, fish and insects, serving in many ways as an inspiration for the designs of simulated swarms.

One early example of simulated swarming behavior is the BOIDS program, which simulates the flocking of birds [Reynolds 1987]. Despite seeming complex, BOIDS models swarming behavior using only three simple behavioral rules: alignment, cohesion and separation:

- Alignment – agents align their heading with their neighbors.
- Cohesion – agents steer towards a group of neighbors.
- Separation – agents steer away from their neighbors if they get too close.

Subsequently added rules include: evasion, pursuit, seeking and fleeing [Reynolds 1999, Slear 2006]:

- Evasion – agents steer away from the predicted location of another agent.
- Pursuit – agents steer towards the predicted location of another agent.
- Seeking – agents steer towards the location of another agent.
- Fleeing – agents steer away from the location of another agent.

Another example is Icosystem's aggressor-protector rules [Bonabeau et al. 2003, Gaudiano et al. 2003, 2005]:

- Rule 1 – agents move to keep a friendly agent between themselves and an aggressor agent.
- Rule 2 – agents move to keep themselves between a friendly agent and an aggressor agent.

When Rule 1 is applied to all of the agents, the agents mill about in a seemingly random manner but when Rule 2 is applied to all agents, they form into a tight cluster. When both rules are applied in varying ratios to agents, other interesting new behaviors can be generated. The behaviors generated by varying rule sets can be hard to predict simply with intuition. Agent-based modeling is a potentially useful solution in finding the effects of various rule sets.

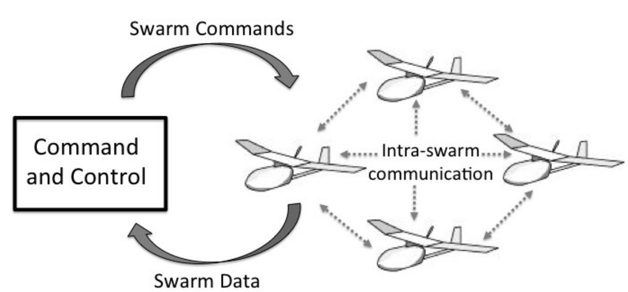

Figure 1: Diagram of Command and Control of a UAV Swarm

This could provide a solution for the command and control problem involved with “flying a swarm” of UAVs; rather than sending direct commands to each UAV, ground control will fly the swarm by adjusting the rules sets and associated parameters to influence the performance of the entire swarm (see Fig. 1). As mentioned above, this would save UAV battery power, long-range bandwidth requirements and reduce operator overload, all of which would be problematic if each individual UAV needed direct control.

3. METHODOLOGY

Agent-based modeling involves simulating each individual agent and its interactions with other agents and the environment. Such simulations can be written in general purpose programming languages, such as Java using supporting libraries such as Repast or MASON [Nikolai and Madey 2009]. We used the simulation toolkit NetLogo for the research in this paper [Wilensky 1999]. NetLogo gives the developer a visual representation of agents' behavior and allows the developer to easily modify parameters with switches, buttons, sliders, etc. as the program is running. This gives the developer an up-to-date view of how different parameters affect the simulation.

We designed two distinct surveillance scenarios to evaluate the feasibility of command and control of a swarm as described in Section 2.2. Our first scenario involves the surveillance and tracking of a contaminant plume, extending the work of [Daniel & Wietfeld 2011, Kovacina, et al. 2002]. Our second scenario centers around the detection and tracking of sea vessels crossing a body of water.

In the contaminant plume scenario, we modeled the plume as a circular area with the highest contaminant concentration in the center, with a decreasing concentration towards the edges of the plume. The contaminant we modeled could represent a chemical, biological or radioactive weapon or an industrial accident as well as a variety of other natural contaminants like volcanic gasses or smoke from forest fires. The objective of our UAV swarm is to get a well distributed map of the plume. Each UAV in our swarm is capable of taking a sensor measurement of the contaminant concentration in its vicinity. The UAVs then broadcast this measurement to their neighbors within a defined range, which can be manipulated by ground control. After broadcasting its own measurements and receiving its neighbors' measurements, the UAVs then use the *cohere rule* from BOIDS (see Section 2.2) to steer towards the neighbor with the highest reading (see Fig. 2). The maximum coherence turn (steering angle) of the UAVs can also be manipulated by ground control. As the UAVs are collecting measurements, the ground pilot is provided with an up-to-date statistical display of the standard deviation of the sensor measurements. Using this information, the ground pilot can make the necessary adjustments required to get the optimal range of measurements depending on the size of the plume.

