Antônio Lucas Sousa Aguiar[1], Vandilberto Pereira Pinto[2], Lígia Maria Carvalho Sousa[2], José Lucas Da Silva Pinheiro[2], José Cleilton do Nascimento Sousa[2]

Federal University of Ceará(1), University of International Integration of Afro-Brazilian Lusophony(2)

doi:10.15199/48.2024.09.43

Route planning for multiple unmanned aerial vehicles

Abstract. *This study addresses efficient task assignment for collaborative systems, with a focus on route planning for multiple Unmanned Aerial Vehicles (UAVs). Using the Ant Colony Optimization algorithm and the A* algorithm for obstacle avoidance, the results show that the proposed method allows route planning with acceptable computational time, providing guidance on the optimal number of UAVs in a mission.*

Streszczenie. *Niniejsze badanie dotyczy efektywnego przydzielania zadań dla systemów współpracujących, ze szczególnym uwzględnieniem planowania trasy dla wielu bezzałogowych statków powietrznych (UAV). Wykorzystując algorytm optymalizacji kolonii mrówek i algorytm A* do omijania przeszkód, wyniki pokazują, że proponowana metoda umożliwia planowanie trasy w akceptowalnym czasie obliczeniowym, zapewniając wskazówki dotyczące optymalnej liczby UAV w misji.**(Planowanie trasy dla wielu bezzałogowych statków powietrznych)***

Keywords: Multiple Traveling Salesmen, UAVs, Route planning, Ant colony, Integer Linear Programming,Circumvent Obstacles.
Słowa kluczowe: Wielu podróżujących sprzedawców, UAV, planowanie trasy, kolonia mrówek, programowanie liniowe, omijanie przeszkód.

Introduction

The cargo transportation sector has shown growing interest and investment in Unmanned Aerial Vehicle (UAV) applications. This interest has been driven by the growth of e-commerce, which has sustained the demand for innovative logistics solutions. UAVs have the ability to take off and land safely in areas close to buildings and people, which significantly improves the quality of transportation services in congested or hard-to-reach regions. This promising technology has the potential to revolutionize the way goods are transported, offering efficiency and agility [4].

In line with the aforementioned advantages, the literature highlights the importance of efficient coordination, especially in complex missions that involve optimizing trajectories for multiple collaborative robots, as evidenced by specialized literature such as [7, 8]. Task allocation in this context requires not only the consideration of energy, capacity and obstacle avoidance constraints, but also the effective integration of robot navigation with robot sensing to avoid collisions and achieve efficient route planning.

In this way, [14] developed an algorithm based on Particle Swarm Optimization (PSO) and Genetic Algorithms (GA) to calculate routes for collaborative robots with energy constraints, using as a model a variant of the team orientation problem that adds reward collection features when passing through waypoints. Similarly, [5] used GA to solve the routing problem for multiple UAVs. The authors observed satisfactory results and reasonable execution times when applying their model to solve a problem involving a fleet of UAVs. In [13], the authors proposed an optimization model using an Integer Linear Programming (ILP) approach to solve a multiple mission planning problem, considering the influence of the wind field and mission duration constraints.

This work aims to use a mathematical model with an emphasis on Ant Colony Optimization (ACO). ACO, inspired by the cooperative behavior of ants, is an optimization technique that stands out in combinatorial optimization problems. With this in mind, the aim of this work is to use a mathematical model of the route calculation problem for multiple collaborative UAVs, considering the constraints of Multiple Traveling Salesmen (MTS) by applying obstacle avoidance. With this problem, this work created Research Questions (RPs) that served as a guide for the development of the research:

- **RP1.** Given the growing demand for applications in various areas, such as agriculture, environmental monitoring and logistics. Is it possible to develop an efficient algorithm for route planning for multiple UAVs?
- **RP2.** Can a hybrid algorithm using ACO and A* for route planning and obstacle avoidance provide results with a satisfactory confidence interval?
- **RP3.** In a fictitious map, what is the difference in time to carry out a mission with different numbers of UAVs?

