Autonomous and Collective Intelligence for UAV Swarm in Target Search Scenario

Luiz Giacomossi Jr., Flavio Souza, Raphael Gomes Cortes, Huascar Mirko Montecinos Cortez, Caue Ferreira, Cesar A. C. Marcondes, Denis S. Loubach, Elton F. Sbruzzi, Filipe A. N. Verri, Johnny C. Marques, Lourenço A. Pereira Jr, Marcos R. O. A. Maximo and Vitor V. Curtis.

2021 Latin American Robotics Symposium (LARS), 2021 Brazilian Symposium on Robotics (SBR), and 2021 Workshop on Robotics in Education (WRE) | 978-1-6654-0761-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/LARS/SBR/WRE54079.2021.9605450

***Abstract*—Unmanned Aerial Vehicle (UAV) swarm, also named drone swarm, has been the study object of many types of research due to its potential to improve applications such as monitoring, surveillance, and search missions. With several drones flying simultaneously, the challenge is to increase their level of automation and intelligence while avoiding collision, reducing communication level with these entities, and improving strategical organization to accomplish a specific task. In this sense, we propose a solution to coordinate a UAV swarm using bivariate potential fields with autonomous and distributed intelligence among drones for a cooperative target search application. Results have shown an improvement in the swarm effectiveness by reducing the number of UAVs blocked at local minima by using distributed decision-making methods, proving to be an effective approach to solve this frequent problem in potential fields.**

***Index Terms*—UAV Swarm, Target Search, potential fields, Swarm intelligence, Collective intelligence.**

I. Introduction

A swarm is called a group of ten or more homogeneous entities working cooperatively towards a common goal [1]. Swarm is a common and natural phenomenon among some groups of animals, such as bees, fishes, and birds, which through simple rules manage to move in space with coordination and cohesion.

The application of Unmanned Aerial Vehicles (UAV) in a swarm has been gaining attention and interest from researchers nowadays. This approach presents numerous challenges, with the capability to communicate among drones that make up the UAV swarm network being an important one [2]. The highest level of UAV swarm autonomy is the ability to perform a task coordinated among multiple UAVs without the intervention of a human operator [3].

Given a large number of drones, the control and autonomy of the UAV swarm are crucial for the success of a mission, i.e., similar to the animal behaviors, a UAV swarm has to cooperate to achieve a goal. Despite the various inherent challenges, the possible applications of a UAV swarm are numerous. Applications that use cameras or other types of remote sensing equipment are the most prominent, including photogrammetry, video surveillance, traffic monitoring, and search and rescue [4]. Another important application is the search for targets, which we address in this paper.

The authors are with the Aeronautics Institute of Technology – ITA, São José dos Campos, SP, Brazil. Corresponding authors: luiz.giacomossi@ga.ita.br, curtis@ita.br.

Search patterns with a UAV swarm are usually associated with some heuristic, since it is not possible to propose an optimal solution [5]. In this work, we develop a simulator to implement and test a set of proposed search heuristics. The dynamics of the autonomous UAVs is modeled with *steering force* method, for guiding the movement, associated with a potential field model to avoid collisions.

This work contributes by developing a novel swarm intelligence model, which also provides a heuristic to avoid the frequent problem of local minima in drone swarm applications that use the potential fields as a method to avoid collisions. Our proposed method is based on distributed decision-making to provide autonomy to UAVs, combined with the development of a collective intelligence module for the coordination of the swarm, with a shared discrete map search strategy for the search mission of a missing target (e.g., an individual).

This paper is organized as follows. In Section II we present the theoretical background. The problem is formulated in Section III. Later, in Section IV we describe our solution using collective and distributed heuristics. Then we proceed to the experiments, results, and discussion in Section V. Lastly, Section VI concludes and shares ideas for future work.

II. Theoretical Background

In this section, we present the main techniques found in the literature regarding the swarm approach.

A. Steering Force Model

A basic theory of autonomous behavior of vehicles is proposed by Shiffman [6], using the *steering force* dynamics for guiding an autonomous vehicle. The modeling is based on a simple principle of the sum of vectors that guide the drone to the desired position by applying a force F calculated by the difference of the desired velocity V_D vector by the current velocity vector V, as indicated in:

$$\mathbf{F}(t+1) = \mathbf{V}_D(t) - \mathbf{V}(t). \tag{1}$$

This force is then applied to the drone model, respecting a maximum threshold according to the model's limitations, and it is used to update the vehicle's velocity and position through numerical integration. This method is used as an attractive force to guide the UAVs to a desired coordinate due to its effectiveness, easy implementation, and fast computation.

