Cooperative Threat Engagement with Heterogeneous Drone Swarms

Luiz Giacomossi Jr.
AB Volvo
12 PUBLICATIONS **44** CITATIONS
SEE PROFILE

Jorge Antonio Ricardo Jr
Embraer
32 PUBLICATIONS **107** CITATIONS
SEE PROFILE

J.F.B. Brancalion
Embraer
16 PUBLICATIONS **83** CITATIONS
SEE PROFILE

Marcos Maximo
Aeronautics Institute of Technology
137 PUBLICATIONS **802** CITATIONS
SEE PROFILE

33RD CONGRESS
OF THE INTERNATIONAL COUNCIL
OF THE AERONAUTICAL SCIENCES
STOCKHOLM, SWEDEN, 4-9 SEPTEMBER, 2022

ICAS
2022
SWEDEN

COOPERATIVE THREAT ENGAGEMENT WITH HETEROGENEOUS DRONE SWARMS

Luiz Giacomossi[1*], Jorge A. Ricardo Jr[2*], José F. B. Brancalion[3], Marcos R. O. A. Maximo[1] & Davi A. Santos[2]

[1] Autonomous Computational Systems Lab (LAB-SCA), Computer Science Division, Aeronautics Institute of Technology (ITA), São José dos Campos - SP, Brazil
[2] Mechatronics Department, Mechanical Engineering Division, Aeronautics Institute of Technology (ITA), São José dos Campos - SP, Brazil
[3] EMBRAER S.A., São José dos Campos - SP, Brazil
[*] First authors

Abstract

The ability of multiple manned and unmanned aircraft systems to cooperatively engage and disable an aerial threat plays a decisive role in modern warfare scenarios. In this paper, we apply key methods to enable the so-called cooperative threat engagement capability among multiple networked agents, *e.g.*, a swarm of drones, with combat and communication capabilities. In particular, this research combines AI-based decision-making and control techniques for a swarm of loyal wingman drones to coordinate efficient defense actions in cooperative and autonomous manner. We apply these concepts in a defense scenario, modeled to analyze the loyal wingman concept, which we consider an interesting testbed for cooperative decision-making as well as low-level control techniques. The methodologies were merged with the creation of a 3D UAV simulator to provide an application and evaluation of behavior strategies and control methods.

Keywords: Cooperative engagement capability; manned-unmanned teaming; loyal wingman UAV; decision-making; sliding mode control.

1 Introduction

The cooperative engagement capability (CEC) is an emerging systems-of-systems capability in which multiple systems coordinate their actions through a network to improve their abilities to effectively perform a given task [1]. The term was coined in the 90s in the Johns Hopkins University Applied Physics Laboratory in an effort to enhance the United States Navy's defense capability. The project aimed to improve the situational awareness of a fleet by sharing tactical information among data link participants and speeding up the decision process regarding the interception, engaging, and neutralization of a flying threat, *e.g.*, a cruising missile or aircraft. With the recent advances in aerial robotics, the CEC concept can be extended to unmanned flying platforms to promote group behavior towards the achievement of a common purpose via a shared communication network. In this context, drone swarms are of particular interest since they enable the use of several low-cost and even disposable platforms in modern defense or surveillance applications.

In particular, manned-unmanned teaming is an essential component of the future unmanned aircraft systems' operational environment. It extends the CEC concept by promoting the seamless integration of unmanned platforms, endowed with some level of autonomy, with manned ones. The unmanned platforms aim to perform given tasks under the command of the manned ones. This integration combines the inherent strengths of both platforms to achieve synergy not seen in single-agent systems, thus enabling a higher degree of effectiveness for the team of networked agents. Specifically, the MUM-T context includes the concept of the loyal wingman (LW), which is an unmanned aerial vehicle (UAV) under the tactical command of a high-level manned or remotely controlled leader [2]. Recently, we have witnessed a rapid emergence of multirotor aerial vehicles (MAVs) [3] in either research and industrial applications where aircraft size, weight, and cost play a critical role in their operation, effectiveness, or feasibility [4]. Therefore, we argue that low-cost experimental MAVs are a suitable alternative to investigate the loyal wingman concept in MUM-T scenarios.

This paper studies the use of loyal-wingman MAVs with combat capabilities to cooperatively engage and disable aerial explosive threats. In particular, we highlight that this work is an extension of a previous work [5], which evaluated a 2D defensive scenario where a MUM-T composed of loyal wingmen protects a leader UAV and a critical infrastructure from aerial threats. Here, we study a more complex version of the same scenario, considering a 3D environment and more realistic flight dynamics. In this paper, the vehicles are modeled as fully actuated MAVs, which can perform independent attitude and position tracking [6]. To tackle the design problems that arise from such context, we proposed a problem breakdown that focuses on both the high-level decision-making tasks and the low-level control aspects to enable the CEC among the leader and the loyal wingmen in a defense scenario. The low-level control addresses the well-recognized robustness requirement for MAV flight control systems against disturbances and uncertainties by designing a joint geometric attitude-position control law using a multi-input first-order sliding mode strategy. On the other hand, the autonomous behavior of the MAVs is designed using state-of-art artificial intelligence (AI) techniques, by applying automated finite-state machines (FSMs) and behavior trees (BTs) decision-making algorithms to select and execute modular behaviors for the loyal wingman and the aerial threats. To the best of our knowledge the main contributions of this paper are 1) an extension of the defensive scenario proposed in [5] considering a three-dimensional environment with realistic flight dynamics and 2) the application of the multi-input sliding mode flight controller and AI-base decision-making techniques with FSMs and BTs into the proposed scenario.

The remaining of this paper is organized as follows. Subsection 1.1 presents the notation used along the paper. Section 2 presents the scenario of interest. Section 3 defines the problem. Section 4 presents the methodology. Section 5 shows the computational architecture used to simulate the method. Section 6 evaluates the proposed method using computer simulations. Finally, Section 7 concludes the paper.

1.1 Notation

Matrices and algebraic vectors are denoted, respectively, by uppercase and lowercase boldface letters, while geometric (Euclidian) vectors are denoted as $\vec{a}$. A Cartesian coordinate system (CCS) is represented as $\mathcal{S}_b \triangleq \{B;\vec{x}_b,\vec{y}_b,\vec{z}_b\}$, with B denoting its origin, and $\vec{x}_b$, $\vec{y}_b$, and $\vec{z}_b$ representing the unit geometric vectors along its orthogonal axes. The algebraic vectors corresponding to the projection of an arbitrary physical vector $\vec{a}$ onto $\mathcal{S}_b$ and $\mathcal{S}_g$ are denoted by $\mathbf{a}_b \in \mathbb{R}^3$ and $\mathbf{a}_g \in \mathbb{R}^3$, respectively. The relation between $\mathbf{a}_g$ and $\mathbf{a}_b$ is $\mathbf{a}_b = \mathbf{D}^{b/g}\mathbf{a}_g$, where $\mathbf{D}^{b/g} \in \mathrm{SO}(3)$ is the attitude matrix of $\mathcal{S}_b$ relative to $\mathcal{S}_g$ and $\mathrm{SO}(3)$ represents the special orthogonal group. The inverse of $\mathbf{D}^{b/g}$ is equal to its transpose, which is denoted by $\mathbf{D}^{g/b}$. The identity matrix of order $n \in \mathbb{Z}_{>0}$ is denoted by $\mathbf{I}_n$. Consider two arbitrary algebraic vectors $\mathbf{x} = (x_1,\ldots,x_n)$ and $\mathbf{y} = (y_1,\ldots,y_n)$. The vector inequality $\mathbf{x} < \mathbf{y}$ means that $x_i < y_i$, $\forall i \in \{1,\ldots,n\}$. The standard basis vectors of $\mathbb{R}^3$ are denoted by $\mathbf{e}_1 \triangleq (1,0,0)$, $\mathbf{e}_2 \triangleq (0,1,0)$, and $\mathbf{e}_3 \triangleq (0,0,1)$. Let $\vec{a}^{\,b/g}$ represent an arbitrary physical quantity of $\mathcal{S}_b$ with respect to $\mathcal{S}_g$; *e.g.*, along the paper, $\vec{v}^{\,b/g}$ will denote the velocity of $\mathcal{S}_b$ relative to $\mathcal{S}_g$. Finally, consider the $\mathcal{S}_g$ representations $\mathbf{a}_g \triangleq (a_1,a_2,a_3)$ and $\mathbf{b}_g$ of $\vec{a}$ and $\vec{b}$, respectively. The vector product $\vec{c} = \vec{a} \times \vec{b}$ is represented in $\mathcal{S}_g$ by $\mathbf{c}_g = [\mathbf{a}_g\times]\,\mathbf{b}_g$, where $[\mathbf{a}_g\times]$ is the following skew-symmetric matrix:

$$[\mathbf{a}_g\times] \triangleq \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}.$$

2 Scenario of Interest

To address the concept of CEC among the loyal wingmen and the leader, we define a defense scenario where there are two teams: a MUM-T composed of loyal wingman MAVs that fly in formation alongside, *i.e.*, escorting the leader, and must defend him as well as a protected area from attacks of an adversarial team, composed by a swarm of kamikaze threats, as depicted in Fig. 1. Therefore, the MUM-T mission is considered to fail if the leader is hit by a single kamikaze, or the protected area is hit by a fixed number of kamikazes. The protected area is equipped with ground assets that provide aerial surveillance capability to the MUM-T members.

