Efficient Swarm Neutralization in Complex Environments Using Multi-Agent Deep Reinforcement Learning and Adaptive Navigation Strategies

Junior Samuel Lopez Yepez
Thales Digital Solutions Inc., 6650 Street Saint-Urbain, Suite 350, Montreal

Antoine Fagette
Thales Digital Solutions Inc., 6650 Street Saint-Urbain, Suite 350, Montreal

Stephanie Allogba
Thales Digital Solutions Inc., 6650 Street Saint-Urbain, Suite 350, Montreal

Jeremy Viel
Thales Digital Solutions Inc., 6650 Street Saint-Urbain, Suite 350, Montreal

Filipe Carvalhais Sanches
Thales Digital Solutions Inc., 6650 Street Saint-Urbain, Suite 350, Montreal

Guillaume Dorion-Racine
ARA Robotics, 2260 Avenue Aird, Suite 204, Montreal
CANADA

junior.lopez@thalesgroup.com

ABSTRACT

This research project presents a novel swarm versus swarm approach for efficiently neutralizing intruding drones, possibly evolving in swarms, using the same or a higher number of collaborative autonomous drones. Each drone in the swarm executes independently its actions in coordination with the rest of the swarm to counteract up to five enemy drones per mission. Our methodology leverages a sequential decision-making process, dividing the mission into distinct phases - approach, tracking, and neutralization- to minimize conflicts during swarm interactions and to enhance control during autonomous operations. In addition, we delimited a protected flight zone where the intrusion of any enemy drone is strictly forbidden and would be considered a mission failure. To ensure safe navigation in complex environments, we propose restricting the defensive swarm's flight to a delimited airspace called a buffer zone, encompassing the protected zone. Any intrusion in the buffer zone triggers a response from our system which in turn allows a coordinated series of actions by the defensive swarm to prevent the enemy drone from entering the protected zone. We use a multi-agent deep reinforcement-learning framework with centralized training on a simulator and decentralized execution, optimizing the decision-making process in simple simulated environments before testing in realistic simulators and deploying real drones for field operations.

Our approach integrates an optimized and dynamic target assignation algorithm for the shortest route assignment able to re-plan at each time step. After assignation, at each time step, each agent of the swarm uses a multi-agent deep reinforcement-learning model to approach and track a target. When an agent meets the criteria to start the neutralization phase, the autonomous navigation control of the selected agent switches to a catching navigation mode, while the rest of the swarm continues tracking if there are no other threats in the buffer zone. Moreover, the navigation decision-making process is combined with adaptive collision avoidance capabilities that depend on the mission phase and decisions from the pre-trained AI navigation model. The collision avoidance capabilities mitigate the risks of colliding against other agents of the same defensive swarm, detected target entities and other detected objects (e.g., birds). This ensures robust navigation and efficient tracking in complex environments. These functions are *coordinated with a sensing one, where an algorithm controls the orientation in yaw and pitch and the zoom of the gimbal mounted EO sensor embedded on each drone.*

The proposed system demonstrates effective performance in neutralizing small swarms through a sequential decision-making process of switching algorithms while maintaining safe navigation, from detection to neutralization, in simulation. During the simulations, the defensive drones use a net mechanism for catching the intruders with high performance. Moreover, we deployed drones in the field facing enemy drones detected by a ground radar on real flight demonstrations and validating this autonomous decision-making process. Our research contributes to advancing the state-of-the-art in swarm-based decision-making tactics and opens new avenues for using collaborative swarms in defense and military operations within the NATO context.

1.0 INTRODUCTION

With the advent of technology and the growing relevance of artificial intelligence (AI) in various armed forces around the world, the concept of unmanned and autonomous military systems has taken centre stage. The world has gradually moved the focus of its military activities from human-centred operations to AI-powered operations. Drones, for instance, have been significantly leveraged in the past decade due to their technological sophistication and efficient manoeuvrability, which serves as a critical component for contemporary military operations. Their commendable surveillance reach, precision targeting, and damage assessment capabilities have combined to make a significant contribution to operational outcomes [1]. Due to their high mobility, easy deployment, and an array of sensor accommodation capabilities, UAVs have found their use in multiple capacities [2]–[5], which has led to new security risks when the UAVs are misused. The danger posed by enemy drone swarms has invigorated the development of counteracting drone swarm systems. The idea of creating defensive drone swarms to counteract invading swarms is not new and has been explored in the literature [6]. Nevertheless, the full potential of UAVs is limited by the consistent challenges presented by autonomous navigation and artificial intelligence integration [7], [8]–[11]. Moreover, the integration of AI, although promising in the aspects of problem-solving and resource management, grapples with obstacles such as lengthy training time, computational power, complexity, information updating, and quick environmental adaptation [11].

