



**HAL**  
open science

## Complete coverage path planning for pests-ridden in precision agriculture using UAV

The Hung Pham, Dalil Ichalal, Saïd Mammar

► **To cite this version:**

The Hung Pham, Dalil Ichalal, Saïd Mammar. Complete coverage path planning for pests-ridden in precision agriculture using UAV. 17th IEEE International Conference on Networking, Sensing and Control (ICNSC 2020), Oct 2020, Nanjing (on line ), China. pp.1-6, 10.1109/ICNSC48988.2020.9238122 . hal-03028878

**HAL Id: hal-03028878**

**<https://hal.science/hal-03028878v1>**

Submitted on 2 Mar 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Complete coverage path planning for pests-ridden in precision agriculture using UAV

<sup>1</sup>The Hung PHAM, <sup>2</sup>Dalil ICHALAL, <sup>3</sup>Said MAMMAR

*IBISC*

*Evry-Val-d'Essonne University, Universite Paris-Saclay*

Evry, France

<sup>1</sup>thehung.pham@univ-evry.fr, thehungpham83@gmail.com; <sup>2</sup>dalil.ichalal@univ-evry.fr; <sup>3</sup>said.mammar@univ-evry.fr

**Abstract**—The contribution of this work focuses on generating the best path for an UAV to distribute medicine to all the infected areas of an agriculture environment which contains non-convex obstacles, pest-free areas and pests-ridden areas. The algorithm for generating this trajectory can save the working time and the amount of medicine to be distributed to the whole agriculture infected areas.

From the information on the map regarding the coordinates of the obstacles, non-infected areas, and infected areas, the infected areas are divided into several non-overlapping regions by using a clustering technique. There is a trade-off between the number of classes generated and the area of all the pests-ridden areas. After that, a polygon will be found to cover each of these infected regions. However, obstacles may occupy part of the area of these polygons that have been created previously. Each polygon that is occupied in part by obstacles can be further divided into a minimum number of obstacle-free convex polygons. Then, an optimal path length of boustrophedon trajectory will be created for each convex polygon that has been created for the UAV to follow. Finally, this paper deals with the process of creating a minimal path for the UAV to move between all the constructed convex polygons and generate the final trajectory for the UAV which ensures that all the infected agriculture areas will be covered by the medicine.

The algorithm of the proposed method has been tested on MATLAB and can be used in precision agriculture.

**Index Terms**—Coverage Path planning, UAVs, Precision Agriculture, Convex Polygon Decomposition, Clustering, K-means.

## I. INTRODUCTION AND STATEMENT OF THE PROBLEM

Currently, humans are facing a very serious challenge in sustainable energy and food production because of limited land resources on the planet and a rapidly growing global population. Therefore, with the same agricultural area and the environmental impacts as well as diseases, crop yields need to be improved. To improve crop yields, we need to create more varieties that can tolerate weather or disease, and besides that, the crop status should also be monitored more frequently. Many types of agricultural robots have been researched and built by scientists. They are widely used to obtain information about crop health, fertilization or harvesting, such as a mobile robot [1] [2], drone (UAV) [3] [4] or the cooperation of both [5].

Due to the capability of high maneuverability and low cost, UAVs are increasingly expanding to commercial, scientific, entertainment and other applications, such as security and

surveillance, product delivery, aerial photography, especially in agriculture.

UAV applications in agricultural research are becoming more noticeable in the literature. In [6], the authors evaluated an aerobatic model plane for high-resolution digital photography used to estimate the nutritional status of corn and crop biomass of corn, alfalfa, and soybeans

A very common application of UAVs in agricultural tasks is crop disease detection. Disease detection using UAV is usually performed with a versatile camera, ultrasound or thermal imaging carried on-board to detect abnormal levels of radiation in the infrared spectrum emitted by plants. The images and spectral data collected are often analyzed at an earth station. Data collection with unmanned aircraft fills the gaps on the observation scale in remote sensing by providing the high spatial and temporal resolution data required in monitoring plant growth. The gathered information about the state of plants facilitates the detection and quantification of intra-field variability to support agricultural management decisions such as the effective application of fertilizers. Physiological parameters such as plant height and biomass are monitored to describe crop growth and act as an indicator of final crop yields.