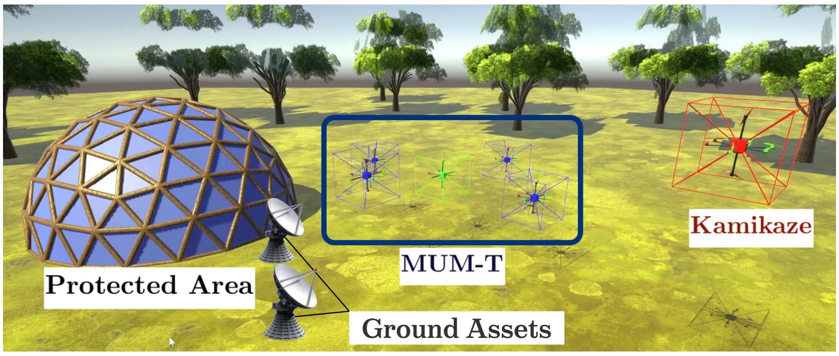

Figure 1 – Scenario of interest where loyal wingman MAVs, highlighted in blue, supported by ground assets, escort a leader MAV and engage kamikaze threats in order to defend a protected area.

The LW is autonomous and capable of making intelligent decisions based on the situational awareness information provided by the ground assets considering they do not have onboard sensors to obtain information about the threats, such as cameras or LIDAR. To be able to neutralize the threats, the LW is equipped with two hypothetical types of weapons, a mid-range *freezing gun* and a short-range *vaporizer gun*, both with a limited number of cartridges and a fixed cool-down time interval, *i.e.*, they become unavailable for a fixed time period after shooting. The *vaporizer gun* can neutralize the threat, while the *freezing gun* slows down the threat by half of its maximum speed. The weapons' model is simplified, being the hitting success calculated by a given probability. Note that, the *freezing gun* is intended to make the decision space more complex, in practice, this type of weapon is infeasible given current technology, and the *vaporizer gun* is also an idealized weapon that uses energy to destroy only the electronic components of the kamikaze.

The leader is remotely controlled by a human and it is in charge of the formation coordination, *i.e.*, it is capable of passing relative coordinate commands to the LW. As a matter of simplification, we assume that the leader can only command the loyal wingmen to fly within one predefined formation pattern. In this paper, we consider this unique pattern as a uniform-circular formation along the local horizontal plane with a desired radius. We also assume that the leader is always capable to command the loyal wingmen whenever required, *i.e*, it can override the decision-making commands of a specific LW in the MUM-T, either by enforcing the loyal wingmen to return to the formation, or selecting an LW to engage or shoot a specific threat.

On the other hand, the kamikaze MAVs are assumed to explode as soon as they collide with either the leader, an LW, or the protected area. When the explosion is triggered, its effect destroys the target vehicle or damages the protected area. Additionally, once a kamikaze selects a target, it will chase this target until the kamikaze explodes itself or is hit by a vaporizer gun. The number of kamikazes is fixed in the simulated scenario, they immediately respawn after being neutralized, keeping a steady stream of attacks. The kamikazes have a simplified AI but are assumed to be faster and more numerous than the MUM-T, thus forcing the MUM-T to cooperatively work in order to effectively neutralize the kamikazes.

The ground assets have a limited detection range and share situational awareness with the MUM-T, *i.e.*, a vector with the state (position, attitude, and linear velocity) of all entities. To simplify the scope of this work, we assume the ground assets to be perfect radars, and that the communication among the MUM-T is perfect, with no delays, packet loss, or bandwidth limitation.

3 Problem Breakdown

In this section, we introduce a problem breakdown according to the scenario of interest. A common approach in robotic cognition involves decomposing the agent into software layers, such as world modeling, decision-making, and control. Figure 2 shows a representation of the software layers present at the ith MAV. In this paper, we use a similar breakdown and focus on both the high-level decision-making tasks and the low-level control aspects, abstracting the world modeling layer represented by the blue block in Figure 2.

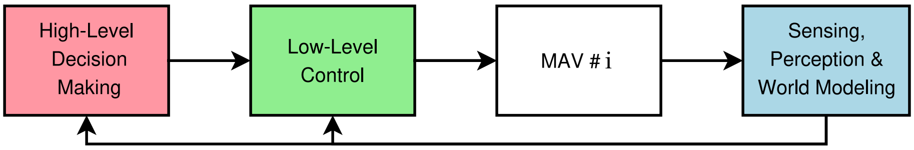
High-Level Decision Making
Low-Level Control
MAV # i
Sensing, Perception & World Modeling

Figure 2 – Problem breakdown indicating the main software layers at the ith MAV.

The high-level decision-making layer receives state information from the world model containing the pose and linear velocity of each object of interest in a global coordinate system. Once this information is received, the high-level layer process the states using AI-based decision-making algorithms that outputs action commands. There are two types of action commands: rotational and translational movement commands and commands regarding the use of weapons. The low-level control, on one hand, translates the movement actions into appropriate state commands that are used to calculate the control input to the MAV, and, on the other hand, allocates the weapon actions using the available resources. Moreover, the MAV module contains the rotational and translational dynamics of the vehicle, the control allocator that translates the control input into actuator commands, the actuator model, and the weapons model. Note that, the weapons' model is simplified, *i.e.*, the aiming and projectile dynamics are abstracted, being the hitting success calculated by a configurable probability. Lastly, we also assume that the MAVs share an internal model of the world in the sense that they are aware of the same threats and other agents state due to the ground-assets radars.

3.1 Low-level Control Problem

This subsection defines the problem of the low-level control layer. To this end, we briefly present the rotational and translational dynamic modeling of a general fully actuated MAV with fixed rotors and define the commanded variables. For a more detailed dynamic modeling please see [7–10].

Consider a ground reference CCS $\mathcal{S}_g \triangleq \{G;\vec{x}_g,\vec{y}_g,\vec{z}_g\}$ located at a known point G on the ground, with $\vec{z}_g$ oriented upwards, parallel to the local vertical. In addition, considering that in the proposed scenario there is a total of N MAVs, we can define a body-fixed CCS $\mathcal{S}_i \triangleq \{B_i;\vec{x}_i,\vec{y}_i,\vec{z}_i\}$ tied to the ith MAV airframe, with B_i coinciding with its center of mass, as depicted in Figure 3.

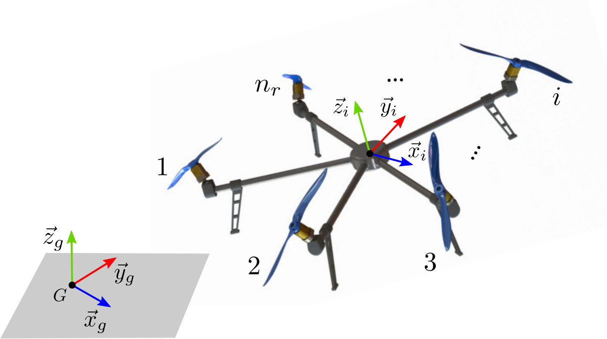

Figure 3 – Schematic representation of a general fully actuated MAV with $n_r \in \mathbb{Z}_{>0}$ fixed rotors along with the used CCSs.

