**Formation Control for a Simplified Distributed System of Disposable, Limited Capability Multi-Domain Autonomous Systems in a Hostile Environment**

Dillon Mead

EECS Department, Embry-Riddle Aeronautical University

UAS 691: Uncrewed and Autonomous Systems Capstone Design Project I

Dr. Richard Stansbury

12 August 2025

I. INTRODUCTION

This report aims to describe the project I am researching, what portion I am responsible for, the relevance of this project to the Uncrewed Autonomous Systems Engineering Masters program, a summary of my accomplishments for the project as of writing, and a schedule detailing the remaining tasks for the second half of the project. This Uncrewed and Autonomous Systems Capstone Design Project is a small view related to a much larger research project being researched and developed at Oak Ridge National Laboratory. This Capstone Design Project has the potential for inclusion in the larger Oak Ridge National Laboratory project at a future date.

A. *What is the project?*

The title itself is descriptive, but more is needed for a proper understanding of my Capstone Design Project’s scope. Given a swarm of autonomous sensing platforms is deployed in some location, there is a desire for the swarm to operate without guidance and maintain effective sensing coverage for the guarded location. Distributed systems – especially for autonomous platforms – are desired and utilized because of their advantages over monolithic systems. Distributed systems can handle more complex tasks, have better physical coverage of a mission area, are scalable and robust, and are likely to be lower in cost from off-the-shelf mass production [1]. Still, distributed systems have their own various design problems. Some of these problems are: coordination and cooperation schemes, negotiation between team and individual goals, collision avoidance, and communication protocols [1]. The problems identified within the scope of this project focus on coordinated formation control of the swarm, coordination and cooperation of individual nodes within a swarm hierarchy, basic collision avoidance, and if possible, a communication protocol that allows nodes to relay transmissions which may have constrained power and lack of infrastructure.

B. *What is my role on the project?*

I am the only person working on this project. As defined above, the scope of the project is the responsibility I have determined for myself as the focus of the Capstone Design Project research.

C. *How is the work relevant to the MS UASE program?*

The content of this project is directly related to Uncrewed Autonomous Systems Engineering. It integrates core program disciplines such as autonomous systems coordination, formation control, swarm cooperation, systems engineering practices – in the form of Model-Based Systems Engineering (MBSE), and modeling the system using simulation to test in a financially inexpensive manner.

D. *What is the summary of activities for UAS 691?*

The activities for the Summer 2025 semester were a review of reference material for the relevant disciplines, a settlement on the Systems Engineering method (MBSE) and development paradigm (prototyping), early design of the system using the chosen methodology, initial development of a leader selection algorithm for the swarm, and a decision on the simulation modeling tool with a structural design of the simulation model.

E. *Highlights or any noteworthy accomplishments:*

No noteworthy accomplishments… yet. A highlight would be the selection to use Systems Modeling Language (SysML) with MBSE. The MBSE approach was chosen over traditional Systems Engineering methods because of its similarity to prototyping as an iterative process and its efficiency when making changes to the system or sharing the produced artifacts.

II. PROBLEM

A. *What research questions or problems am I trying to address? Provide a more detailed overview of my tasks.*

1) *Coordinated Formation Control:*

How can a swarm of autonomous nodes, deployed without explicit instructions, collectively decide on an appropriate geometric formation and then successfully maneuver?

2) *Cooperation and Collision Avoidance:*

How do individual nodes coordinate their movements to achieve and maintain this formation while avoiding collisions?

3) *Adaptability and Resilience:*

How does the swarm intelligently reconfigure itself to maintain communication and sensing coverage when some of its nodes are destroyed or fail in a hostile environment?

4) *Resilient Communications (Time Permitting):*

How can the swarm effectively relay command, control, and communication (C3) data when faced with limited power and poor connectivity?

B. *Hardware Elements*

No hardware is used for this project. The design of the system focuses on system function and maintains a generic, neutral solution for the system form. There is an assumption that sensing, propulsion, and RF communication are working components without failure.

C. *Software Elements*

A modeling tool and simulator are required for this project. As of now, draw.io is the modeling tool used to draw MBSE diagrams. Otherwise, the free tool *Modelio* could also be used. The simulator I chose is Adevs, a discrete-event simulator framework [2]. Discrete-event systems are a generalization of discrete-time systems and therefore allow time to be continuous – meaning that time only progresses when an event occurs and most of the time components of a discrete-event system do not have a change in state. This distributed system of autonomous nodes is a multitude of individual agents and the nodes only have meaningful events during discrete actions, therefore the simulator manages resources well for a large simulated swarm of this project.