In our second scenario, we simulated a wide channel with sea vessels periodically crossing it. The channel could represent a border crossing, and the vessels could represent possible smugglers, pirates or hostile vessels. The objective for the UAVs is to find and then track as many of the vessels as possible. The swarm is made up of two different heterogeneous UAV types: *Searchers* and *Pursuers*.

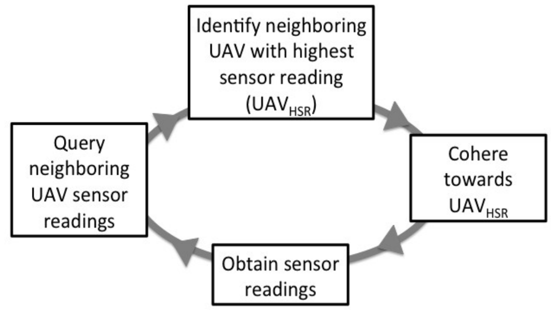

Figure 2: Diagram of individual UAV behavior in contaminant plume scenario.

The ratio of *Searchers* to *Pursuers* can be modified by ground control, possibly depending on alternate intelligence sources or on data collected directly by the swarm. *Searchers* navigate using only the original three BOIDS rules, modified for maximum coverage. If a *Searcher* detects a vessel, it transmits the vessel's location, speed and heading to a set number of local *Pursuers* (see Fig. 3).

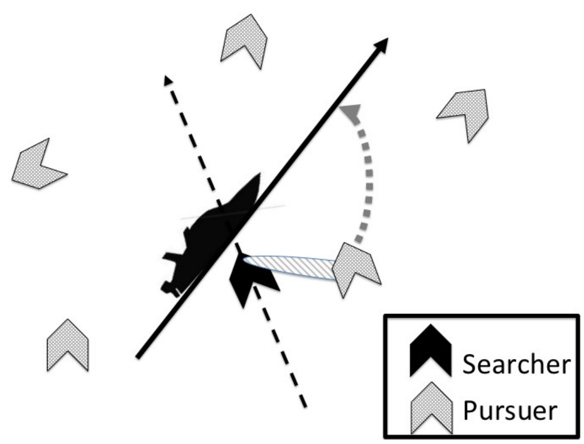

Figure 3: Diagram showing the Searcher UAV and Pursuer UAV behaviors. When a Searcher discovers a target vessel, it calls nearby Pursuers to converge on and track the vessel. The ground controller can modify the number of Pursuers that are designated to track each detected vessel.

The number of *Pursuers* can be determined by ground control. The *Pursuers* then use the extended BOIDS *seeking rule* to intercept the projected path of the detected vessel. The P*ursuers* then continuously circle around the vessel until it reaches the other side of the channel. If a P*ursuer* sees a new vessel itself, it will immediately track the vessel and call in more *Pursuers* if necessary. Our performance metrics are the percentage of vessels detected and successfully tracked and the average number of times each vessel is seen, representing how well each vessel is tracked.

4. RESULTS

4.1. Contaminant Plume Scenario

The objective of our contaminant plume scenario was to simulate the mapping of a contaminant plume by a UAV swarm. In this scenario, the ground controller can control the maximum *coherence turn* of the UAVs. A higher maximum *coherence turn* results in a tighter grouping for the swarm, while a lower value results in a more dispersed swarm. Figure 4a demonstrates a sub-optimal maximum *coherence* value, while Figure 4b demonstrates an improved value. Figure 4c shows a graph of the standard deviation of the entire swarm's sensor measurements, representing how well distributed the measurements are. Ground controllers would be provided with an up-to-date version of the graph, giving them feedback and allowing them to find optimal parameters.

