

Coverage Path Planning for Spraying Drones

E. Viridiana Vazquez-Carmona^a, J. Irving Vasquez-Gomez^{b,a,*}, Juan Carlos Herrera-Lozada^a, Mayra Antonio-Cruz^c

^a*Instituto Politécnico Nacional (IPN), Centro de Innovación y Desarrollo Tecnológico en Cómputo (CIDETEC), , Av. Luis Enrique Erro S/N, Ciudad de México, 07738, México*

^b*Consejo Nacional de Ciencia y Tecnología (CONACYT), Av. Insurgentes Sur 1582, Ciudad de México, 03940, México*

^c*Instituto Politécnico Nacional, UPIICSA, SEPI, Av. Té 950, Granjas México, Iztacalco, Mexico City, 08400, Mexico*

Abstract

The pandemic by COVID-19 is causing a devastating effect on the health of global population. There are several efforts to prevent the spread of the virus. Among those efforts, cleaning and disinfecting public areas have become important tasks. In order to contribute in this direction, this paper proposes a coverage path planning algorithm for a spraying drone, a micro aerial vehicle that has mounted a sprayer/sprinkler system, to disinfect areas. In contrast with planners in the state-of-the-art, this proposal presents i) a new sprayer/sprinkler model that fits a more realistic coverage volume to the drop dispersion and ii) a planning algorithm that efficiently restricts the flight to the region of interest avoiding potential collisions in bounded scenes. The drone with the algorithm has been tested in several simulation scenes, showing that the algorithm is effective and covers more areas with respect to other approaches in literature. Note that the proposal is not limited to disinfection applications, but can be applied to other ones, such as painting or precision agriculture.

Keywords: path planning, coverage, spraying, drone, COVID-19

PACS: 0000, 1111

2000 MSC: 0000, 1111

*Corresponding author

Email address: jvasquezg@ipn.mx (J. Irving Vasquez-Gomez)

1. Introduction

The pandemic by COVID-19 is causing a devastating effect on global public health. Because of that, various efforts have made to face it. Among those efforts, cleaning and disinfection of public areas have been important. Therefore, tools that facilitate the task are needed. For this purpose, in the automation field some tools has been provided, example of that are the diagnostic systems using computer vision and artificial intelligence techniques, human support robots for the cleaning and maintenance of door handles, and robotic automation for distribution of food and essentials [1, 2, 3]. Another example is the disinfection task, which involves spreading a liquid that cleans surfaces. To carry out the disinfection task in an autonomous fashion, a positioning system or robot moves an active sprinkler in such a way all regions of interest (ROI) are covered [4]. Within positioning systems, the Unmanned Aerial Vehicles (UAVs), also known as drones, have the ability to move in three dimensions while they have a relative low-cost. In consequence, relative large areas such as buildings, courts or halls can be disinfected quickly. See Fig. 1 as an example of the disinfection of a basketball court.

An autonomous disinfection requires to integrate several techniques into one single application that is executed before flight and also during flight. Among them, this paper focuses on the planning technique, which is carried out before flying and receives as input the ROI, the sprinkler system and the vehicle's capabilities in order to to determine a set of waypoints that will be loaded in the on-board controller, so that in a next stage, the vehicle will flight autonomously following the planned path. The core problem in the planning stage is the coverage path planning (CPP). Which is defined as determining a path for a vehicle so that the ROI is covered [5]. In addition, planning for disinfection requires an adequate modeling of the sprinkler system, otherwise disinfection is compromised.

1.1. Related work

Literature dealing with CPP for UAVs is as follows. M.E *et al.* [6] developed a system for spraying pesticides and fertilizers in open crop fields. The spraying UAV was controlled through manual flight plans. The UAV had a camera to capture remote sensing images, with the purpose of identifying green fields and the edges of the crop areas. The remote sensing images were analyzed by QGIS software to generate a map of the area. Keller *et al.* [7]

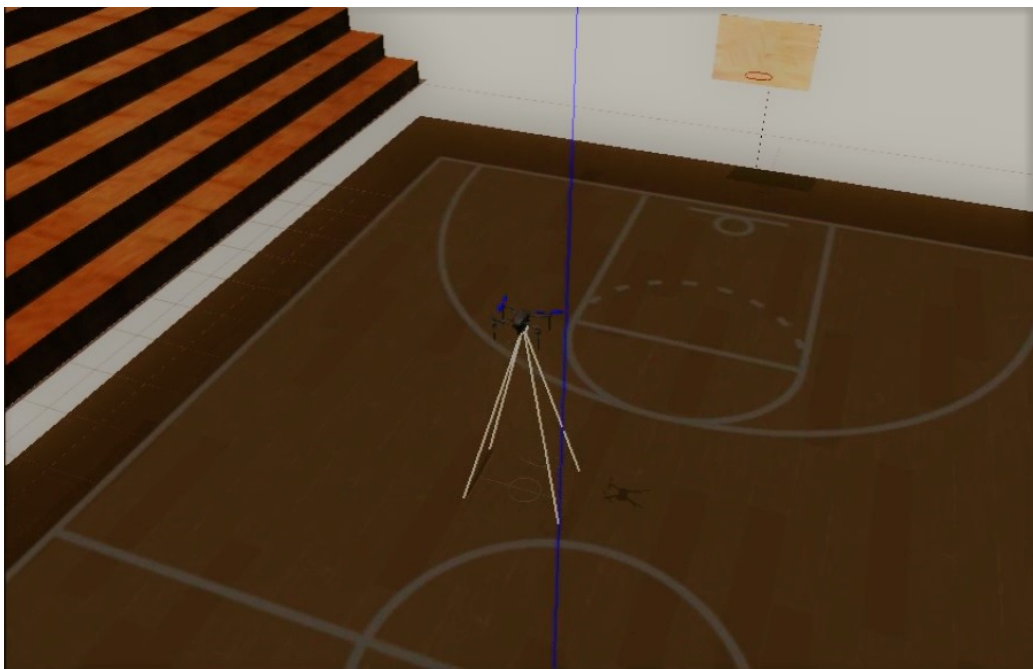


Figure 1: Software in the loop simulation of a basketball court disinfection using the proposed method and Gazebo simulator.