In light of this, the main contributions of this work are:

- Proposal of a new hybrid heuristic algorithm for the UAV routing problem, considering the following points:
  - application of the multi-objective ACO algorithm to route planning for multiple UAVs;
  - integration of ACO with the A* algorithm for the avoidance of fixed obstacles;
  - analysis of routes with different numbers of UAVs based on time and distance.

Problem Formulation

To formulate the problem, given a group of N_v UAVs $\mathcal{V} = \{1, 2, 3, ..., N_v\}$, and a set of N_t waypoints that can be explored, represented by $\mathcal{T} = \{1, 2, 3, , N_t\}$, where 1 and N_t denote the start and end point, respectively, the objective of the task assignment challenge is to find a conflict-free configuration in which the waypoints are traversed by the UAVs and minimize the overall costs. Where the set $P = \{P_1, P_2, P_3, ..., P_{N_t}\}$ indicates the routes taken by each vehicle. To do this, it is assumed that each waypoint is visited by a single UAV, and that each UAV can visit multiple points in a specific order. The UAVs must collaborate with each other, which means that each waypoint needs to be assigned to at most one UAV [16].

In addition, the sequence of execution of their activities for each vehicle must be optimized. Considering the activities carried out by a vehicle, the cost can vary significantly if the activities are carried out in different orders. Therefore, the sequence of execution of the activities must be optimized to minimize the total cost. The cost of each vehicle is calculated as the distance traveled to carry out the assigned activities, while the total cost is determined by adding up the individual costs of the UAVs.

The point of departure and arrival can be the location of a base station, also known as a depot, shed or store. The allocation of tasks is formulated as a multi-objective MTS, in which the objectives are to minimize the total cost traveled to execute the mission assigned to the UAVs [16, 14]. In this way, the optimization problem is represented by (1) [16], where x_{ijk} takes on the value 1 if the UAV i visits the way-point k, and takes on the value 0 otherwise. The variable $c_{jk}^{(i)}$ represents the cost of UAV i going from waypoint j to maximum partial cost of the vehicles.

$$\min F = \{f_1(P), f_2(P)\} \tag{1}$$

$$f_1(P) = \sum_{i=1}^{N_v}\sum_{j=1}^{N_t}\sum_{k=1}^{N_t} x_{ijk} c_{jk}^{(i)}(P_i) \tag{2}$$

$$f_2(P) = \max_{1\le i\le N_v} \sum_{j=1}^{N_t}\sum_{k=1}^{N_t} x_{ijk} c_{jk}^{(i)}(P_i) \tag{3}$$

Subject to the restrictions [16, 14]:

$$\sum_{i=1}^{N_v}\sum_{j=2}^{N_t-1} x_{ijk} = 1, \forall k \tag{4}$$

$$\sum_{i=1}^{N_v}\sum_{k=2}^{N_t-1} x_{ijk} = 1, \forall j \tag{5}$$

$$\sum_{i=1}^{N_v}\sum_{j=1}^{N_t} x_{ij1} = N_v \tag{6}$$

$$\sum_{i=1}^{N_v}\sum_{k=1}^{N_t} x_{ink} = N_v, n = N_t \tag{7}$$

$$\sum_{i=1}^{N_v}\sum_{j=1}^{N_t}\sum_{k=1}^{N_t} x_{ijk} = N_t \tag{8}$$

$$f_1(P) \ge \lambda f_2(P) \tag{9}$$

The constraints (4) and (5) are indicating that each waypoint, except the depot, can be visited at most once. The constraints (6) and (7) guarantee that all vehicles start their routes at the initial depot and return to the final depot after finishing their respective tasks. (8) is the constraint that ensures that all waypoints can be visited [16, 14]. And (9) indicates the workload balance in route planning [16].