To monitor or spray a robot-grown plant, a trajectory needs to be generated for the robot. This trajectory must meet a number of properties, such as, (i) ensuring that robots will monitor or spray the entire area of the plant to be monitored while avoiding obstacles, (ii) optimizing the area to be monitored, and/or (iii) optimize the travel distance or working time of the robot.

Coverage path planning (CPP) is one of the most important tasks for robot movement. CPP is the identification of a trajectory that a robot must follow to pass each point in an environment while avoiding obstacles. Based on the generated trajectory, the robot can accomplish pre-defined tasks. For generating trajectory for the robots, first, the environment needs to be divided into smaller obstacle-free regions, then the trajectory in each region has to be generated. CPP has been extensively studied in recent years for applications such as vacuum cleaning robots [7], painter robots [8], path planning for autonomous underwater vehicles [9], demining robots [10], lawn mowers [11].

The purpose of this study is to use UAVs to spray pesticides

on all infected agricultural areas with non-convex obstacles while minimizing the amount of medication needed and the trajectory of the robot.

The remainder of this paper is organized as follows. In Section II, the problem is stated and decomposed in several steps. In Section III, the system model is introduced. Based on the coordinates of infected areas and obstacles, the obstacle-free infected areas are generated in section IV. Then, the trajectory for the UAV to cover entire the infected areas is calculated in section V. Section VI is dedicated to simulation results. The section VII concludes the paper and gives some suggestions for future works.

## II. PROBLEM STATEMENT AND DECOMPOSITION

To fix ideas, suppose that the agricultural area of interest is as depicted in Fig. 1. In this figure, there are obstacles (the green areas), uninfected plants (the white areas), and infected plants (the dotted red area). One can see that the infected area are close to some obstacles and an UAV which wants to move from one area to another will meet an obstacle on its path. To tackle the stated problem, we propose the algorithm illustrated in Fig. 2. In view of this algorithm, the problem could be subdivided into four main tasks.

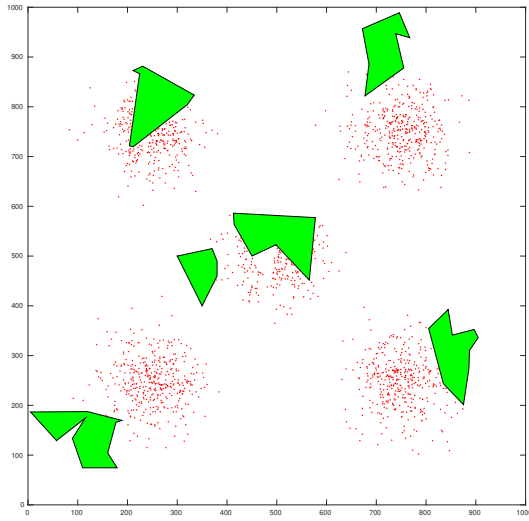


Figure 1: Problem definition

- The first task is MAP DIVISION (MD). From the information of the map about the coordinates of obstacles and infected areas, we divide the infected areas into several non-overlapping regions (clusters) using a clustering technique [12]. Intuitively, the smallest shape covering each cluster is formed by the boundary of the cluster. Nevertheless, this shape does not necessarily minimize the trajectory of the UAVs. Hence we propose another approach for this cluster-covering step. In particular, we characterize the minimal convex polygon covering each cluster. Note that to ultimately determine the trajectory of the UAVs, we need to divide each covering polygon into convex regions. For this reason, the number of

convex regions produced by the former approach might be larger than that of the later. In fact, which method is more efficient (i.e. producing fewer number of convex regions, smaller coverage region and shorter trajectory of the UAVs) is not known a priori; this is problem-specific. We provide an algorithm for the comparison in the simulation section of the paper. However, the covering polygon of each cluster might be occupied by obstacles, which we eliminate using available coordinates data. Then, we divide each obstacles-free covering polygon into a minimum number of convex polygons.

