2023 International Conference on Unmanned Aircraft Systems (ICUAS)
June 6-9, 2023 | Lazarski University, Warsaw, Poland

Spiral coverage path planning for Multi-UAV photovoltaic panel inspection applications

Marco Andrés Luna*, Mohammad Sadeq Ale Isaac*, Miguel Fernandez-Cortizas*, Carlos Santos†, Ahmed Refaat Ragab‡, Martin Molina§, Pascual Campoy *
* Computer Vision and Aerial Robotics Group (CVAR), Centre for Automation and Robotics (C.A.R.), Universidad Politécnica de Madrid (UPM-CSIC), Calle Jose Gutierrez Abascal 2, 28006 Madrid, Spain
† Department of Signal Theory and Communications, University of Alcala (UAH), 28805, Alcalá de Henares, Madrid, Spain
‡ Department of Network, Faculty of Information Systems and Computer Science, October 6 University, Giza 12511, Egypt
§ Department of Artificial Intelligence, Universidad Politécnica de Madrid, 28040 Madrid, Spain

2023 International Conference on Unmanned Aircraft Systems (ICUAS) | 979-8-3503-1037-5/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICUAS57906.2023.10156085

Abstract—This paper deals with the problem of coverage path planning for multiple UAVs in disjoint regions. For this purpose, a spiral-coverage path planning algorithm is proposed. Additionally, task assignment methods for multi-region inspection with a swarm of UAVs are applied. The centralized system architecture is described, and an adaptive sliding mode controller is designed. Furthermore, we evaluate the performance of the proposed techniques by obtaining numerical results and simulations with the controller. The results show that the spiral pattern optimizes the cost of the mission and improves the task distribution of the mission planning system. Additionally, the performance of the proposed controller is robust to simulated disturbances.

Index Terms—Path Planning, Swarms, UAS Applications

I. Introduction

Over the years, industrial applications of UAVs (Unmanned Aerial Vehicles) have grown exponentially due to their versatility, payload integration capabilities, and low operating cost. UAVs have become a popular tool for various types of inspections, such as building inspections, bridge inspections, power line inspections, and oil rig inspections. These inspections are critical to ensure the safety and reliability of these structures and to detect potential problems before they become more serious. UAVs provide a cost-effective and efficient solution to inspect large structures that would otherwise be difficult to access, and they can gather data much more quickly and safely than traditional inspection methods. For these reasons, these vehicles have driven the development of several lines of research such as low-level flight controllers, trajectory planning and control, communications, decision-making systems, and others. In the case of photovoltaic (PV) plants, UAVs have become increasingly important due to the growth in renewable energy production. PV plants can cover large areas and have many individual solar panels, making manual inspections time-consuming and potentially dangerous. UAVs equipped with cameras, thermal imaging sensors, and other specialized equipment can quickly inspect each panel to detect any issues and provide accurate data for maintenance and repair. With regular UAV inspections, PV plant owners and operators can ensure that their plants are operating at peak efficiency and maximize their return on investment. The use of UAVs in PV inspections is, therefore, a critical component of effective renewable energy management.

In this context, multi-UAV systems and UAV swarms have gained interest due to their advantages over single-UAV systems, such as fault tolerance, time efficiency, and flexibility [1]. The current challenges of this discipline include formation control [2], [3], task assignment [4], [5], communication architecture [6], [7], multi-agent path planning [8], and others.

Based on this, controlling multi-agent systems is a recently proposed issue that investigates optimized stability for a desired formation. Mostly, challenges appear while swarm agents are interacting and there is no estimated position for individual nodes. Simplifying the problem, several models are observed: position-based, displacement-based, and distance-based controllers [9]. The first one stabilizes agents' positions independently, with respect to the global coordinate, leading to a harmonic swarm. The second one controls the dispersion of each node, according to a relative system based on other nodes. The latter maintains the distance among all UAVs to stabilize the whole formation.

The authors of [10] introduced spatial predictive control (SPC) for the attitude level of a swarm model. They implemented an optimization function by observing the position of the agents, which computes the cost function gradient and aims to estimate the next position. They claimed that SPC is robust enough to compensate for communication latency and sensor noise; however, their model was not implemented as an outdoor platform. Whereas through previous works of current authors [11], [12], a robust controller is presented using a Lyapunov candidate, the calculation processes were outfit due to the complicated control law and the simulation time was too

long; in this paper, a new sliding method is proposed that reduces the computation dramatically by considering all the nonlinear uncertainties as a unique function to be reduced.

Leon-Blanco et al. [13] worked on a formation topology by randomizing ground and flying vehicles to target non-predefined locations, called the truck-multi-drone team logistic problem (TmDTL). This paper copes with defining routes that are estimated after one agent passes and announces for others to switch for other locations dynamically; however, this paper did not implement a control scenario with a realistic model.

Yang et al. [14] concentrated on a desired formation with constraints and controlling the motion inside the confined area. Considering the neighborhood's local positions, they implemented a novel method similar to the displacement-based method, supposed projection and formation elements in their control law, and proved theoretically that all agents are asymptotically stabilized inside the constrained area to reach the formation. This study included two numerical case studies; however, no practical work was conducted.

