INTERNATIONAL CONFERENCE ON ADVANCES IN TECHNOLOGY AND COMPUTING (ICATC-2021)
Faculty of Computing and Technology (FCT), University of Kelaniya, Sri Lanka
18th December 2021

Index No: ES-14-35

Drone Technology for Rice Agriculture at the Fertilizer Spraying Process

Akarshani Amarasinghe (1st Author)
University of Colombo School of Computing
University of Colombo
Colombo, Sri Lanka
m.c.a.amarasinghe@gmail.com

Lakshman Jayaratne (2nd Author)
University of Colombo School of Computing
University of Colombo
Colombo, Sri Lanka
klj@ucsc.cmb.ac.lk

Viraj B. Wijesuriya (3rd Author)
Department of Computer Science
University of Oxford
Oxford, United Kingdom
viraj.wijesuriya@cs.ox.ac.uk

***Abstract*— This paper presents a drone simulator to spray fertilizer for rice farmland with minimum human intervention. The proposed drone simulator is capable of suggesting the optimal path to spray fertilizer, indicating the current altitude and the current battery level of the drone at flight. The obtained results at the evaluation stage show that the proposed path planning algorithm outputs the optimal path for given farmland with minimum execution time. This solution will cherish the use of drone technology for rice agriculture while supporting the economic growth of a country.**

Keywords - drone simulator, q-learning, path planning algorithm

I. INTRODUCTION

Rice cultivation is one of the main income generating methods in most if the developing countries. Due to increasing demand for rice, farmers need to produce a higher quantity of rice every year. Fertilizing crops is the most difficult part of crop production.

A drone has considerably larger popularity rather than other unmanned equivalents due to its smaller size and flexibility. Also, drone technology has been successfully applied to agriculture through soil and field analysis, planting, crop spraying, crop monitoring, irrigation, and health assessment [1].

Using individual drones for spraying fertilizer is not a novel approach. Operating a drone in a suitable manner can minimize excessive fertilizer usage. Using an autonomous drone to spray fertilizer has several advantages over manual fertilizer spraying process as it allows to cover a larger area within a small time, which ultimately leads to low cost by reducing labor and time.

Through this research, we address the question of “How to develop a drone simulator for spraying fertilizer in a rice farmland?”. This solution proposes a path planning algorithm for an autonomous drone to spray fertilizer appropriately to arable areas with minimum human intervention [11, 12]. The proposed solution has 4 modules: the 2D Grid Maker module, the Path Planning Algorithm module, the Altitude Generation Service module and the Battery Monitoring Service module. Here, when the farmer inputs the relevant details of the farmland that he/she wants to spray fertilizer, the proposed solution outputs the 2D grid of the farmland using the 2D Grid Maker module. Then the Path Planning Algorithm module outputs the optimal path for the generated 2D grid and the Altitude Generation Service module represents the current altitude of the drone when it flies along the proposed optimal path. Afterwards, the Battery Monitoring Service module monitors the current battery level of the drone in flight.

When using the proposed solution, the farmer only needs to identify the safe and unsafe areas of the farmland using its aerial image and input these details to the system. The farmer does not have to worry about the spraying process. He/she can monitor the overall spraying process through a mobile application while the drone is executing its task. Moreover, this solution can spray fertilizer to multiple arable areas with minimum human intervention.

II. OBJECTIVES

Manually spraying fertilizer takes a considerable amount of time and manpower. Hence, using an autonomous drone for the fertilizer spraying process can cover large farmland within a minimum time while reducing human intervention. This approach can diminish the extravagant cost for the labourers while saving time and money of the farmers.

Since the proposed drone simulator is for a fully customized autonomous drone, it sprays only the approved amount of fertilizer. Hence, it helps to reduce fertilizer wastage, health and environmental issues and save money spent on excessive amounts of fertilizer.

Most of the available path planning drone solutions are available only to achieve a single target from a starting location [2]. However, our research solution is capable of visiting and covering multiple arable areas (multiple targets) in a single drone flight. As mentioned in [3], some successful path planning methods are only for unknown environments. However, the proposed solution is for a fully observable environment. A key feature here in is the ability of the drone to avoid unsafe areas which can harm the drone flight [4]. Interestingly, the proposed drone simulator will be a better platform to the fertilizer industry since it can regulate the fertilizer usage, reduce fertilizer over usage, save farmer's time, speed up the fertilizer spraying process and minimize the human intervention at the spraying process.