B. Potential Fields for Collision Avoidance

A traditional technique for the interaction of multiple entities is the potential fields method [7]. In this method, obstacles exert repulsive forces ($|\overrightarrow{F_r}|$) on the agent, base on the distance d to the obstacles and a repulsion coefficient Q. The formula is derived from the electric potential field, which is given by

$$|\overrightarrow{F_r}| = \frac{Q}{d^2}. \quad (2)$$

An alternative implementation is proposed by Barnes [1], using the bivariate normal function as a potential field, where the potential function is applied to contain multiple drones within an elliptical area for a formation flight [1]. The normal bivariate function is

$$f(x,y) = e^{-\alpha(x-x_c)^2-\gamma(y-y_c)^2}, \quad (3)$$

where α and γ are constants to be tuned, $[x_c, y_c]^T$ is the center of the field and $[x,y]^T$ is a position on which the potential field has an influence. The x and y partial derivatives of (3) create a velocity field that is used for the movement of drones within the formation according to

$$\begin{bmatrix} \frac{\partial f(x,y)}{\partial x} \\ \frac{\partial f(x,y)}{\partial y} \end{bmatrix} = \begin{bmatrix} -2\alpha f(x,y) & 0 \\ 0 & -2\gamma f(x,y) \end{bmatrix} \begin{bmatrix} (x-x_c) \\ (y-y_c) \end{bmatrix}. \quad (4)$$

along the x and y axis, respectively. In the simulator, $\alpha = \gamma$ in order to generate a symmetrical field.

The normal bivariate function has the advantages of allowing easy tuning of the field's actuation region by the constants α and γ along the x and y axis, respectively. We can see in Fig. 1, where α and γ are equal, generating a symmetrical field in x and y. The maximum $f(x,y)$ value is unitary for any values of α and γ, which occurs when the entities are very close, facilitating the adjustment of the maximum force value.

The disadvantage of potential fields technique is the possibility of local minima in the scenario [8], which can make a UAV stationary by the cancellation of the repulsion and attraction forces of the various objects present in a scenario.

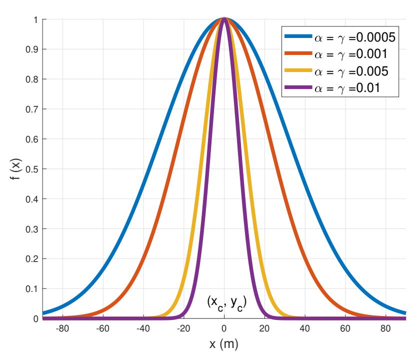

Fig. 1. One-dimensional normal bivariate function.

C. Search Algorithms and Swarm Intelligence

Using UAVs for search operations is not a new concept, but using systems of several UAVs is a subject still not fully explored and with very few works discussing it [4]. Aunon and Cruz [5] performed a study of heuristic algorithms for the search task using UAV swarms and concluded that the search by line and with a displacement to unvisited cells presented the greatest simplicity in the implementation and the most efficient algorithm for target search among the studied methods. Thus, this is the method that we explore in this paper.

In the line search method proposed by Aunon and Cruz [5], the search area is divided into cells, and each drone is designated to start the search in a specific line, as illustrated in Fig. 2. The drone navigates in search of the target to the end of the initial line. At the end of its respective line (and considering the non-localization of the target), the UAV restarts the search in a new line not yet visited by another UAV in the swarm. There is also a communication effort to be considered for the UAV swarm network, as UAVs must share with the swarm which lines and cells were previously visited. Therefore, they need to have a world model, such as a discrete map, which must be continuously shared by all agents.

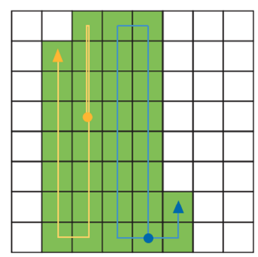
Fig. 2. Line search method with two agents, yellow and blue points [5].

III. Problem Formulation

In this section, we present the problem formulation, the description of the scenario, and the research question.