On the other hand, multi-UAV coverage path planning (CPP) is a novel research field within the path planning discipline that aims to find the optimal path to cover the free workspace with multiple UAVs equipped with limited footprint sensors [15]. Thus, the authors in [16] identify decomposition and non-decomposition methods for 2D-CPP. For example, Barrientos et al. [17] decompose the free workspace into cells and place a waypoint at its center. They calculate the most efficient route by optimizing the graph that includes all cells. Moreover, non-decomposition methods generate paths by taking the total workspace area as a reference. In this context, the research described in [18], [19] uses the back-and-forth method for planning. The authors perform the route assessment using the Mixed Integer Linear Programming technique. Additionally, the authors in [20] apply the spiral-like coverage path planning in a single-UAV system, and Balampis et al. [21] use this pattern for multiple UAVs by dividing the entire region into sub-regions.

Furthermore, PV inspection applications may require monitoring several disjoint areas. Whereby, the determination of the visiting order of the regions arises as an additional problem. Xie et al. [22] refer to this issue as Travel Salesman Problem CPP (TSP-CPP). They propose an algorithm based on dynamic programming with a grid navigation pattern. Besides, the study presented in [23] states a method based on area centroid calculation, solving the order of visits by applying genetic algorithms as a simple TSP. However, both techniques were focused on single UAVs. In [24], the authors propose a solution for multi-UAV systems using a genetic algorithm approach. Through this research, we present a method for generating a spiral trajectory, and an algorithm for UAV swarm task assignment based on the Nelder-Mead optimization technique [25] is implemented. In addition, this approach is analyzed and compared with our previous research [26].

Hence, the present study focuses on algorithms for generating spiral coverage routes for multiple disjoint regions and task distribution for a fleet of UAVs. While in [21], the authors present an approach based on Constrained Delaunay Triangulation and cell decomposition, the authors in [16] use geometric methods to calculate the distances and angles of each waypoint along the trajectory. Our algorithm presents two main variants compared to previous approaches, considering that the scope of the problem has been extended to the exploration of multiple areas. Firstly, the vertices of the polygon are reordered based on the previous area visited. This stage has been implemented to optimize mission time, and reduce the turning angles and the distance traveled by the UAVs. Secondly, the polygon angles are calculated once, and parallel lines are generated from these angles until the number of rings is covered. Then, the intersections of the lines are found in an orderly manner to generate the waypoints. This last step helps to improve the computation time of each spiral route. In addition, it describes communications for a centralized architecture and controller design for navigation. For this phase, numerical and simulated results are presented. The rest of the paper is organized as follows: Section II describes the materials and methods used; then, Section III presents the obtained results; next, Section IV discusses the results obtained; and finally, Section V concludes the paper.

II. Methodology

A. System Architecture

For this system, a centralized architecture is presented, i.e., the central node is responsible for the control and planning of the UAV swarm, as shown in Figure 1.

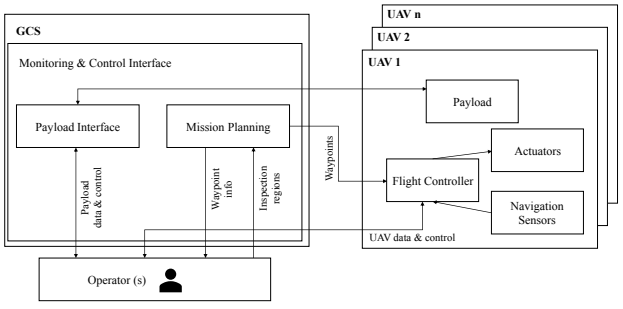

Fig. 1. System architecture schema

The ground control station (GCS) contains multiple subsystems, such as communications, mission planning, flight control, and payload management. The communications module sends and receives flight data during the operation. The planning module calculates the trajectories of each aircraft for a user-defined mission. The flight

control module monitors the state of the flight allowing the operator to make decisions in real-time; finally, the payload management module contains specific software and a communication system focused on processing the images obtained during the inspections. On the other hand, each UAV is composed of a communications system, navigation sensors, flight controllers, actuators, and a special payload for inspection tasks.

B. Controller Design

As mentioned in the introduction, the controller developed in this paper is an adaptive sliding mode controller (ASMC) to maintain robustness in the closed-loop system and to converge in a finite time. By this method, the chattering problems which could be harmful to the actuators are solved by a boundary layer defined by the dynamic adaptation of the controller gains, which reduces the discontinuity. Meanwhile, to obtain smooth results from the controller, some of the harsh waypoints that lead to yaw changing of more than 90° were saturated and a continuous trajectory packet is exported to the controller, which is summarized in Algorithm 1, in which t refers to the time, i to mission waypoints, and k to waypoints lists.

