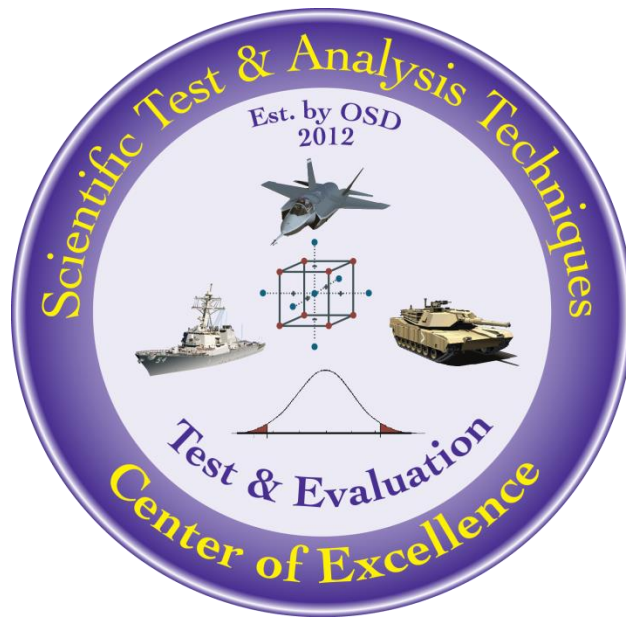


# FAMUS Planning Case Study

---



*Cory Natoli  
Gina Sigler  
November 2018*

**The goal of the STAT COE is to assist in developing rigorous, defensible test strategies to more effectively quantify and characterize system performance and provide information that reduces risk. This and other COE products are available at [www.afit.edu/STAT](http://www.afit.edu/STAT).**

DISTRIBUTION STATEMENT A. Approved for public release; distribution unlimited.  
Case number: 88ABW-2023-0744; CLEARED 03 Oct 2023

## Table of Contents

Executive Summary .....	2
Background .....	2
Project Scoping .....	2
Behavior Algorithms .....	3
Vector Steering .....	4
Station Keeping.....	5
Time on Target.....	5
Ground Engagement.....	6
Test Strategy .....	7
Conclusions .....	7
Appendix A. FAMUS Behavior Breakdown Document .....	8
Station Keeping.....	8
Time to Target.....	9
Ground Engagement.....	9

## **Executive Summary**

The STAT COE has been heavily involved in scoping the test strategy for Flexible Assured Manned-Unmanned Systems (FAMUS). Problem statements and goals have been defined for each testing event for the entire FAMUS program. A collaboration between the STAT COE and FAMUS group leads led to a more comprehensive, sequential test strategy. More detailed work in behavior algorithm testing led to objectives, responses, and factors being identified for each behavior type as a result of mission decomposition. 14 test designs have been created in an effort to support behavior algorithm testing. Both descriptive and inferential analysis has been produced by the STAT COE and provided to FAMUS test leads to assist in decision making. The purpose of this document is to summarize the work provided by the STAT COE in support of AFRL/RQQ.

## **Background**

The STAT COE was introduced to the FAMUS program with a kickoff meeting detailing the goals of the program. Several different stages of the test process existed and would benefit from STAT COE support. The test process includes software development, basic algorithm testing, flight sim testing, small UAS flight testing, and full scale and/or surrogate testing. The testing naturally lends to a sequential testing approach as all testing can map to a future or past test. Utilizing the information learned in early phases of testing will help narrow the scope of the more costly full scale testing. Also, the full scale testing will be used to validate behaviors in the simulation environment. The STAT COE was tasked with providing support for the behavior algorithm testing and scoping the overall test strategy.

## **Project Scoping**

The initial scope had laid out a series of tests that would be conducted, but did not establish goals for each test. The test schedule included simulation testing, a small UAS test, a pilot familiarization exercise, a surrogate risk reduction simulation, and a surrogate test during Phase 1 of testing. In collaboration with test leads and SMEs, the STAT COE held discussions on how each test was connected and what information needed to be gained. These discussions led to the development of goals for each phase of testing. Table 1 shows the defined goals for each test.

