**Overview / Goals**

The primary purpose of the simulations in this section are to determine safe distances to begin maneuvering in order to avoid violations. These tests are parameterized to explore the various ways in which the two vehicles may encounter one another during the physical test, accounting for the variability in their behavior such as speed, displacement, and turn rate.

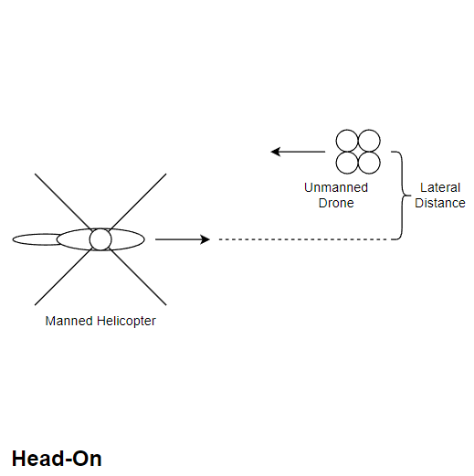
The primary goals of this testing battery is to identify A) The minimum distance necessary to avoid a violation, given the ideal physical test’s parameters, B) The worst-case scenario according to the variability of the physical test’s parameters, and C) The minimum distance necessary to avoid a violation for the worst-case scenario.

These distances will be referred to as “response distances,” i.e. the distance between crafts where a response (maneuver) must be made.

**2.3 ERAU Sim**

Agents.jl is a Julia-based framework designed for agent-based modeling (ABM), a simulation methodology where autonomous agents respond to their environment based on predefined rules. Noteworthy features of Agents.jl include its remarkable speed, surpassing that of MASON, NetLogo, or Mesa. The framework is distinguished by its simplicity, boasting a short learning curve and minimal code requirements. With an extensive interface offering thousands of out-of-the-box agent actions, Agents.jl seamlessly facilitates simulations on Open Street Maps. Embry-Riddle Aeronautical University (ERAU) has been using this platform in scenario-based validation testing of decision-making algorithms for UAS.

**Simulation Methodology**

The simulations were built using Agents.jl which is a library for the Julia programming language. This simulation software is a agent-based simulator where each agent (Vehicle for this use case) is given behavior which allows it to update over time. Each agent then interacts with one another using this behavior. In These simulations are composed of a Standard Agent Based Model (StandardABM) using a continuous three-dimensional space. The agents are placed within the space according to the scenario and its parameters.

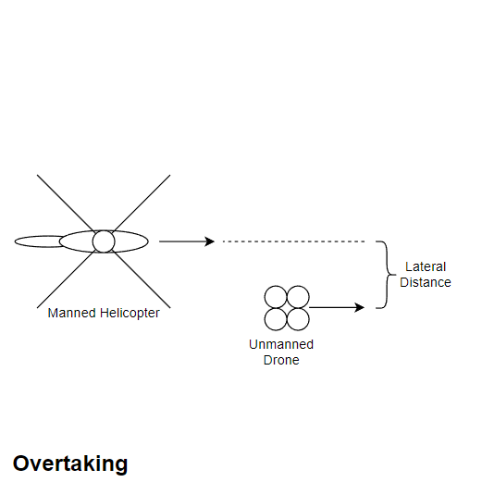
A diagram of a helicopter

Description automatically generatedThere are three scenarios, the first is the Head-On Scenario, illustrated in Figure 1, where the helicopter and drone start on opposite sides of the space and face one another. During this scenario, there are three fundamental parameters that can vary: Drone Speed, Helicopter Speed, and Lateral Separation Distance. The lateral separation distance is lateral distance between the helicopter and the drone.

Figure : Head-On Scenario Layout

The second scenario is the Converging Scenario, illustrated in X, where the drone is positioned to the left of the Manned Helicopter where its heading is clockwise from the helicopter’s heading, leading to a intersection between the vehicles. There are three parameters for this scenario: Drone Speed, Helicopter Speed, and Longitudinal Separation Distance. The longitudinal separation distance is the measure of how far the drone is ahead of the helicopter.

Figure : Converging Scenario Layout

The third and final scenario is the Overtaking Scenario, illustrated in X, where the drone is positioned in front of the helicopter with the same heading. Due to the drone always being slower than the helicopter, this leads to the helicopter intercepting the drone. The parameters that can vary are: Drone Speed, Helicopter Speed, and Longitudinal Separation Distance.

These scenarios have variations which define how the drone responds to the scenario. Specifically, these variations are Horizontal and Vertical variations. The horizontal variation defines a right-hand-turn response for the drone, where the drone only rotates horizontally to avoid the violation. This variation adds a Drone Turn Rate parameter to the scenario. The vertical variation defines a descent response for the drone, where the drone only descends to avoid the violation. This variation adds a Drone Descent Rate to the scenario’s parameters.

Figure : Overtaking Scenario Layout

**Parameters**

The vehicle parameters were varied around the target parameters of the physical test according to expert experience. How they were varied is summarized in Table X.

|  |  |  |
| --- | --- | --- |
| Parameter | Value | Variability |
| Drone Speed | 55fps | +/- 15fps |
| Helicopter Speed | 169fps | +/- 15fps |
| Drone Turn Rate | 13 deg/sec | +/- 7 deg/sec |
| Drone Descent Rate | 17fps | +/- 9fps |

Agents.jl is a Julia-based framework designed for agent-based modeling (ABM), a simulation methodology where autonomous agents respond to their environment based on predefined rules. Noteworthy features of Agents.jl include its remarkable speed, surpassing that of MASON, NetLogo, or Mesa. The framework is distinguished by its simplicity, boasting a short learning curve and minimal code requirements. With an extensive interface offering thousands of out-of-the-box agent actions, Agents.jl seamlessly facilitates simulations on Open Street Maps.

**Results**

For each scenario,

**Discussion**