D. *Discuss any other challenges or constraints to your work*

The major challenge I have encountered is time constraint. Since I work full time and also completing my remaining schooling I have a tight schedule and I have had some difficulties

III. DETAILED SUMMARY OF WORK TO DATE

1. Review of reference material for the relevant disciplines
   1. The focus of the review was primarily on formation control. Three primary references were identified with many additional references. [1] and [3] give detailed explanations of using graph theory for formation control.
   2. Development of the leader-follower algorithm is largely taken from [4]. Some of the additional references also cover the control theory of leader-follower algorithms in more detail.
2. Settlement on the Systems Engineering method (MBSE) and development paradigm (prototyping)
   1. Delligatti L. states in [5], “The [traditional] document-based approach to systems engineering is expensive… you incur a significant percentage of total life cycle cost maintaining that disjoint set of artifacts.” As a quicker and easier option, Model-based Systems Engineering (MBSE) will be utilized for the design of the system. The INCOSE Handbook explains, “In an MBSE approach, much of the information that has been traditionally captured in informal diagrams, text, and tables is captured in a descriptive system model.” [6].
   2. Continuing to follow [5], I will use the Systems Modeling Language, or SysML, as the MBSE modeling language. SysML is a dialect of the Unified Modeling Language (commonly used in software development) and is nearly contained within the UML, except for some additional artifacts [7]. SysML should help to identify logical errors of the system design easier and make sharing the artifacts and iterations on system design a quicker process.
   3. Aligned with the iterative process of MBSE, I believe this project would benefit most from rapid development using a prototyping modeling method. The deliverable schedule is structured with this iterative approach in mind.
3. Early design of the system using the chosen methodology
   1. MBSE artifacts created for this project are located in APPENDICES C-F below. These artifacts focus primarily on the system’s behavior and creating a visual flow that aids the development of algorithms.
4. Initial development of a leader selection algorithm for the swarm
   1. APPENDICES A and B give a depiction of the leader selection behavior. The algorithm awaits implementation in the simulation model and validation through testing.
5. Decision on the simulation modeling tool with a structural design of the simulation model.
   1. Discrete-time and discrete-event simulators were considered for use in this project. [2] was used as the foundation for weighing the differences of the two simulator types. The conclusion is on A Discrete Event Simulator Specification (Adevs) utilizing C++, which fulfills this role well and uses resources efficiently [8].

IV. NEXT STEPS

A. *Clearly define the tasks planned for UAS 692.*

The tasks as currently planned are:

1. Building the simulation model environment – the structure and basic behavior of the simulation is already developed. This may also include an implementation of a user interface to use as better visualization of the system behavior.
2. Implementing the leader selection algorithm. Leaders and followers exist in pairs, but any node may be a leader even if it follows another node. Only the node “n1” is a leader and does not follow.
3. The follower shall maintain the distance *d* at a minimum from any other node. This will count as successful collision avoidance for the project.
4. Testing and validation for successful completion of the deployment sequence. This is achieved when the individual nodes collectively form in the correct pattern and stay outside their minimum inter-node distance.
5. Next, the formation algorithm will be tested as swarm reformation when nodes are removed by random selection or rate. The swarm shall recognize a node is lost and respond within a reasonable amount of time. The response timeframe shall be defined by node’s communication and propulsion characteristics.
6. Implementation of an optimization algorithm for the sensing coverage area. This will validate mission success and can also be used to inform reformation needs.
7. Finally, the research project report shall be completed for review in November.

B. *Identify the timeline for remaining tasks.*

Table I  
Project deliverables and timeline

|  |  |  |
| --- | --- | --- |
| **#** | **Deliverables** | **Date** |
| 1 | Leader Selection Algorithm | 18/08/2025 |
| 2 | Deployment Sequence Completion and Simple Collision Avoidance | 25/08/2025 |
| 3 | Random Loss and Reformation Algorithm | 01/09/2025 |
| 4 | Optimization of Coverage Surface Area | 15/09/2025 |
| 5 | Optional: Communication Relay Algorithm | TBD |
| 6 | Draft Report | 04/11/2025 |
| 7 | Final Report | 25/11/2025 |

C. *Identify any risks/ issues that may impact the progress of your work and how you would intend to mitigate them.*

1. **Risk:** The algorithms for reformation and optimization could be more complex than anticipated, causing delays.
   * **Mitigation:** The use of a rapid prototyping methodology allows for quick iteration. Each iteration is its own finished state and problems should be identified earlier and easier. There are two individuals that are experts in simulation and control theory I can ask for assistance. They would likely be able to clarify any confusion with complex algorithms.
2. **Risk:** Project falls behind schedule because of strict time schedule.
   * **Mitigation:** Some buffer time is built into the project schedule. Bi-weekly meetings with Dr. Watson and weekly email updates should also help with pacing.