Algorithm 1: Trajectory navigation algorithm

```
Data: pos{x_1, y_1, z_1, t_1}, ...., pos{x_n, y_n, z_n, t_n}, vel
Result: desired values{(x_1, y_1, z_1), (x_2, y_2, z_2), ...}
t_i, d_{t_i} ← timeStamps;
for i ← 1 to timeSeries do
    for t ← 1 to length(wps) do
        crntLine ← prevPoint ∩ curntPoint(t, i);
        wpList_k ← crntLine;
        if atan(y_i − y_{i−1})/(x_i − y_{i−1}) − atan(y_{i−2} −
          y_{i−1})/(x_{i−1} − y_{i−2}) > 2 then
            crntPoint ← nextPoint;
        end
        fullTrajectory ← lineSeries(t, i);
    end
end
```

After planning suitable trajectories for each UAV, the desired values are utilized to generate desired velocity and acceleration terms, and then, produce the error functions to bring stability to the sliding surface. Assuming that the model is of a rigid body, asymmetric and the center of gravity (CoG) is the same as the origin of the Body fixed frame. In addition, it is supposed that the Z axis is toward the gravitational direction, X is forward, and Y is toward the right side of the UAV. The rotational dynamics motion of the model, given by Euler angles is as Eq. 1;

$$\vec{M} = d\vec{H}/dt + \vec{\omega} \times \vec{H} \tag{1}$$

Where, $\vec{M}$ is the total external torque, $\vec{H}$ is the angular torque generated by the body rotation in the Body frame, and the last term is produced by the propeller's angular velocity. Considering $\vec{H}$ as a 3D momentum matrix times angular velocities, the total momentum could be represented as,

$$\vec{M} = \begin{bmatrix} I_x\dot{\omega}_x + \omega_y\omega_z(I_z - I_y) + J_r\Omega\omega_y \\ I_y\dot{\omega}_y + \omega_x\omega_z(I_x - I_z) - J_r\Omega\omega_x \\ I_z\dot{\omega}_z + \omega_x\omega_y(I_y - I_x) \end{bmatrix} \tag{2}$$

Then, solving for the roll, pitch, and yaw torques (τ_x, τ_y, and τ_z, respectively), the relation between the angular velocities about body axes and Euler angles rates could be obtained as follows,

$$\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} \dot{\phi} - sin(\theta)\dot{\psi} \\ cos(\phi)\dot{\theta} + sin(\phi)cos(\theta)\dot{\psi} \\ -sin(\phi)\dot{\theta} + cos(\phi)cos(\theta)\dot{\psi} \end{bmatrix} \tag{3}$$

Hence, presenting the Eq. 3 in the Euler form, it yields,

$$\begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} \frac{\tau_x}{I_x} + \dot{\theta}\dot{\psi}\left(\frac{I_y - I_z}{I_x} + 1\right) - \frac{J_r\omega}{I_x}(\dot{\theta} + \phi\dot{\psi}) + \ddot{\psi}\theta \\ \frac{\tau_y}{I_y} + \dot{\phi}\dot{\psi}\left(\frac{I_z - I_x}{I_y} - 1\right) - \frac{J_r\omega}{I_y}(\dot{\theta} + \phi\dot{\psi}) + \ddot{\psi}\theta \\ \frac{\tau_z}{I_z} + \dot{\phi}\dot{\theta}\left(\frac{I_x - I_y}{I_z} + 1\right) + \ddot{\theta}\phi \end{bmatrix} \tag{4}$$

Where Ω is the total angular velocity of the propellers and J_r is the rotor inertia. Then, to define the Jacobian state matrix of the system, twelve elements are assumed as $X = \begin{bmatrix} x & \dot{x} & y & \dot{y} & z & \dot{z} & \phi & \dot{\phi} & \theta & \dot{\theta} & \psi & \dot{\psi} \end{bmatrix}^T$, and the input matrix is rewritten as $U = \begin{bmatrix} \tau_x & \tau_y & \tau_z & \delta \end{bmatrix}^T$, in which, δ is the uncertainty term. Considering the reference model of a second-order system as Eq. 5, the control aim is to estimate the reference model and adapt the parameters dynamically.

$$G_m(s) = \frac{w^2}{s^2 + 2\xi\omega s + \omega^2} \tag{5}$$

Improving the last works done [11], [12], a proposed candidate second-order transfer function was assumed as $G(s) = 1/s^2$, the error function could be obtained and the Lyapunov candidate presented as $V = 1/3e^Te + e$. To continue, it is proven this candidate is negatively definite and the controller's response would be asymptotically stable.

$$\begin{aligned} &\dot{V} = \tfrac{1}{3}(\dot{e}^Te) + \dot{e} \\ &\rightarrow \dot{V} = \tfrac{1}{3}(\dot{x} - \dot{x}_m)e + 2\dot{x} - \dot{x}_m \\ &\rightarrow \dot{V} = \tfrac{1}{3}(Ke - e^TIe + \delta) \end{aligned} \tag{6}$$