Multi-objective Ant Colony System

The ACO was created by Dorigo in the 1990s with the aim of solving the traveling salesman problem [10]. The idea is inspired by the behavior of ants, which leave a trail of pheromones while searching for food. Subsequent ants are attracted to this trail and are more likely to follow suit [2]. Over time, the trail is reinforced by the ants that find shorter paths, leading to an optimal solution, where a group of ants leave an anthill in search of food [9]. To deal with multiple objectives and vehicles, the optimization process involves three main steps: basic settings and initialization, solution construction and pheromone updating, which are described below:

Step 1: basic settings and initialization

In this stage, N_g groups of ants are employed, each group builds a solution at each iteration, and N_v ants are employed in each group, representing the UAVs used. For a link between the waypoints r and s, there are two pheromones $\tau_{rs}^{(1)}$ and $\tau_{rs}^{(2)}$, assigned the objective functions (2) and (3).

To initialize the algorithm and obtain the pheromone matrix, an ant is selected at random and the tasks closest to it are assigned. The procedure is repeated until all the tasks have been assigned to the ants in a group. The heuristic information between points r and s for ant i in a group is initialized as the inverse of the cost, i.e. $\eta_{rs} = 1/c_{rs}$. The pheromone matrix is assigned as $\tau_0^{(1)} = 1/f_1(P^{(0)})$ and $\tau_0^{(2)} = 1/N_v/f_2(P^{(0)})$ [16].

Step 2: solution construction

Tasks are added iteratively one by one; an ant must be selected first to add an unassigned task. The partial cost of each ant in an ant group can be obtained at each stage of the solution construction phase because tasks are assigned one by one [16, 11]. The ant i that will be added to the next unassigned task is selected based on the criteria in (10). Where $\hat{P}_i$ is a partially constructed solution ant i in the group, q is a random number between [0, 1], q_0 is the probability that the ant with minimum partial cost is selected, while q_1 is the probability that the ant with maximum partial cost is selected. And Int(1,N_v) is a random integer between 1 and N_v, and $C(\hat{P}_i)$ the partial cost of the group, is defined by (11).

$$\hat{I} = \begin{cases} \text{arg} \min\limits_{1\le i\le N_v} C(\hat{P}_i), \text{ if } q < q_0 \\ \text{arg} \max\limits_{1\le i\le N_v} C(\hat{P}_i), \text{ if } q > 1 - q_1 \\ \text{Int}(1,N_v), \text{ otherwise} \end{cases} \tag{10}$$

$$C(\hat{P}_i) = \sum_{j=1}^{N_t}\sum_{k=1}^{N_t} c_{jk}^{(i)} x_{ijk} \tag{11}$$

When the ant i in a group is chosen, r becomes the last task in $\hat{P}_i$, the task s is selected according to (12). Where C is the set containing the tasks that have not yet been assigned, α and β are parameters, used to determine the relative importance of the pheromone and the info. heuristic, respectively, p is a random number between $[0, 1]$, and p_0 is the probability of choosing the path with maximum η_{rs}. The term $[\prod_{k=1}^{2}[\tau_{rs}^{(k)}]^{\alpha k}] \cdot [\eta_{rs}^{(i)}]^{\beta}]$ indicates the task's decision information, used to decide the next task to be assigned. If $p \le p_0$, the task with the maximum decision information is assigned. And S is a random variable selected [11].

$$s = \begin{cases} \text{arg} \max\limits_{u\in C}[\prod\limits_{k=1}^{2}[\tau_{rs}^{(k)}]^{\alpha k}] \cdot [\eta_{rs}^{(i)}]^{\beta}], \text{ if } p < p_0 \\ \text{S, otherwise} \end{cases} \tag{12}$$

Once the task has been selected, it is assigned to ant i and removed from the unassigned set C. The procedures are repeated until all the tasks have been assigned. At the end, the final depot is assigned to each ant to return to.