The translational kinematics and dynamics of $\mathcal{S}_i$ w.r.t. $\mathcal{S}_g$ are, respectively, described by

$$\dot{\mathbf{r}}_g^{i/g} = \mathbf{v}_g^{i/g}, \tag{1}$$

$$\dot{\mathbf{v}}_g^{i/g} = -g\mathbf{e}_3 + \frac{1}{m_i}\mathbf{f}_g^{c,i} + \frac{1}{m_i}\mathbf{f}_g^{d,i}, \tag{2}$$

where $\mathbf{r}_g^{i/g} \in \mathbb{R}^3$, $\mathbf{v}_g^{i/g} \in \mathbb{R}^3$, and $m_i \in \mathbb{R}$ are, respectively, the ith MAV position, velocity, and mass; $g \in \mathbb{R}$ is the gravity acceleration magnitude; and $\mathbf{f}_g^{c,i} \in \mathbb{R}^3$ and $\mathbf{f}_g^{d,i} \in \mathbb{R}^3$ are, respectively, the control and disturbance forces of the ith MAV.

On the other hand, the rotational kinematics and dynamics of $\mathcal{S}_i$ w.r.t. $\mathcal{S}_g$ are, respectively, represented by [11]

$$\dot{\mathbf{D}}^{i/g} = -\left[\omega_i^{i/g}\times\right]\mathbf{D}^{i/g}, \tag{3}$$

$$\dot{\omega}_i^{i/g} = (\mathbf{J}_i)^{-1}\left[\mathbf{J}_i\omega_i^{i/g}\times\right]\omega_i^{i/g} + (\mathbf{J}_i)^{-1}\left(\tau_i^{c,i} + \tau_i^{d,i}\right), \tag{4}$$

where $\mathbf{D}^{i/g} \in \mathrm{SO}(3)$ is the attitude, $\omega_i^{i/g} \in \mathbb{R}^3$ is the angular velocity, $\mathbf{J}_i \in \mathbb{R}^{3\times 3}$ is the inertia tensor calculated w.r.t. $\mathcal{S}_i$, $\tau_i^{c,i} \in \mathbb{R}^3$ is the control torque, and $\tau_i^{d,i} \in \mathbb{R}^3$ is the disturbance torque acting on the ith MAV.

Let us denote a command with an overbar in the corresponding variable. Particularly $\bar{\mathbf{r}}_g^{i/g}$ and $\bar{\mathbf{D}}^{i/g}$ denote, respectively, the position and attitude commands.

The main problem of the low-level control is now defined.

Problem 1. The low-level control problem is to design a joint attitude-position control law to make the plant described by (1)– (4) robustly track the commands $\bar{\mathbf{r}}_g^{i/g}$ and $\bar{\mathbf{D}}^{i/g}$ with respect to the disturbance force and torque.

To tackle the above problem, we design the joint attitude-position control law using a first-order sliding mode approach (SMC) [12]. It uses a high-frequency switching control to drive the plant output to the so-called sliding manifold, where the plant ideally becomes insensitive to bounded disturbances of the matched type.

3.2 High-Level Decision-Making Problem

In this section, the problem to be solved by the decision-making of LW agents will be described.

The main objective of the LW agents' mission is to defend both the protected area and the remotely-controlled leading drone against multiple incursions from a swarm of kamikaze drones. Therefore, an LW must effectively engage and disable threats to ensure the safety of both. Furthermore, for this mission to be successful, LW MAVs are expected to have the ability to make intelligent decisions autonomously. Thus, we can define necessary capabilities for this agent, such as the ability to fly in formation with cohesion, and the ability to engage and neutralize imminent threats to guarantee the defense of both the protected area and the leader. To effectively achieve neutralization, agents must have the capability to use embedded weaponry, *i.e*, use weapons intelligently and strategically, and even in extreme cases, sacrifice themselves to ensure the mission's success.

The main problem in high-level decision-making can now be defined.

Problem 2. The problem is to develop an autonomous intelligence module for the LW MAV, in which we need to model basic behaviors and design a decision-making architecture to successfully achieve the mission objective, aiming at the smallest loss of LW agents during the maneuvers of attack and defense, thus ensuring the security of the protected area and the manned leader.

4 Methodology

This section presents the methods used for addressing Problems 1–2. Subsection 4.1 describes the method used to solve Problem 1. Subsection 4.2 describes the methods used to solve Problem 2.

4.1 Low-level Control Methodology

To design the controller, we first obtain the equations describing the error kinematics and dynamics. To this end, define a command-related CCS $\mathcal{S}_{\bar{i}} \triangleq \{\bar{B}_i; \vec{x}_{\bar{i}}, \vec{y}_{\bar{i}}, \vec{z}_{\bar{i}}\}$ representing the commanded position and orientation for the ith MAV body-fixed frame $\mathcal{S}_i$. The attitude and angular velocity control errors can be defined, respectively, as [11]

$$\tilde{\mathbf{D}}^i \triangleq \mathbf{D}^{i/\bar{i}} \equiv \mathbf{D}^{i/g} \left(\bar{\mathbf{D}}^{i/g}\right)^{\mathrm{T}} \in \mathrm{SO}(3), \tag{5}$$

$$\tilde{\omega}^i \triangleq \omega_i^{i/\bar{i}} \equiv \omega_i^{i/g} - \tilde{\mathbf{D}}^i \bar{\omega}_i^{i/g} \in \mathbb{R}^3, \tag{6}$$

where $\bar{\mathbf{D}}^{i/g} \triangleq \mathbf{D}^{\bar{i}/g}$ and $\bar{\omega}_i^{i/g} \triangleq \omega_{\bar{i}}^{\bar{i}/g}$ are, respectively, the ith MAV attitude and angular velocity commands.

Similarly, let us define the position and linear velocity errors, respectively, as

$$\tilde{\mathbf{r}}^i \triangleq \mathbf{r}_g^{i/g} - \bar{\mathbf{r}}_g^{i/g}, \tag{7}$$

$$\tilde{\mathbf{v}}^i \triangleq \mathbf{v}_g^{i/g} - \bar{\mathbf{v}}_g^{i/g}, \tag{8}$$

where $\bar{\mathbf{r}}_g^{i/g} \in \mathbb{R}^3$ and $\bar{\mathbf{v}}_g^{i/g} \in \mathbb{R}^3$ are, respectively, the position and velocity commands of the ith MAV.

The time derivatives of (5)–(8) can put into the following state-space model:

$$\begin{aligned} \dot{\mathbf{x}}_1^i &= \mathbf{f}_1^i\left(\mathbf{x}_1^i, \mathbf{x}_2^i\right), \\ \dot{\mathbf{x}}_2^i &= \mathbf{f}_2^i\left(\mathbf{x}_1^i, \mathbf{x}_2^i\right) + \mathbf{B}^i \mathbf{u}^i + \mathbf{B}^i \mathbf{d}^i, \end{aligned} \tag{9}$$

where $\mathbf{x}_1^i \triangleq (\tilde{\mathbf{r}}^i, \tilde{\mathbf{g}}^i)$, with $\tilde{\mathbf{g}}^i \in \mathbb{R}^3$ being the Gibbs vector representing $\tilde{\mathbf{D}}^{i/g}$, and $\mathbf{x}_2^i \triangleq (\tilde{\mathbf{v}}^i, \tilde{\omega}^i)$ are the states, $\mathbf{u}^i \triangleq (\mathbf{f}_g^{c,i}, \tau_i^{c,i})$, $\mathbf{d}^i \triangleq (\mathbf{f}_g^{d,i}, \tau_i^{d,i})$,

$$\begin{aligned} \mathbf{B}^i &\triangleq \begin{bmatrix} \mathbf{I}_3/m_i & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & (\mathbf{J}_i)^{-1} \end{bmatrix}, \\ \mathbf{f}_1^i\left(\mathbf{x}_1^i, \mathbf{x}_2^i\right) &\triangleq \begin{bmatrix} \tilde{\mathbf{v}}^i \\ \frac{1}{2}\left(\tilde{\mathbf{g}}^i \left(\tilde{\mathbf{g}}^i\right)^{\mathrm{T}} + [\tilde{\mathbf{g}}^i \times] + \mathbf{I}_3\right) \tilde{\omega}^i \end{bmatrix}, \\ \mathbf{f}_2^i\left(\mathbf{x}_1^i, \mathbf{x}_2^i\right) &\triangleq \begin{bmatrix} -\dot{\bar{\mathbf{v}}}_g^{i/g} - g\mathbf{e}_3 \\ (\mathbf{J}_i)^{-1}\left[\mathbf{J}_i \omega_i^{i/g} \times\right] \omega_i^{i/g} - \tilde{\mathbf{D}}^i \dot{\bar{\omega}}_i^{i/g} + \left[\tilde{\omega}^i \times\right] \tilde{\mathbf{D}}^i \bar{\omega}_i^{i/g} \end{bmatrix}. \end{aligned} \tag{10}$$

From (9), it can be immediately seen that the term $\mathbf{d}^i$ is of the matched type, *i.e.*, it belongs to the span of the input matrix $\mathbf{B}^i$. Consequently, a first-order sliding mode control (SMC) law is able to guarantee the robustness (or invariance) of the closed-loop system with respect to $\mathbf{d}^i$ [13].