A. Scenario

As an application of the search problem, we propose a scenario where a UAV swarm is applied in a search mission to find a missing individual in a forest region of about $60,800m^2$. The scenario area is divided into $(38 \times 16\ m)$ squares, where each square is equivalent to $100\ m^2$ $(10 \times 10\ m)$. N is the number of UAVs performing the search mission in this area. Considering a vision system on each UAV capable of identifying a target at less than 20 meters, we restrict the UAVs to flying at heights less than the treetops, allowing the UAV to successfully identify the target using its vision system.

We assume as the swarm's primary hypothesis that the UAVs do not have previous information about the position of the missing target. Also, the UAVs must not collide with each other and avoid surrounding obstacles, represented as trees. Drones are considered to have a common UAV swarm network in which they share mission information such as positions of all mission components and target location when identified. The target is considered found when a drone is less than two squares away (less than 20 meters), illustrated in Fig. 3 as a circle around the target.

The simulator was developed in Python with the aid of the *Pygame* library for visual representation, as seen in Fig. 3. Our simulator aims to facilitate future simulations and different scenarios through an easy to use and modify open-source code. Interested readers can watch our simulator presentation video[2] and also find more implementation details in our repository on Github[3].

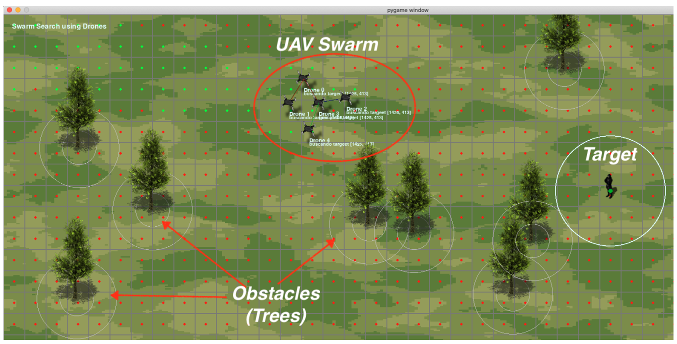

Fig. 3. Simulation view of the search scenario using only 8 obstacles.

B. Research Question

The proposed mission objective is to identify the location of a previously unknown target. Thus, considering the swarm search and intelligence heuristic implemented, we evaluate how the search time of an unknown target is affected by the number of drones and obstacles. It is considered a reference for the shortest time to locate the target if the drones known the target and move directly to its position.

In addition, the influence of the proposed heuristics and implemented swarm intelligence is investigated against metrics of interest such as mission time, i.e., when all drones reach the unknown target location, and the amount of occurrence of local minimum when the drones cannot move.

IV. Distributed and Cooperative Decision-Making

Our decision-making can be divided into two categories: autonomous intelligence, referring to a single UAV, and collective swarm intelligence, referring to the group. Therefore, decisions are decentralized to each UAV, and cooperative, as the group must cooperate to attain the mission. The implemented intelligence modules are described in Table I.

A. Autonomous Intelligence

In the target search module, the UAVs are equally distributed among the scenario lines according to the number of drones used, and each drone begins the search for the target autonomously, by searching in a line on the neighboring cells. Thus, the drones perform a horizontal sweep across the scene.

The behavior module was developed to control the autonomous behavior of the drones, consisting of a finite state machine (FSM), with each state representing a behavior, only

[2]https://www.youtube.com/watch?v=l07YPjrnLNw

[3]https://github.com/luizgiacomossi/Search_Drone_Swarms/tree/grid_search

TABLE I
Proposed Heuristics for Swarm Intelligence

Swarm Intelligence	Module	Description
Autonomous Intelligence	Target Search	Target search is performed on a discrete map.
	Behavior	Intelligent drone behavior through a finite state machine.
	Collision Avoidance	Potential fields for collision avoidance between drones and obstacles.
Collective Intelligence	Swarm Search	Target search in regions mapped as not visited using a common map.
	Blockage Prevention	Prevents drones from getting stuck in local minimums present in the scenario.
	Mission Closure	When a drone finds the target the coordinates are shared and all drones move to this position.

one state is executed at a time, with events leading to behavior changes [9]. Our FSM for a UAV can be seen in Fig. 4, structured with two main states: *SearchState* and *SeekState*. There are also two auxiliary states: *GoToClosestDrone* and *RandomTarget*, used to avoid the blockage by local minimums. Note that there is a FSM for each drone in the swarm.