Step 3: updating the pheromones

Each time a task r in a group selects s as the next task, the local pheromone update rule is applied to the link between task r and s. The pheromone level is updated according to (13), where the parameter ρ which takes on values between $[0, 1]$ and is used to balance the newly added pheromone. The global pheromone update, on the other hand, takes place after all the groups have built their solutions during each iteration. The pheromone matrix is updated based on (14), where $\Delta\tau_{rs}^{(k)}$ is the newly added pheromone in the link, being defined in (15), where $n_k = 1$ if $k = 1$ and $n_k = N_v$ if $k = 2$ [15, 9].

$$\tau_{rs}^{(k)} = (1-\rho)\cdot\tau_{rs}^{(k)} + \rho\cdot\tau_0^{(k)}, k = 1,2 \tag{13}$$

$$\tau_{rs}^{(k)} = (1-\rho)\cdot\tau_{rs}^{(k)} + \rho\cdot\Delta\tau_{rs}^{(k)}, k = 1,2 \tag{14}$$

$$\Delta\tau_{rs}^{(k)} = \begin{cases} \tau_0^{(k)} + \displaystyle\sum_{l=1}^{|P|} \frac{1}{n_k \cdot f_k(P^{(l)})}, \text{ se } p < p_0 \\ \qquad\qquad\qquad\qquad \tau_0^{(k)}, \text{ otherwise} \end{cases} \tag{15}$$

Obstacle Avoidance With A*

In the context of multi-robot collaborative task allocation and route planning, the problem of finding a path from a point to a specific destination while avoiding collisions with obstacles in the environment is still a significant challenge [1]. This problem is known as the shortest path problem with collision constraints. The goal is to find a route that minimizes the distance travelled by the UAV while avoiding collisions with obstacles in the environment. There are several approaches to solving this problem, such as graph-based algorithms, heuristic search and optimization algorithms [6].

Heuristic search algorithms, such as the A* algorithm, are able to find routes that avoid collisions, but may not guarantee the optimality of the solution [17]. It uses a heuristic function to estimate the cost of the shortest path from the current node to the destination node and then expands the node with the lowest estimated cost. This allows the algorithm to quickly find a path close to the optimum, avoiding the expansion of many unnecessary nodes. However, the result is not always the shortest possible path [12].

In this problem, the state space is represented as a matrix, where each cell of this matrix is a state (node), as illustrated in Fig. 1. The rules that can be applied are the 8 directions that the robot can move from the cell, according to the obstacles [3].Based on the problem description, the idea is to perform a search in this state space to find the shortest route to the destination. The robot's position must be known.

The formula for the traditional algorithm is shown in (16), where $f(n)$ indicates the estimate from the start point to the destination point, $g(n)$ the actual cost from the start point to point n and $h(n)$ the estimated value from node n to the destination node.

$$f(n) = g(n) + h(n) \tag{16}$$

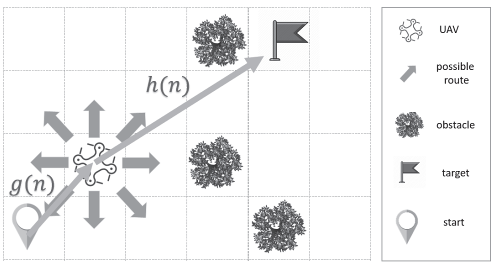

Fig. 1. Representation of the possible paths of A*

Methodology

Using the concepts discussed in the previous section, the UAVs will follow the route defined by the A* algorithm, visiting each waypoint defined by the ACO algorithm, as shown in Fig. 2. To do this, a fictitious map with obstacles was created for the experiments, as shown in Fig. 3, since there are currently no reference examples for MTS to be used in the performance evaluation of the approaches for comparison with the optimal solution.

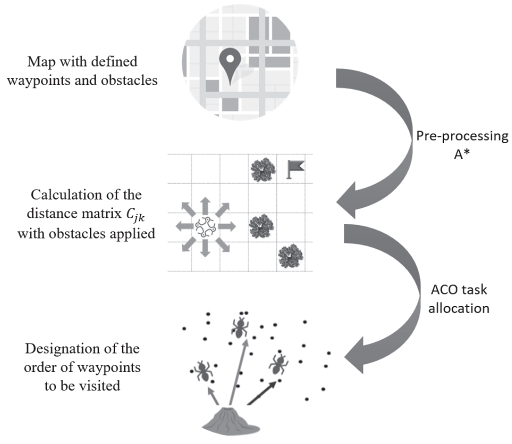

Fig. 2. Steps for allocating tasks

Table 1 presents the position of the waypoints on the x and y axes, where W_1 indicates the start depot and W_{16} indicates the final depot. In addition, the layout of the waypoints is sparse. Table 2 presents the coordinates of the predefined obstacles in the simulations, since they have an associated length, the table illustrates the initial and final position on the x axis, being x_i and x_f respectively, and their position on the y axis, showing the linear barrier effect.