planned G^2 feasible paths for an UAV provided with a camera by concatenating C^2 curves in order to achieve persistent surveillance missions. An augmented A* algorithm was used to find a cycle sequence for the surveillance area of interest. This sequence was used in a B-spline curve generation algorithm to develop smooth paths that satisfy curvature constraints. For the same task, Xiao *et al.* [8] proposed a path planning algorithm based on the continuous updating of the virtual regional field and its local gradients. This virtual field incorporated a Boolean function which contains the information of the target regions and the obstacle information that forms a logical map. When a nonzero gradient at each point is in the regional field, the UAV finds the following target regions. Also, Stefas *et al.* [9] introduced an autonomous aerial system on a multi-rotor UAV navigating on an orchard for crop inspection. With a stereo camera yield data was collected from the area of interest. For this, components for the UAV navigation, obstacle detection and avoidance based on vision, and CPP were developed. The autonomous aerial system used a global planner to enter, exit, and navigate to the next tree, also integrates a local planner to navigate on the tree

rows. By their part, Freitas *et al.* [10] combined area coverage planning with path planning for biological pest control by releasing capsules with an UAV. The locations to release the capsules in the infected areas were calculated using the hexagon as base geometry. The capsule was placed only if its center was within the infected areas. Then, a capsule was placed in the center of mass of each remaining sub-polygon of the previous step. Whereas the path planning was achieved with ant colony optimization, guided local search, and Lin-Kernighan algorithms. Furthermore, Gao *et al.* [11] proposed a method to solve CPP for precise spraying in peach orchards. For that, a binocular color depth sensor was used to acquire video images. Then, a color depth fusion segmentation method based on the leaf wall area of the color depth images was proposed. Additionally, image erosion was used to delineate the two largest leaf wall areas as a ROI. The path of the spraying UAV was planned by detecting the midpoint of the ROI spacing as the end of the spray path. Also, Dong *et al.* [12] proposed an artificially weighted spanning tree coverage algorithm for the trajectory planning of flying robots. The robots simultaneously built their spanning tree, which grew toward the center of inertia of the uncovered area while stayed away from the trees of its partners. To go forward the tree, each robot iteratively evaluates the discovered cells and then selects the one having a maximum weight. According with this weighting, the selected cell is added to the spanning tree covering an area. Furthermore, Skorobogatov *et al.* [13] presented an algorithm open-source to divide any convex and non-convex polygon area into multiple parts, including any number of no-fly zones. Later, trajectories for the UAV were assigned. For the task, the UAV took pictures that were joined to obtain a complete map after a flight. The algorithm had as input a polygon defining the area of interest, the initial positions of the UAV, the mission parameters, and a con-vox divisor function based on Hert and Lumelsky algorithm. Moreover, Tang *et al.* [14] introduced a CPP method based on a Region Optimal Decomposition (ROD) using a multi-rotor UAV in a maritime port for concave polygons. They applied the ROD to a Google Earth image of a port and combining the resulting sub-regions through an improved depth-first-search algorithm. Then, a genetic algorithm determined the traversal order of all sub-regions and connect the coverage paths. By the part, Yao *et al.* [15] proposed a mission assignment scheme for the farmland spraying problem by using multi-quadcopters. To solve the problem a mathematical model for the mission assignment and a sequential quadratic programming method to obtain the optimal solution were used. Thus, quadcopters can

spray pesticides covering the farmland, but spraying is prohibited over areas covered by different quadcopters. Phung *et al.* [16] formulated the inspection path planning problem as an extended Travelling Salesman Problem (TSP) for an UAV. They introduced a discrete particle swarm optimization (DPSO) algorithm using deterministic initialization, random mutation, and edge exchange to resolve the problem. The UAV has a CCD camera to detect potential defects or damages in the inspected area and, then, find the shortest path for the inspection of the planar surface. They used parallel computing for the velocity, position, and aptitude of the particles. The parallel program was implemented on a Jetson board mounted on the UAV using the MAVLink protocol. Muliawan [17] *et al.* proposed a path planning approach for an UAV to carry out the spraying process in a plantation. A Modified Particle Swarm optimization (PSO) algorithm is used for the spraying task. The spraying process is carried out depending on the severity of disease of the plantation, which ranges from moderate to low. Recently, Gonzalez *et al.* [18] presented an approach to CPP for zigzag paths performed by an UAV in a three-dimensional environment. An optimization process based on the Differential Evolution (DE) algorithm is used in combination with the fast marching square scheduler. The UAV kept a fixed altitude to obtain images of a terrain, maintaining a homogeneous pixel size without overlapping. From the obtained images, a method to generate the zigzag path was used. Then, the DE algorithm optimizes zigzag path so that the steering angle of the UAV is optimal, ensuring a minimal distance cost. Campo *et al.* [19] proposed a data acquisition system using a low-cost Lightweight UAV (LUAV) with a camera to cover areas of interest and obtain a continuous map in crops. The LUAV optimized the coverage paths using a heuristic strategy, where a waypoint for the navigation of the LUAV agent was the center of the footprint of the camera. Zuo *et al.* [20] introduced a linear programming model to maximize coverage area and minimize coverage time for intelligence, surveillance, and reconnaissance missions with UAVs. In the first stage, a mission planner determined the search pattern, point of interest (POI), and the area of interest (AOI) for each UAV. In the second stage, the mission planner assigns some AOIs and flight paths for each UAV in the mission. Lastly, the aggregated mixed-integer linear programming for the path planning problem is solved by using the branch-and-bound algorithm in the CPLEX solver. Shang *et al.* [21] presented a co-optimal CPP approach to generate an aerial inspection path that optimized coverage of a 3D surface and quality of the captured images, and reduced the computational complex-

ity of the solver. The approach found the feasible paths for complete visual coverage. Later, a collection of the sampled paths were introduced in a PSO, which integrated the quality and efficiency of a coverage path in an objective function. Then, the calculated path was transferred to a flight trajectory using the rapidly exploring random tree to avoid obstacles. Tamayo *et al.* [22] designed and implemented a software system to plan low-cost drone coverage for surveillance in agricultural or forested polygonal areas. The user specifies a start location and charging station (CS) locations. Then, the drone took a video of the area and creates a binary grid representation of the observed field. To minimize time to loading stations and field configurations, the branch-and-bound algorithm was implemented, which finds a minimal set of CSs that minimized distance and decomposed the field into Voronoi regions. After that, it computed a path using the modified TSP algorithm and constructs the paths. More recently, Biundini *et al.* [23] presented a framework for CPP for inspection with UAVs. They designed a metaheuristic algorithm based on point cloud data to inspect structures and coupling 3D reconstructions. First, the camera of the UAV captured a moving image of the surface or ROI. The data of the structure is imported in point cloud or mesh format. Then, the data removed outliers and reduced the points of the optimization algorithm to identify the surface shape. After that, a genetic algorithm created a waypoint mission. Lastly, the path was sent to the UAV to start the flight, avoiding the UAV to waste energy. Popescu *et al.* [24] designed a remote system to determine flooded areas through image processing with fixed-wing type UAVs. The system integrated terrestrial and aerial components. The first one is a coordinator at a distance of the UAV, which has a ground control station that communicated with more ground data terminals, via internet, through a network of nodes for data acquisition and communication. The second component corresponded to mobile nodes and the UAV, which must perform area coverage and image processing tasks during the mission. Thus, a deep neural network was used for texture analysis, color extraction, selection, and classification to provide the segmented image and the relative size of the flood. Liu *et al.* [25] presented a navigation solution for UAVs working as a team. The UAVs were mobile base stations that flew around a ROI, performed the coverage task, and provided data services to a set of ground POIs. Each UAV needed to maintain connectivity with at least another UAV to avoid being isolated in the network. Then, a framework based on reinforcement learning was introduced to control each UAV. The UAVs were trained using environment state information, with the

objective being to maximize the coverage for each vehicle, maximize the geographical fairness of all considered POIs or service points, and minimize total power consumption without leaving the edge of the area using a reward function. Godio *et al.* [26] introduced an approach based on bio-inspired neural networks to solve a CPP problem for surveillance and exploration in critical areas with a fleet of rotary-wing UAVs. The bio-inspired neural network was based on the propagation of the neuron dynamics of unvisited areas in all the map to guide the vehicles toward unexplored locations. Each neuron had a local cost and connection with neighboring neurons, considering the unvisited areas, obstacles, and UAV position in the fleet.