**Table 1: Goals for each test**

Surrogate Flight Testing	Small UAS Testing	Surrogate Risk Reduction	Computer Simulation
<ul style="list-style-type: none"><li>• Verify simulation assumptions</li><li>• Characterize performance of pilot interface</li><li>• Characterize the algorithm performance</li></ul>	<ul style="list-style-type: none"><li>• Characterize the algorithm performance</li><li>• Feedback to inform simulation and algorithm development</li></ul>	<ul style="list-style-type: none"><li>• Characterize performance of pilot interface</li><li>• Feedback on integrated system performance</li></ul>	<ul style="list-style-type: none"><li>• Characterize the algorithm performance</li></ul>

Understanding the goals of each test will inform what data needs to be collected. Additionally, the overlapping of objectives allows for a sequential testing approach that involves heavily testing in simulation environments and narrowing the scope based on the simulation results for the more costly surrogate phase. This sequential approach will allow for a more efficient overall test strategy and highlight any problems or concerns early on in the program. The early information gained here can have a big impact as correcting mistakes is much easier early in development. Initial testing was, and will continue to be, conducted on behavior algorithms to understand the performance of different algorithms and to inform future testing.

## Behavior Algorithms

A behavior algorithm can be tested through the use of a simulation in which an autonomous system is given a command and follows the command. A number of different commands can be given and each causes a different behavior in the autonomous system. The goal of these tests is to characterize algorithm performance. Each behavior algorithm has a different set of factors (or inputs into the simulation) and responses (outputs) that need to be measured in order to determine the performance of the autonomous system when given a command. Several different behaviors needed to be tested.

## Vector Steering

The vector steering command is the simplest behavior for the autonomous systems. The autonomous system is given an altitude, speed, and heading to achieve. Once it has achieved those parameters, it should maintain speed, altitude, and heading until another command is given. The objective of testing the vector steering behavior algorithm was to determine reasonable speeds, altitudes, and headings for the autonomous system. In addition to determining the settings the system could handle, a baseline time at which it takes to achieve a certain speed, altitude, and heading was desired. In order to determine when these objectives were met, the time to speed, altitude, heading, and the time to achieve all three were measured as response variables. The original design received by the STAT COE was a 567-run test matrix. Each run would last 10 minutes and would last for two intervals (each response recorded twice). The design focused on all possible combinations of variables, similar to a full factorial design.

This design provided valuable data, but it was not the most efficient possible design. The simulation environment is deterministic, meaning the response remains constant for the same input variables, which prompted the use of space filling designs. The STAT COE provided a 60-run example space filling design with one interval. Doing only one interval reduced the time it takes to run each simulation to 5 minutes. However, the 567-run design had already been executed and had been given to the STAT COE for data management and analysis. This served as a chance to expedite the data handoff process. An efficient method for sharing data was implemented as a result of this first dataset.

Additional designs were created based on the analysis conducted by STAT COE. Empirical models were built in JMP (statistical software) on each response. The models were used to make predictions for performances when entering values for each factor.

After the in-depth analysis results for vector steering behavior algorithm were presented, a larger FAMUS group meeting was held with team leads, systems experts, and the STAT COE to determine the objectives, responses, and factors for additional behavior algorithms. Initially the group discussed the remaining behavior algorithms to test, and came up with a list of 10 algorithms:

1. Way point following
2. Way point steering
3. Airspace constrained navigation
4. Single Ground engagement
5. Multi-ground engagement
6. Time on target
7. Rendezvous

8. Combat air patrol
9. Synchronized combat air patrol
10. Formation following/Station keeping

However, discussion led to identifying four algorithms that need to be tested. The four chosen algorithms were Time on Target, Station Keeping/Formation Following, Multi Ground Engagement, and Synchronized combat air patrol. Together these algorithms would use all of the other function algorithms that were not specifically being tested. Using a top down approach, the top level functions can be tested to see if they work as a whole. Then, if issues are found, the behaviors can be broken down into alternative functions. This limits testing if the behaviors are found to be working correctly. More details on all the behaviors can be found in the appendix.