Table 1. Location (x_i, y_i) of waypoints W_i

W_i	x_i	y_i	W_i	x_i	y_i	W_i	x_i	y_i	W_i	x_i	y_i
1	1	1	7	1	4	9	11	7	13	7	5
2	6	1	8	3	4	10	5	3	14	8	1
3	8	9	5	13	7	11	4	6	15	12	5
4	4	8	6	2	9	12	10	2	16	13	9

Table 2. Location (x_i, x_f, y) of obstacles

Ob_i	x_i	x_f	y	Ob_i	x_i	x_f	y	Ob_i	x_i	x_f	y
Ob_1	9	11	4	Ob_4	5	6	8	Ob_7	6	7	4
Ob_2	5	5	6	Ob_5	3	4	9	Ob_8	12	13	8
Ob_3	9	10	8	Ob_6	2	4	7				

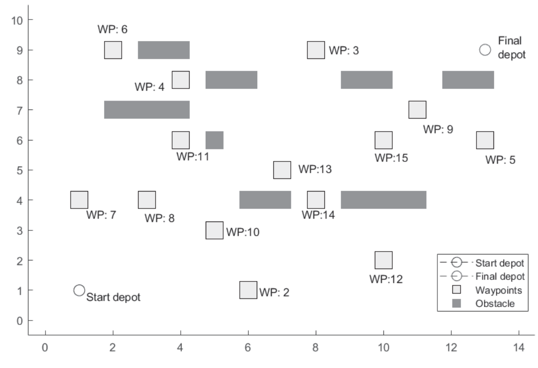

Fig. 3. Scenario used for the experiments

The UAVs are also set at a constant speed of $20m/s$ at a height of $15m$ and the unit of distance between the waypoints is in meters. The figures from the experiments show the displacements of the UAVs on the x and y axes, with behavior similar to that of a particle.

The ACO used the parameters shown in Table 3 with λ in (9) equal $N_v/2$. The experiments were carried out in order to analyze the behavior in different situations. As this is a heuristic method, 20 simulations were carried out with 10^3 iterations in each experiment, since the results of heuristic methods can be different even with the same input between different runs, so that the mean, standard deviation, minimum and maximum values of the performance metrics and the confidence interval of the experiments could be verified.

Table 3. Parameters used

Param.	Description	Value
N_g	Number of ant groups	24
q_0	Prob. of choosing the ant with min. $C(\hat{P}_i)$	0.9
q_1	Prob. of choosing the ant with max. $C(\hat{P}_i)$	0.05
α	Degree of importance of the pheromone	1
β	Degree of importance of η_{rs}	2
p_0	Prob. of choosing path with max. η_{rs}	0.9
ρ	Pheromone evaporation rate	0.5

Results

In this section, we present the simulations carried out to illustrate the application of the proposed models to the problem of collaborative UAV route planning. The simulations of the experiments were carried out in Matlab® software on a computer with an Intel(R) Core(TM) i7-7500U CPU @2.70GHz-2.90 GHz, 16GB of RAM.

The simulations varied in the number of UAVs in each instance, with experiments ranging from 2 to 8 UAVs. In all the experiments, the collaborative UAVs start the tasks from the initial depot and end the mission by visiting the final depot. The Table 4 illustrate the results and comparisons of the experiments. The results show that an increase in the number of UAVs increases the maximum distance traveled; on the other hand, the maximum mission time drops with this increase. There is also a subtle increase in CPU time between 2 and 6 UAVs, with an abrupt increase with 8 UAVs.