On the other hand, papers concerning to spraying UAVs without considering CPP are introduced in the following. Huang *et al.* [27] developed a low volume spraying UAV to apply crop protection products on specific growing areas indicated to the MAV through GPS coordinates or pre-programmed locations. Qin *et al.* [28] developed a similar system to spray pesticides in fields and mountainous areas. Suryawanshi *et al.* [29] introduced a system for agricultural fields which included a water pump connected to a fertilizer tank monitored with a sensor. When a certain threshold was reached the user was notified to fill the tank. In the work of Rao *et al.* [30] a semi-autonomous agricultural spraying system for pesticides was developed. The spraying system was connected to a control board, which in turn was connected to an Arduino board, which generates pulses to activate a DC motor and thus spray the pesticide. By their part, Uddin *et al.* [31] designed a system to clean windows of high-rise buildings with an UAV (Quadcopter) based on open source autopilot software and a LIDAR sensor. The drone sprayed water and a microfiber brush did the washing of the windows. Vempati *et al.* [32] presented an UAV for painting 3D surfaces in desired locations with a spraying gun. A virtual reality interface is used to operate the system in a room, where the user could walk and interact with virtual objects. Zhang *et al.* [33] used a four-rotor drone to spray pesticides on sugarcane crop. The parameters taken into account were spraying volume and height and speed of the flight. Lastly, some trade spraying UAVs are described in Table 1.

Lastly, other important contributions associated with flight controllers for UAVs and spraying methods are found in [39, 40, 41] and [42, 43, 44, 45], respectively.

Table 1: Main features of commercial sprinklers.

Sprinkler	Features
Sprinkler SS600 Commercial Agriculture (Hexacopter) [34]	23 L tank. Covers up to 10 Ha/h. Integrates 4 nozzles. Flight time: 15-20min. 6 rotors of 100KV. Weight: 46.5kg. Payload: 24kg
Agras MG-1S Commercial agriculture (Octocopter)[35]	10 L tank. Covers up to 6 Ha/h. Integrates 4 nozzles. Flight time: 10-24min. 4 rotors of 130rpm. Weight: 8.0kg. Payload: 10kg
DJI Agras T16 Commercial Agriculture (Hexacopter)[36]	16 L tank. Covers up to 10 Ha/h. Integrates 8 nozzles. Flight time: 10-18min. 6 rotors of 75rpm. Weight: 18.5 kg. Payload:40.5kg.
DRONEHEXA-AG Phytosanitary treatments (Hexacopter)[37]	16 L tank. Covers up to 2 Ha/v. Integrates 4-8 nozzles. Flight time: 10-18min. Weight: 12.4 kg. Payload:32kg
Sprinkler 160AG GAIA Commercial Agriculture (Hexacopter)[38]	22.5 L tank. Covers up to 10 Ha/h. Integrates 4 nozzles. Flight time: 18min. 6 rotors of 100KV. Weight: 24 kg. Payload:46.5kg.

1.2. Proposal

After reviewing the literature on CPP for UAVs and spraying UAVs, it was found two fashions: i) The one focusing on solving the CPP problem for UAVs, and ii) the second one concerning the spraying task without considering the CPP problem. However, the first fashion is mainly dedicated to applications on surveillance, obstacle avoidance, inspection, mapping, and reconnaissance of areas. Only four papers were dedicated to the CPP problem addressing the spraying task [6, 11, 15, 17]. In general, for crop areas, they do not consider the model of the sprinkler **neither** the collisions that **may** occur in the surrounding area. Also, those papers do not model the

footprint of the sprinkler and some are limited to certain shape of the terrain to perform the spraying task. In order to contribute in this direction, particularly to provide an spraying system for the autonomous disinfection useful to face the pandemic by COVID-19, this paper integrates the aforementioned fashions, developing a CPP method which allows disinfecting convex regions with an UAV. The method consist on a polygonal decomposition through an improvement of the algorithms introduced in [46, 47, 48]. The improvement erodes the polygon area avoiding collisions and visits regions omitted by the original method. For that, the sprinkler is modeled as a 3D paraboloid whose intersection with the ground creates a disinfection footprint, which is an input parameter of the CPP algorithm. Another inputs of the algorithm are the kinematic model of the spraying, the shape of the terrain, and the UAV parameters. Then, the CPP algorithm computes the waypoints of the shortest coverage path including take-off and landing points for the area of interest. Lastly, the computed path is sent to an on-board controller via MAVLink to be executed by the UAV. To the authors' best knowledge, this is one of the first algorithms for automatically disinfecting urban areas that avoids collisions with the surrounding building.

1.3. Paper organization

The rest of the article is presented as follows. In section, 2 we present our novel method for path planning. Section 3 describes the experiments that were carried out including a software in the loop simulation due to mobility restrictions. Finally, section 4 presents our conclusion and future research directions.

2. Coverage path planning.

In this section, we present our method for automatically disinfect a two dimensional area with a unmanned aerial vehicle that carries a sprinkler. The proposed method is composed by several parts, see Fig. 2. First, we compute a sprinkler model using a regression approach that fits data obtained from the drops distribution to a paraboloid shape. Then, the sprinkler model the ROI shape and the vehicle's capabilities are used to plan a path. The path, given by a set of waypoints, is loaded to the on-board controller that guides the vehicle through the set of waypoints. Below, we first introduce the sprinkler modeling and then we present the coverage path planning.