Assume that the disturbance input $\mathbf{d}^i$ is unknown, but it is bounded according to $|\mathbf{d}^i| \leq \mathbf{d}^{i,\max}$, where $\mathbf{d}^{i,\max} \in \mathbb{R}^6$ is a vector with positive components.

Now, define the sliding variable $\mathbf{s} \in \mathbb{R}^6$ as

$$\mathbf{s}^i \triangleq \mathbf{C}^i \mathbf{x}_1^i + \mathbf{f}_1^i(\mathbf{x}_1^i, \mathbf{x}_2^i),$$

where $\mathbf{C}^i \in \mathbb{R}^{6\times 6}$ is a design parameter matrix.

Note that in the sliding manifold $\mathbf{s}^i = \mathbf{0}_6$, the system dynamics are reduced to $\dot{\mathbf{x}}_1^i = -\mathbf{C}\mathbf{x}_1^i$. Therefore, by choosing $\mathbf{C}^i$ as a positive-definite diagonal matrix, the point $(\mathbf{x}_1^i, \dot{\mathbf{x}}_1^i) = (\mathbf{0}_6, \mathbf{0}_6)$ is made asymptotically stable. Inspecting (10), we can see that $\mathbf{f}_1^i(\mathbf{0}_6,\mathbf{x}_2^i)=\mathbf{0}_6$ if $\mathbf{x}_2^i=\mathbf{0}_6$. Then, since $(\mathbf{x}_1^i,\dot{\mathbf{x}}_1^i)\to(\mathbf{0}_6,\mathbf{0}_6)$, it also holds that $\mathbf{x}_2^i\to\mathbf{0}_6$, and consequently the point $(\mathbf{x}_1^i,\mathbf{x}_2^i)=(\mathbf{0}_6,\mathbf{0}_6)$ is asymptotically stable. Therefore, the tracking objective of Problem 1 is accomplished by robustly keeping the system states on the sliding manifold $\mathbf{s}^i=\mathbf{0}_6$. To this end, we adopt a first-order SMC, which steers $\mathbf{s}^i$ to zero in finite time.

By choosing the control law

$$\mathbf{u}^i=-\left(\frac{\partial\mathbf{f}_1^i}{\partial\mathbf{x}_2^i}\mathbf{B}^i\right)^{-1}\left(\mathbf{C}\mathbf{f}_1^i+\frac{\partial\mathbf{f}_1^i}{\partial\mathbf{x}_1^i}\mathbf{f}_1^i+\frac{\partial\mathbf{f}_1^i}{\partial\mathbf{x}_2^i}\mathbf{f}_2^i+\mathbf{K}^i\text{sign}\left(\mathbf{s}^i\right)\right),$$

where $\mathbf{K}^i\in\mathbb{R}^{6\times6}$ is a constant diagonal matrix satisfying

$$\mathbf{K}\mathbf{1}_6=\mathbf{d}^{i,\max}+\varepsilon,$$

with $\varepsilon\in\mathbb{R}^6$ being a vector with positive components, $\mathbf{s}^i\to\mathbf{0}_6$ in finite time, thus completing the objective of Problem 1. The proof is omitted here but is analogous to the one presented in Section 3.1 of Reference [14].

4.2 High-level Decision-Making Methodology

In this section, we discuss the methodology used to address Problem 2. For that, we focus on the artificial intelligence of both types of agents present in the scenario of interest. The development of the decision-making architectures of the agents described is based on our previous work [5], where a prototype of the scenario was developed, in order to have an initial decision architecture of the LW and kamikaze agents.

Many approaches have been devised for robotic decision-making, in modern applications such as autonomous cars, the agents are highly deliberative [15]. In these complex scenarios, the decision-making is further broken down into modules tailored to solve sub-tasks, called behaviors [16]. Once this decomposition is performed, we need to orchestrate the execution of these behaviors. Many behavior selection mechanisms exist, with finite-state machines (FSMs) for decision-making [17] and behavior trees (BTs) [18] being the most popular techniques. The design of these mechanisms is empirical and is mainly based on intuition, creativity, experience, and good practices [17]. Furthermore, this process is iterative, with the agent performance being evaluated by an expert or through statistics, in order to select the best agent architecture [18]. On the other hand, optimization methods and machine learning techniques may be employed to tune decision parameters [5] or to learn complete behaviors with no prior knowledge [19].

To begin to address the solution of Problem 2, we need the kamikaze agents to be functional through simplified but effective AI. Consequently, we selected the finite state machine (FSM) technique to develop the decision-making module for the kamikaze. The FSMs are the most common model of computation for modeling decision-making architectures for simple AI [18, 20], due to its easy implementation and intuitive structure. In this technique, a state represents a behavior for the agent. An FSM can switch between behaviors in response to events. In Fig. 4, we present the decision-making developed for the kamikaze MAV, note that each MAV contains an identical decision module.

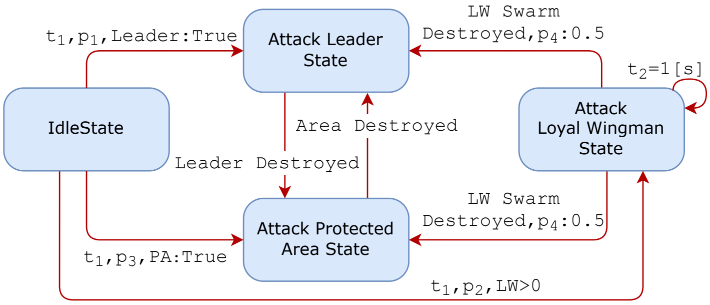

Figure 4 – Decision-Making module for a kamikaze MAV using FSM.

The kamikaze has four behaviors, the initial behavior is *IdleState*, where the agent is idle for t_1 seconds and then selects a target to attack based on the probabilities p_1, p_2 and p_3, in this work equally distributed probabilities were used for each target type. Once a target is selected, the FSM will transition to the representative state of the selected target and will remain in that state until the agent is destroyed. The exception is *Attack Loyal Wingman*, this behavior targets the closest LW, and will select the closest LW every $t_2 = 1\,[s]$, to avoid frequent target switching. Once there are no more LW, the FSM transitions to attack the leader or the protected area, with a probability of 50% each.

In contrast, the behaviors to be executed by the LW MAV are more complex given their desired capabilities, consequently, its AI architecture requires further elaboration. To effectively solve Problem 2, we implement its decision-making module using the behavior tree (BT) technique, due to its inherent advantages in comparison to FSMs, such as behaviors that are highly modular, reactive, and flexible to changes [18, 21]. We briefly present the basic BT framework used, but keep in mind that alternatives commonly employ extensions [21, 22]. A BT is composed by nodes of two types: *composite or leaf*. *Composite* nodes controls the BT logic, while *leaf* nodes execute the behavior modules or condition checks. When executed, each node returns one of the following execution statuses: *Success, Failure, or Running*. Table 1 presents the return status logic of each node type.

Table 1 – Node types of a BT.

Node type	Success	Failure	Running
Selector	If one child succeeds	If all children fail	If one child returns running
Sequence	If all children succeeds	If one child fails	If one child returns running
Parallel	If N children succeeds	If M-N children fail	If all children return running
Action	Upon completion	When impossible to complete	During completion
Condition	If true	If false	Never
Inverter (Decorator)	If `Failure`	If `Success`	-

Therefore, based on Problem 2 description, we can identify the modular behaviors for the decision-making of a LW, in Table 2. We present three movement behaviors: *Chase Threat*, *Go To formation*, and *Approach Formation*, and three offensive behaviors: *Vaporize Threat*, *Freeze Threat*, and *Sacrifice Attack*.