- The second task is WAYPOINTS INFECTED AREA (WIA). We create the way-points for each obstacle-free convex polygon (generated from the MD task) such that when the UAV follows these points, the whole area of the obstacle-free convex polygon will be covered.
- The third task is TRAJECTORY GENERATION (TG). This task aims at creating a boustrophedon trajectory for each convex polygon that has been created in MD. After that, by applying the Traveling Salesman Problem (TSP) method, we can find the shortest path for the aerial robot to cover the whole polygon.
- The fourth task consists in finding the best path for the UAV to change between infected regions without colliding with obstacles.

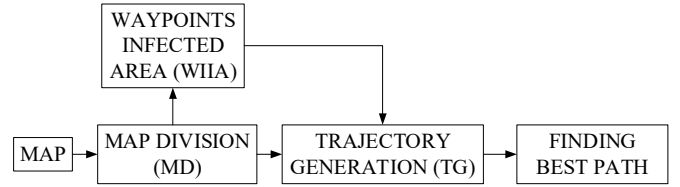


Figure 2: Path planning Algorithm

## III. SYSTEM MODEL

### A. UAV platform

This study deals with UAVs that are equipped with a mechanism that can spray the crop. The distance from the UAV to the fixed plant under  $h$  and a radial area of radius  $R$  (Fig. 3) of the crop below the UAV can be covered by the medicine.

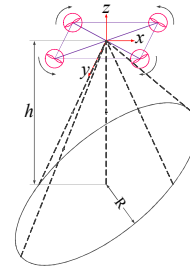


Figure 3: UAV with frame picture capture

Without loss of generality, we suppose that the working area is rectangular. If the area is not a rectangle, it is possible to form a rectangular area by adding virtual obstacles to the original area.

### B. Augmented the size of the quadcopter to obstacles

In the process of performing its tasks, the UAV has to avoid the obstacles. Therefore, the distance from the center of the UAV to any point in the obstacles should be greater or equal to the radius of the smallest circle which covers the UAV. For simplicity, consider the UAV as a point, it is necessary to augment the size of the obstacles by a radius depending on the shape of the drone. At each border point of the obstacle, the operator draws a circle with radius  $r$  and creates the new obstacle inside the red curve (in Fig. 4a). However, the augmented obstacle now is no longer a polygon.

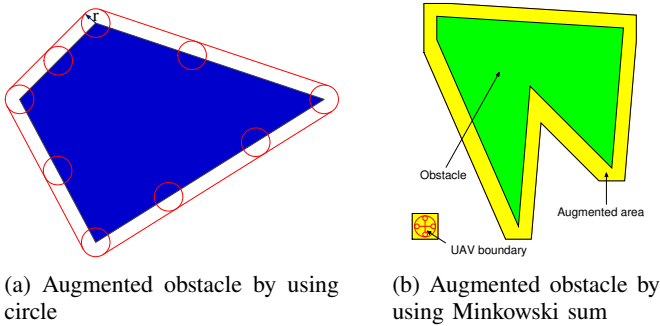


Figure 4: Augmentation of the size of obstacles

In this study, we use another solution to augment the size of obstacle such that the augmented obstacle is a polygon. Suppose that the shape of UAV is a square that its edge is greater than diameter of the UAV (Fig. 4b). Let us denote the set of vertices of this square by  $A$ , and the set of vertices of the obstacles by  $B$ . The augmented obstacle is generated is the Minkowski sum [13] of  $A$  and  $B$ .

$$augmented\_obstacle = A \oplus B = \{x + y | x \in A, y \in B\}$$

It is easy to prove that in the worst case scenario, when the center of the UAV is on the edge of augmented obstacle, the distance from the center of the UAV to the real obstacle (green polygon in Fig 4b) is always greater than the radius of the UAV and the UAV will not collide with the obstacle.

## IV. MAP DIVISION

The purpose of this section is to generate convex polygons which cover all the pest-ridden areas in the agriculture area with non-convex obstacles (Fig. 1).