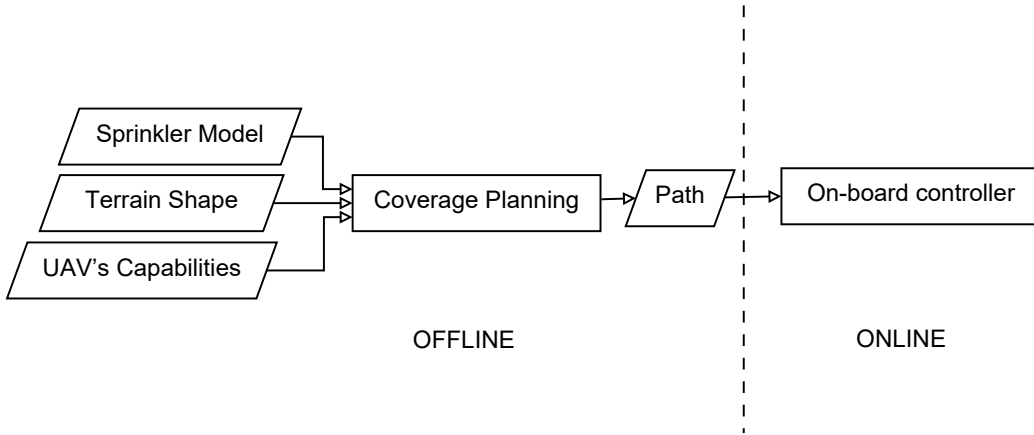


Figure 2: General workflow diagram of coverage path planning for area disinfection.

2.1. Sprinkler modeling

There are several ways to model a sprinkler system depending on the application and system characteristics [49]. In this work, we assume that only one sprinkler is mounted on the MAV and that the vehicle’s altitude is relative small; for closed areas, the ceiling limits the altitude ; and for open areas the flight altitude should be also small, otherwise the liquid effectiveness can be compromised. See Fig. 3. Therefore, we model the kinematics of the sprinkler’s coverage field as a rigid body governed by the equations of a inverse paraboloid:

$$z = -Ax^2 - By^2 + h \quad (1)$$

where h is the drone altitude and A and B are parameters that define the amplitude of the paraboloid. Fig. 3 shows the proposed model.

This model is closer to the reality w.r.t. previous models such as squares or cones. However, a deterministic model is still limited because in real sprinklers, the dispersion effect is much more complex given that many variables contribute to the drop’s fall [49]. To insert such variability in our model, we consider the drop’s fall as a stochastic process that affects the paraboloid shape, in consequence, we update eq. (1) with a random noise.

$$z' = -Ax^2 - By^2 + h + \epsilon_\sigma \quad (2)$$

where ϵ_σ is a random error from a normal distribution with zero mean, $\mu = 0$, and standard deviation equal to σ . Fig. 4b shows an example of this noisy

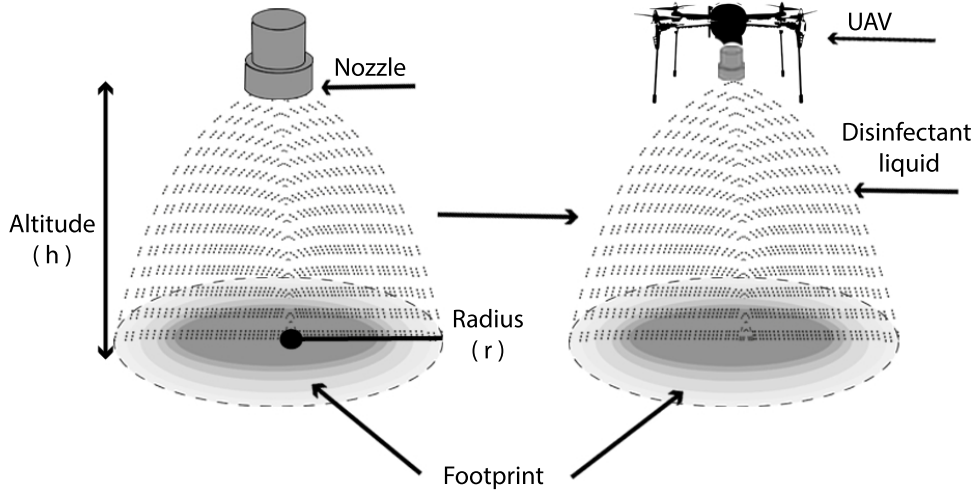


Figure 3: Paraboloid model of the disinfecting area. The footprint of the sprinkler is given by a circle of radius equal to r .

model.

In many cases, the sprinkler is not characterized by the maker, however, a model fitting can be carried out so that the parameters A and B can be inferred using data from the drop's fall for a given sprinkler. In this method, we use the Levenberg-Marquardt Algorithm [50] for finding the paraboloid parameters (\hat{A}, \hat{B}) . Fig. 4c shows an example of a fitted model for a given set of observed drops.

Next, let us assume that the terrain is horizontal, therefore, given the position of the vehicle, the spray footprint on the ground is determined by the intersection of the fitted paraboloid, equation (1), and the floor plane. In general, such footprint is described by an ellipsoidal shape. However, we restrict the intersection between the paraboloid and the floor to a circle. Look at Fig. 3. The circle is the biggest one that fits into the ellipsoidal intersection. This restriction has several vantages: i) the plan is conservative, meaning that in the worst case more area is disinfected, ii) the path planning is simplified and iii) the processing efficiency is improved. A more detailed plan considering irregular footprints is left for future work. In summary, the footprint forms a circle, where its limit is given by the circumference with

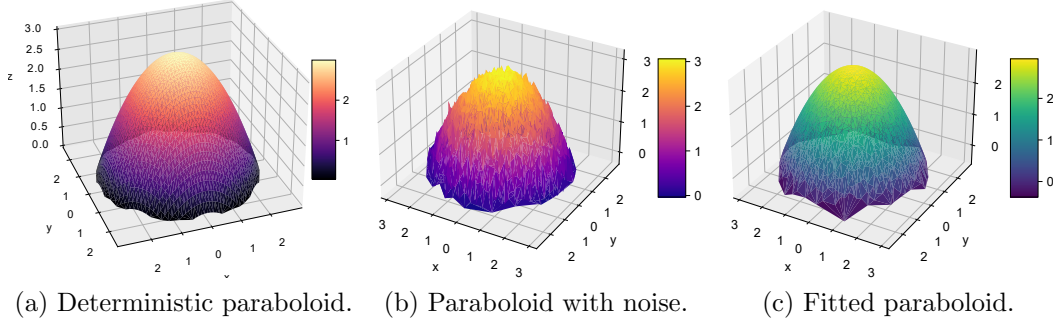


Figure 4: Sprinkler model based on paraboloid equations for a MAV at three meters. The paraboloid bounds the volume covered by the sprinkler. Units are in meters. Figure best seen in color.

equation:

$$r^2 = (x - a)^2 + (y - b)^2 \quad (3)$$

where r is the circle’s radius. The proof of the intersection between the paraboloid and the plane is straightforward, so it is omitted. Then, from now on, the area that is sprayed inside the circle will be called as the drone’s footprint. Once the vehicle is moving, the footprint evolves to a disinfected lane.