Current research also exhibits substantial focus on collision avoidance systems that aim to forewarn operators of imminent collisions for autonomously control systems to steer clear of incoming threats [13], [14]. These systems can be divided into global path planning, which considers the entire environment, and local path planning, which focuses on changes detected in the immediate environment [15]. Despite significant advancements in obstacle detection, collision avoidance, path planning, localization, and control, ensuring safe navigation still presents a poignant challenge, especially in volatile environments involving multiple UAVs and moving obstacles [16], [20]– [22], [12], [23], [25].

Swarms of UAVs have received considerable attention in recent times due to their collaborative and cooperative functionality, offering substantial benefits over single UAV systems [17]– [19]. However, the present state-of-the-art has ample room for improvement of the overall system performance and to ensuring the safety and success of autonomous UAV for real flight operations.

These persisting issues prompt the need for unconventional approaches and innovative perspectives that could contribute a new solution to the shortcomings of the current systems. In this article, we present our solution to neutralize a swarm of drones by leveraging our own fleet of drone and utilizing an adaptive navigation system which includes multi-agent deep reinforcement learning. By optimizing the utilization of multi-agent deep reinforcement learning (MARL) algorithms and heuristics, this research presents a promising pathway to enhancing safer adaptive navigation measures.

1.1 Overview of the Proposed Approach

This research proposes an end-to-end autonomous decision-making navigation procedure encompassing different mission phases that intricately interplay with sensing and neutralization capabilities. At each phase, the AI agent of each drone has the capability of using different navigation models or rules to increase its performance and control along a mission. The interplay of different navigation algorithms has been tested on simulations, as well as during real flight demonstrations.

Crucially integrated within these phases is an advanced collision avoidance mechanism, based on the artificial field potentials concept [25], that changes dynamically depending on the reliability of data information. This system is designed to prevent drones from colliding with other drones from the defensive swarm as well as with detected target entities and unexpected detected objects, such as birds. Such collision avoidance capabilities ensure the integrity of our drones while maintaining an efficient targeting system. The assignation of targets to each member of the defending swarm is managed by a combinatorial optimal algorithm. Furthermore, an autonomous sensing control algorithm was designed, tested and deployed in simulations and real flight demonstrations. It controls the pan, tilt and zoom of a gimbal-mounted camera to track visually the targets. In the next section, we further define and detail each decision-making component of the system and mission phase.

2.0 METHODOLOGY

The design principle segregates the mission into different phases - approach, tracking, and neutralization. The idea behind this segregation is to allow unique processes for each phase. For the approaching and tracking phases, a Multi-Agent Deep Reinforcement Learning (MARL) model is used for navigation to achieve higher performance in following the target. For the neutralizing phase, a rule-based algorithm is used instead. This algorithm's deterministic nature allows for more direct control, a crucial aspect when attempting to neutralize a target safely and efficiently. The navigation is coupled with autonomous sensing control to provide a visual tracking of the targets by the operator. Since a defending swarm might face multiple intruders, a dynamic target assignation combinatorial algorithm is set in place. The objective is to counteract the drone threats that can be presented inside the buffer zone before reaching a protected area. The buffer zone refers to a three-dimensional geodetic polygon, bound by a minimum and maximum altitude, within which the autonomous defending drones operate. When one or more intruder drones (i.e., targets) are detected inside the buffer zone, the defending swarm starts the counteracting operation entering an "approaching" phase to an assigned target.

2.1 Dynamic Target Assignation Algorithm

Every autonomous drone in the defensive swarm is strategically assigned to one target drone within the enemy swarm. The composition of the antagonist swarm can range from one to five drones, whereas the defending force always maintains a strength of five drones. The drone-to-target association relies on a rule-based combinatorial strategy that calculates the best plan of action by computing the minimal collaborative distance each drone needs to traverse with respect to every target drone. The primary objective lies in counteracting each intruding drone with at least one from our swarm.