III. METHODOLOGY

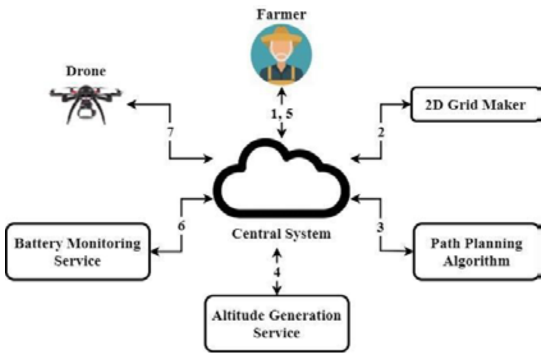

Fig. 1. The overview of the proposed solution

Fig. 1 illustrates the overview of the proposed solution, and the stepwise details are as follows:

A. There is a central system to manage drone operations and a mobile application for the farmer to communicate with the central system. The farmer has an aerial image of the farmland (Fig. 2(a)), and he/she identifies the safe and unsafe areas of the farmland and sends these details to the central system. Here, the safe areas are the areas with crops and the unsafe areas are traps, forbidden areas (dams, ponds, wastelands, streams, and private properties), and dead zones.

B. The central system sends the farmer's input to the 2D Grid Maker module. It generates the 2D grid based on the input as illustrated in Fig. 2(b). Here, the purple cell is the initial distribution (the starting location of the drone flight), the green cells are the safe areas, the red cells are the unsafe areas, and blue cells are the obstacle-free areas. The 2D grid details are passed to the central system.

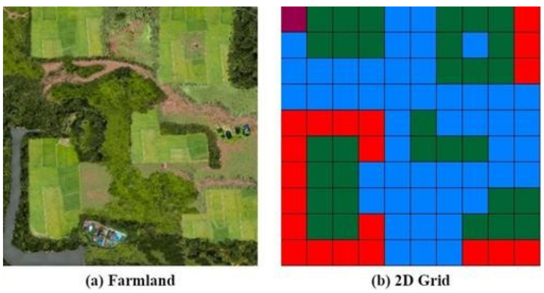

Fig. 2. The aerial image and the 2D grid of the farmland

C. The central system passes the obtained 2D grid details to the Path Planning Algorithm module. It outputs the optimal path as illustrated in Fig. 3. Here, the yellow line shows the proposed path for the drone flight, which starts from the initial distribution and covers all the safe areas in a single visit. A white dot in the middle of each safe area represents the fertilizer spraying locations. The drone needs to hover in-place at the location, which is represented by the white dot, spray the approved amount of fertilizer, and move forward along the proposed path.

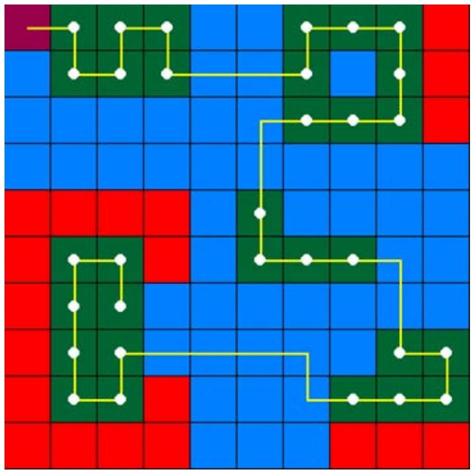

Fig. 3. The 2D grid with the optimal path

The steps of the Path Planning Algorithm module are as follows:

i. Create the transition matrix. Here, it contains the transitions from safe areas to obstacle-free areas and from obstacle-free areas to safe areas. We avoid the transitions to and from unsafe areas to speed up the learning and training process.