### A. Classification of infected areas

The question now is that from the data of the coordinates of the pests, we need to group them into smaller non-intersecting areas. This task can be completed using clustering algorithms [12] [14]. Cluster analysis or clustering is a Machine Learning technique whose task is grouping a set of objects in such a

way that objects in the same group (called a cluster) are more similar (in some sense) to each other than to those in other groups (clusters). Each data point of a given set of data points can be classified into a specific cluster. Data points in the same cluster should have analogous features or properties, while data points in different clusters should have highly dissimilar features or properties. This technique is widely applied in exploratory data mining, pattern recognition, image analysis, information retrieval, bioinformatics, data compression, and computer graphics.

In this study, we adopt the K-means algorithm [15], the output of which is illustrated In Fig. 5 where distinct clusters are represented by different colors.

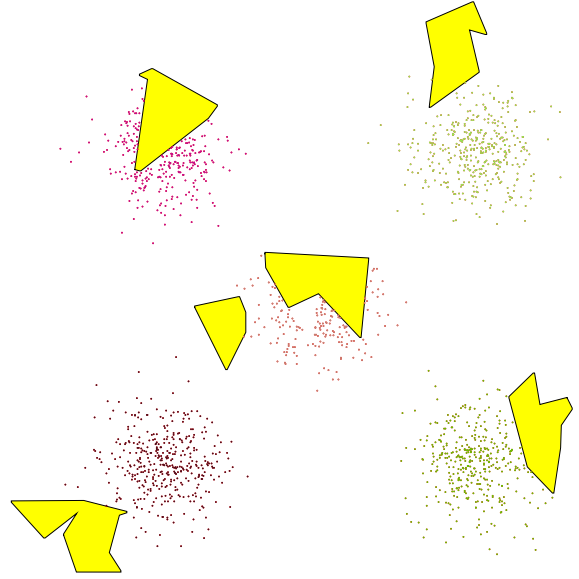


Figure 5: Division infected areas to several smaller regions

### B. Calculation of polygons for covering all the infected areas

The aim of this task is to generate a polygon that contains all the infected points of each generated cluster. We propose two approaches: (i) find the minimal convex polygon which covers all the infected points in each cluster, and (ii) find the boundary polygon which covers all the infected points in each cluster. As mentioned earlier, an algorithm to compare the efficiency of the two methods will be presented in Section VI.

### C. Minimal Convex Partitions

As shown in Fig. 6, the polygon covering all infected points may occupy part of the obstacle. Therefore, a new obstacle-free polygon needs to be generated as shown in Fig. 7.

A convex partition by segments of a polygon is a decomposition into convex polygons obtained by introducing arbitrary segments [16] [17] [18] [19]. By applying Greene's dynamic programming algorithm [16], the non-convex polygon can be divided into a minimum number of convex polygons (Fig. 7).

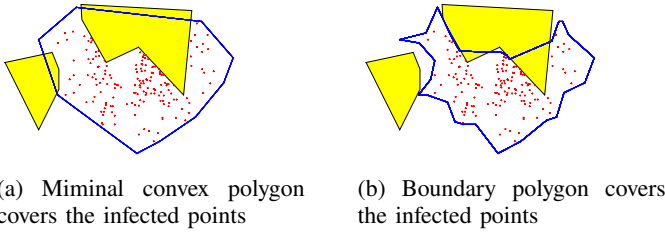


Figure 6: Polygon covers the infected points

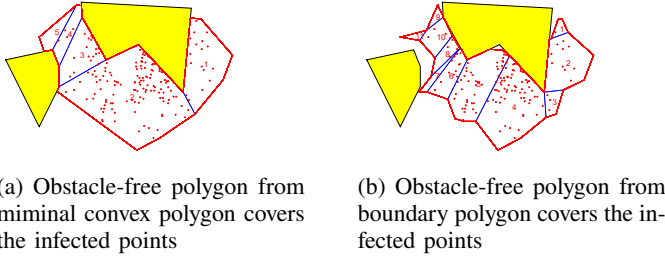


Figure 7: Minimal convex polygon decomposition

The algorithm that describes the process of dividing an infected area into obstacle-free convex polygons is shown in Fig. 8.