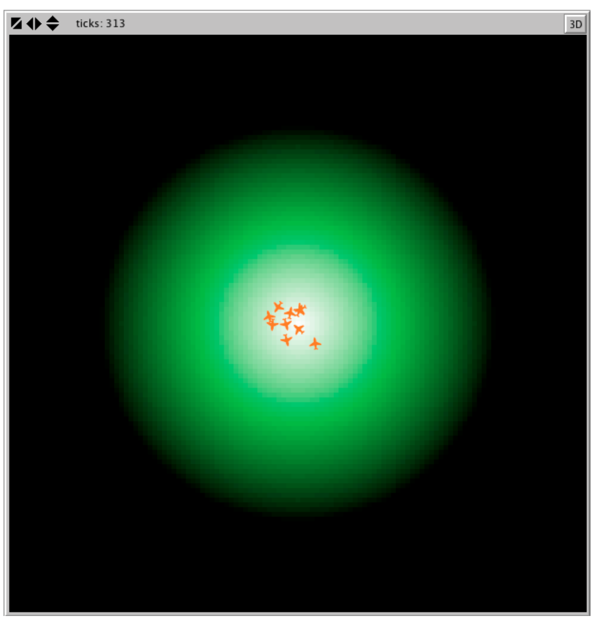

Figure 4a: Screen capture of the swarm mapping the contaminant plume. The swarm's maximum coherence turn value is too high, causing the UAVs to clump in the center instead of being distributed across the plume.

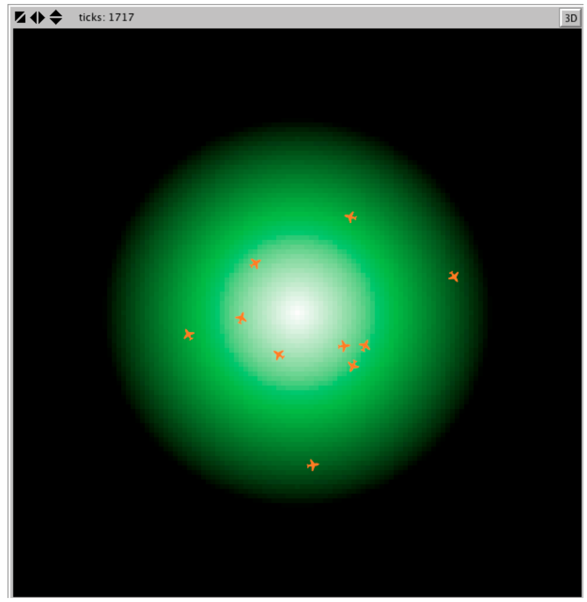

Figure 4b: Screen capture of the swarm mapping the contaminant plume. The swarm's max coherence turn value has been reduced by ground control, causing the UAVs to better disperse themselves for more distributed measurements.

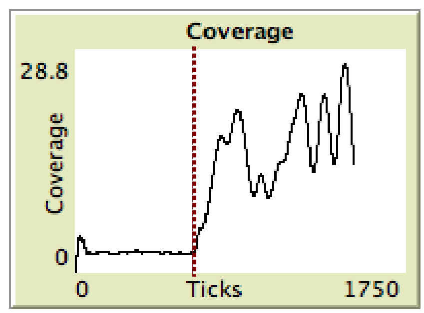

Figure 4c: A simulation performance graph in the contaminant plume scenario. It shows the standard deviation of the swarm's sensor measurements every tick. The vertical line represents a decrease in max coherence turn similar to what is seen in Figures 4a and 4b. This graph would be kept up to date and displayed to ground control, allowing any necessary adjustments to be made.

4.2. Vessel Tracking Scenario

The objective of our vessel tracking scenario was to simulate the detection and tracking of vessels crossing a channel using two types of UAVs: *Searchers* and *Pursuers*. The ground pilot is able to control two main swarm variables: the ratio of *Pursuers* to *Searchers* and the number of *Pursuers* assigned to track each detected vessel. We used two performance metrics: the percent of vessels detected and tracked and the average number of times each vessel is seen by UAVs, representing how well each vessel is tracked.