Table 4. Comparison between experiments

N_v	experiment	mean	min.	max.	std. dev.
2	$Dist(m)$	49.63	47.36	55.66	1.92
4	$Dist(m)$	70.35	68.43	72.43	1.03
5	$Dist(m)$	83.61	81.68	86.01	1.04
6	$Dist(m)$	98.71	97.41	100.21	0.76
8	$Dist(m)$	128.20	126.80	129.47	0.62
2	$T_{max}(s)$	60.54	56.36	67.69	3.24
4	$T_{max}(s)$	49.46	45.20	50.79	1.35
5	$T_{max}(s)$	46.60	44.33	49.39	2.12
6	$T_{max}(s)$	44.91	44.33	45.25	0.31
8	$T_{max}(s)$	45.07	44.33	48.49	0.97
2	$CPU_{time}(s)$	102.73	94.82	121.07	6.22
4	$CPU_{time}(s)$	104.50	103.14	109.48	1.37
5	$CPU_{time}(s)$	108.49	106.92	110.12	0.95
6	$CPU_{time}(s)$	113.03	111.52	116.34	1.23
8	$CPU_{time}(s)$	148.30	137.90	170.08	11.85

Fig. 4 shows the value of the cost function used in the ACO algorithm over the course of 10^3 iterations, with a confidence interval of 90%. It is clear from the graph that the meta-heuristic used tends to reduce the cost with each iteration, and also shows high reliability in reproducing the results, as it has a narrow confidence interval. For the experiment with 6 UAVs there was a momentary saturation where the cost remained constant from 3000 to 8000 iterations, however, afterwards there was a subtle drop in the cost and it was saturated again until the end of the iterations.

The Fig. 5 shows the best routes generated with different numbers of UAVs. As a result, the experiment that obtained the best mean T_{max} was with 6 UAVs, with $44.91s$. However, looking at the minimum values of the experiments,

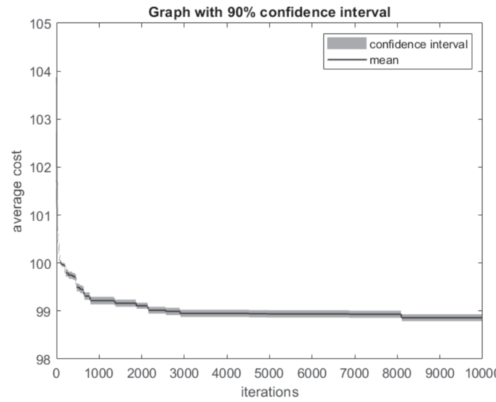

Fig. 4. Cost over iterations in the experiment with 6 UAVs

.

in the cases with 5, 6 and 8 UAVs, all obtained $44.33s$ as a result, but the experiment with 6 UAVs stood out, as it had the lowest standard deviation, with a value of 0.31. In view of this, the simulation with 6 UAVs would be sufficient to carry out the mission in good time.

Conclusion

This work proposed a new hybrid heuristic algorithm using ACO and A* for the UAV routing problem, automating the creation of routes with deviation from fixed obstacles on a fictitious map. In view of the results, it is possible to answer the RP in Introduction, which served as a guide for the development of the research:

- **RP1.** Given the growing demand for applications in various areas, such as agriculture, environmental monitoring and logistics. Is it possible to develop an efficient algorithm for route planning for multiple UAVs?
  Answer RP1: It is no secret that route planning for multiple collaborative UAVs is highly complex, however, the proposed hybrid algorithm was able to carry out the planning in a timely manner, processing up to 8 UAVs in a maximum time of $170.08s$, as shown in Table 4.
- **RP2.** Can the use of a hybrid algorithm using ACO and A* for route planning and obstacle avoidance provide results with a satisfactory confidence interval?
  Answer RP2: The use of heuristic methods was able to provide satisfactory results, because through the 20 simulations of each experiment it was possible to obtain similar results in each one, guaranteeing the reproducibility of the experiment. In addition, Fig. 4, shows a narrow 10% confidence interval, guaranteeing the reliability of the proposed method.
- **RP3.** In a fictitious map, what is the difference in time to carry out a mission with different numbers of UAVs?
  Answer RP3: The map used shows that the number of UAVs is not directly related to minimizing mission time. When analyzing the variation from 2 to 8 UAVs, the experiment that obtained the lowest mean value of T_{max} was carried out with 6 UAVs, with a lower standard deviation. In this way, the proposed algorithm is able to help choose the number of UAVs in a mission.