Formally speaking, we determine what it is the assignation $x_{ij} \in \{0,1\}$ of a defending drone i to an enemy drone j that minimizes the function:

$$f = \sum_{i=1}^{N} \sum_{j=1}^{M} d_{ij} \cdot x_{ij}, \forall j = 1,2, \dots, M \qquad , \forall i = 1,2, \dots, N$$

This function, however, is subject to conditions to ensure that each target has at least one assigned defending drone i.e., $\sum_{i=1}^{N} x_{ij} \geq 1$ and that each defending drone has only one target to handle i.e., $\sum_{j=1}^{M} x_{ij} = 1$. Note that these conditions are only valid when our number of defending drones N is equal to or higher than the number of intruding drones M . One way to solve this optimization problem is by using a grid search at each timestep. Nevertheless, this method is only feasible for flight trials when there is not a high number of drones to compute all possible combinations since its computational complexity is $O(N!)$. In our experiments, the computational complexity of this method did not present a problem when computed at each time step for experiments run with $M = 5$ and $N = 5$.

This allocation process ensures that the defensive drones on a neutralization mission are strictly dedicated to that task, unaffected by other drones in the swarm. The shortest route assignment and re-planning happen at every timestep, providing an adaptive response to rapidly evolving field situations. Once a drone completes its mission of securing a target and relocating it in a retrieval area, it can re-align itself with the rest of the swarm to support tracking of any remaining targets, leveraging its embedded camera system.

2.2 Autonomous Collaborative Swarm Navigation

A multi-agent deep reinforcement learning (MARL) model is used during the approaching and tracking mission phases for autonomous collaborative drone navigation. The approaching phase marks the start of the mission, beginning when the defending drones are getting in motion towards a new target and ending when they get close to that target. In this phase, our main objective is to navigate the defensive swarm from their point of origin to the intruder's vicinity. We accomplish this by employing MARL model to calculate the best trajectory for each drone. The success of this phase is defined by the swarm's ability to cover the distance to the incoming drone(s) effectively while maintaining safe navigation. Following the approaching phase, the drones enter the tracking phase. In this phase, the drones stay near the target, observing its movement while waiting for a rule or command to transition into the neutralizing phase. Moreover, adaptive collision avoidance capabilities come into play intensively, preventing collisions with other defensive drones, targeted entities or detected objects.

2.2.1 Deep Multi-Agent Reinforcement Learning Algorithm for Navigation

The choice of MARL is guided by its strengths in providing solutions in environments characterized by continuous action spaces and a high number of agents, a typical scenario in drone swarm vs. swarm applications.

In the context of MARL, we adopted different off-policy algorithms with centralized training and decentralized execution. A well-known representation of this algorithm is the Multi-Agent Deep Deterministic Policy Gradient (MADDPG) algorithm [26]. In essence, MADDPG allows an agent to incorporate policies of other agents while maximizing its own expected reward. This sharing in turn leads the agents of the swarm to exhibit collaborative behaviour when the expected reward is shared among the agents of the swarm. The key principles of MADDPG can be synthesized as maximizing the reward function by learning from experiences collected in a buffer. MADDPG utilizes two main neural networks, with other neural networks for learning stability. One of the networks is known as the Critic Q_{ω}. The Critic is trained to estimate the cumulative reward r along timesteps t based on the current observations, known as the multi-agent state $x_t = (s_1, \ldots, s_N)$ and the actions a_t of all N agents in the swarm. The estimated value for agent i is known as the Q value, where

$$Q^i = E[r_t + \gamma r_{t+1} + \cdots | x_t, a_t^1, \ldots a_t^N].$$

In the case of the critic network, the update function of the weights ω is as follows:

$$\Delta\omega = \rho E_{(x,a_1 \ldots a_N,r,x')\sim D}\left[\left(y - Q_{\omega}(x, a_1 \ldots a_N)\right)^2\right]$$

In this equation, y is the target value defined as:

$$y = r + \gamma Q'_{\omega}\left(x', \mu'_{\theta}(x')\right)$$

The Critic network, however, is artificially created and it works only for training purposes. This is because it has access to the current actions of all the agents of the swarm, which might not be possible in real flights. This is known as centralized training. The execution must be able to be carried out with no centralization, meaning that the agent has only the information about its own states and actions. For that, the other neural network, the Actor μ_{θ}, can make decisions to take an action a_t based uniquely on its current local state s_t. The Actor model, during training, uses the value produced by the Critic and it updates its neural network by the gradients of the Critic network and the Actor's actions as:

$$\Delta\theta = \rho E_{s \sim D}\left[\nabla_a Q_{\omega}(s, a)|_{a=\mu_{\theta}(s)} \nabla_{\theta}\mu_{\theta}(s)\right].$$

Here, ρ represents the learning rate, D denotes the replay buffer, s and s' are the current and next states, a is the action, r is the reward and γ is the discount factor. Note that this type of MARL algorithm can provide decentralized execution to drones, while working collaboratively. The mission goal could then be encoded as the reward function that the neural networks using MARL attempt to optimize at each iteration.