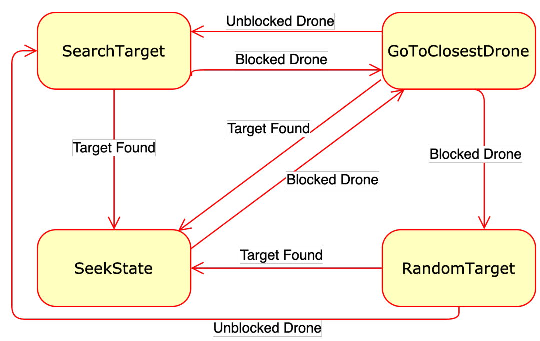

Fig. 4. Finite State Machine controller the drone's behavior.

By default, the drone is in the *SearchTarget* state, which represents the search behavior for the missing target. If the target is found, by this UAV or another agent in the swarm, the behavior will change to *SeekState*, in which the target's coordinates are now found and are transmitted to all agents in the swarm, then the drone stops the search and heads to those coordinates directly. If the drone gets blocked in any region of the map, the FSM identifies that the drone is not moving and transitions to the *GoToClosestDrone* behavior, in which the drone heads to the closest agent coordinates as a first heuristic to solve the situation. As a final solution, if the drone is still blocked, the agent transition to *RandomTarget* state which defines random targets on the map until it can unlock itself. When the agent is unblocked the FSM returns to its default search state. Note that the drone can transit to the *SeekState* state from any state.

As autonomous intelligence, there is also the module to avoid collisions. The potential field technique, using the normal bivariate function, was applied to calculate repulsive forces among drones and obstacles, efficiently solving this problem. By experimentation, a value of $\alpha = \gamma = 0.0001$

was found to be efficient to the UAVs field force and $\alpha = \gamma = 0.005$ for the collision avoidance with obstacles.

Thus, based on the distance between drones ($\overrightarrow{d}_{i,k}$) and in the distance between the drone and the obstacles ($\overrightarrow{d}_{i,j}$ and $\overrightarrow{d}_{k,j}$), a resulting force (attraction force using the steering force method and repulsion's using potential fields) is applied to the drone, resulting in movement to avoid collision. In Fig. 5 we see a graphics representation of how it works, we see two drones i and k, and one obstacle$j$, as force peaks (repulsion forces), note that the influence of the field depends on the distances between these elements.

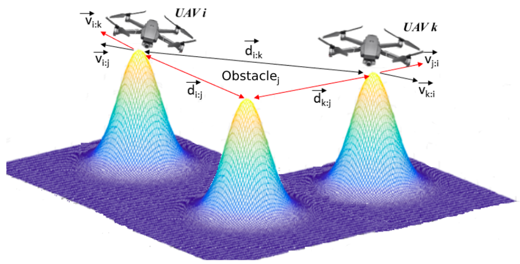

Fig. 5. Interaction of potential fields.

In Fig. 6 we see the technique being executed in simulation, the trajectory of each drone is drawn on the map, showing that the drones avoid collision between agents and with obstacles. Note that the area of action of the potential field is highlighted by the circumference around the tree and drones in gray.

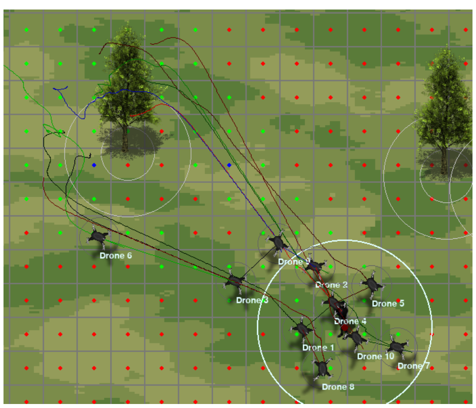

Fig. 6. Drones avoiding collision during simulation execution.

B. Collective Intelligence

In the collective intelligence module, we have the swarm search module, responsible for selecting unvisited cells of the shared map. Each cell $h(w, l)$ have two possible states, indicated as

$$h(w, l) = \begin{cases} 1, & \text{if visited,} \\ 0, & \text{Otherwise,} \end{cases} \quad (5)$$

where w varies from 1 to 38 and l varies from 1 to 16, according to the discrete area of the scenario. The default state is unvisited, when a cell is visited, the agent informs the swarm and the cell state is changed.

Fig. 7 demonstrates the marking on the map of the visited cells, with a green center dot, and the cells not visited by any member of the swarm, in red. This map is collectively built and shared with all agents. The map is collectively updated at every new cell visited and it is constantly being shared, so all agent have the same map during the execution.