As shown in Eq. 6, three terms are involved, the first two ones are negative definite, and the only uncertainty term remained that could be considered suitably negative to stabilize the algorithm. Hence, the control law is represented as,

$$\begin{cases} \ddot{x} = -\lambda\dot{e}_x - \mu\tanh(\delta_x) \\ \ddot{y} = -\lambda\dot{e}_y - \mu\tanh(\delta_y) \\ U_z = m(\ddot{z}_d - \lambda\dot{e}_z + \xi_{ad}e_z) - \mu\tanh(\delta_z) \\ U_\phi = I_x/l(\ddot{\phi}_d - \lambda\dot{e}_\phi + \xi_{ad}e_\phi) - \mu\tanh(\delta_\phi) \\ U_\theta = I_y/l(\ddot{\theta}_d - \lambda\dot{e}_\theta + \xi_{ad}e_\theta) - \mu\tanh(\delta_\theta) \\ U_\psi = I_z(\ddot{\psi}_d - \lambda\dot{e}_\psi + \xi_{ad}e_\psi) - \mu\tanh(\delta_\psi) \end{cases} \tag{7}$$

Where U_z, U_ϕ, U_θ, and U_ψ are the total thrust force, rolling torque, pitching torque, and yawing torque, respectively. Moreover, λ is the modifiable controller's parameter to stabilize the system, μ is the uncertainty parameter, l is the rotors' arm long to the CoG, and the subscript $\{\}_d$ targets the desired value. Furthermore, ξ_{ad} is the adaptation parameter, $\delta = \dot{e} + \lambda e$ defines the sliding principle, and $tannh$ reduces the chattering phenomenon by smoothing the sliding surface. Elaborately, the adaptation term aims to all the nonlinear uncertainties as a unique noise and converges to stability faster than conventional model reference adaptive controller (MRAC).

C. Mission Planning System

1) Problem Statement: For a given set of UAVs with different initial positions and a group of polygonal regions separated from each other in a. solar power plant (Figure 2), the objective is to find the optimal trajectory for each UAV that minimizes the mission cost. Therefore, the

Fig. 2. Mission operation example.

algorithm must calculate the order of visitation of the areas, determine each region's coverage flight pattern and generate the individual trajectories for each UAV. Also, we assume that the aircraft are homogeneous and their flight altitude is the same (consequently, their footprint is equal). Hence, the analysis of the parameters that define the sensor footprint, such as height, the field of view (FOV), and overlap, is not part of this research.

2) Trajectory Generation: Considering the needs of the problem, the determination of the visit order of the areas was performed using a similar approach to the studies presented in [23], [26]. For this case, we calculate the centroid of the polygon formed by the set of UAVs and then the centroid of each region to use them as TSP coordinates (as shown in Figure 3).

Unlike previous works, we use the Cristofides algorithm [27] to obtain the TSP solution due to its computational efficiency. According to [28], this is one of the most optimal algorithms to date for solving the TSP problem.

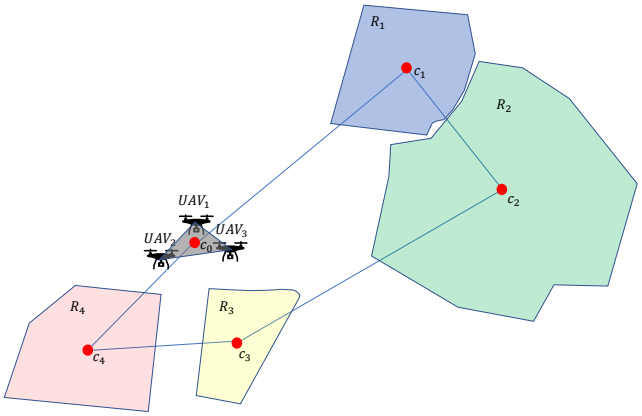

Fig. 3. TSP calculation schema.

On the other hand, the coverage trajectories were generated by a spiral-like coverage pattern; according to [20], this type of route is more energy efficient than back-and-forth. For ease of use, the polygon is assumed to be convex.

Algorithm 2 presents the pattern generation process.

Algorithm 2: Spiral pattern generation algorithm

```
Data: R_0{v_1, v_2, ...}, R{v_1, v_2, ...}, l
Result: wpList{(x_1, y_1), (x_2, y_2), ...}
c_0 ← centroid(R_0);
R' ← OptimalArrange(R, c_0);
c ← centroid(R');
dc_min ← getMinDist(R, c);
rings ← dc_min / l;
paralels ← generateParalels(R', rings);
prevLine ← Line_0;
for i ← 1 to rings do
    for j ← 1 to length(R') do
        crnttPoint ← prevLine ∩ paralels(j, i);
        wpList_k ← crntPoint;
        prevLine ← paralels(j, i);
    end
end
```

In Algorithm 2, the inputs are given by the region to explore R, the previous region R_0, and the width of the lanes l. Due to its application in multiple zones, the centroid of the previous zone c_0 defines the beginning of the spiral in the current region. Thus, the algorithm rearranges the vertices of R to form R'. For the first region, we consider the centroid of the polygon formed by the UAVs. This step, called OptimalArrange, calculates the minimum Euclidean distance between the previous centroid and the vertices of the current polygon:

$$\min(|v_i - c_0|) \tag{8}$$

The starting point of the spiral is then taken as the vertex with the shortest distance, and all the vertices of the

polygon are reorganized according to this vertex. Then, the number of rings is obtained by dividing the minimum distance between one side of the polygon and the centroid [21]. Next, a set of internal parallel lines is defined by calculating the slopes of the sides. Finally, we intersect the edges generated in each step and the corresponding parallel to get the points of the spiral.