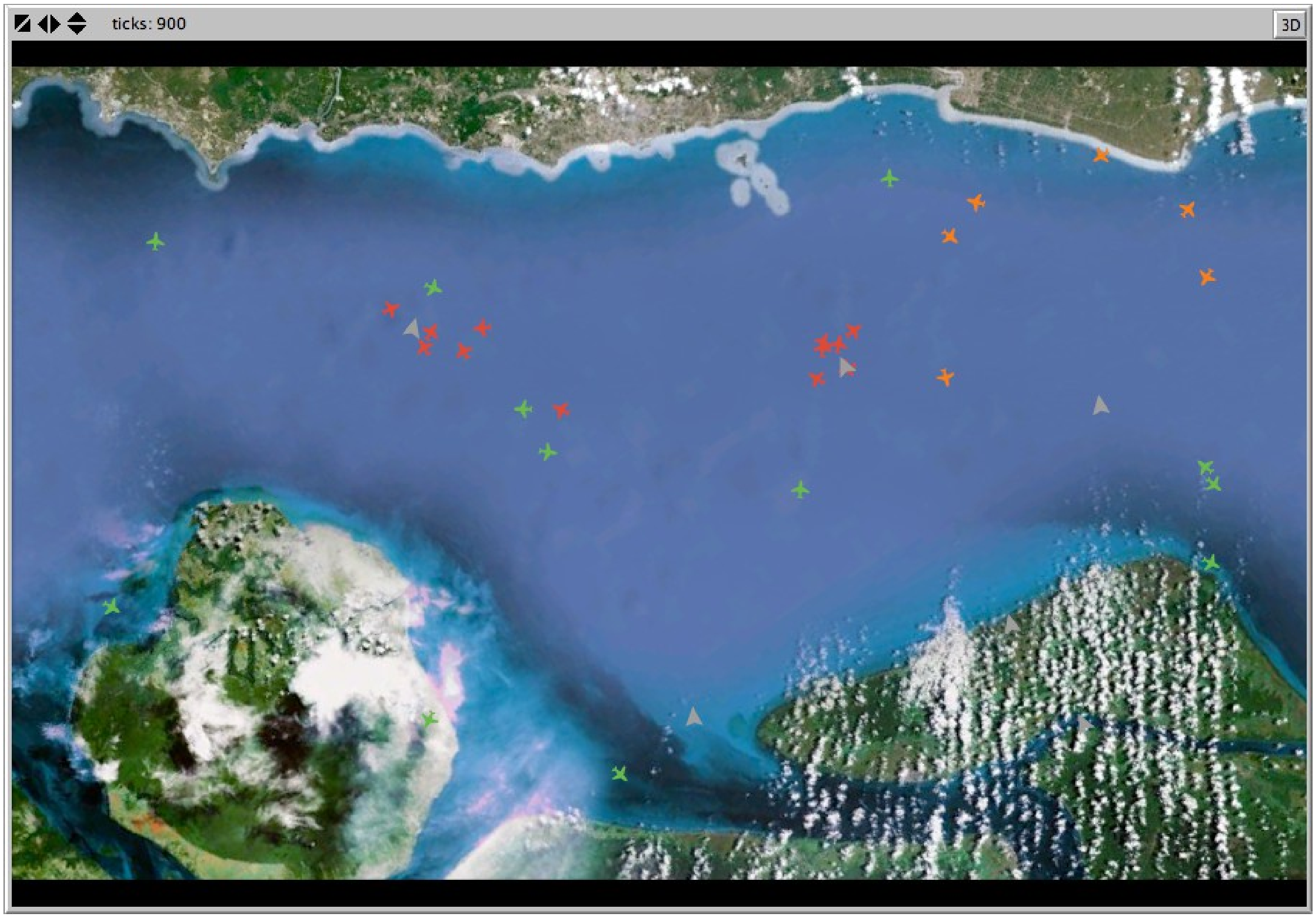

Figure 5a: Screen capture of the vessel tracking scenario. Gray arrowheads represent target vessels, green airplanes represent searchers, orange airplanes represent unassigned pursuers and red airplanes represent pursuers which are actively tracking vessels. (background from Google Earth)