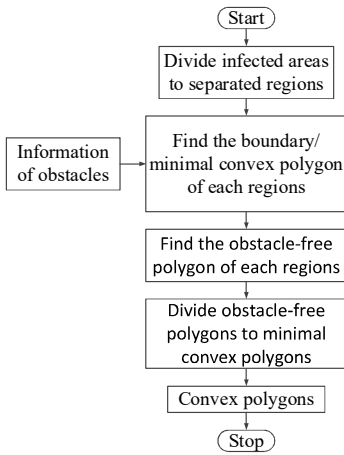


Figure 8: Map division

## V. TRAJECTORY GENERATION

The task of this section is to create a trajectory for the UAV to cover the entire infected area in a convex polygon. There exist two common trajectories for covering a convex polygon, the first is the Boustrophedon trajectory, and the second is spiral. In this paper, we use the boustrophedon.

### A. Waypoints generation for convex polygon

Consider the convex polygon  $ABCDEFG$  with two sides  $AB$  and  $DE$  are parallel to each other and separated by a distance of  $2R$  as shown (in Fig. 9). Draw a parallel line at distance  $R$  to the edge  $AB$ . This line intersects the sides of the convex polygon  $ABCDEFG$  at  $M$  and  $N$ . Form the rectangle

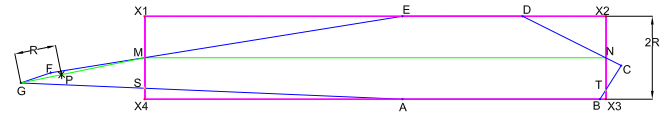


Figure 9: Trapezoid

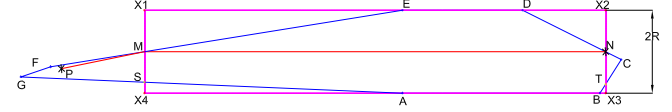
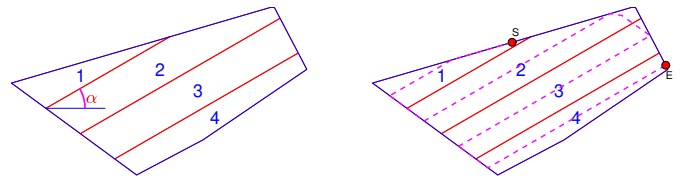


Figure 10: Way-points of trapezoid

$X_1X_2X_3X_4$  with two edges  $X_1X_2$  and  $X_3X_4$  parallel to edge  $AB$  and  $X_1X_2 = X_3X_4 = MN$ . When the UAV moves from  $M$  to  $N$ , overlapping region between the convex polygon  $ABCDEFG$  and the rectangle  $X_1X_2X_3X_4$  will be covered. It remains to find the path for the UAV in the uncovered regions of the  $ABCDEF$  polygon. In Fig. 9, these regions are made up of the triangle  $NTC$  and the quadrangle  $MEGS$ . Consider first the triangle  $NTC$  to the right of  $X_2X_3$ . Observe that the furthest vertex of this triangle to point  $N$  is less than  $R$ . Consequently, when the UAV is at  $N$ , it will cover the area of this triangle. For the quadrangle  $MFGS$  to the left of  $X_1X_4$ , notice that its furthest vertex is  $G$ . Let  $P$  be a point on the segment  $GM$  at distance  $R$  from  $G$ . Since  $ABCDEF$  is a convex polygon, the point  $P$  and the segment  $PM$  belong to the polygon  $A$ . When the UAV moves from  $P$  to  $M$ , the whole area of  $MFGS$  can be covered. Thus,  $PMN$  (the red lines in Fig. 10) is the shortest path the UAV must follow to cover the entire area of polygon  $ABCDEF$ .

As calculated above, a moving trajectory can be calculated for convex polygons with two parallel sides of a distance  $2R$  in between.



(a) Split convex polygon into several trapezoids with  $\alpha$  angle (b) Path way with  $\alpha$  angle

Figure 11: Augmentation of the size of obstacles

The task now is to create the motion trajectory of the UAV over an entire convex polygon so that when the UAV moves in this orbit, its entire area can be covered. Consider a convex polygon that has been subdivided into smaller convex polygons as illustrated in Fig. 11a by parallel lines at distance  $2R$  from one another and at an angle  $\alpha$  to the horizontal axis.