3) Task Assignment: With the global route generated, the BINPAT method introduced in [29] is used to assign the paths to each UAV. This technique divides the trajectory into sub-routes with a heuristic similar to the bin packing problem in which each UAV is weighted and receives its sub-route according to its weight; equating the time mission cost given by:

$$c^k = \sum_i \sum_j \frac{D_{ij}}{v^k} + \frac{dl^k}{v^k} \tag{9}$$

Assuming constant velocity, it can be simplified to:

$$c^k = \sum_i \sum_j D_{ij} + dl^k \tag{10}$$

Where the individual cost is given by the distance of the part of the route traveled D_{ij} plus delays caused by the initial position of the aircraft dl^k that could unbalance this cost and make the task allocation less optimal. On the other hand, the global cost of the mission is defined as the maximum individual cost:

$$C = \max(c^k) \tag{11}$$

For this reason, it is necessary to optimize the initial weights to reduce the effect of dl^k; for this purpose, the Nelder-Mead technique [26] uses a simplex method to minimize the global cost function C. Numerical and simulation results can be found in section III.

III. Results

A. Numerical Results

To obtain numerical results, we perform several tests using two to five UAVs and two to eight inspection areas with spiral and back-and-forth navigation patterns. Graphical results are shown in Figure 4.

Additionally, if the disjointed regions are too far apart, the optimization stage of the algorithm will correct any possible planning distribution errors and calculate the most optimal mission for each aircraft (as shown in Figure 5).

Tables I and II report the results of the mission cost, that is, the maximum single mission cost, as mentioned in Section II-C3.

Furthermore, the distribution of the mission among the UAVs was analyzed quantitatively using the coefficient of variation CV given by:

$$CV = \frac{SD}{Av} \times 100 \tag{12}$$

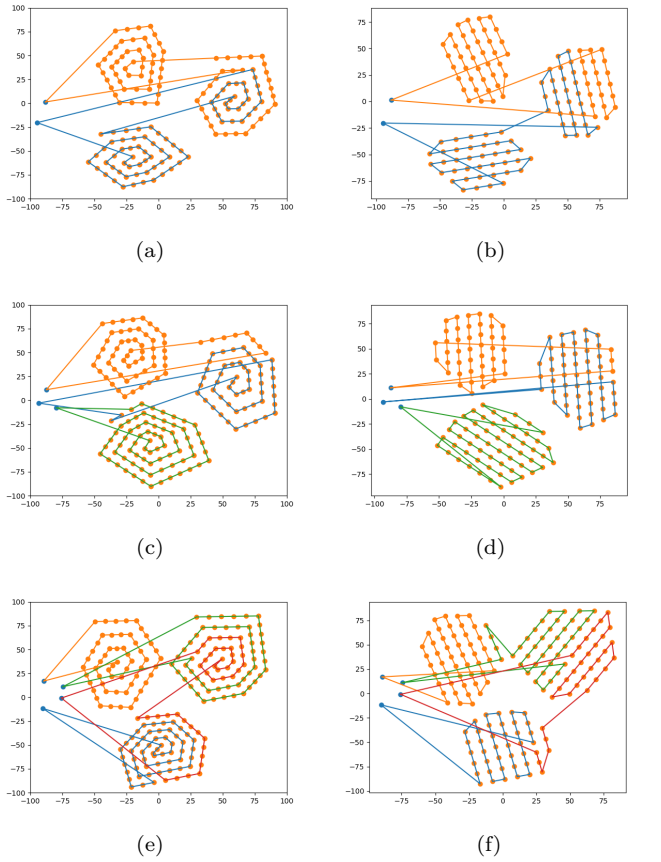

Fig. 4. Graphical results for spiral and back-and-forth patterns in three regions (a) spiral pattern for 2 UAVs (b) back-and-forth pattern for 2 UAVs (c) spiral pattern for 3 UAVs (d) back-and-forth pattern for 3 UAVs (e) spiral pattern for 4 UAVs (b) back-and-forth pattern for 4 UAVs.

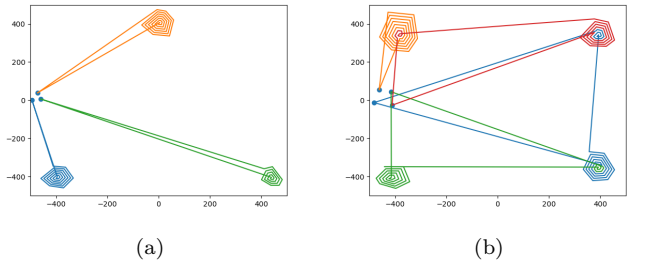

Fig. 5. Graphical results of a spiral pattern for far-apart disjoint areas. (a) 3 UAVs and 3 regions (b) 4 UAVs and 4 regions

Where SD is the standard deviation of the individual costs and Av is the average cost. Results are shown in Tables III and IV.