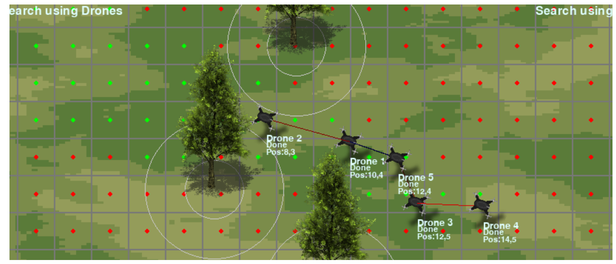

Fig. 7. Cells visited in green and not visited in red.

The mission begins with the UAVs being equally distributed in the rows of the map, as seen in Fig. 8. Then, the search for unvisited cells is performed, with the agent selecting an unvisited cell among the eight neighboring cells of its location (eight connected) to proceed.

Fig. 8. Mission begins, UAVs marked in yellow, distributed in the map.

If all eight neighboring cells were already visited, the algorithm chooses a random not visited cell on the map as the next destination. Fig. 9 illustrates this case, where a UAV with 8 already visited cells is seen in the bottom dashed rectangle. The agent then selects a random cell not visited on the map to continue (yellow arrows).

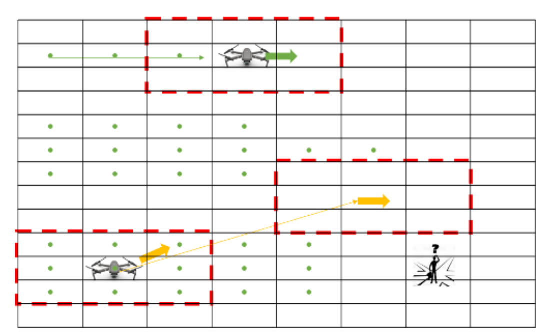

Fig. 9. Implementation of the unvisited cell search.

The collective intelligence also implements heuristics to avoid the potential field's local minima, called *Stationary State* by the drone, which prevents the drone from being blocked in a given location in the scenario. In this sense, if the UAV identifies that it can no longer move because it is stuck in a local minimum i.e enters the *Stationary State*, it will search in the UAV swarm network for the nearest UAV and try to head to its coordinates, as seen in Fig. 10. If the *Stationary State* UAV still can not move, after this attempt, this agent does a new search for the second closest drone, and tries to move towards it. Finally, if the drone remains blocked, the agent chooses a random cell on the map and repeats this procedure until it leaves *Stationary State*, returning to the default *SearchTarget* state.

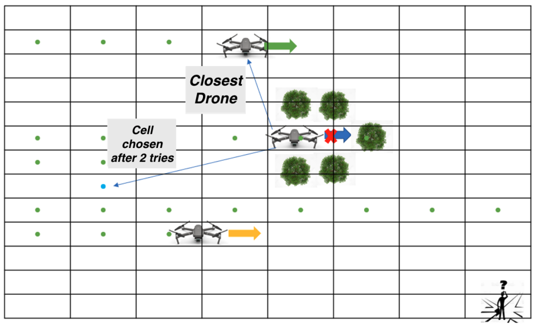

Fig. 10. Drone stuck in obstacle looking for the nearest drone.

Once the target is located by a drone i.e. distance less than or equal to two squares from the target, all UAVs in the swarm will transit the mission closure phase. In this phase, the coordinates of the located target are shared in the swarm network by the UAV that located the target, so that all drones must move to this coordinates, as illustrated in Fig. 11.

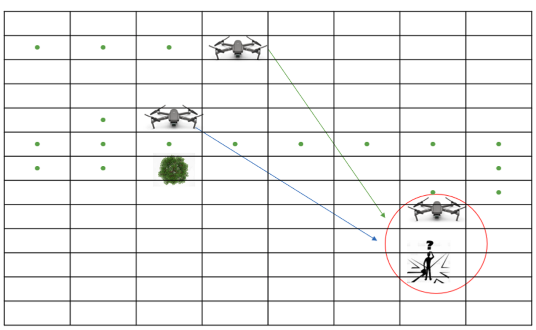

Fig. 11. Mission closure phase, target is located by one UAV in the swarm.

The simulation with the swarm in mission closure phase can be seen in Fig. 12, we observe that the target location was successfully shared in the swarm network, with all UAVs successfully moving to this coordinate, ending the mission. Note that the swarm is in formation around the target with cohesion and without collisions among UAVs.