Figure 5a is a screen capture from the vessel tracking scenario. In this figure, gray arrowheads represent target vessels, green airplanes represent *Searchers*, orange airplanes represent unassigned *Pursuers* and red airplanes represent *Pursuers* which are actively tracking vessels. This color coding system makes it easy to visualize the behavior of the *Pursuers* and *Searchers* and their interactions with each other. Figures 5b and 5c are simulation performance graphs. Figure 5b displays the percent of vessels successfully detected and tracked over time in simulation ticks. Figure 5c displays the average number of times each vessel is seen by UAVs. The vertical lines in each figure represent a command from ground control to reduce the maximum number of UAVs allocated to track each detected vessel. As can be seen in these graphs, the percent of vessels detected increases while the average number of times each vessel is seen goes down. In other words, a higher percentage of vessels are detected and tracked, but the vessels are not tracked as well. Careful observation of the simulation in action showed that this is because when too many *Pursuers* follow each vessel, despite tracking those vessels very well, there are not enough *Pursuers* available to track newly detected vessels. This demonstrates how changing a parameter in one potential situation within the scenario can improve the swarm's performance in one metric, but hurt it in another.

We found that in situations where there was a high number of UAVs and a low frequency of vessels, a higher number of *Pursuers* could be assigned to track each vessel, as there were still sufficient numbers of *Pursuers* to track new vessels. In situations where there was a lower number of UAVs and a higher frequency of vessels, we found that fewer *Pursuers* should be assigned to each vessel, because, as mentioned in the previous paragraph, there thus are not enough *Pursuers* to follow every vessel, and too many vessels slip past without being tracked. It is up to ground controllers to determine the ideal parameters for each situation, depending on how well they want each vessel to be tracked and on how many vessels they are willing to allow to slip through.

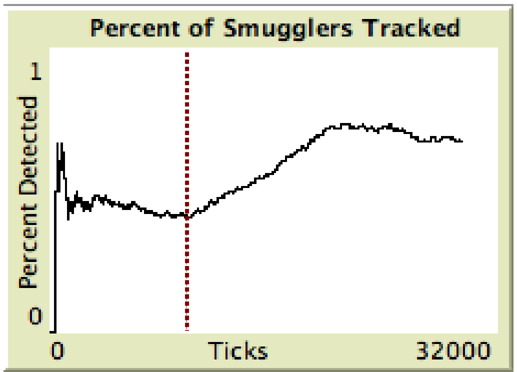

Figure 5b: A simulation performance graph in the vessel tracking scenario. It displays the percent of vessels successfully detected and tracked over time in simulation ticks. The vertical line represents a command from ground control to reduce in the maximum number of UAVs allocated to track each detected vessel. This demonstrates a way in which the ground controller can improve the swarm's performance.

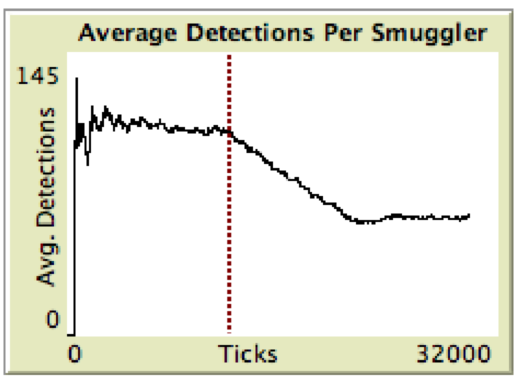

Figure 5c: Another simulation performance graph in the vessel tracking scenario. It displays the average times each vessel is seen by the UAVs representing how well each vessel is tracked. The vertical line represents a decrease in the number of UAVs allocated to each vessel, similar to figure 6. This demonstrates that while changing a parameter may improve one metric, it may hurt another. Ground controllers have to find the optimal parameters depending on the situation for the best swarm performance.

5. RELATED WORK

The work described in this paper is part of a larger AFSOSR funded project investigating the application of DDDAS principles to the command and control UAV swarms. On that project, Purta et al. developed a modular testbed that permits two agent-based simulations to run in parallel. One is a simulation intended to be used for the command and control of UAVs by ground pilots while the other simulates swarms of UAVs in various scenarios [Purta et al. 2013b]. McCune applied UAV swarm search methods based on a cooperative cleaning algorithm and the SWEEP algorithm for use in the testbed [McCune and Madey 2013]. Wei et al. developed and presented an agent based control framework for efficient swarm monitoring and mission planning. The framework employs a control agent for task assignment and multiple UAV agents for distributed task scheduling [Wei et al. 2013].