From Tables, I, II, III, and IV spiral obtains lower mission costs in most cases; and also distributes the mission equally according the coefficient of variation.

B. Simulation Results

Based on the simulation conducted for distinct navigation patterns, as back-and-forth and spiral, the numerical

TABLE I
Mission cost results for spiral (SP) and back-and-forth (B&F) patterns with two and three UAVs

Zones	UAVS			
	2		3	
	SP	B&F	SP	B&F
2	2572.35	2785.59	1212.56	1285.98
3	2629.95	2988.90	1529.81	1617.01
4	3330.61	3281.44	2167.11	2268.72
5	3429.97	3502.74	2494.25	2546.90
6	3457.27	3636.47	2415.82	2494.04
7	3928.27	4129.97	2918.46	3053.34
8	5268.29	5256.78	2908.79	3168.43

TABLE II
Mission cost results for spiral (SP) and back-and-forth (B&F) patterns with four and five UAVs

Zones	UAVS			
	4		5	
	SP	B&F	SP	B&F
2	869.04	969.28	745.19	840.85
3	1293.42	1438.28	1155.99	1271.96
4	1744.01	1742.09	1502.00	1516.71
5	1945.91	2239.52	1735.04	1667.34
6	2332.11	2498.55	1747.21	1915.74
7	2038.00	2071.84	1540.86	1635.14
8	2450.55	2702.15	2292.69	2611.43

TABLE III
Coefficient of variation results for spiral (SP) and back-and-forth (B&F) patterns with two and three UAVs

Zones	UAVS			
	2		3	
	SP	B&F	SP	B&F
2	0.28	2.03	2.15	0.57
3	0.29	0.12	5.11	9.88
4	3.63	4.23	2.20	3.75
5	3.21	0.09	0.32	2.03
6	2.67	2.08	0.43	2.01
7	0.38	0.14	0.23	2.93
8	0.14	0.16	0.21	0.30

TABLE IV
Coefficient of variation results for spiral (SP) and back-and-forth (B&F) patterns with four and five UAVs

Zones	UAVS			
	4		5	
	SP	B&F	SP	B&F
2	1.51	3.79	2.60	8.10
3	0.97	5.10	0.67	1.81
4	9.96	5.70	3.80	3.62
5	0.38	5.08	5.09	5.72
6	0.38	2.88	6.34	4.94
7	0.38	4.46	0.39	0.77
8	0.31	1.48	0.29	4.16

results are demonstrated in Figs. 6 and 7. As shown through Fig. 6, four main zones are mapped to be covered by three UAVs in a total area of $100 \times 100\ m^2$, and the controller performance are quietly acceptable, in presence of the wind disturbance. Nonetheless, the variations of the paths are due to the large lines with a few waypoints exported for the controller. Overall, this mission lasted four minutes, UAVs launched within two meters of distance, and the flight altitude was supposed to be $20m$; however, since the lines in some points overlapped, a $5m$ difference in drones' altitudes is considered. Through the simulation, various wind disturbances are implemented that are obviously seen in the trajectory of drone2, then, a medium noise for the drone0, and without noise for the drone1. In addition, Fig. 7 displays the same

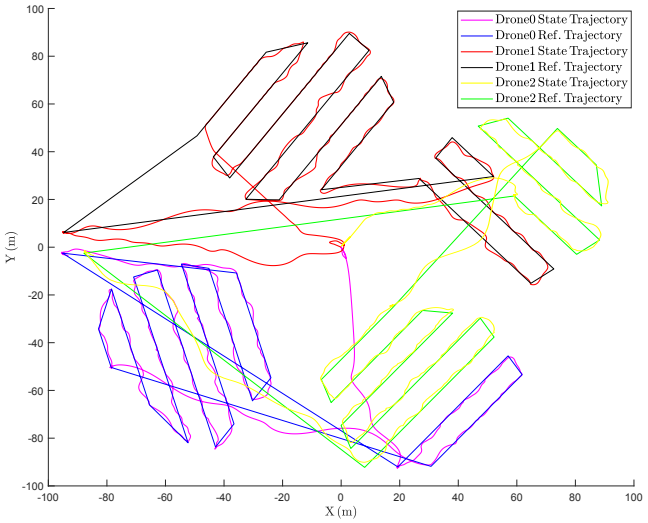

Fig. 6. back-and-forth desired and state trajectories, considering wind disturbance for drone2 and drone0.

area with the same nodes, but different planning, which enhanced the controller's efficiency due to the smoother lines through the mission. Likewise, the latter navigation pattern, drone2 and drone0 are exposed to the wind gusts for a double amount of the other pattern to examine the controller's performance dependent on the spiral pattern.