2.2. Path planning

Once the footprint has been established, the coverage path planning is defined as computing a vehicle’s path in such a way that the ROI is completely covered by the footprint. The ROI is defined as a convex polygon, whose area is bigger than the footprint. Otherwise, the path is a single point. In addition, a constraint is that the vehicle must stay inside the ROI, e.g., for indoor scenes, the robot must avoid collision with doors and walls.

To solve the problem, we propose a method based on back and forth paths. This strategy simplifies the problem to finding a path composed by straight movements, called flight lines, in a similar way than a boustrophedon covers a terrain [51]. The method is summarized in algorithm 1 and described next. The requirements are the polygon (M), the footprint radius (r), the takeoff point (s) and the landing point (e). First, the method establishes the distance between flight lines, δ , as two times the foot print radius. By setting

Algorithm 1: Coverage path planning for spraying drones.

Data: Polygon (M), radius (r), starting point or home (s), final point(e), distance between lines (δ)

Result: Ruta (ρ)

$\delta \leftarrow 2r$;

$M' \leftarrow \text{Erode}(M, r)$;

$\rho \leftarrow \text{RCPP}(M', \delta, s, e)$;

$\rho \leftarrow \rho \cup \text{Corners}(M')$;

$\delta = 2r$, we guarantee that there is no gaps between flight of lines. Unlike previous literature [46], we do not require overlap between flight lines since we assume that a single exposition to the spray is enough for disinfecting the area. Second, the ROI's polygon is eroded by a circular kernel of diameter equal to r . See Fig. 5a. The new polygon is called M' . M' facilitates the planning given that its frontiers can be reached by the MAV without colliding, in addition, the erosion diameter is equal to r because when the drone reaches the frontier of M' the footprint is touching the frontier of M . Then, we use a modified version of the rotating calipers path planner (RCPP)[48] for computing a path, ρ , that covers M' . See Fig. 5b. RCPP optimizes the flight of lines direction so that it gets the optimal path in linear time w.r.t. the number of vertices in the polygon. In our implementation, we connect flight lines over the frontier of M' , unlike the original method where the flight lines are connected by perpendicular sections, such perpendicularity might lead the MAV outside M' , in consequence the drone could collide with the environment. Finally, we modify the computed path by adding a tour of the M' vertices before e , this tour covers some areas left in the path ρ which is an implicit defect of the back-and-forth planners that deal with delimited regions [5]. See Fig. 5c.

3. Experiments

In this section, we present several experiments for validating our method. First, we test the method with several polygon shapes, then, we simulate the execution of the paths using the Gazebo simulator. At the same time, we compare the computed paths against the Torres' algorithm [46].

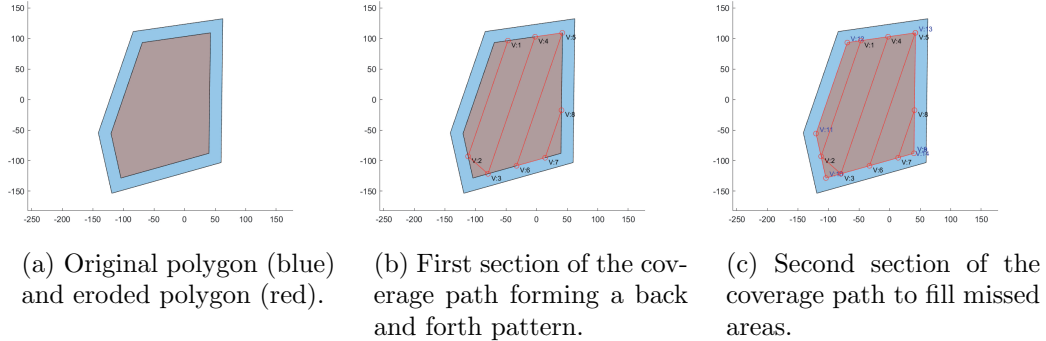


Figure 5: Illustrations of the path computed by the proposed algorithm.

3.1. Random polygons experiment

In this experiment, the proposed method is tested with six different polygons, the polygons were constructed randomly but the planning parameters were set by using close to reality parameters. The objective is to show that the method can deal with several shapes. The polygons and computed paths are displayed in Fig. 6. Additionally, the distance traveled and the flight time in simulation are shown for each random polygon using the Torres’ algorithm [46] which is a popular choice for CPP.

Table 2 summarizes the results of the method with six polygons that have a different number of vertices and a constant sensor radius. The table also includes the data obtained from SITL(software in the loop) simulation with Gazebo. The columns in the table display the name of the polygon, the number of vertices, the sensor radius, the distance between flight lines, the total traveled distance of our method, the simulation flight time of our method, the total traveled distance of the computed path by Torres and the simulation flight time of the Torres’ path.

In Figure 6, we can see that the paths cover the polygons and that they are oriented to reduce the distance traveled. The erosion function is carried out on the polygon, represented by the border in blue, in order to avoid collisions during the vehicle’s trajectory. The processing time for the proposed approach is 0.06 seconds in average, while the time for Torres’s algorithm is 0.136 seconds.

The algorithm works for different shapes or different number of vertices. It is also observed that to the right of each polygon are the polygons generated