REFERENCES

[1] M. de Queiroz, X. Cai, M. Feemster, *Formation Control of Multi-Agent Systems: A Graph Rigidity Approach*, in Dynamics and Control of Electromechanical Systems, Hoboken, NJ, USA: Wiley & Sons, Inc., 2019.

[2] J. J. Nutaro, *Building Software for Simulation: Theory and Algorithms with Applications in C++*. Hoboken, NJ, USA: John Wiley & Sons, Inc., 2011.

[3] H.S. Ahn, *Formation Control: Approaches for Distributed Agents*, in Studies in Systems, Decision and Control, Vol. 205, Cham, Switzerland: Springer Nature, 2020.

[4] S. He, M. Wang, S. Dai, F. Luo, “Leader-Follower Formation Control of USVs with Prescribed Performance and Collision Avoidance”, *IEEE*, Vol. 15, No. 1, pp. 572-581, Jan 2019.

[5] L. Delligatti, *SysML Distilled: a Brief Guide to the Systems Modeling Language*. Upper Saddle River, NJ, USA: Addison-Wesley, 2014.

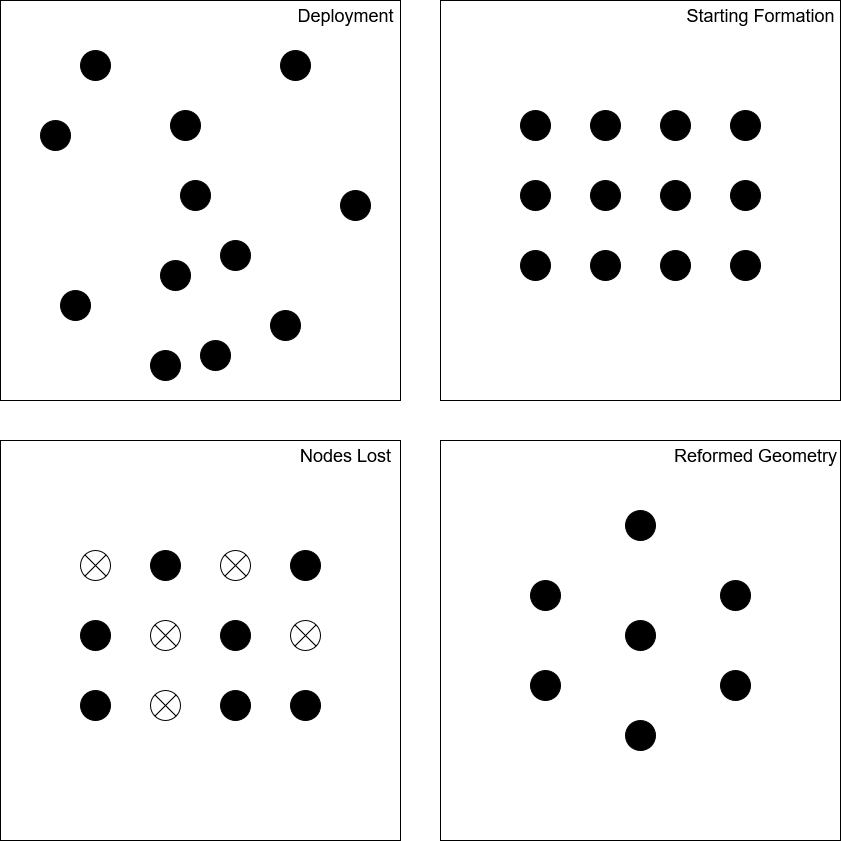
[6] *Systems Engineering Handbook*, 5th ed., International Council on Systems Engineering, Hoboken, NJ, USA, 2023.

[7] Object Management Group. SysML Open Source Project. <https://sysml.org/>. (accessed Aug 10, 2025).