IV. Discussion

Regarding the numerical results, Tables I and IV show that the spiral pattern results in lower mission cost; this is due to two main variations in the method; on the one hand, the starting criterion of the spiral and the use of the Nelder-Mead optimization technique instead of Powell in the last step. On the other hand, the spiral pattern has a lower coefficient of variation (Tables III and IV); therefore, the mission cost is more equitable. Both criteria could be translated into a lower energy expenditure for each UAV. In addition, simulation results are shown in Figs. 6 and 7 demonstrate that even implementing a double wind perturbation exposed against the UAVs movement,

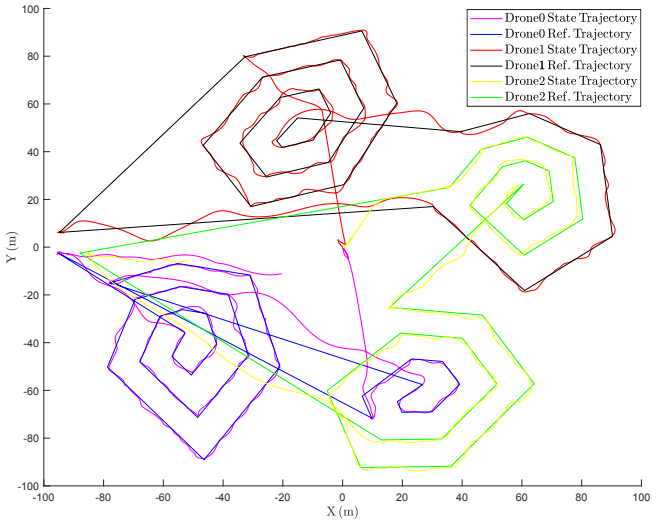

Fig. 7. spiral desired and state trajectories, considering wind disturbance for drone2 and drone0.

the spiral pattern navigates the drones with a lesser error of attitude and position as well rather than the back-and-forth pattern.

V. Conclusions and Future Work

This paper deals with the spiral path planning algorithm for Multi-UAV PV inspection tasks. Considering its application in multiple regions, we apply a TSP-CPP strategy to find the order of visiting; then, each route starts at the point closest to the centroid of the previous one.

Results demonstrate improved system controllability and fair task distribution when applying this type of route compared to the back-and-forth approach; also, the proposed algorithms for path planning show promising results in energy efficiency and can be improved. Thus, this suggests that other ways of covering areas should be researched. Furthermore, the nonlinear controller functions adequately hasten to converge toward stability. For future work, the route generation problem in non-convex regions will be addressed. Finally, we recommend the implementation and testing in real scenarios.

Acknowledgment

This work has been supported by the project COPILOT ref. Y2020/EMT6368 “Control, Monitoring and Operation of Photovoltaic Solar Power Plants by mean of synergic integration of Drones, IoT, and advanced communication technologies”, funded by Madrid Government under the R&D Synergic Projects Programa. We also acknowledge the support of the European Union through the Horizon Europe Project No. 101070254 CORESENSE. This work has also been supported by the project INSERTION ref. ID2021-127648OBC32, ”UAV Perception, Control and Operation in Harsh Environments”, funded by the Spanish Ministry of Science and Innovation under the program ”Projects for Knowledge Generating”. This work has been funded by the Community of Madrid through the project REF: IND2020/IND-17478 ”Ayudas para la realización de Doctorados Industriales”, especially regarding the work of the first author.

References

[1] Y. Zhou, B. Rao, and W. Wang, “UAV swarm intelligence: Recent advances and future trends,” IEEE Access, vol. 8, pp. 183 856–183 878, 2020.

[2] H. T. Do, H. T. Hua, M. T. Nguyen, C. V. Nguyen, H. T. Nguyen, H. T. Nguyen, and N. T. Nguyen, “Formation control algorithms for multiple-UAVs: a comprehensive survey,” EAI Endorsed Transactions on Industrial Networks and Intelligent Systems, vol. 8, no. 27, pp. e3–e3, 2021.

[3] Y. Wu, J. Gou, X. Hu, and Y. Huang, “A new consensus theory-based method for formation control and obstacle avoidance of UAVs,” Aerospace Science and Technology, vol. 107, p. 106332, 2020.

[4] M. A. Luna, A. R. Ragab, M. S. A. Isac, P. F. Peña, and P. C. Cervera, “A new algorithm using hybrid UAV swarm control system for firefighting dynamical task allocation,” in 2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC). IEEE, 2021, pp. 655–660.

[5] J. Zhang and X. Jiahao, “Cooperative task assignment of multi-UAV system,” Chinese Journal of Aeronautics, vol. 33, no. 11, pp. 2825–2827, 2020.

[6] M. Campion, P. Ranganathan, and S. Faruque, “A review and future directions of UAV swarm communication architectures,” in 2018 IEEE international conference on electro/information technology (EIT). IEEE, 2018, pp. 0903–0908.

[7] A. R. Ragab, M. S. A. Isaac, M. A. Luna, and P. F. Peña, “Unmanned aerial vehicle swarming,” in 2021 International Conference on Engineering and Emerging Technologies (ICEET). IEEE, 2021, pp. 1–6.

[8] M. Monwar, O. Semiari, and W. Saad, “Optimized path planning for inspection by unmanned aerial vehicles swarm with energy constraints,” in 2018 IEEE Global Communications Conference (GLOBECOM). IEEE, 2018, pp. 1–6.