Table 2 – Behavior description for a Loyal Wingman MAV.

Behavior	Description
Chase Threat	The loyal wingman leaves the formation to pursue a threat in order to neutralize it.
Go To Formation	The loyal wingman returns to formation.
Approach Formation	The loyal wingman approaches the leader from a safe distance from the formation.
Vaporize Threat	Performs an attack using the vaporizer gun.
Freeze Threat	Slows down a kamikaze for a limited period of time.
Sacrifice Attack	When there is no ammunition left, the LW will disable the threat by sacrificing itself as a weapon of last resort.

Figure 5 presents the behavior tree for an LW MAV using the behaviors described in Table 2. Note that every LW has an identical and independent copy of this decision-making module.

This decision-making module will run by default the `Go To Formation` behavior, where the LW keeps a flying formation surrounding the leading drone while there are no threats identified. Whenever a kamikaze drone crosses the engagement range, the LW enters into the `Chase Threat` behavior, for which the vaporizer weapon must be available. In this case, the agent leaves formation to pursue the threat and tries to neutralize it using available weapons. Once a kamikaze is in range of use of the weapons, the LW selects a neutralization method. The selected method may be the

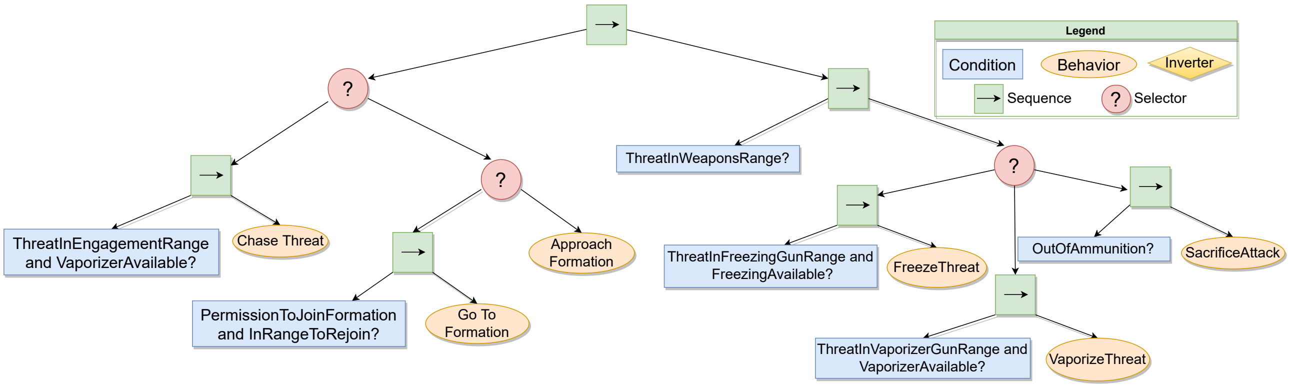
Legend
Condition
Behavior
Inverter
Sequence
Selector
ThreatInEngagementRange and VaporizerAvailable?
Chase Threat
PermissionToJoinFormation and InRangeToRejoin?
Go To Formation
Approach Formation
ThreatInWeaponsRange?
ThreatInFreezingGunRange and FreezingAvailable?
FreezeThreat
OutOfAmmunition?
SacrificeAttack
ThreatInVaporizerGunRange and VaporizerAvailable?
VaporizeThreat

Figure 5 – Behavior tree for a loyal-wingman MAV.

vaporizer gun, the freezing gun or by sacrifice as last resort (no ammo left). A weapon is *available* if there is still ammunition and if it is not cooling down after being fired. Note that the engagement distance and the range of use of both weapons are decision-making parameters, with the latter being less or equal than the nominal range of the weapons, thus leaving the use of weapons as a criterion of strategy, not necessarily equal to the nominal range of the weapon.

The neutralization strategy embedded in this BT is to first select the closest threat within range of his freezing gun (mid-range) and then try to freeze it. Once the threat is slower, it is safer to approach it, then the LW uses its vaporizer gun (short-range) to eliminate the threat. Note that the freezing and vaporizer gun must be available at the time that the weapons need to be used. The `SacrificeAttack` behavior is executed when there are no ammunition left in both weapons, it causes the LW to collide with an imminent threat, thus using its body-frame as a protective weapon. Immediately after the threat is eliminated or becomes out of range, the LW returns to formation, thus ensuring the protection of the leader.

In order to rejoin the formation, the LW must first approach the formation from a safe distance and send a request to the leader asking for permission to rejoin. the leader receives and processes the request, allowing or not the entry into formation. The LW waits for the permission to be granted and for the coordinates in the formation. The leader can arbitrarily refuse the request, so the LW will remain at a safe distance from the formation. The safe distance is also a parameter in the decision module.

Note that the decision-making modules described in this section will make decisions based only on its internal state and the states of the other agents present in the scenario, which are received through messages from the ground assets. Also, it is important to reiterate that the leader is capable of override the decisions made by the module of a specific LW, by either enforcing the loyal wingmen to return to formation, or selecting an LW to engage a selected kamikaze.

5 Computational Architecture

This section describes the computational architecture used to implement the proposed scenario. A simulation framework is built as a proof of concept for the development of the proposed techniques. The infrastructure is divided into two PCs, as seen in Figure 6. The first is responsible to compute the low-level control, physics, 3D visualization, and interface with the human pilot through a joystick. The second one is responsible to run the high-level decision-making modules of the agents being simulated. The computers communicate on a local network using the Robot Operating System 2 (ROS 2) middleware which is built on an open-source data distribution standard (DDS) middleware that provides features such as discovery, serialization, and transportation[1].

The low-level control, physics, and interface with the human pilot through a joystick are coded in MATLAB. The 3D simulator was developed using the Unity engine to allow a proper evaluation of the

[1] ROS2 Documentation: https://www.ros.org

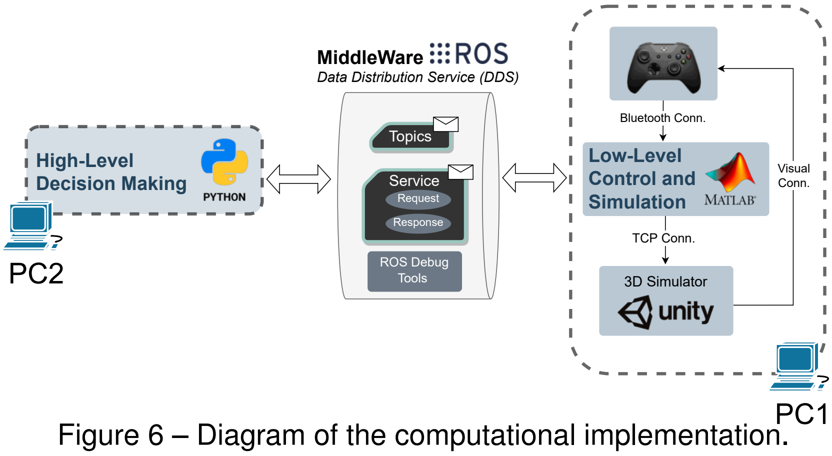

Figure 6 – Diagram of the computational implementation.

used methods. On the other hand, the decision-making algorithms were implemented in the Python programming language, due to its large community, support, and availability of libraries of AI and machine learning algorithms.

The data communication interface between the high-level decision-making layer and the low-level control layer is implemented using the framework provided by ROS 2 [23]. It is based on topics used to publish and receive customized messages in a publish-subscribe pattern that allows the layers to share important information asynchronously. The customized messages contain information regarding the action commands from the high-level decision-making module and the status update of the vehicles calculated by the low-level control and simulation sub-module, as described in Tab. 3. For ease of implementation, each vehicle is identified by a unique ID. The low-level control and simulation sub-module also sends unidirectional information containing the pose of the vehicles via socket to the 3D simulator that works as a visual feedback to the pilot.

Table 3 – Table of commands.