In this fashion, the optimal trajectory for each convex polygon 1, 2, 3, 4 can be created. Combining these trajectories, we obtain the motion trajectory between the points  $S$  and  $E$

(Fig. 11b). When the UAV moves along this trajectory, the entire area of the polygon can be covered.

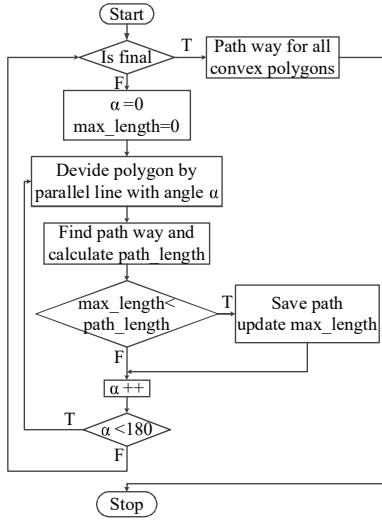
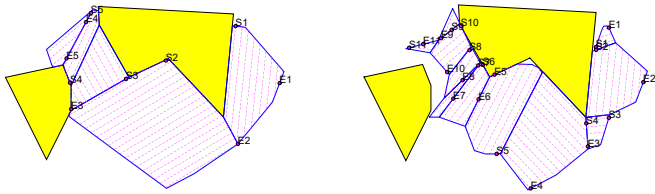


Figure 12: Algorithm for way-point of infected area

### B. Trajectory generation for each convex polygon

As shown in Fig. 11a, the generated trajectory depends on the slope of the parallel lines. Therefore, we vary the angle of these parallel lines to numerically compute the shortest path. This algorithm is shown in Fig. 12. There is an optimal  $\alpha$  for each convex polygon.

By using the algorithm in Fig. 12, we can create the shortest trajectory for each convex polygon region so that when the UAV follows that path, the entire area of the convex polygon will be covered (Fig. 13).



(a) Obstacle-free polygon generated by minimal convex polygon (b) Obstacle-free polygon generated by boundary polygon

Figure 13: Waypoints generation for infected areas

### C. Trajectory generation for entire agriculture area

As a result of the previous section, the Boustrophedon trajectory was generated for each convex polygon. Each orbit of these convex polygons has two points S and E. This means that when UAV comes from point E, it goes to point S when moving on the Boustrophedon trajectory or vice versa from S to E, the entire area of that convex polygon will be covered. However, how the UAV can scan the entire area of a pest, also means that the UAV must move through all the generated Boustrophedon trajectory inside generated convex polygons.

An algorithm that allows finding the shortest path for UAV has been previously developed by the author [20]. By using Genetic Algorithm with some modification on the swap, flip and slide operations and the Traveling Salesman Problem with some additional constraints, the shortest path for changing the cells has been developed. This algorithm guarantees that the solution always exist and the time for calculation is decreasing.

## VI. SIMULATION RESULTS

### A. The covered area and length of Boustrophedon trajectory

In this simulation, we compare the covered area and the trajectory length of generated Boustrophedon trajectories when each obstacle-free cluster is generated by minimal convex polygon (Fig. 13a) and boundary polygon (Fig. 13b).

Table I: Minimal convex polygon vs Boundary polygon

		Length(m)	Area(m <sup>2</sup> )
Experiment 1	Minimal convex polygon	29860	8119
	Boundary polygon	26275	7482
Experiment 2	Minimal convex polygon	26089	7222
	Boundary polygon	23035	6841
Experiment 2	Minimal convex polygon	26146	7301
	Boundary polygon	22397	6664

From the results in Table I, we can see that the method for generating the obstacle-free by using boundary is better than the method for generating the obstacle-free by using minimal convex polygon. The area and the trajectory length for the later method are smaller than the former.

### B. Number of classes vs covered area and length of Boustrophedon trajectory

In this simulation, we are going to compare the trajectory length when the pest-ridden areas of an agricultural area are divided to different number of classes. The result is shown in Fig. 14.