2.2.2.1 Reward Function

In MARL, rewards play a pivotal role as they define the quality of actions thus guiding the following ones. In the context of our drone swarm system, we have designed a dual reward criterion: an individual reward and team rewards for approaching and tracking phases.

In the approaching and tracking phases, an individual drone is awarded based on its proximity with the target; specifically, the Euclidean distance ($d_{euclidean}$) between the agent's i position and the enemy's j position is included in the computation of the reward. Mathematically, this distance-based individual reward function can be simply formulated as:

$$R_{\text{individual}} = d_{i,j} = -d_{euclidean}\left(agent_i, enemy_j\right)$$

The objective of having a negative "reward" value is to encourage the agent to minimize this value, thus, the distance to the detected enemy as soon as possible. There is a navigation collaborative reward based on the distance, to promote smart collaborative behaviour among the agents of the swarm. It is described as the cumulative distance of the M detected enemies with respect to their closest defending drone:

$$R_{swarm} = -\sum_{j=1}^{M} \min_i\left(d_{i,j}\right)$$

The neural network models were trained with a reward "neutralization" bonus, meaning that if any agent of the team reached a Euclidean distance below a neutralization threshold (e.g., 30 meters), there is a positive bonus (e.g., +50) to each member of the team. Therefore, the reward function for an agent is represented by:

$$R = R_{individual} + R_{swarm} + Bonus_{neutralization}.$$

The union of these reward strategies contributes to the synergy and efficiency of the defensive swarm during the whole mission lifecycle.

2.2.2 Adaptive Collision Avoidance

In the real-world deployment of our defensive drone swarm, one of the most critical aspects of their operational efficiency and safety hinges on collision avoidance algorithms. In our deployment, collision avoidance fields include interactions with friendly entities (drone peers within the same swarm), hostile entities (target drones), neutral objects (birds or other flying objects), and the borders of the buffer zone. The latter is crucial as it prevents our drones from straying out of the allowed flight area. The collision avoidance fields (i.e., local fields) are dynamic depending on the uncertainty provided by the sensors on the position of the entities inside the buffer zone. When computation errors or malfunctions occur, a local field could potentially produce a very high repulsion, pushing the drones out of the buffer zone. To prevent this, we have set a threshold value (maximum safety distance). When the computed safety distance exceeds this threshold, the movement of the drone is halted, effectively freezing its position. This ensures that the drone swarm remains within the defined operational area and avoids being repelled outward due to excessive uncertainty or computational error. A “minimum safety distance” is parametrized to ensure that, even without uncertainty in the estimated positions, the drone respects a safety distance to prevent accidental collisions.

In the range in between the minimum and maximum local field radius, we dynamically adapt the local field to the position uncertainty collected by the sensors. We use the position uncertainties of each observed entity as a covariance ellipsoid in a 3D matrix form U. Then, we use our current 3D position x and compute the Mahalanobis distance [27] from an entity j as:

$$d_{Mahalanobis,\,j} = \sqrt{x^T U_j^{-1} x}\,.$$

The radius of the collision avoidance field r_j is then the proportion of the distance between the defending drone position and the entity of interest and its Mahalanobis distance to this entity. This is represented as:

$$r_j = \frac{|x - x_j|}{d_{Mahalanobis,\,j}}$$

Moreover, we define the repulsion of the entity j to be cubic, following the logic of navigation within a 3D space, and it is computed as:

$$repulsion_j = -r_j^3(\kappa)u = -r_j^3\left(\frac{1}{|x-x_j|^3}\right)\left(\frac{x-x_j}{|x-x_j|}\right),$$

where κ is a constant function that determines the repulsion factor based on the cubic norm $|x - x_j|^3$, and u is the unit vector that will provide direction of the repulsion.

To be noted that when the uncertainty of the positions is near zero, the Mahalanobis distance will be high for even small separations between the defending drone and the entity j. Thus, the collision avoidance radius becomes small, hence, the repulsion is minimal. The influence of all the entities, including, friendly, enemy and neutral entities on a given defending drone i is then:

$$total\ repulsion_i = \sum_{j=1}^{K} repulsion_j$$

The risk of collision is minimized by navigating in the direction where the sum is minimal on top of the navigation vectors provided by the rule based or MARL algorithm. It is important to mention that, under collision risk circumstances, the vectors of the total repulsion might be much higher than the navigation vectors which are bounded by default. This mechanism allows the integration of autonomous decision-making for navigation control while favouring safe flights through our independent collision avoidance mechanism.