Type of command	***Command***	***Parameters***
cmd_movement	`GO_TO_POSE`	Target position and attitude.
	`SET_FORMATION_POSE`	Leader ID and relative position and attitude w.r.t. to the leader.
	`CHASE`	Threat ID to be pursued and neutralized.
cmd_weapon	`SHOOT`	Orders the firing of a specified gun on a desired target

6 Scenario Simulation and Results

In this section, we describe the experiment conducted using the proposed computational architecture for a use-case with 4 LW against 2 kamikazes. The results described can also be seen on this video[2]. The simulation objective is to test the effectiveness of the proposed low-level control method and the LW BT architecture. Moreover, the simulation is used to empirically find an effective LW decision-making architecture, while developing the behaviors and tuning the decision-making parameters. To this simulation, we consider that the vehicles are equal and modelled as non-planar fully actuated octocopters, as depicted in Figure 1. All the MAVs have a total mass of $1\,$kg, arm length of 0.5 m, and inertia matrix equal to $\mathrm{diag}(0.015, 0.015, 0.015)$ kgm^2. The MAVs are subject to sinusoidal force disturbances with the same amplitude of 0.6 N but with different phase shifts, and to sinusoidal torque disturbances with the same amplitude of 0.05 Nm but also with different phase shifts.

Table 4 shows the maximum velocities of the MUM-T and kamikazes, the control parameters of all MAVs, the weapon parameters of the loyal wingmen, and the parameters of the decision-making algorithm.

The low-level control and simulation, coded in MATLAB, use the Euler integration method, runs with a frequency of 100 Hz, and sends the updated data to the 3D simulator with a frequency of 20 Hz. The actuator dynamics are modeled by first-order differential equations with a time constant of 0.01 s

[2] https://youtu.be/YQjDRBW-0nk

Table 4 – Maximum velocities of the MUM-T and kamikazes, control parameters of all MAVs, weapon parameters of the loyal wingmen, and parameters of the decision-making algorithm.

MAV Parameters	
Description	*Value*
MUM-T maximum velocity	2.0 m/s
Kamikazes maximum velocity	3.0 m/s

Control Parameters	
Description	*Value*
Controller gain matrix	$\mathrm{diag}(1.44, 1.44, 1.44, 1.5, 1.5, 3)$
Sliding variable coeff. matrix	$0.5\mathbf{I}_6$

Weapon Parameters	
Description	*Value*
Vaporizer gun nominal range	10 m
Vaporizer gun cooldown time	1 s
Vaporizer gun ammo	20
Freezing gun nominal range	20 m
Freezing gun cooldown time	2 s
Freezing gun ammo	20
Weapons hitting probability	90 %

Decision Parameters	
Description	*Value*
Engagement distance	8 m
Formation radius	3 m
Decision frequency	5 Hz
Vaporizer distance threshold	5 m
Freezing distance threshold	6 m
Kamikaze *IdleState* t_1 Time	3 s

Since the MAVs use the same controller, we choose to evaluate the tracking performance of the low-level control method by plotting the tracking errors for the LW 1. A simulation of this flight can be seen in this video [3]. To better illustrate the proposed control method, we performed a formation flight test with four loyal wingmen following the controlled leader. In this sense, Fig. 7 shows the position and velocity tracking performance of the LW 1. The references are the leader position and velocity. In plot (a) it can be seen that the vehicle performs an accurate position tracking, while in plot (b) it can also be seen an accurate velocity tracking but in this case the chattering becomes more evident.

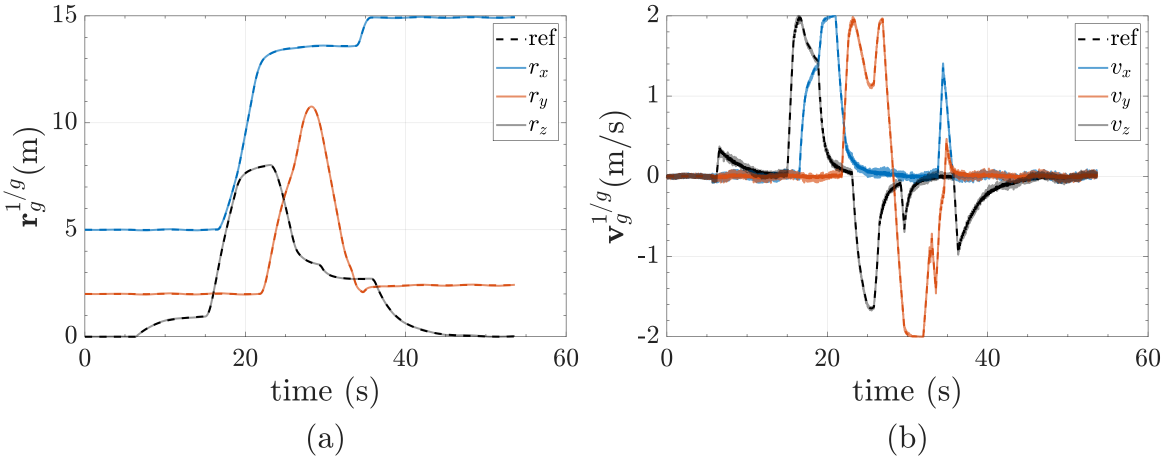

Figure 7 – Position and velocity tracking performance of the LW 1.

Figure 8 shows the attitude and angular velocity tracking performance of the LW 1. In plot (a) it can be seen that the attitude presents chattering, *i.e.*, oscillations of the system motion around the sliding surface, with a magnitude smaller than two degrees in each attitude parameterization component. This is mainly caused by the presence of unmodeled actuator dynamics. Also, note that at the time instant 5.6 s the vehicle is called to formation, *i.e.*, it receives the leader attitude and angular velocity references, and consequently it starts amplifying the leader chattering. In plot (b) we observe the same described behavior, and additionally, it can be that the components of the angular velocity have high-frequency profiles due to the presence of chattering.

Subsequently, we demonstrate operating sequences found during the execution of the scenario. First, in Fig. 9, we see the simulation result of a case where a kamikaze threat is identified and neutralized. Initially, in frame (a) we see the MUM-T in formation and an identified kamikaze threat in red. The LW MAV closest to the threat leaves the formation and chases the kamikaze by executing the `chase threat` behavior. As a result, all remaining LW MAV members in formation will readjust to new positions defined by the leader, so no gaps for attacks are present.

Consequently, in frame (b), we have the LW that left the formation attacking the kamikaze with the freezing gun, this strategy aims to facilitate the approach for neutralization since this weapon considerably reduces the kamikaze's speed. Now in frame (c) the LW gets even closer to the frozen kamikaze and uses his vaporizer gun to effectively destroy it. Consequently, we see in frame (d)

[3] https://youtube.com/clip/UgkxHcMf1eOj0JNFoK2u1dw6rOV$_{6}R38mfEs$

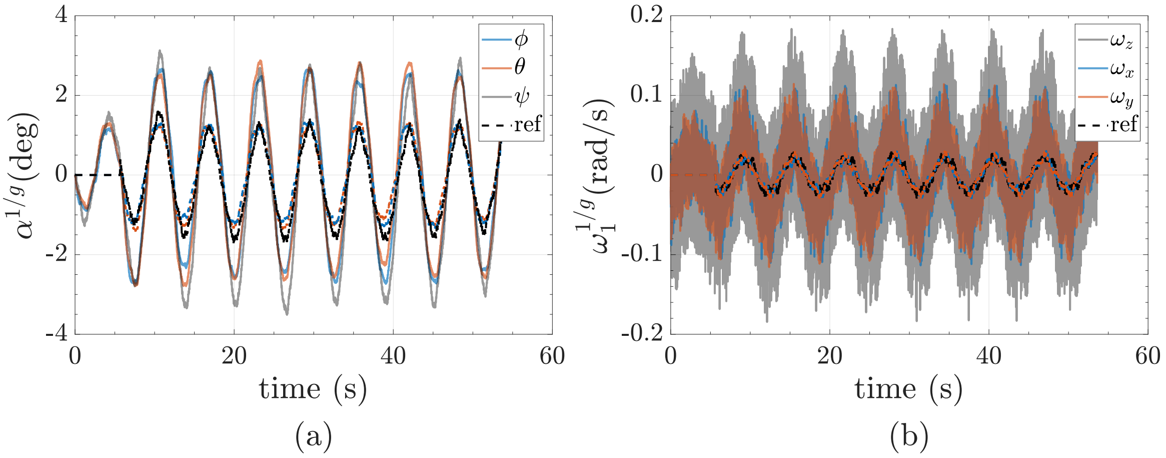

Figure 8 – Attitude and angular velocity tracking performance of the LW 1. The symbol $\alpha^{1/g}$ contains the Euler angles in the 1-2-3 sequence corresponding to $\mathbf{D}^{1/g}$.