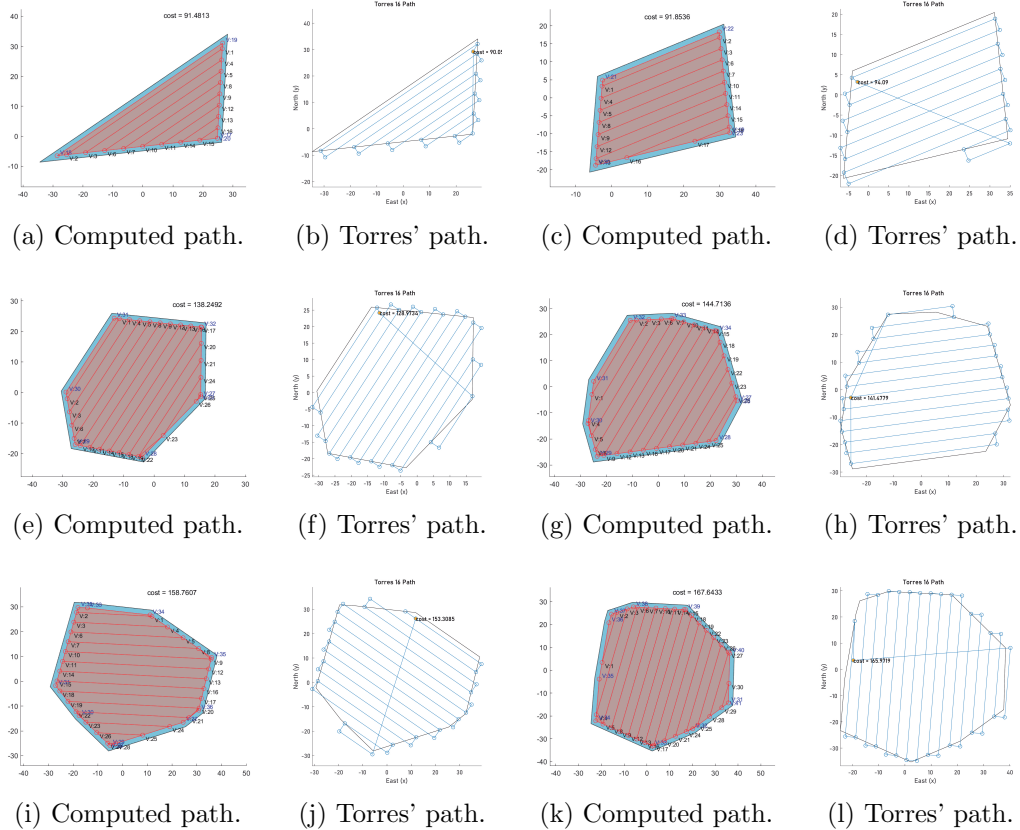


Figure 6: Comparison of computed paths by Torres et al. [46] versus our proposed approach. Torres paths are displayed in blue lines and our computed paths are displayed in red lines. Figure best seen in color.

with Torres' algorithm [46], the paths are represented in lines in blue, flight lines go outside of the area of the polygon, which would represent collisions in a real environment.

3.2. 3D simulation

The objective of this experiment is to validate that the method works in a more realistic simulation. See Fig. 7. Due to the current situation, it was not possible to run the experiments in a real vehicle. However, the Gazebo simulator was configured with closest to reality parameters. In addition, we want to provide a qualitative display of the planned task.

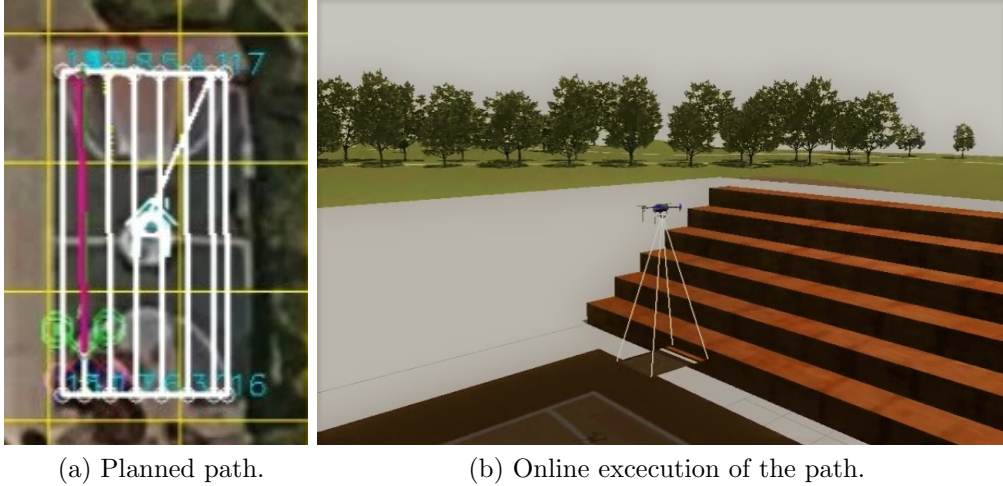


Figure 7: Software in the loop simulation (SITL) where the vehicle in Gazebo simulator is commanded through the MAVLink protocol. Once the path is loaded in the on-board controller, it flights autonomously.

We draw a polygon on the edge of the area surrounding one of the Bicentennial Park basketball courts located in Mexico City using the Mission Planner software. Later, the coordinates define the ROI. Next, the coordinates are converted from geodetic coordinates (latitude, longitude, and height) to NED coordinates (North, East, and Down). The erosion function in the algorithm is in charge of reducing the border of the area that will depend on the radius of the sensor, and once it has the area, it continues to the creation of the flight lines. However, it is required to visit all the areas so that as much as possible is covered. The algorithm adds the nodes of all the polygon vertices to cover areas without visiting. Then a conversion from NED coordinates to geodetic coordinates is performed again to add these coordinates to a waypoints file. The Gazebo simulator uses the waypoints file to carry out the path over the court.

Subsequently, for this simulation, SITL was used in collaboration with the Gazebo 9 simulator to see the area in 3D. In this simulation, the Quadcopter 3DR Iris was used. The height set during the simulation was 10 meters. The speed of the drone is $2\text{m} / \text{s}$. The start takeoff set with the coordinates $19.467468, -99.193345$. Additionally, a basketball court model made for Gazebo allows to evaluate the behavior of the UAV in a virtual environment.

Table 2: Results for each tested polygon. The columns indicate N for number of vertices, r for sensor radius, δ for distance between flight lines, sim. t for simulation flight time and d total for total traveled distance,

Polygon	N	r	δ	D to- tal	sim. t	D total [46]	sim t [46]
A (6a)	3	1.5	3.0	91.4m	6:09 min	90.0m	6:20 min
B (6c)	4	1.5	3.0	91.8m	5:44 min	94.0m	5:26 min
C (6e)	6	1.5	3.0	138.2m	7:53 min	128.9m	8:11 min
D (6g)	8	2.0	4.0	144.7m	8:56 min	141.4m	9:20 min
E (6i)	9	2.0	4.0	158.7m	9:30 min	153.3m	9:36 min
F (6k)	10	2.0	4.0	167.6m	10:07 min	165.9m	10:17 min

It was observed in the experiment that the vehicle covered most of the basketball court area without going out of the contour, thus avoiding some collision. In total, it covered an area of 59.4966 meters in a time of 3:46 minutes. In the video, you can see the detailed simulation of this experiment.

3.3. Discussion

The distances computed by both methods are similar and it cannot be declared that one algorithm is better in terms of total distance. However, our algorithm makes a final tour around the area, to cover missing areas, in addition, this proposal, the vehicle does not go outside the ROI compared versus Torres et al.[46]. This is an important characteristic, because our method will avoid collisions in a real scene. About the simulation time, our algorithm performs the path in less time, which represents an advantage.

4. Conclusions

A method for planning a two dimensional area disinfection with a micro aerial vehicle has been presented. Our method extends previous research by including a more precise sprinkler modeling and restricts the vehicle flight inside the region of interest. Through several experiments, we have validated our method; in those experiments, we observed that the vehicle maximizes the area of disinfection in despite of the area shape. The method has shown to be efficient enough for being implemented in on-board systems with low computational capacity. For future work, we plan to extend our method for non-convex polygons and to carry out experiments with a physical platform.