## Station Keeping

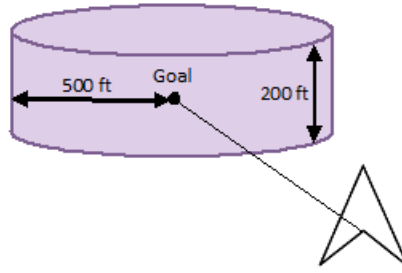
The station keeping command (also called formation following) gives a command to the lead aircraft and the autonomous system is expected to maintain a specified formation in relation to the lead aircraft. The autonomous system will behave differently depending on the lead aircrafts behavior, so several different behaviors need to be tested. However, the objectives for each behavior remain constant. The working group decided on four objectives:

1. How closely does the follow aircraft maintain position relative to the lead aircraft?
2. What percent of the time does the follow aircraft stay within so many feet of the assigned waypoint?
3. Do inputs exceed the ability of the aircraft?
4. What mode is the follow aircraft in?

Responses to answer the objectives included maximum distance of follow from waypoint, root mean square error (RMSE) of distance of follow from waypoint, average distance of follow from waypoint, minimum distance of follow from lead, percentage of time follow is within feet of waypoint, rate of change of direction of speed, rate of change of direction of bank angle, and reported follow mode.

## Time on Target

The time on target algorithm gives a time and target command to the autonomous system. The autonomous system should arrive at the target at the given time for a run to be a “success.” For the time on target algorithm, the group only defined one objective: Does the aircraft reach the appropriate point at the appropriate time? The responses used to answer the objective were defined to be time until possible arrival in seconds, distance of closest point of approach, and percent of runs that did not achieve the goal. The group also set boundaries in the shape of a puck around the goal point to define “reaching” the goal point. Figure 23 shows the tolerance boundaries of the puck.



**Figure 23: Time on target tolerance puck around goal point**

An aircraft could be within a 500 foot radius or at an altitude of  $\pm 100$  feet and still be considered to have reached the goal point. Factors were again chosen based on possible inputs to the behavior algorithm.

### Ground Engagement

After initial simulations for the time on target and station keeping algorithms was completed, it was decided that the ground engagement function instead of the multi ground engagement function would be tested for simplicity. Removing “multi,” in this case, meant that only one aircraft at a time would be used for the simulation; the number of targets can still vary in the ground engagement algorithm and will be accounted for as a factor in the test. For the ground engagement behavior algorithm, the group decided on four objectives.

1. Does the aircraft approach from the proper angle?
2. Does the aircraft approach from the proper distance?
3. Does the aircraft reach all targets within a given tolerance range?
4. Does the aircraft maintain a given speed and altitude?

Responses to answer the objectives included angle offset of the aircraft at waypoint approach, percent of runs within some degrees at way point approach, minimum distance of aircraft from each waypoint approach, number of targets reached, and distance of aircraft from tolerance range. The tolerance range was defined to be the same as the time on target. An aircraft could be within a 500 foot radius or at an altitude of  $\pm 100$  feet and still be considered to have reached the goal point.

## **Test Strategy**

The STAT COE has been tasked to help develop an overarching test program that includes the different testing phases of basic algorithm testing, flight simulation testing, small UAS testing, and surrogate flight testing. This provides an opportunity to integrate all of the different test teams with STAT COE support to better define constraints and goals at each phase of testing. The STAT COE has advocated for a sequential testing approach for the entirety of the FAMUS program in order to leverage the information learned at each test event. The STAT COE will continue to provide input on the overall test strategy for the next phase of FAMUS testing.

## **Conclusions**

STAT COE involvement with the FAMUS program has been heavily focused on project scoping and behavior algorithm testing. The STAT COE has helped clearly define goals and objectives that have led to a more rigorous test strategy. The designs and analysis provided by the STAT COE have provided a great deal of insight into the capabilities of the behavior algorithms that have been tested. Also, several issues have been identified and corrections made due to this exhaustive test strategy. These results will extend throughout all testing and can be used to save resources in future testing events. The sequential testing approach being employed now will be a more efficient use of testing resources while also maintaining effectiveness.