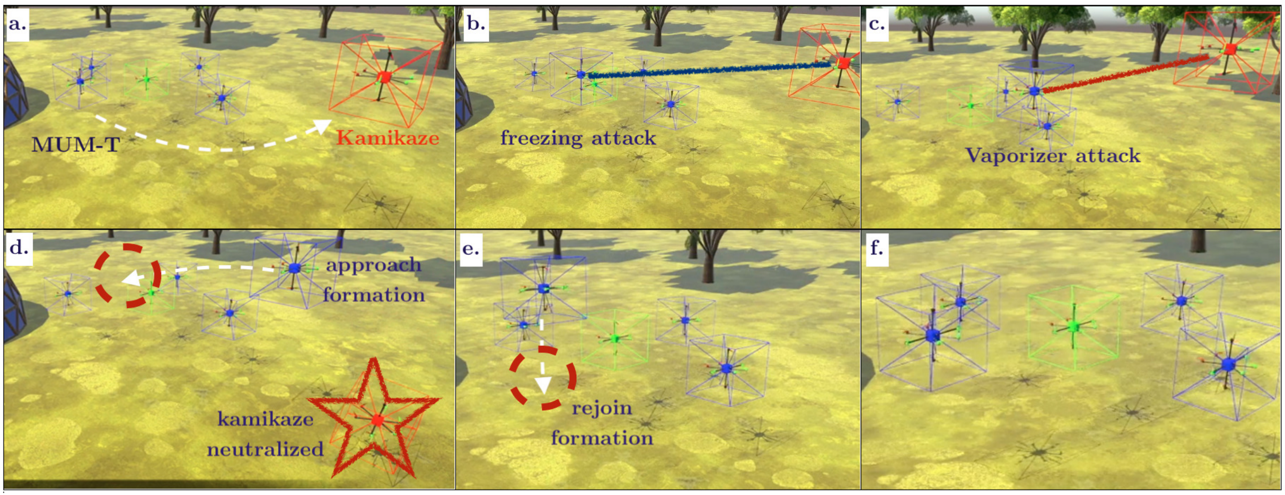

Figure 9 – Demonstration sequence of engagement to a kamikaze threat, in red. The leader is seen in the center, in green, and four loyal wingman in blue.

that the kamikaze is destroyed and the LW returns to formation, initially the LW must approach the formation from a safe distance executing the `approach formation` and request re-entry for the leader, who must allow its entry and readjust the formation with the addition of this agent. In frame (e) we see the LW successfully re-entering the formation and sequentially executing the behavior `go to formation`. Finally, in frame (f) we see all the agents keeping formation after the success of the mission.

Furthermore, in Fig. 10, we observe two sequences of cooperative neutralization that emerged during the simulations, where two loyal wingmen performed tasks together to neutralize kamikazes. In the first sequence, frames (a)-(c), we have LW1 freezing the kamikaze from a longer distance, consequently, we observe LW2 in (b), which also identified the same threat, neutralizing it with the use of its vaporizer gun. In the second sequence, frames (d)-(f), we see a similar case from a different view. In both cases, the participating LW1 and LW2 take complementary actions to eliminate a common threat.

Now we present the sequence leading to the sacrifice of an LW, in Fig. 11. The case begins in (a) with three loyal wingmen in formation, after LW4 is destroyed. In (b), we observe the LW1, which only contains one cartridge remaining in its weapons, engaging two kamikazes, note that after the neutralization of the first kamikaze, it collides with the other one present in the attack, thus eliminating two threats. In frame (d) we have LW1 out of ammo and starting his sacrifice attack after a new threat enters the area. In (e) we have the successful elimination of the kamikaze, and consequently, the

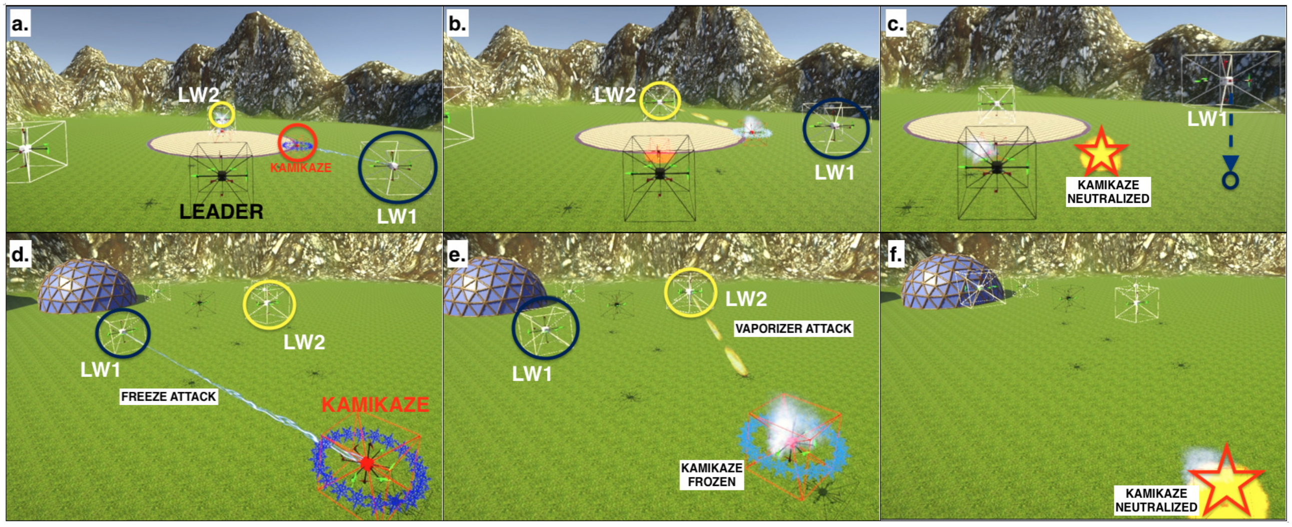

Figure 10 – Demonstration of a cooperative neutralization sequence. In this simulation, the leader is in seen in black. The LW1 are represented in the blue circle, LW2 in the yellow circle and the kamikazes in red.

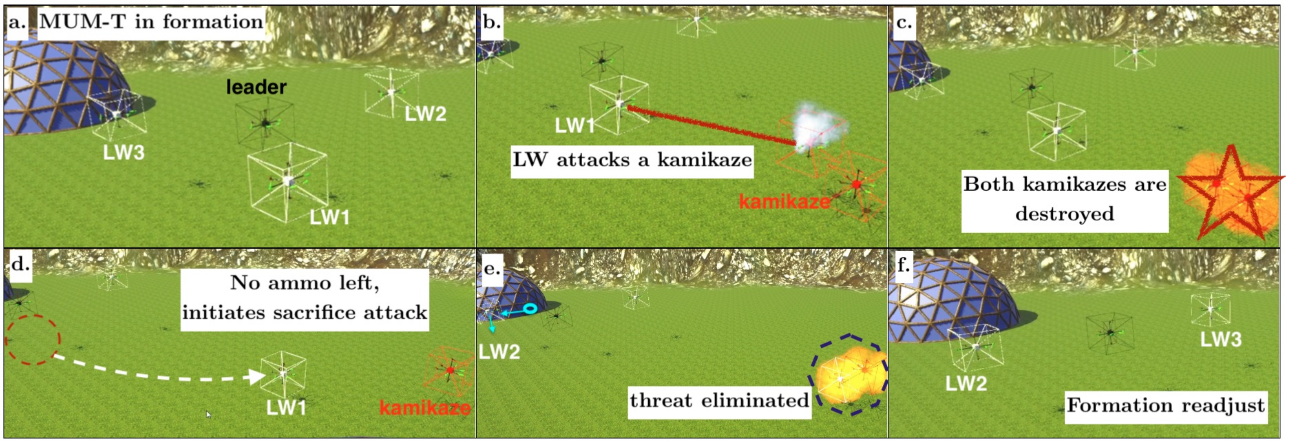

Figure 11 – Demonstration of the sacrifice neutralization sequence. In this simulation, the leader is in seen in black. The LW are represented in white and the kamikazes in red.

leader readjusts the formation of LW2. Finally, in (f) we have the MUM-T with only LW2 and LW3 remaining in its new formation.