Given these results, although positive, there is still room for improvement in the project, allowing the algorithm to be further optimized and made more commercial. Therefore, this work has some points for improvement, such as:

- Insert a friendly graphical interface for the end user;
- Insert an energy constraint involving the flight time;
- Apply obstacle avoidance to dynamic objects.

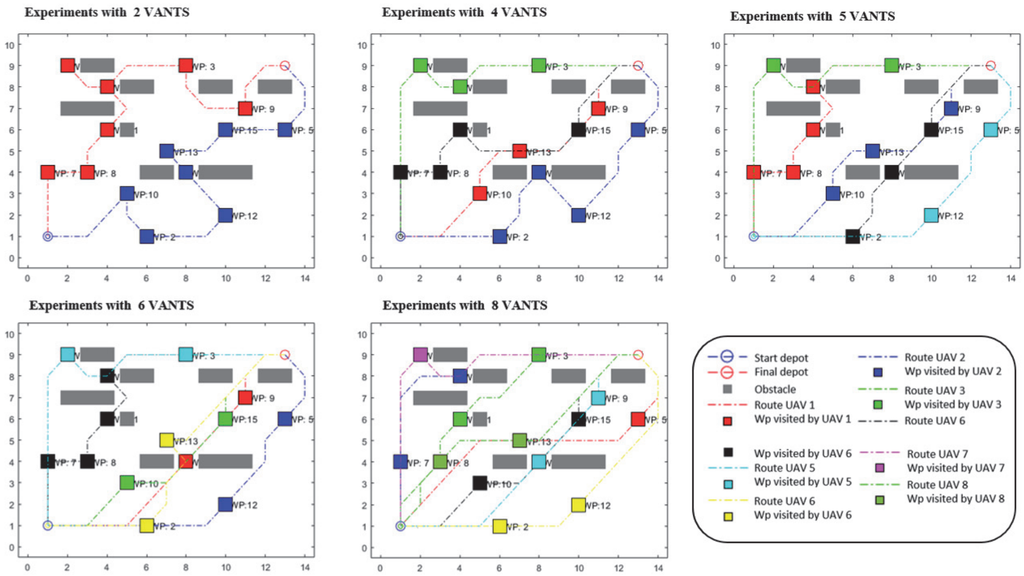

Fig. 5. Routes generated by the best experiments

***Authors*: *Antônio Lucas Sousa Aguiar:* has a degree in Electrical Engineering from the Federal University of Ceará (2022). He is currently a regular Master's student in the Postgraduate Program in Electrical and Computer Engineering at the Federal University of Ceará (UFC). E-mail: lucas.aguiar@alu.ufc.br. *Vandilberto P. Pinto:* Post-doctorate at the Instituto Tecnológico de Aeronáutica - ITA (2017). PhD in Electrical Engineering from the Federal University of Ceará. Master's in Electrical Engineering from the Federal University of Ceará. Specialization in Mathematics from the State University of Ceará. Degree in Electrical Engineering from the Polytechnic School at the Uninter International University Center. Graduation in Mathematics from the Federal University of Ceará.He is currently Associate Professor at the University of International Integration of Afro-Brazilian Lusophony.He has experience in alternative sources, industrial control and automation, robotics, optimization and control theory,artificial intelligence, unmanned aerial vehicles and fault detection.E-mail: vandilberto@unilab.edu.br *Lígia Maria Carvalho Sousa:* She is currently Associate Professor at the Institute of Engineering and Sustainable Development at the University of the International Integration of Afro-Brazilian Lusophony. Her areas of interest are: digital signal processing; modeling, analysis and control of dynamic systems;artificial intelligence, automation and robotics. E-mail: ligia@unilab.edu.br. *José Lucas Da Silva Pinheiro:* Computer Engineering student at the University of International Integration of Afro-Brazilian Lusophony E-mail: lucaspinheiro19072000@gmail.com. *José Cleilton do Nascimento Sousa:* Computer Engineering student at the University of International Integration of Afro-Brazilian Lusophony E-mail: cleiltonsousa00@gmail.com.*

Acknowledgments

The authors would like to thank FUNCAP for the support (process BP5-0197-00058.01.00/22) for financial support.