From the results in Table II, we can see that the number of clusters effects to the area of polygon needed to be covered and generated trajectory. When the number of cluster is higher, the area of polygon needed to be covered and generated trajectory are lower. Beside that, the method for generating the obstacle-free by using boundary polygon is better than the method for generating the obstacle-free by using minimal convex polygon. However, the number of divided convex polygons with the boundary polygon is much higher than the number of divided convex polygons with the minimal convex polygon decomposition. This number affects the length of trajectory for changing between divided convex polygons.

## VII. CONCLUSION AND FUTURE WORKS

In this paper, we have proposed a method for generating a trajectory which allows the UAV put the medicine to the entire pest-ridden area of an agricultural area. First, the pest-ridden areas have been divided into several smaller areas (clusters) by using clustering technique. After that, each cluster is divided

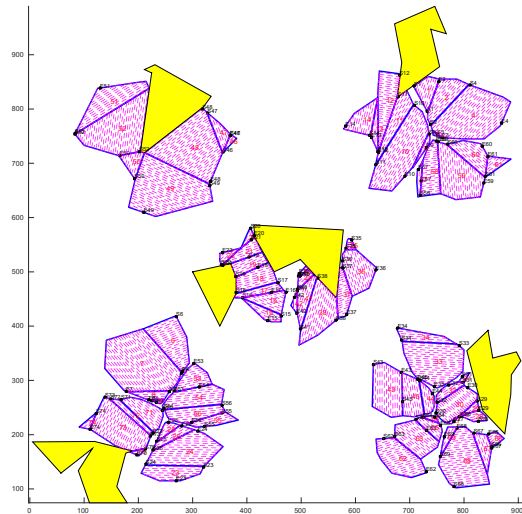


Figure 14: Simulation result

Table II: Classes number vs Trajectory path length

No Cl.	Division method	No con. polygon	Area( $m^2$ )	Trajectory length(m)
5	Min. con. polygon	14	240762	63357
	Boundary polygon	37	204158	59574
10	Min. con. polygon	20	227956	60764
		19	227681	60588
		19	229330	60965
		15	223351	59009
	Boundary polygon	18	229149	60893
		54	204158	56700
		55	206239	57272
		56	203441	56567
15	Min. con. polygon	56	206246	57423
		59	205353	57357
		24	218952	58573
		23	214078	57310
		23	208604	55935
	Boundary polygon	22	215226	57651
		26	221521	59350
		77	191593	54729
		74	191657	55345
		68	181859	50956
	Boundary polygon	79	193424	55266
		76	196747	54123

again to several obstacle-free convex polygons. Then the shortest Boustrophedon trajectory is created in each obstacle-free convex polygon. Finally, the shortest trajectory for changing between obstacle-free convex polygons is generated to form the final trajectory. The results provided in this paper have been verified under Matlab/Simulinks.

Several extensions from this research are possible. One might consider the non-convex shape obstacle, recalculating the trajectory under windy condition of the environment, or trajectory generation for an UAV team. Field tests are also subject of future work.

#### ACKNOWLEDGMENT

We would like to express our greatest gratitude toward the late Professor Yasmina BESTAOUI-SEBBANE, Pro-

fesseur des Universités à l'UFR Sciences et Technologies de l'Université d'Evry-Val-d'Essonne, without whom this project could not have gotten this far.