Unexpected behaviors emerge, as illustrated in Fig. 12, we observe two kamikazes heading toward the leader, in frame (a). So, in frame (b), we have LW1 identifying and freezing the closest threat K1. As a consequence, we observe that K1 collides with K2, since K1 has been frozen and K2 which is closely following K1 cannot avoid the collision in time. In this way, we can highlight this emergent behavior as an indirect neutralization method.

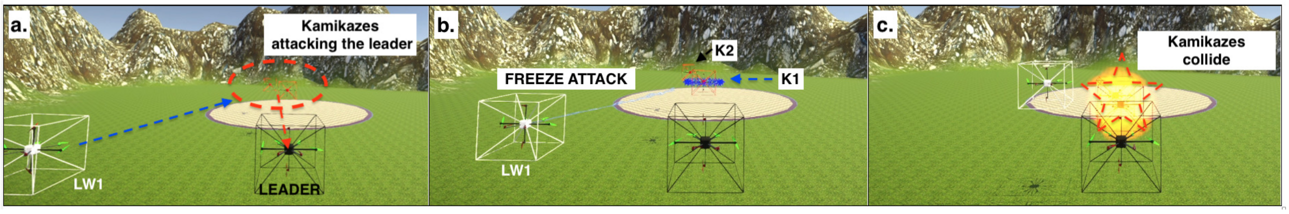

Figure 12 – Freezing attack being used as a neutralization method. In this simulation, the leader is seen in black. The LW1 in white and the kamikazes K1 and K2 in red.

7 Conclusions

This paper evaluated the concept of CEC applied to MUM-T. To this end, we proposed a defensive scenario where loyal wingman MAVs cooperate to defend a manned leader and protect a critical infrastructure from a swarm of kamikazes with explosive capabilities. To deal with such threats the loyal wingmen have the capability to engage and neutralize the threats, using two idealized weapons, a vaporizer gun representing a short-range weapon capable of neutralizing a threat with a single shot, and a freezing gun, a non-lethal weapon of mid-range, capable of slowing down the threat. To reduce the complexity of this problem, all the vehicles are modeled as fully actuated MAVs, and the problem is broken down into two parts, one involving a low-level control layer, and another involving a high-level layer of intelligent software capable of making autonomous and decentralized decisions. The low-level control designs a joint attitude-position control law using the first-order sliding mode control that guarantees the closed-loop system robustness with respect to model uncertainties and disturbances. On the other hand, the decision-making for loyal wingman agents was developed using the behavior tree technique and reduced to modular behaviors, such as: chasing a threat, remaining in protective formation, approaching formation to initiate rejoin procedure, and attack behaviors using the available weapons or using the sacrifice attack in the extreme case where there are no more weapons available. The method is analyzed using a high-fidelity 3D simulator and shows to be effective. In future works, we plan to increase the complexity of the kamikazes' behaviors to improve the difficulty of the proposed scenario. We also plan the development of consensus algorithms and task allocation using a world model that is not synchronized among the MUM-T agents. Moreover, more complex scenarios can be defined involving different types of threats and drones.

Acknowledgement

Luiz Giacomossi Jr. acknowledges EMBRAER S.A. for his scholarship and Jorge A. Ricardo Jr. acknowledges the Coordination of Superior Level Staff Improvement (CAPES) and EMBRAER S.A. for the doctorate scholarship under the Academic-Industrial Graduate Program (DAI). The authors are grateful for the support of the Funding Authority for Studies and Projects (FINEP).

References

[1] APL-Team, The cooperative engagement capability, JOHNS HOPKINS APL TECHNICAL DIGEST 16 (4) (1995).

[2] C. Humphreys, R. Cobb, D. Jacques, J. Reeger, Optimal mission path for the uninhabited loyal wingman, in: 16th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, 2015, p. 2792.

[3] F. Santoso, M. A. Garratt, S. G. Anavatti, Visual–inertial navigation systems for aerial robotics: Sensor fusion and technology, IEEE Transactions on Automation Science and Engineering 14 (1) (2016) 260–275.

[4] V. Kumar, N. Michael, Opportunities and challenges with autonomous micro aerial vehicles, The International Journal of Robotics Research 31 (11) (2012) 1279–1291.

[5] L. Giacomossi, S. Schwanz Dias, J. Brancalion, M. Maximo, Cooperative and decentralized decision-making for loyal wingman UAVs, in: IEEE Latin American Robotics Symposium (LARS), IEEE, 2021, pp. 78–83.

[6] D. Brescianini, R. D'Andrea, Design, modeling and control of an omni-directional aerial vehicle, IEEE International Conference on Robotics and Automation (ICRA) (2016) 3261–3266.

[7] A. L. Silva, D. A. Santos, Fast nonsingular terminal sliding mode flight control for multirotor aerial vehicles, IEEE Transactions on Aerospace and Electronic Systems 56 (6) (2020) 4288–4299.

[8] J. A. Bezerra, D. A. Santos, On the guidance of fully-actuated multirotor aerial vehicles under control allocation constraints using the receding-horizon strategy, ISA Transactions (2021).

[9] D. A. Santos, J. A. Bezerra, On the control allocation of fully actuated multirotor aerial vehicles, Aerospace Science and Technology 122 (2022) 107424.

[10] J. A. Ricardo Jr, D. A. Santos, Smooth second-order sliding mode control for fully actuated multirotor aerial vehicles, ISA Transactions (2022).

[11] F. L. Markley, J. L. Crassidis, Fundamentals of Spacecraft Attitude Determination and Control, Space Technology Library, Springer-Verlag, New York, 2014.

[12] S. V. Drajunov, V. I. Utkin, Sliding mode control in dynamic systems, International Journal of Control (1991) 1029–1037.

[13] V. I. Utkin, Sliding Modes in Control and Optimization, Springer-Verlag, 1992.

[14] J. A. Ricardo Jr, D. A. Santos, T. Roux Oliveira, Attitude tracking control for a quadrotor aerial robot using adaptive sliding modes, Proceedings of the XLI Ibero-Latin-American Congress on Computational Methods in Engineering (11 2020).

[15] B. Paden, M. Čáp, S. Z. Yong, D. Yershov, E. Frazzoli, A survey of motion planning and control techniques for self-driving urban vehicles, IEEE Transactions on Intelligent Vehicles 1 (1) (2016) 33–55.

[16] T. P. Baldão, M. R. Maximo, C. d. A. C. Cesar, Decision-making for 5x5 very small size soccer teams, in: 2020 Latin American Robotics Symposium (LARS), IEEE, 2020, pp. 1–6.

[17] M. Buckland, Programming Game AI by Example, Jones & Bartlett Publishers, Burlington, Massachusetts, USA, 2004.

[18] M. Colledanchise, P. Ogren, Behavior Trees in Robotics and AI: An Introduction, Chapman Hall/CRC Press, 2018.

[19] L. C. Melo, D. C. Melo, M. R. Maximo, Learning humanoid robot running motions with symmetry incentive through proximal policy optimization, Journal of Intelligent & Robotic Systems 102 (3) (2021) 1–15.

[20] L. Giacomossi, F. Souza, R. G. Cortes, H. Mirko Montecinos Cortez, C. Ferreira, C. A. C. Marcondes, D. S. Loubach, E. F. Sbruzzi, F. A. N. Verri, J. C. Marques, L. A. Pereira, M. R. O. A. Maximo, V. V. Curtis, Autonomous and collective intelligence for uav swarm in target search scenario, in: 2021 Latin American Robotics Symposium (LARS), IEEE, 2021, pp. 72–77.

[21] M. Iovino, E. Scukins, J. Styrud, P. Ögren, C. Smith, A survey of behavior trees in robotics and ai, Robotics and Autonomous Systems (2022) 104096.

[22] G. L. Silva, M. R. Maximo, L. A. Pereira, A minimalist open source behavior tree framework in c++, in: 2021 Latin American Robotics Symposium (LARS), IEEE, 2021, pp. 306–311.

[23] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, A. Y. Ng, et al., Ros: an open-source robot operating system, in: ICRA workshop on open source software, Vol. 3, Kobe, Japan, 2009, p. 5.

COPYRIGHT STATEMENT