2.2.3 Rule-Based Algorithm for Controlled Navigation for Threat Neutralization

The final phase in this process is the neutralizing phase. Upon verification of the conditions for capture and receiving authorization to engage, the navigation switches from the reinforcement-learning model to a rule-based system. The switch to a deterministic, rule-based algorithm provides a higher degree of control during this critical phase. Such target capturing requires precise manoeuvres, thus warranting enhanced control beyond what the probabilistic RL model can provide. Initial focus prioritizes positioning the drone at an altitude higher than the target, assuring an optimal path for net deployment while maintaining a secure distance as per the collision avoidance algorithm. After achieving an optimal altitude, the drone can reduce its safety distance. This strategic placement, directly above the intended target, allows for a swift and precise capture attempt.

2.3 Sensing and Tracking Mechanisms

Sensing and tracking are integral components of the swarm-based operation, allowing our drone swarm to detect, approach, and neutralize enemy entities rapidly. For efficient sensing, each drone in our swarm is equipped with a gimbal. This gimbal, bearing an electro-optic sensor, is controlled to optimize the detection and tracking of enemy entities.

Given the yaw and pitch attitudes of each individual drone, a rule-based algorithm is implemented to determine the optimal pan and tilt of the gimbal. The estimated difference between the yaw attitude and current pan of the gimbal to direct the target estimated position to the centre of the Field of View of the camera is Δpan_{total}. Similarly, the pitch deviation from the tilt is $\Delta pitch_{total}$. The estimated position of the target can vary depending on the quality of the estimations, including the precision of the sensors. Due to this, the actual sensing control on pan and pitch of the gimbal uses a damping factor γ to increase smoothness during the movement of the gimbal, hence, the camera. Thus, the system optimizes the positioning of the sensor, ensuring effective sensing and subsequent stable visual tracking. Moreover, the zoom function of the camera is also fine-tuned given various parameters. These include the CMOS specs of the camera, the desired distance to the object, the estimated size of the object (i.e., the target), and the estimated position of the target. This calculated control helps in maintaining focused and efficient visual tracking by adjusting to the dynamic nature of the moving targets.

3.0 SIMULATION AND RESULTS

3.1 Simulation Results

In the initial stages of this research project, primary training and testing were conducted on a multi-particle environment (MPE) characterized by simplistic dynamics. The fast-processing nature of the MPE allowed for more extensive experimentation and model fine-tuning. It was here that our model successfully demonstrated the potential to neutralize a smaller, faster drone using a defense swarm of five drones. The drones within these simulators are misrepresented as simple 3D round particles possessing momentum and velocity, making the MPE an ideal training ground for preliminary model development.

Figure 1 presents the evolution of the rewards gained per episode throughout the training process on a 3D MPE simulator, each episode having a fixed duration. For this article, the same algorithm was used to train a swarm in a 2D environment on a use case of five defending drones versus one intruder. The Figure 2 includes snapshot of the 2-D MPE simulator with train models to showcase different collaborative defending behaviors.

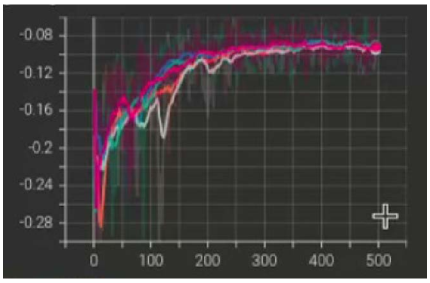

Figure 1: Total reward collected (vertical axis) along episodes (horizontal axis) of 50 timesteps. Each color represents the total reward for navigation of each defending drone of the swarm.

We enriched the model's generalization capabilities by randomizing the starting positions of the agents, allowing for the accommodation of an array of diverse use cases. Following these steps, the model was then introduced into a more realistic and comprehensive simulator for further testing and adaptation.

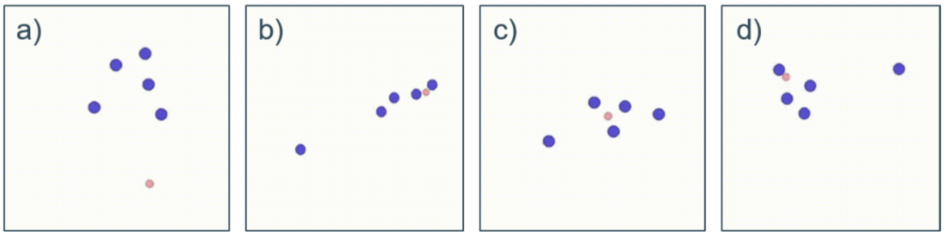

Figure 2: Collaborative MARL trained model for navigation exhibiting different defensive behaviors on a 2D simulator for a use case of five defending drones (blue) versus one target drone (red). The drones approach the target in (a) with a concave formation. A linear barrier is spawn on some episodes (incise b). A typical behavior to catch the target is to surround it, as depicted in incise (c). At least one drone acting as a backup of the swarm is also common to observe (d).