V. Results and Discussions

In this section, we describe the experiments and their results.

A. Evaluation Planning

The evaluation is based on the computational performance evaluation methodology, proposed by Jain [10]. The Table II

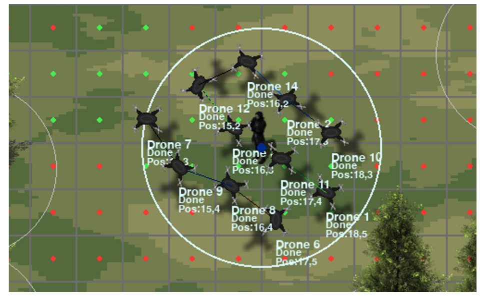

Fig. 12. Target is located and a swarm is formed around him.

presents the factors and levels considered for the evaluation. We applied a complete factorial design for the experiments, with 27 possible combinations for ten repetitions each, totaling 270 executions.

TABLE II
Experiments description.

Factors	Drones	Obstacles	Search Strategy
Level 1	5, 10, 15	10, 15, 30	Target location previously known and without heuristics - *TNH*.
Level 2	5, 10, 15	10, 15, 30	No knowledge of target location and without blockage heuristics - *NTNH*.
Level 3	5, 10, 15	10, 15, 30	No knowledge of target location and using blockage heuristics - *NTH*.

In this evaluation, the metrics observed are: mission completion time and and the number of observed local minima. The mission is considered completed when all drones form a swarm around the located target, see Fig. 12.

At level 1, we run the experiments with the coordinates of the target previously known, using this result as reference, as it can be considered as the best possible case, with the UAVs heading directly to the target.

B. Comparison of mission competition time

in Fig. 13 we have the average time of mission completion bar chart with a confidence interval of 95%. For all cases, the strategies in level 2 and 3 presented a longer mission completion time compared to the level 1 reference. This result is expected, since there is no guarantee of optimal time with the use of heuristics for search. Therefore, we note that when the swarm uses our search strategy, there is a tendency for them to spread across the area, with some UAVs searching in regions far from the target location or in complex situations, with several obstacles on the way to the target, causing this increased of mission time. Also, statistically, there are no differences in mission times between level 2 and level 3 strategies, the latter with agents using our blockage prevention heuristic.

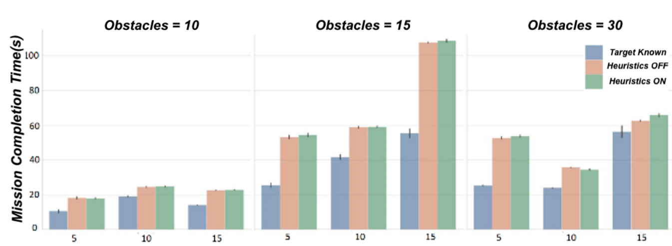

Fig. 13. Mission completion time by the number of drones used.

Therefore, we note that when the swarm uses our search strategy, there is a tendency for them to spread across the area, with some UAVs searching in regions far from the target location or in complex situations, with several obstacles on the way to the target, causing this increased of mission time. Also, note that, statistically, there are no differences in mission times between level 2 and level 3 strategies, the latter with agents using our blockage prevention heuristic.

It is interesting to note that with 15 obstacles there are two cases, with 5 and 15 drones, where there was a large increase in time with the heuristic on and off. This can be explained by the complexity of the simulated scenario, counter-intuitively from fewer obstacles, you can get a more difficult search scenarios, with the target being found in regions with very close obstacles, creating big barriers.

C. Comparison of local minima quantities

In Fig. 14 we present the average amount of local minima chart bar. Notice that, with the increase in obstacles and amount of UAVs, there is a tendency for the increase in the average amount of local minima. However, when the swarm is using our search heuristics, the amount of local minima is considerably reduced, as seen in the level 2 and 3 strategies. Note that in the reference level 1 case, the UAVs tended to get more blocked, this is justifiable as the UAVs are heading to the target directly in swarm formation, thus having little space to maneuver. The force field of other UAVs in the surroundings also can influence negatively, aggravating the amount of blocked drones.