REFERENCES

[1] Bai, X., Jiang, H., Cui, J., Lu, K., Chen, P., & Zhang, M. (2021). UAV Path Planning Based on Improved A* and DWA Algorithms. International journal of aerospace engineering, 2021.
[2] De Castro Pereira, S., Solteiro Pires, E. J., & de Moura Oliveira, P. B. (2023). Ant-Balanced Multiple Traveling Salesmen: ACO-BmTSP. Algorithms, 16(1), 37.
[3] Farlei J. Heinen, F. S. O. (2002). Sistema de controle híbrido para robôs móveis autônomos.
[4] Floreano, Dario, and Robert J Wood.: Science, technology and the future of small autonomous drones. Nature vol. 521,7553 (2015): 460-6. doi:10.1038/nature14542.
[5] Freitas, E. and Carvalho, J. (2015). Genetic algorithm approach for a class of multicriteria, multi-vehicle planner of uavs. volume 9019, pages 234—248.
[6] Gonçalves, M. A. (2013). Algoritmo a-estrela de estado híbrido aplicado à navegação autônoma de veículos.
[7] Ju, C. and Son, H. I. (2018). Multiple uav systems for agricultural applications: Control, implementation, and evaluation. Electronics, 7(9):162.
[8] Liu, Y., Zhang, Y., Zhang, W., Zhang, Y., and Zhang, J. (2019). Reinforcement learning in multiple-uav networks: Deployment and movement design. IEEE Transactions on Vehicular Technology, 68(8):8037–8049.
[9] Lu, L.-C. and Yue, T.-W. (2017). Mission-oriented ant-team aco for min-max mtsp. In 2017 International Conference on Information, Communication and Engineering (ICICE), pages 522–525. Necula, R., Breaban, M., and Raschip, M. (2015). Tackling the bi-criteria facet of multiple traveling salesman problem with ant colony systems. In 2015 IEEE 27th International Conference on Tools with Artificial Intelligence (ICTAI).
[10] M. Dorigo, V. Maniezzo and A. Colorni.: Ant system: optimization by a colony of cooperating agents, in IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), vol. 26, no. 1, pp. 29-41, Feb. 1996, doi: 10.1109/3477.484436.
[11] Necula, R., Breaban, M., and Raschip, M. (2015).: Tackling the bi-criteria facet of multiple traveling salesman problem with ant colony systems. In 2015 IEEE 27th International Conference on Tools with Artificial Intelligence (ICTAI).
[12] Niu, C., Li, A., Huang, X., Li, W., and Xu, C. (2021). Research on global dynamic path planning method based on improved a algorithm. Mathematical Problems in Engineering, 2021.
[13] Pinto, V., Galvão, R., Rodrigues, L., and Gomes, J. P. (2020). Mission planning for multiple uavs in a wind field with flight time constraints. Journal of Control, Automation and Electrical Systems, 31.
[14] Santana, K. (2020). Método metaheurístico multi-objetivo para o cálculo de rotas de robôs colaborativos com restrições energéticas.
[15] Wang, M., Ma, T., Li, G., Zhai, X., and Qiao, S. (2020). Ant colony optimization with an improved pheromone model for solving mtsp with capacity and time window constraint. IEEE Access, 8:106872–106879.
[16] Wang, S., Liu, Y., Qiu, Y., Zhang, Q., Huo, F., Huangfu, Y., Yang, C., and Zhou, J. (2022). Cooperative task allocation for multi-robot systems based on multi-objective ant colony system. IEEE Access, 10:56375–56387.
[17] Wang, Z., Xiang, X., Yang, J., Yang, S. (2017, December). Composite Astar and B-spline algorithm for path planning of autonomous underwater vehicle. In 2017 IEEE 7th International Conference on Underwater System Technology: Theory and Applications (USYS) (pp. 1-6). IEEE.