This reinforced simulator presented a platform inclusive of collision avoidance algorithms, physical quadcopter dynamics, and environmental factors such as wind patterns. It replicates the actual geographic locality of our live flight demonstrations, offering a significant proximity to real-world conditions. Moreover, this simulator integrates control from the same Ground Control Station system used in actual flight demonstrations. These enhancements ensure an authentically realistic testing environment for the model, contributing to the refinement and validation of its accuracy and efficacy.

We captured significant elements from the simulation videos and translated them into Figure 3, snapshots illustrating different points in the scenario where our approach demonstrated robustness and effectiveness. Figure 3 shows the initial approach of the defensive swarm against an offensive one. The figure includes the defending swarm as blue points, and intruders as red with a faded track of their most recent trajectory. Moreover, Figure 3 reveals a snapshot during the tracking phase. This figure illustrates the dynamic target assignation in action, agents' coordination, and how adaptive collision avoidance prevents interferences between friendly entities while tracking hostile entities. Moreover, Figure 3 depicts the subsequent moments of enemy neutralization. It reveals the spatial coordination and the implemented strategy for the enemy drone's neutralization by the selected drone.

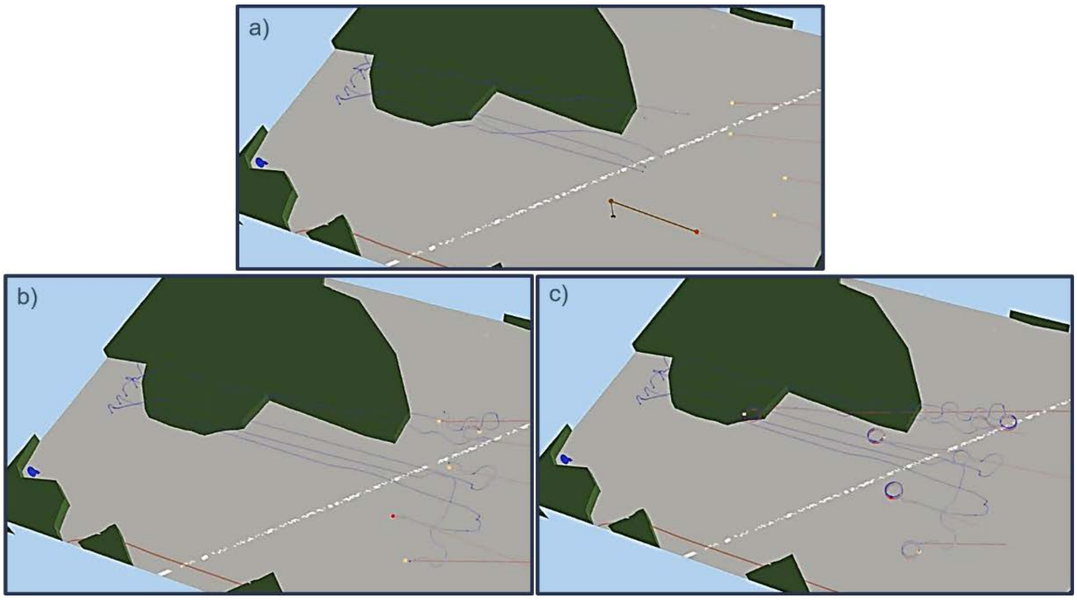

Figure 3: Different mission phases of an autonomous defending swarm with five agents responding to a threat of five intruders observed in a 3D realistic simulator. The target assignation and Deep MARL model for navigation are depicted for the approaching phase (a) and the tracking phase (b). For visualization purposes, the defending drones fly idle in circles when they have captured a threat, as observed in (c).

3.2 Real-flight Demonstration Results

The execution and performance of our navigation algorithms, specifically those with adaptive navigation strategies, were successfully deployed in the field and tested against various threat scenarios. These algorithms were utilized across different mission phases: approach, tracking, and neutralization. Our field tests involved scenarios with different swarm configurations. In one scenario, the tested field consisted of a defensive swarm of five drones working in unison against one incoming intruding drone. Meanwhile, the other scenario entailed a matching two-on-two drone configuration; both scenarios demonstrated promising results which affirm the efficiency and safety of our swarm versus swarm methodology for autonomous collaborative decision-making.