#### REFERENCES

- [1] R. R. Shamshiri, C. Weltzien, I. A. Hameed, I. J. Yule, T. E. Grift, S. K. Balasundram, L. Pitonakova, D. Ahmad, and G. Chowdhary, "Research and development in agricultural robotics: A perspective of digital farming," *International Journal of Agricultural and Biological Engineering*, pp. 1 – 14.
- [2] R. Vidoni, R. Gallo, G. Ristorto, G. Carabin, F. Mazzetto, L. Scalera, and A. Gasparetto, "Byelab: An agricultural mobile robot prototype for proximal sensing and precision farming," *ASME 2017 International Mechanical Engineering Congress and Exposition*, pp. 1 – 7.
- [3] Z. C and K. J. M, "The application of small unmanned aerial systems for precision agriculture: a review," *Precision agriculture*, Springer, pp. 693 – 712.
- [4] A. S. Natu and K. S. C, "Adoption and utilization of drones for advanced precision farming: A review," *International Journal on Recent and Innovation Trends in Computing and Communication*, pp. 563 – 5652.
- [5] P. Tokekar, J. V. Hook, D. Mulla, and V. Isler, "Sensor planning for a symbiotic uav and ugv system for precision agriculture," *IEEE TRANSACTIONS ON ROBOTICS*, pp. 1 – 14.
- [6] E. R. Hunt, M. Cavigelli, C. S. T. Daughtry, J. E. McMurtrey, and C. L. Walthall, "Evaluation of Digital Photography from Model Aircraft for Remote Sensing of Crop Biomass and Nitrogen Status," *Precision Agriculture*, vol. 6, no. 4, pp. 359–378, Aug. 2005.
- [7] Z. Yuyi, Z. Yu, L. Huanxin, L. Yunjia, and L. Liang, "Control system design for a surface cleaning robot," *International Journal of Advanced Robotic Systems*, vol. 10, no. 5, p. 220, 2013. [Online]. Available: <http://dx.doi.org/10.5772/56200>
- [8] P. N. Atkar, A. Greenfield, D. C. Conner, H. Choset, and A. A. Rizzi, "Uniform coverage of automotive surface patches," *The International Journal of Robotics Research*, vol. 24, no. 11, pp. 883–898, 2005. [Online]. Available: <http://dx.doi.org/10.1177/0278364905059058>
- [9] L. Paull, Z. Saeedi, and H. Li, "Path planning for autonomous underwater vehicles." Springer, New York, NY, 2013, pp. 177–223, DOI: 10.1007/978-1-4614-5659-9\_4.
- [10] E. Acar, H. Choset, Y. Zhang, and M. Schervish, "Path planning for robotic demining: Robust sensor-based coverage of unstructured environments and probabilistic methods," *The International Journal of Robotics Research*, vol. 22, no. 8, pp. 441–466, 2003.
- [11] M. Bosse, N. Nourani-Vatani, and J. Roberts, "Coverage algorithms for an under-actuated car-like vehicle in an uncertain environment," *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pp. 698–703.
- [12] J. A. Hartigan, "Clustering algorithms," *John Wiley & Sons New York, London, Sydney, Toronto*, p. 351.
- [13] H. Barki, F. Denis, and F. Dupont, "Contributing vertices-based minkowski sum computation of convex polyhedra," *Computer-Aided Design*, pp. 525–538.
- [14] F. U.M, G. P. Shapiro, P. Smyth, and R. Uthurusamy, "Advances in knowledge discovery and data mining," *Aaai Press/The Mit Press*, pp. 573 – 592.
- [15] M. Jianliang, S. Haikun, and B. Ling, "The application on intrusion detection based on k-means cluster algorithm," *IEEE International Conference On Information Technology And Applications*, pp. 150 – 152.
- [16] D. H. Greene, "The decomposition of polygons into convex parts," *Advances in Computational Mathematics*, vol. 1, pp. 235–259, 1983.
- [17] J. Fernandez, L. Canovas, and B. Pelegrin, "Algorithms for the decomposition of a polygon into convex polygons," *European Journal of Operational Research*, vol. 121, pp. 330–342, 2000.
- [18] P. K. Agarwal, E. Flato, and D. Halperin, "Polygon decomposition for efficient construction of minkowski sums," *Computational Geometry*, vol. 21, pp. 39–61, 2002.
- [19] D. Adjashvili and D. Peleg, "Equal-area locus-based convex polygon decomposition," *Theoretical Computer Science*, vol. 411, pp. 1648–1667, 2010.
- [20] T. H. Pham, Y. Bestaoui, and S. Mammam, "Aerial robot coverage path planning approach with concave obstacles in precision agriculture," *Proceedings 2107 Workshop on Research, Education and Development of Unmanned Aerial Systems (RED-UAS)*, pp. 43 – 48.