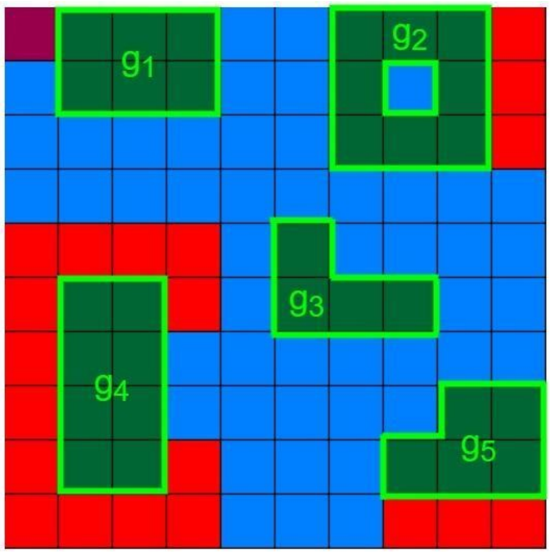

Fig. 4. The goal clusters

ii. Merge successive safe areas into a single goal cluster (Fig. 4). Otherwise, there are a lot of safe areas to cover by a single path from the initial distribution and it will take more time to give the final output.

iii. Acquire the relevant Linear Temporal Logic (LTL) [5] formula to cover all the goal clusters as represented by Equation (1).

$$\blacklozenge g1 \wedge \blacklozenge g2 \wedge \blacklozenge g3 \wedge \blacklozenge g4 \wedge \blacklozenge g5 \qquad (1)$$

iv. Obtain the relevant Deterministic Büchi Automata (DBA) [5] to satisfy the obtained LTL formula.

v. Update the transition matrix according to the newly formed goal clusters.

vi. Apply the Q learning algorithm [6] to the updated transition matrix. The reason behind that is to identify a single optimal path from the initial distribution to multiple goal clusters.

vii. However, the Q learning algorithm can only approach the goal clusters and it cannot visit the safe areas inside a goal cluster. To solve that, we apply the Hamiltonian Cycle [7] for each goal cluster while the Q learning process is ongoing.

viii. Obtain the optimal path for the relevant 2D grid as shown in Fig. 3 and then send the path details to the central system.

D. According to the obtained path, the central system directs the Altitude Generation Service module to create the relevant altitude graph. Using this module, the farmer can monitor the flying altitude of the drone at the spraying process, in real time. This module helps to fly the drone above the ground level without colliding on the ground.

We identify the altitude of the proposed path according to the geo-locations with the help of Google Earth [8]. The farmer can monitor the current altitude of the drone using the altitude curve illustrated in Fig. 5 while the drone is in flight via the mobile application. Here, the x-axis and the y- axis represent geo-location (latitude and longitude) and altitude in meters of the flying path respectively. Also, the green curve shows the true elevation function (the altitude curve of the farmland) and the red curve shows the drone altitude function (the flying altitude of the drone). To obtain the drone altitude function, we apply a Polynomial Regression [9]. It helps the drone to fly a little bit higher than the ground level without colliding on the ground. Moreover, the purple dot in Fig. 5 shows the current altitude value of the drone. The farmer can observe this dot moving when the drone moves along the proposed path. However, the movement of the drone is shown only until the end of the proposed path. If the battery dies before finishing the spraying process, the system shows the movement of the drone only until that time. At the moment, the proposed solution does not have an option to identify the path from the stopping location of the drone to the initial distribution.

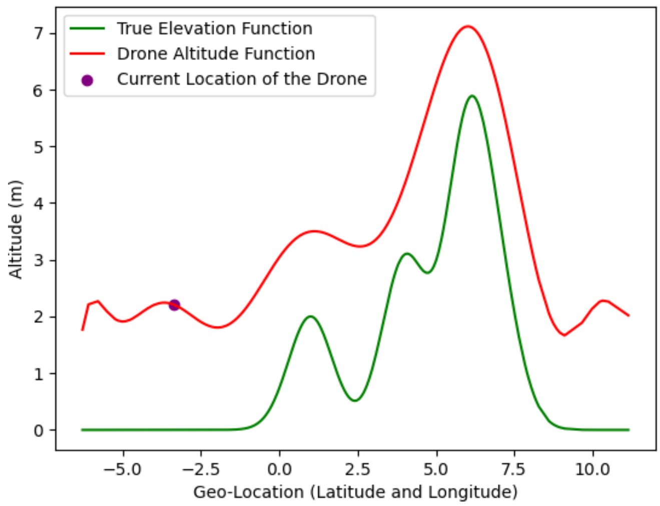

Fig. 5. A sample altitude curve

E. Afterwards, the central system informs the farmer, that the drone is ready to fly. The farmer can command the central system to fly the drone to spray fertilizer.