References

- [1] A. Bhargava, A. Bansal, Novel coronavirus (covid-19) diagnosis using computer vision and artificial intelligence techniques: a review, *Multi-media Tools and Applications* (2021) 1–16.
- [2] B. Ramalingam, J. Yin, M. R. Elara, Y. K. Tamilselvam, M. M. Rayguru, M. A. V. J. Muthugala, B. F. Gomez, A Human Support Robot for the Cleaning and Maintenance of Door Handles Using a Deep-Learning Framework, *Sensors* 20 (12) (JUN 2020). doi: {10.3390/s20123543}.
- [3] A. Sharma, P. Zanotti, L. P. Musunur, Drive Through Robotics: Robotic Automation for Last Mile Distribution of Food and Essentials During Pandemics, *IEEE Access* 8 (2020) 127190–127219. doi: {10.1109/Access.2020.3007064}.
- [4] R. Al Jaber, M. S. Sikder, R. A. Hossain, K. F. N. Malia, M. A. Rahman, Unmanned aerial vehicle for cleaning and firefighting purposes, in: 2021 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), IEEE, 2021, pp. 673–677.
- [5] E. Galceran, M. Carreras, A survey on coverage path planning for robotics, *Robotics and Autonomous systems* 61 (12) (2013) 1258–1276.
- [6] S. M. M.E, R. Maguteeswaran, N. G. BE, G. Srinivasan, Quadcopter uav based fertilizer and pesticide spraying system, *International Academic Research Journal of Engineering Sciences*, ISSN 1 (2414-6242) (2016).

- [7] J. Keller, D. Thakur, M. Likhachev, J. Gallier, V. Kumar, Coordinated Path Planning for Fixed-Wing UAS Conducting Persistent Surveillance Missions, *IEEE Transactions on Automation Science and Engineering* 14 (1) (2017) 17–24. doi:{10.1109/TASE.2016.2623642}.
- [8] Z. Xiao, B. Zhu, Y. Wang, P. Miao, Low-Complexity Path Planning Algorithm for Unmanned Aerial Vehicles in Complicated Scenarios, *IEEE Access* 6 (2018) 57049–57055. doi:{10.1109/Access.2018.2873084}.
- [9] N. Stefan, H. Bayram, V. Isler, Vision-based monitoring of orchards with UAVs, *Computers and Electronics in Agriculture* 163 (Aug 2019). doi:{10.1016/j.compag.2019.05.023}.
- [10] H. Freitas, B. S. Faical, A. V. Cardoso e Silva, J. Ueyama, Use of UAVs for an efficient capsule distribution and smart path planning for biological pest control, *Computers and Electronics in Agriculture* 173 (JUN 2020). doi:{10.1016/j.compag.2020.105387}.
- [11] G. Gao, K. Xiao, Y. Jia, A spraying path planning algorithm based on colour-depth fusion segmentation in peach orchards, *Computers and Electronics in Agriculture* 173 (2020) 105412. doi:https://doi.org/10.1016/j.compag.2020.105412.
- [12] W. Dong, S. Liu, Y. Ding, X. Sheng, X. Zhu, An Artificially Weighted Spanning Tree Coverage Algorithm for Decentralized Flying Robots, *IEEE Transactions on Automation Science and Engineering* 17 (4) (2020) 1689–1698. doi:{10.1109/TASE.2020.2971324}.
- [13] G. Skorobogatov, C. Barrado, E. Salami, E. Pastor, Flight planning in multi-unmanned aerial vehicle systems: Nonconvex polygon area decomposition and trajectory assignment, *International Journal of Advanced Robotic Systems* 18 (1) (JAN 2021). doi:{10.1177/1729881421989551}.
- [14] G. Tang, C. Tang, H. Zhou, C. Claramunt, S. Men, R-DFS: A Coverage Path Planning Approach Based on Region Optimal Decomposition, *Remote Sensing* 13 (8) (APR 2021). doi:{10.3390/rs13081525}.
- [15] Luo Yao, Yan Jiang, Zhao Zhiyao, Yang Shuaishuai, Quan Quan, A pesticide spraying mission assignment performed by multi-quadcopters and

- its simulation platform establishment, in: 2016 IEEE Chinese Guidance, Navigation and Control Conference (CGNCC), 2016, pp. 1980–1985.
- [16] M. D. Phung, C. H. Quach, T. H. Dinh, Q. Ha, Enhanced discrete particle swarm optimization path planning for UAV vision-based surface inspection, *Automation in Construction* 81 (2017) 25–33. doi:{10.1016/j.autcon.2017.04.013}.
- [17] I. W. Muliawan, M. A. Ma’sum, N. Alfiany, W. Jatmiko, Uav path planning for autonomous spraying task at salak plantation based on the severity of plant disease, in: 2019 IEEE International Conference on Cybernetics and Computational Intelligence (CyberneticsCom), 2019, pp. 109–113.
- [18] V. Gonzalez, C. A. Monje, S. Garrido, L. Moreno, C. Balaguer, Coverage Mission for UAVs Using Differential Evolution and Fast Marching Square Methods, *IEEE Aerospace and Electronic Systems Magazine* 35 (2) (2020) 18–29. doi:{10.1109/MAES.2020.2966317}.
- [19] L. Viviana Campo, A. Ledezma, J. Carlos Corrales, Optimization of coverage mission for lightweight unmanned aerial vehicles applied in crop data acquisition, *Expert Systems with Applications* 149 (JUL 1 2020). doi:{10.1016/j.eswa.2020.113227}.
- [20] Y. Zuo, R. Tharmarasa, R. Jassemi-Zargani, N. Kashyap, J. Thiya-galingam, T. T. Kirubarajan, MILP Formulation for Aircraft Path Planning in Persistent Surveillance, *IEEE Transactions on Aerospace and Electronic Systems* 56 (5) (2020) 3796–3811. doi:{10.1109/TAES.2020.2983532}.
- [21] Z. Shang, J. Bradley, Z. Shen, A co-optimal coverage path planning method for aerial scanning of complex structures, *Expert Systems with Applications* 158 (NOV 15 2020). doi:{10.1016/j.eswa.2020.113535}.
- [22] L. V. Tamayo, C. Thron, J. L. K. E. Fendji, S.-K. Thomas, A. Foerster, Cost-Minimizing System Design for Surveillance of Large, Inaccessible Agricultural Areas Using Drones of Limited Range, *Sustainability* 12 (21) (NOV 2020). doi:{10.3390/su12218878}.