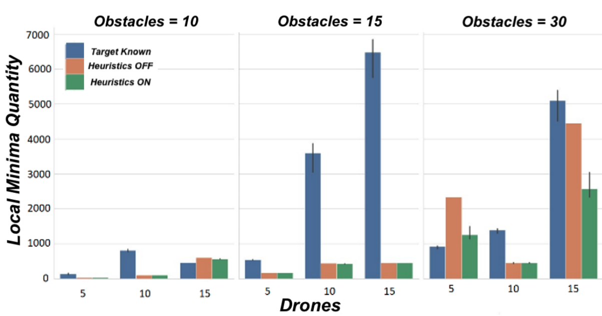

Fig. 14. Local minimums observed in each tested scenarios.

This metric demonstrates that when our heuristics is applied. it significantly reduces the number of UAVs blocked, when there is a significant amount of local minima in the scenario. By using our blockage prevention heuristics, we note that adding more UAVs to the swarm increases the probability of a nearby neighboring drone helping a blocked UAV, so fewer drones got blocked, this can be seen in the experiments with 10 and 15 UAVs using 30 obstacles.

VI. Conclusions and Future Work

In this paper, we study the use of UAV swarms in the context of target search for an individual of unknown location, in a region with trees. A UAV swarm simulator was developed to evaluate this study. We developed intelligent search strategies based on autonomous and collective decision-making for the UAVs. The UAVs were simulated with individual autonomous intelligence through a FSM, which controls the behaviors in decentralized manner. We also developed a collective intelligence strategy using a discrete map of the region, collectively built and shared, with the UAVs working as a team by sharing information for the search coordination and distributing the search task. We also contributed by creating a blockage prevention heuristic, to reduce the number of UAVs blocked In local minima.

After the performance evaluation, varying the number of obstacles and UAVs in the swarm, we observed that our swarm intelligence proved to be effective, also reduced the number of UAVs blocked in local minima, showing signs of being an effective tool for solving this frequent problem in the potential fields technique. Our results showed mission completion times proximate to reference time i.e level 1 strategy, which is a promising result. Note that there is no guarantee of optimal time when using heuristics for search. Thus, it was verified that forms of collective swarm intelligence should be evaluated to improve the target search mission, its usage and improvement may bring significant advantages in such contexts.

Future research should consider applying optimization algorithms to the decision-making parameters to improve the results. Another suggestion is the evaluation of additional effectiveness metrics, such as redundant paths and revisited cells, that UAVs may occasionally revisit. In order to have deeper evaluations, a simulated scenario with realistic physics is suggested. Alternative ideas to avoid blocking at local minima can also be further evolved.

Acknowledgment

Luiz Giacomossi Jr. acknowledges Embraer S.A for his scholarship. The authors are also grateful to Embraer for supporting this research.

References

[1] L. Barnes, *A Potential Field Based Formation Control Methodology for Robot Swarms*. University of South Florida, 2018.
[2] M. Campion, P. Ranganathan, and S. Faruque, “Uav swarm communication and control architectures: a review,” *Journal of Unmanned Vehicle Systems*, vol. 7, no. 2, pp. 93–106, 2019. [Online]. Available: https://doi.org/10.1139/juvs-2018-0009
[3] H.-M. Huang and E. Messina, “Autonomy levels for unmanned systems (alfus) frameworkvolume ii: Framework models initial version,” 2007.
[4] G. Skorobogatov, C. Barrado, and E. Salamí, “Multiple uav systems: A survey,” *Unmanned Systems*, vol. 08, no. 02, pp. 149–169, 2020. [Online]. Available: https://doi.org/10.1142/S2301385020500090
[5] A. Aunon, P. Cruz, *Comparison of Heuristic Algorithms in Discrete Search and Surveillance Tasks Using Aerial Swarms.* Swarm Robotics: MDPI, 2018.
[6] D. Shiffman, *The Nature of Code*, 5th ed. Shiffman, 2012. [Online]. Available: https://natureofcode.com
[7] Y. K. Hwang, N. Ahuja *et al.*, “A potential field approach to path planning.” *IEEE Transactions on Robotics and Automation*, vol. 8, no. 1, pp. 23–32, 1992.
[8] M. Silva, “Campos potenciais modificados aplicados ao controle de robôs em ambiente tridimensionais,” *Universidade de São Paulo*, 2019.
[9] M. Colledanchise, “Behavior trees in robotics and ai,” *KTH Royal Institute of Technology*, 2018.
[10] R. Jain, *The Art of Computer System Performance Analysis: Techniques for Experimental Design, Measurement, Simulation and Modeling*. John Wiley & Sons, 1991.