The field observations were simultaneously mirrored onto simulators to provide a comparative basis for the navigation algorithms' performance. In the simulations, the defensive drone effectively captured the target in a safe manner. During the real-life testing, to prevent damage to drones and to facilitate continuous experiments, we marked the target as 'captured' when they met the specific conditions of capture, rather than using a physical net.

In terms of sensing, our results highlighted the drones' ability to visually track targets by controlling the pan, pitch, and zoom of an onboard camera. This refined control enhances the effectiveness of the swarm in tracking and neutralizing enemy drones, thus asserting the promise of our work in advancing swarm-based decision-making strategies, particularly within a NATO context. Our research unlocks the potential of autonomous collaborative swarms in both defensive and military operations.

4.0 DISCUSSION

The primary advantage of the proposed system is its ability to perform a dynamic target assignation algorithm for the shortest route assignment capable of re-planning at each time step. This unique feature aids efficiency and quick response, facilitating timely neutralization of hostile drones. Moreover, the centralized training on simulators ensures the development of robust AI models, while decentralized execution provides flexibility during real field operations.

Despite its promising benefits, our approach faces several challenges. One such limitation is the need for well-calibrated simulators that accurately depict real-life scenarios to ensure effective learning by the drones. Errors in calibration could lead to discrepancies in performance when transitioning from the simulated environment to the field. Additionally, hostile drones' unpredictable behaviours pose significant challenges to maintaining an efficient tracking and neutralization system.

Our research's implications for NATO operations are manifold. First, it can enhance airspace security, providing an efficient system for neutralizing enemy drones without risking human operators. Furthermore, this autonomous decision-making process can be potentially integrated into other NATO defense systems, contributing to the enhancement of overall defense capabilities.

5.0 CONCLUSIONS

5.1 Summary of Major Findings

Our proposed swarm versus swarm methodology demonstrates an effective means for neutralizing intruding drones within a defined airspace. The process of using a sequential decision-making model enhanced efficiency and safety during autonomous operations. Real-flight demonstrations validated the strategy's effectiveness, contributing to the development of robust tactics for neutralizing groups of hostile entities.

Our approach provides significant contributions to swarm-based decision-making tactics. The unique process of dynamically assigning targets, coupled with an efficient algorithm for navigation and tracking elevates the state-of-the-art drone swarm tactics. Overall, the research offers promising solutions for military operations within the NATO context, shaping the future of autonomous defense systems.

Future research should focus on developing intuitive simulators to facilitate the learning process of the AI models. Studies should also investigate refining algorithms to deal with the unpredictable behaviors of the hostile drones. Moreover, considering other neutralizing mechanisms apart from the net-related approach could prove beneficial.

6.0 REFERENCES

[1] Scharre, P., & Horowitz, M. C. (2015). Autonomy in Weapon Systems. Center for a New American Security Working Paper.

[2] P. Grippa, D. Behrens, C. Bettstetter and F. Wall, "Job selection in a network of autonomous UAVs for delivery of goods", pp. 1-10, Jul. 2017.

[3] Z. Huang, C. Chen and M. Pan, "Multiobjective UAV path planning for emergency information collection and transmission", IEEE Internet Things J., vol. 7, no. 8, pp. 6993-7009, Aug. 2020.

[4] M. Y. Arafat and S. Moh, “A survey on cluster-based routing protocols for unmanned aerial vehicle networks”, IEEE Access, vol. 7, pp. 498-516, 2018.

[5] M. Liu, J. Yang and G. Gui, “DSF-NOMA: UAV-assisted emergency communication technology in a heterogeneous Internet of Things”, IEEE Internet Things J., vol. 6, no. 3, pp. 5508-5519, Jun. 2019.

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

[7] Y. Lu, X. Zhucun, G.-S. Xia and L. Zhang, “A survey on vision-based UAV navigation”, Geo-spatial Inf. Sci., vol. 21, no. 1, pp. 1-12, Jan. 2018.

[8] M. E. Morocho-Cayamcela, H. Lee and W. Lim, “Machine learning for 5G/B5G mobile and wireless communications: Potential limitations and future directions”, IEEE Access, vol. 7, pp. 137184-137206, 2019.