- [23] I. Z. Biundini, M. F. Pinto, A. G. Melo, A. L. M. Marcato, L. M. Honorio, M. J. R. Aguiar, A Framework for Coverage Path Planning Optimization Based on Point Cloud for Structural Inspection, *Sensors* 21 (2) (JAN 2021). doi:{10.3390/s21020570}.
- [24] D. Popescu, L. Ichim, F. Stoican, Unmanned Aerial Vehicle Systems for Remote Estimation of Flooded Areas Based on Complex Image Processing, *Sensors* 17 (3) (MAR 2017). doi:{10.3390/s17030446}.
- [25] C. H. Liu, X. Ma, X. Gao, J. Tang, Distributed Energy-Efficient Multi-UAV Navigation for Long-Term Communication Coverage by Deep Reinforcement Learning, *IEEE Transactions on Mobile Computing* 19 (6) (2020) 1274–1285. doi:{10.1109/TMC.2019.2908171}.
- [26] S. Godio, S. Primatesta, G. Guglieri, F. Dervis, A Bioinspired Neural Network-Based Approach for Cooperative Coverage Planning of UAVs, *Information* 12 (2) (FEB 2021). doi:{10.3390/info12020051}.
- [27] Y. Huang, W. C. Hoffmann, Y. Lan, W. Wu, B. K. Fritz, Development of a spray system for an unmanned aerial vehicle platform, *Applied Engineering in Agriculture* 25 (6) (2009) 803–809.
- [28] W.-C. Qin, B.-J. Qiu, X.-Y. Xue, C. Chen, Z.-F. Xu, Q.-Q. Zhou, Droplet deposition and control effect of insecticides sprayed with an unmanned aerial vehicle against plant hoppers, *Crop Protection* 85 (2016) 79–88.
- [29] V. K. Suryawanshi, J. Ashok, S. A. Rajmane, S. S. Mali, Design & development of agricultural fertilizer spraying drone with remote controller and autonomous control with low weight aluminium alloy frame structure, *Journal of Remote Sensing GIS & Technology* 5 (2019). doi:10.5281/zenodo.2631047.
- [30] V. P. Subba Rao, G. S. Rao, Design and modelling of an affordable uav based pesticide sprayer in agriculture applications, in: 2019 Fifth International Conference on Electrical Energy Systems (ICEES), 2019, pp. 1–4.
- [31] S. M. Uddin, M. R. Hossain, M. S. Rabbi, M. A. Hasan, M. S. R. Zishan, Unmanned aerial vehicle for cleaning the high rise buildings, in: 2019

International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), IEEE, 2019, pp. 657–661.

- [32] A. S. Vempati, H. Khurana, V. Kabelka, S. Flueckiger, R. Siegwart, P. Beardsley, A virtual reality interface for an autonomous spray painting uav, *IEEE Robotics and Automation Letters* 4 (3) (2019) 2870–2877.
- [33] X.-Q. Zhang, X.-P. Song, Y.-J. Liang, Z.-Q. Qin, B.-Q. Zhang, J.-J. Wei, Y.-R. Li, J.-M. Wu, Effects of spray parameters of drone on the droplet deposition in sugarcane canopy, *Sugar Tech* (2020) 1–6.
- [34] Skysolutions.mx, Servicio de agricultura de precisión con drones: Sky solutions méxico, [Online]. Available: <https://www.skysolutions.mx/agricultura>, [Accessed: 01-May-2020] (2020).
- [35] Drone, C., Agras mg-1s drtk drone para agricultura, [Online]. Available: <https://cielito-drone.myshopify.com/products/agras-mg-1-dron-para-agricultura>, [Accessed: 01-May-2020] (2020).
- [36] Tuxtla, Hobby, Dji t16 dron para fumigar de 16l, [Online]. Available: https://www.hobbytuxtla.com/drones-agricultores/dji-agras-t16/?utm_source=tienda, [Accessed: 02-May-2020] (2019).
- [37] S. User, Uav vtol and nbsp;dronehexaag, [Online]. Available: <https://www.dronetools.es/index.php/dronehexa-ag>, [Accessed: 02-May-2020] (2020).
- [38] F , Foxtech, Gaia 160ag-agricultural spraying drone arf combo, [Online]. Available: <https://www.foxtechfpv.com/gaia-160-ag-hexacopter-arf-combo.html>, [Accessed: 02-May-2020] (2008).
- [39] H. Zhu, Y. Lan, W. Wu, W. C. Hoffmann, Y. Huang, X. Xue, J. Liang, B. Fritz, Development of a pwm precision spraying controller for unmanned aerial vehicles, *Journal of Bionic Engineering* 7 (3) (2010) 276–283.

- [40] X. Xue, Y. Lan, Z. Sun, C. Chang, W. C. Hoffmann, Develop an unmanned aerial vehicle based automatic aerial spraying system, *Computers and electronics in agriculture* 128 (2016) 58–66.
- [41] Spoorthi S., B. Shadaksharappa, Suraj S., V. K. Manasa, Freyr drone: Pesticide/fertilizers spraying drone - an agricultural approach, in: 2017 2nd International Conference on Computing and Communications Technologies (ICCCT), 2017, pp. 252–255.
- [42] Y. Meng, J. Song, Y. Lan, G. Mei, Z. Liang, Y. Han, Harvest aids efficacy applied by unmanned aerial vehicles on cotton crop, *Industrial Crops and Products* 140 (2019) 111645.
- [43] G. Potrino, N. Palmieri, V. Antonello, A. Serianni, Drones support in precision agriculture for fighting against parasites, in: 2018 26th Telecommunications Forum (TELFOR), 2018, pp. 1–4.
- [44] Y. Meng, Y. Lan, G. Mei, Y. Guo, J. Song, Z. Wang, Effect of aerial spray adjuvant applying on the efficiency of small unmanned aerial vehicle for wheat aphids control, *International Journal of Agricultural and Biological Engineering* 11 (5) (2018) 46–53.
- [45] G. Wang, Y. Lan, H. Qi, P. Chen, A. Hewitt, Y. Han, Field evaluation of an unmanned aerial vehicle (uav) sprayer: effect of spray volume on deposition and the control of pests and disease in wheat, *Pest management science* 75 (6) (2019) 1546–1555.
- [46] M. Torres, D. A. Pelta, J. L. Verdegay, J. C. Torres, Coverage path planning with unmanned aerial vehicles for 3D terrain reconstruction, *Expert Systems with Applications* 55 (2016) 441–451. doi:{10.1016/j.eswa.2016.02.007}.
- [47] J. I. Vasquez-Gomez, J.-C. Herrera-Lozada, M. Olguin-Carbajal, Coverage path planning for surveying disjoint areas, in: 2018 International Conference on Unmanned Aircraft Systems (ICUAS), IEEE, 2018, pp. 899–904.
- [48] J. I. Vasquez-Gomez, M. Marciano-Melchor, L. Valentin, J. C. Herrera-Lozada, Coverage path planning for 2d convex regions, *Journal of Intelligent & Robotic Systems* 97 (1) (2020) 81–94.

- [49] D. J. Grose, Mathematical modelling and simulation of irrigation sprinklers, Ph.D. thesis, Cranfield University (1999).
- [50] K. Levenberg, A method for the solution of certain non-linear problems in least squares, Quarterly of applied mathematics 2 (2) (1944) 164–168.
- [51] H. Choset, P. Pignon, Coverage path planning: The boustrophedon cellular decomposition, in: Field and service robotics, Springer, 1998, pp. 203–209.