## Appendix A. FAMUS Behavior Breakdown Document

*Constructive Sim Goal: Characterize algorithm performance.*

### Station Keeping

---

*Objectives: How closely does the follow maintain position relative to the lead?*

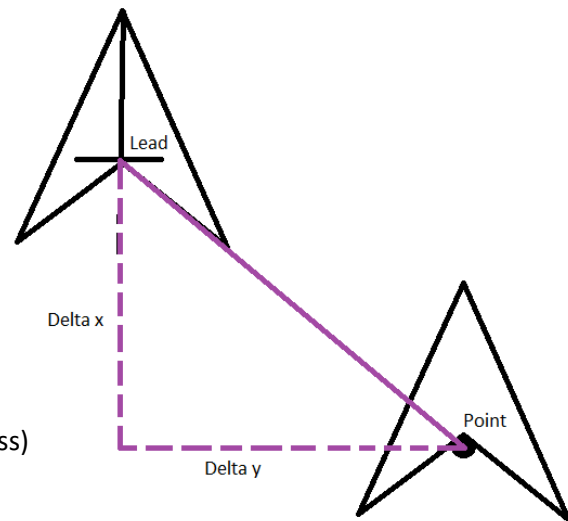
*What percent of the time does the follow stay within so many feet of the point?*

*Do inputs exceed ability of the aircraft?*

*What mode is the follow in?*

#### Responses:

- Maximum distance from point
- RMSE from point
- Average distance from point
- Minimum distance from lead
- % of time within **feet** of point
- Changes in directions of speed (smoothness)
- Changes in directions of bank angle (smoothness)
- Mode (rejoin or station keeping)



#### Factors:

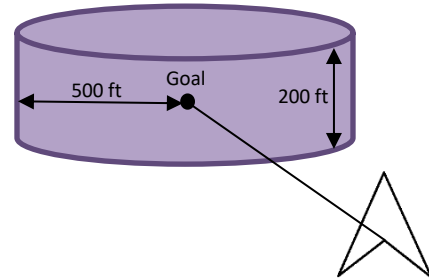
- Range offset – continuous variable
- Angle offset – continuous variable
- Altitude offset – continuous variable
- Starting altitude – continuous variable
- Formation type – categorical variable
- Starting speed – hold constant
- Lead aircraft behavior
  - Straight and level with speed change – continuous variable
  - Ascending/descending altitude change – discrete variable
  - Ascending and descending pitch angle change – discrete variable
  - Turning bank angle – discrete variable
  - Roll-reversal bank angle – discrete variable

## Time to Target

*Objective: Do we reach the appropriate point at the appropriate time?*

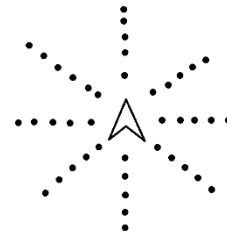
### Responses:

- Time till arrival  $\pm$  seconds
- Closest point of approach
- % not achieved



### Factors:

- Distance from target – continuous variable
- Heading to turn towards target – discrete variable
- Altitude change of target – continuous variable
- Wind speed – discrete variable
- Starting speed - hold constant
- Starting altitude – hold constant
- Starting heading – hold constant
- Goal time – hold constant
- Wind direction – hold constant



## Ground Engagement

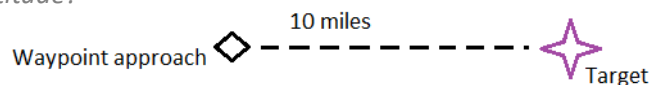
*Objectives: Do we approach from a proper angle?*

*Do we approach from a certain distance?*

*Do we reach all targets within the given tolerance range?*

*Do we maintain/keep a given speed and altitude?*

### Responses:



- Angle offset at way point approach
- % of runs within so many degrees at way point approach
- Minimum distance away from each waypoint approach
- Number of targets reached
- Distance from tolerance ranges (hockey puck)

*Factors:*

- # of aircraft – hold constant
- # of targets – discrete variable
- Distance from start to waypoint 1– continuous variable
- Starting position off of waypoint 1 – continuous variable
- Starting heading – discrete variable
- Distance from target 1 to waypoint 2– continuous variable
- Position of waypoint 2 from target 1 – continuous variable
- Starting altitude – hold constant
- Starting speed – hold constant
- Angle of target 1 to waypoint 1 – hold constant
- Distance of any waypoint to associated target – hold constant
- Angle of target 2 to waypoint 2 – hold constant