[9] M. Sheraz, M. Ahmed, X. Hou, Y. Li, D. Jin, Z. Han, et al., “Artificial intelligence for wireless caching: Schemes performance and challenges”, IEEE Commun. Surveys Tuts., vol. 23, no. 1, pp. 631-661, 1st Quart. 2021.

[10] [10] G. Villarrubia, J. F. De Paz, P. Chamoso and F. D. la Prieta, “Artificial neural networks used in optimization problems”, Neurocomputing, vol. 272, pp. 10-16, Jan. 2018, [online] Available: https://www.sciencedirect.com/science/article/pii/S0925231217311116.

[11] B. Liu, L. Wang and M. Liu, “Lifelong federated reinforcement learning: A learning architecture for navigation in cloud robotic systems”, IEEE Robot. Autom. Lett., vol. 4, no. 4, pp. 4555-4562, Oct. 2019.

[12] C. Zhuge, Y. Cai and Z. Tang, “A novel dynamic obstacle avoidance algorithm based on collision time histogram”, Chin. J. Electron., vol. 26, no. 3, pp. 522-529, May 2017.

[13] [13] A. Vahidi and A. Eskandarian, “Research advances in intelligent collision avoidance and adaptive cruise control”, IEEE Trans. Intell. Transp. Syst., vol. 4, no. 3, pp. 143-153, Sep. 2003.

[14] Z. Liu, Y. Zhang, C. Yuan, L. Ciarletta and D. Theilliol, “Collision avoidance and path following control of unmanned aerial vehicle in hazardous environment”, J. Intell. Robot. Syst., vol. 95, no. 1, pp. 193-210, Jul. 2019.

[15] A. Mujumdar and R. Padhi, “Evolving philosophies on autonomous obstacle/collision avoidance of unmanned aerial vehicles”, J. Aerosp. Comput. Inf. Commun., vol. 8, no. 2, pp. 17-41, Feb. 2011.

[16] A. Foka and P. Trahanias, “Real-time hierarchical POMDPs for autonomous robot navigation”, Proc. IJCAI Workshop Reasoning Uncertainty Robot., pp. 1-8, 2005.

[17] R. Murray, “Recent research in cooperative control of multi-vehicle systems”, J. Dyn. Syst. Meas. Control, vol. 129, pp. 571-598, Sep. 2007.

[18] G. Ladd and G. Bland, “Non-military applications for small UAS platforms”, Proc. AIAA Infotech Aerosp. Conf., pp. 2046, Apr. 2009.

[19] L. He, P. Bai, X. Liang, J. Zhang and W. Wang, “Feedback formation control of UAV swarm with multiple implicit leaders”, Aerosp. Sci. Technol., vol. 72, pp. 327-334, Jan. 2018, [online] Available: http://www.sciencedirect.com/science/article/pii/S1270963816309816.

[20] S. S. Esfahlani, “Mixed reality and remote sensing application of unmanned aerial vehicle in fire and smoke detection”, J. Ind. Inf. Integr., vol. 15, pp. 42-49, Sep. 2019, [online] Available: http://www.sciencedirect.com/science/article/pii/S2452414X18300773.

[21] K. P. Valavanis, Unmanned Aircraft Systems: The Current State-of-the-Art, Cham, Switzerland: Springer, 2016.

[22] C. A. Wargo, G. C. Church, J. Glaneueski and M. Strout, “Unmanned aircraft systems (UAS) research and future analysis”, Proc. IEEE Aerosp. Conf., pp. 1-16, Mar. 2014.

[23] X. Wang, V. Yadav and S. N. Balakrishnan, “Cooperative UAV formation flying with obstacle/collision avoidance”, IEEE Trans. Control Syst. Technol., vol. 15, no. 4, pp. 672-679, Jul. 2007.

[24] S. Huang, R. S. H. Teo and K. K. Tan, “Collision avoidance of multi unmanned aerial vehicles: A review”, Annu. Rev. Control, vol. 48, pp. 147-164, 2019, [online] Available: http://www.sciencedirect.com/science/article/pii/S1367578819300598.

[25] J. Sun, J. Tang and S. Lao, “Collision Avoidance for Cooperative UAVs With Optimized Artificial Potential Field Algorithm,” in IEEE Access, vol. 5, pp. 18382-18390, 2017, doi: 10.1109/ACCESS.2017.2746752.\

[26] Lowe, Ryan, et al. “Multi-agent actor-critic for mixed cooperative-competitive environments.” Advances in neural information processing systems 30 (2017).

[27] McLachlan, Goeffrey J. “Mahalanobis distance.” Resonance 4.6 (1999): 20-26.