6. SUMMARY, CONCLUSIONS AND FUTURE RESEARCH

UAV swarms have been receiving increased attention in recent years from the military. We used the NetLogo agent-based simulation platform to create two distinct scenarios: mapping a contaminant plume and detecting and tracking vessels crossing a body of water. We then engineered UAV swarming behaviors and explored ways in which they can be controlled by ground pilots to accomplish both of these scenarios. Certain parameters can be applied to the individual UAVs which alter the swarm's performance as a whole. The swarm behaviors generated by different individual UAV behaviors and the parameters which control them can be hard to predict simply with intuition. For example, in the vessel tracking scenario, we found through watching the simulation which parameters work best in certain situations. To enable replication of this work by other researchers, we have posted to the web the NetLogo source code for the two simulated scenarios: Plume.nlogo and Smuggle.nlogo [Madey 2013].

There are many areas for potential improvement in our current simulations. In the plume detection scenario, the shape of the contaminant plume was unrealistic, as it did not take factors like wind into account, the plume did not diffuse in real time and it was only 2-dimensional. With the given behaviors, the UAVs might not be able to map a more realistic plume efficiently. This is a potential area for further improvement. In the vessel tracking scenario, the vessels could be simulated more realistically. Currently, the vessels have a fixed heading and speed, but real vessels might make turns or change their speed. The *Pursuers* might have trouble tracking the vessels if they changed heading or speed significantly at any point. The simulation can be used to test different potential solutions to this problem. Despite the fact that these models are still preliminary, and lacking in full realism, this work has demonstrated the potential usefulness of agent-based modeling in the engineering of UAV swarms.

Acknowledgments

We thank Dr. Patrick Mooney (Trinity School) and Rachel Purta (Notre Dame) for valuable discussion. This research was supported in part under grants from the Air Force Office of Scientific Research, award No. FA9550- 11-1-0351, and the National Science Foundation, award No. 1063084.

References

Bonabeau, Eric, Carl W. Hunt, and Paolo Gaudiano. Agent-based modeling for testing and designing novel decentralized command and control system paradigms. *8th International Command and Control Research and Technology Symposium*, June 17 – 19, 2003, National Defense University Washington, DC, 2003.

Bugajski, Gabriel T. Architectural Considerations for Single Operator Management of Multiple Unmanned Aerial Vehicles. Master Thesis No. AFIT/GSE/ENV/10-M03. Air Force Inst of Tech Wright-Patterson AFB OH School of Engineering, 2010.

Daniel, Kai, and Christian Wietfeld. Using Public Network Infrastructures for UAV Remote Sensing in Civilian Security Operations. *Homeland Security Affairs, Supplement 3 (March 2011)*.

Darema, Frederica. "Dynamic data driven applications systems: New capabilities for application simulations and measurements." *Computational Science–ICCS 2005,* 661-712, 2005.

Gaudiano, Paolo, B Shargel, E Bonabeau, BT Clough. Swarm Intelligence: A New C2 Paradigm with an Application to Control Swarms. *8th ICCRTS Command and Control Research and Technology Symposium,* Washington DC. 2003

Gaudiano, Paolo, Eric Bonabeau, and Ben Shargel. "Evolving behaviors for a swarm of unmanned air vehicles." *Swarm Intelligence Symposium, 2005. Proceedings IEEE*. 2005.

Ha, Taegyun. The UAV Continuous Coverage Problem. Master's Thesis No. AFIT-OR-MS-ENS-10-03. Air Force Inst of Tech Wright-Patterson AFB OH Dept of Operational Sciences, 2010.

Hexmoor, Henry, Brian Mclaughlan, and Matt Baker. "Swarm Control in Unmanned Aerial Vehicles." *In Proceedings of International Conference on Artificial Intelligence (IC-AI), CSREA*. 2005.

Kovacina, Michael A., et al. "Multi-agent control algorithms for chemical cloud detection and mapping using unmanned air vehicles." *Intelligent Robots and Systems. IEEE/RSJ International Conference on*. Vol. 3. IEEE, 2002.