[8] J. J. Nutaro. “smiz/adevs”. GitHub. <https://github.com/smiz/adevs/tree/main>. (accessed Aug 08, 2025).

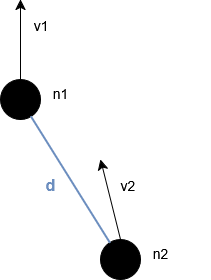
APPENDIX A

Deployment, Formation and Reformation



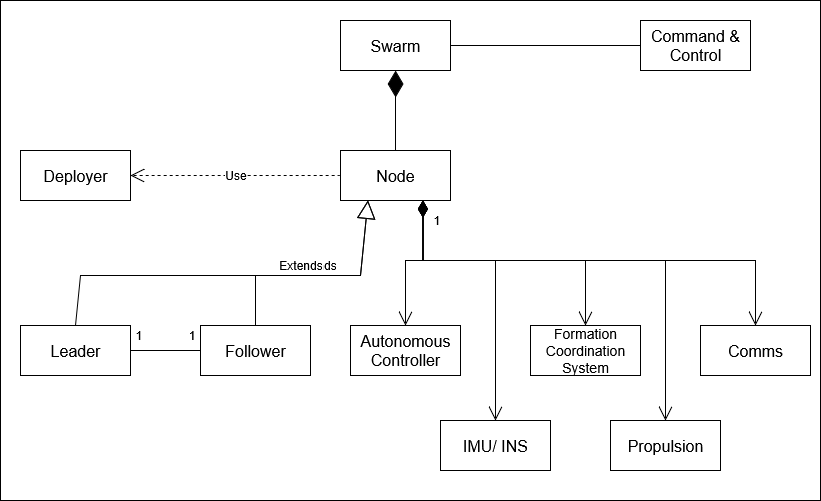
APPENDIX B

Leader-Follower Structure



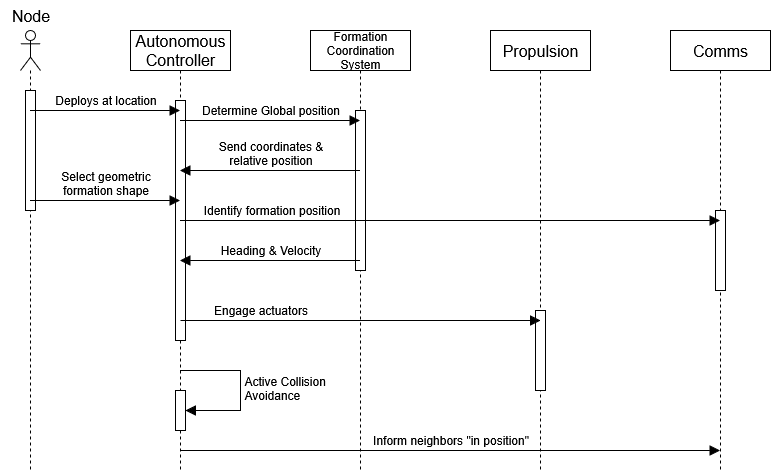
APPENDIX C

MBSE Structural Diagram of Limited Capability Autonomous Node



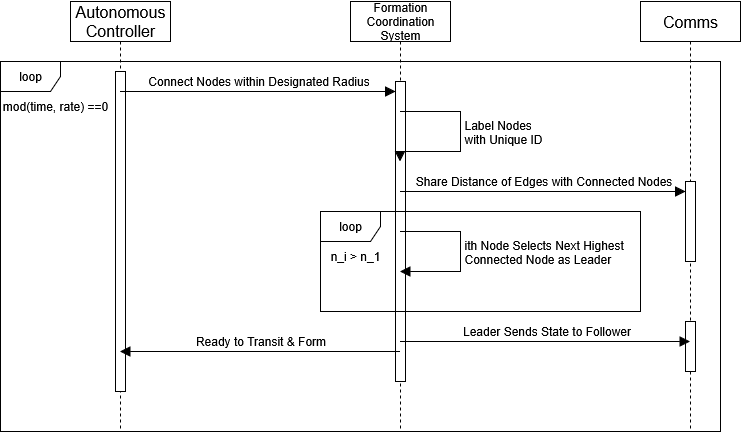
APPENDIX D

Deployment Sequence Diagram



APPENDIX E

Leader Selection Sequence Diagram



APPENDIX F

Reformation Sequence Diagram