[9] K.-K. Oh, M.-C. Park, and H.-S. Ahn, “A survey of multi-agent formation control,” Automatica, vol. 53, pp. 424–440, 2015.

[10] A. Brandstätter, S. A. Smolka, S. D. Stoller, A. Tiwari, and R. Grosu, “Multi-agent spatial predictive control with application to drone flocking (extended version),” arXiv preprint arXiv:2203.16960, 2022.

[11] M. S. Ale Isaac, M. A. Luna, A. R. Ragab, M. M. Ale Eshagh Khoeini, R. Kalra, P. Campoy, P. Flores Peña, and M. Molina, “Medium-scale uavs: A practical control system considering aerodynamics analysis,” Drones, vol. 6, no. 9, p. 244, 2022.

[12] M. Isaac, A. Naghash, and S. Mirtajedini, “Control and guidance of an autonomous quadrotor landing phase on a moving platform,” in Proceedings of the IMAV Annual Conference of Autonomous Vehicles, Madrid, Spain, vol. 29, 2019.

[13] J. M. Leon-Blanco, P. L. Gonzalez-R, J. L. Andrade-Pineda, D. Canca, and M. Calle, “A multi-agent approach to the truck multi-drone routing problem,” Expert Systems with Applications, vol. 195, p. 116604, 2022.

[14] Z. Yang, Q. Zhang, and Z. Chen, “Formation control of multi-agent systems with region constraint,” Complexity, vol. 2019, 2019.

[15] A. Otto, N. Agatz, J. Campbell, B. Golden, and E. Pesch, “Optimization approaches for civil applications of unmanned aerial vehicles (UAVs) or aerial drones: A survey,” Networks, vol. 72, no. 4, pp. 411–458, 2018.

[16] T. M. Cabreira, L. B. Brisolara, and F. J. Paulo R, “Survey on coverage path planning with unmanned aerial vehicles,” Drones, vol. 3, no. 1, p. 4, 2019.

[17] A. Barrientos, J. Colorado, J. d. Cerro, A. Martinez, C. Rossi, D. Sanz, and J. Valente, “Aerial remote sensing in agriculture: A practical approach to area coverage and path planning for fleets of mini aerial robots,” Journal of Field Robotics, vol. 28, no. 5, pp. 667–689, 2011.

[18] G. S. Avellar, G. A. Pereira, L. C. Pimenta, and P. Iscold, "Multi-UAV routing for area coverage and remote sensing with minimum time," Sensors, vol. 15, no. 11, pp. 27 783–27 803, 2015.
[19] Y. Hong, S. Jung, S. Kim, and J. Cha, "Autonomous mission of multi-UAV for optimal area coverage," Sensors, vol. 21, no. 7, p. 2482, 2021.
[20] T. M. Cabreira, C. Di Franco, P. R. Ferreira, and G. C. Buttazzo, "Energy-aware spiral coverage path planning for UAV photogrammetric applications," IEEE Robotics and automation letters, vol. 3, no. 4, pp. 3662–3668, 2018.
[21] F. Balampanis, I. Maza, and A. Ollero, "Spiral-like coverage path planning for multiple heterogeneous uas operating in coastal regions," in 2017 International Conference on Unmanned Aircraft Systems (ICUAS). IEEE, 2017, pp. 617–624.
[22] J. Xie, L. R. G. Carrillo, and L. Jin, "An integrated traveling salesman and coverage path planning problem for unmanned aircraft systems," IEEE control systems letters, vol. 3, no. 1, pp. 67–72, 2018.
[23] J. I. Vasquez-Gomez, J.-C. Herrera-Lozada, and M. Olguin-Carbajal, "Coverage path planning for surveying disjoint areas," in 2018 International Conference on Unmanned Aircraft Systems (ICUAS). IEEE, 2018, pp. 899–904.
[24] J. Xie and J. Chen, "Multiregional coverage path planning for multiple energy constrained UAVs," IEEE Transactions on Intelligent Transportation Systems, 2022.
[25] J. A. Nelder and R. Mead, "A simplex method for function minimization," The computer journal, vol. 7, no. 4, pp. 308–313, 1965.
[26] P. Flores Peña, M. A. Luna, M. S. Ale Isaac, A. R. Ragab, K. Elmenshawy, D. Martín Gómez, P. Campoy, and M. Molina, "A proposed system for multi-UAVs in remote sensing operations," Sensors, vol. 22, no. 23, p. 9180, 2022.
[27] N. Christofides, "Worst-case analysis of a new heuristic for the travelling salesman problem," Carnegie-Mellon Univ Pittsburgh Pa Management Sciences Research Group, Tech. Rep., 1976.
[28] R. van Bevern and V. A. Slugina, "A historical note on the 3/2-approximation algorithm for the metric traveling salesman problem," Historia Mathematica, vol. 53, pp. 118–127, 2020.
[29] M. A. Luna, M. S. Ale Isaac, A. R. Ragab, P. Campoy, P. Flores Peña, and M. Molina, "Fast multi-uav path planning for optimal area coverage in aerial sensing applications," Sensors, vol. 22, no. 6, p. 2297, 2022.