Madey, Alexander. Netlogo Programs: plume.nlogo and smuggle.nlogo, www3.nd.edu/~agent/swarm/swarms.zip 2013

Madey, Gregory R., Blake, M. Brian, Poellabauer, Christopher, Lu. Hongsheng., McCune, R. Ryan., and Yi Wei. "Applying DDDAS Principles to Command, Control and Mission Planning for UAV Swarms," *Procedia Computer Science* 9, 1177-1186, 2012.

McCune, R. Ryan and Gregory R. Madey, "Agent-Based Simulation of Cooperative Hunting with UAVs," In Proceedings of the 2013 Symposium on Agent Directed Simulation (ADS '13/SpringSim 2013). Society for Computer Simulation International, San Diego, CA, 2013.

Milam, Kevin M. *Evolution of control programs for a swarm of autonomous unmanned aerial vehicles*. Master's Thesis No. AFIT/GCS/ENG/04-15. Air Force Inst Of Tech Wright-Patterson AFB OH School Of Engineering And Management, 2004.

Miller, Patrick M. "Mini, Micro, and Swarming Unmanned Aerial Vehicles: A Baseline Study." Library Of Congress Washington DC Federal Research Div, 2006.

Nikolai, Cynthia and Madey, Gregory (2009). 'Tools of the Trade: A Survey of Various Agent Based Modeling Platforms'. *Journal of Artificial Societies and Social Simulation* 12(2)2 <http://jasss.soc.surrey.ac.uk/12/2/2.html>.

Office of the Secretary of Defense, Roadmap, Unmanned Aerial Vehicles. "Roadmap 2002-2027." *US DoD, December* (2002).

Phillips, Adrian N. *A secure group communication architecture for a swarm of autonomous unmanned aerial vehicles*. Master's Thesis No. AFIT/Gce/Eng/08-09. Air Force Inst Of Tech Wright-Patterson AFB OH Graduate School Of Engineering And Management, 2008.

Purta, Rachael, Saurabh Nagrecha, and Gregory Madey, "Multi-hop Communications in a Swarm of UAVs," In Proceedings of the 2013 Symposium on Agent Directed

Simulation (ADS '13/SpringSim 2013). Society for Computer Simulation International, San Diego, CA, 2013a.

Purta, Rachael, Mikolaj Dobski, Artur Jaworski, and Gregory Madey, “A Testbed for Investigating the UAV Swarm Command Problem Using DDDAS.” *International Conference on Computational Science (Procedia Computer Science),* Barcelona 2013b.

Reynolds, Craig W. "Flocks, herds and schools: A distributed behavioral model." *ACM SIGGRAPH Computer Graphics*. Vol. 21. No. 4. ACM, 1987.

Reynolds, Craig W. "Steering behaviors for autonomous characters." *Game Developers Conference. http://www. red3d. com/cwr/steer/gdc99*. 1999.

Slear, James N. *AFIT UAV swarm mission planning and simulation system*. Master's Thesis No. AFIT/GCE/ENG/06-08. Air Force Inst Of Tech Wright-Patterson AFB OH Dept Of Electrical And Computer Engineering, 2006.

Wei, Yi, Gregory R. Madey, and M. Brian Blake, "Agent-based Simulation for UAV Swarm Mission Planning and Execution," In Proceedings of the 2013 Symposium on Agent Directed Simulation (ADS '13/SpringSim 2013). Society for Computer Simulation International, San Diego, CA, 2013.

Wilensky, Uri. (1999). NetLogo. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL. http://ccl.northwestern.edu/netlogo/.

Appendix – Acronyms Used in Paper

ABM – Agent-based Modeling
ABS – Agent-based Simulation
AFIT – Air Force
AFOSR – Air Force Office of Scientific Research
AFRL – Air Force Research Laboratory
C2 – Command and Control
C2ISR – Command, Control, Intelligence, Surveillance and Reconnaissance
DC2 – Distributed Command and Control
DDDAS – Dynamic Data Driven Application System
ISR – Intelligence, Surveillance and Reconnaissance
MAV – Manned Aerial Vehicle, Micro Aerial Vehicle
UAV – Unmanned Aerial Vehicle
WPAFB – Wright-Patterson Air Force Base