F. If the farmer wants to monitor the real-time battery level of the drone at its flight, he/she can enable the Battery Monitoring Service module through the mobile application. This module functions in real-time.

Fig. 6 illustrates a battery draining curve that we have used. Here, the x-axis represents the time in minutes and the y-axis represents the battery life (the battery level) as a percentage. Also, the green curve and the red curve show the true battery draining function and the 10% level of the battery respectively. We interpolate [10] the first 10 data points of the battery reading of the drone and extrapolate [10] the future battery readings based on the interpolated values.

Fig. 7 shows the battery reading at 34 minutes and the remaining cells (number of hops) are 9. The number of hops shows the remaining cell count that the drone can fly with the current battery level. The blue dotted line illustrates the current battery value reading and the purple dotted line shows the predicted battery values based on the extrapolation. If the battery level becomes 10%, the drone stops the flight, and the system stops its execution. Currently, we do not have an option to plan the path or follow any safety steps at the battery level at 10%.

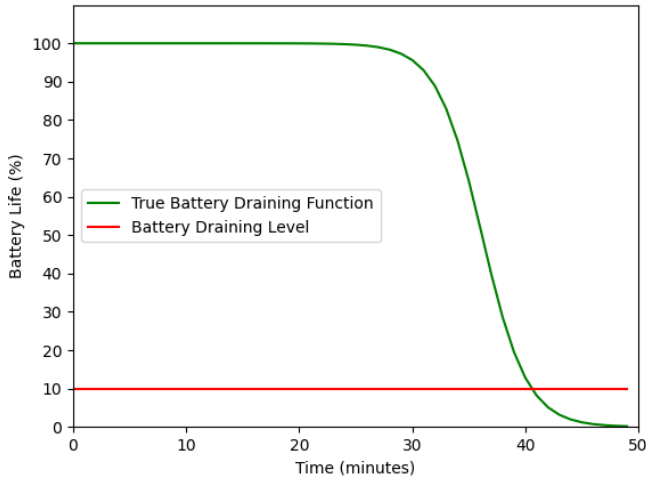

Fig. 6. The battery draining curve

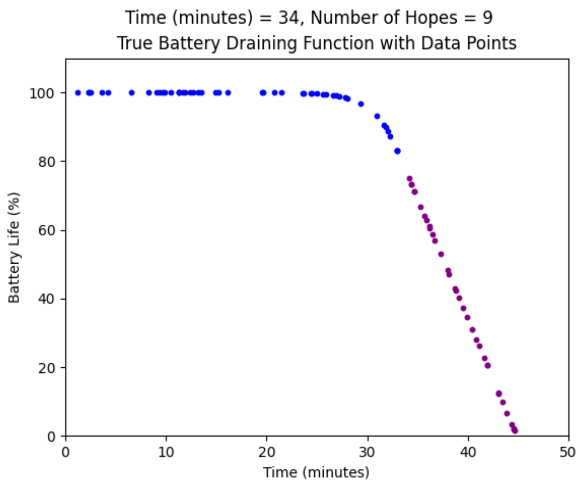

Fig. 7. Real-time battery reading

G. Finally, the central system commands the drone to spray fertilizer according to the proposed path. The farmer can view the spraying process in real-time via the mobile application. Here, the drone has a GPS sensor for geo-location identification, a General Packet Radio Service (GPRS) sensor to communicate via internet with the central system, a spray tank to spray fertilizer and a charging port to recharge its battery.

IV. RESULTS AND DISCUSSION

The overall evaluation was carried out in 2 phases with the purpose of evaluating accuracy and speed. The evaluation phases are described as follows:

A. Suitability of the Proposed Algorithm

The execution speed of our Path Planning Algorithm (PPA) was evaluated with the Without Clustering approach (WC). WC has similar steps as PPA except for steps 3(ii) and 3(vii) in Section III. In WC, successive safe areas were not merged into a single goal cluster, due to each safe area being considered as a separate goal and there was no need of using step 3(ii) in Section III. In PPA, the Hamiltonian Cycle was applied only to visit all the safe areas inside a goal cluster. However, in WC there are no goal clusters and it is useless to use step 3(vii) in Section III for a single goal (a safe area).

As illustrated in Fig. 8, PPA and WC for 4 test cases were evaluated. According to Fig. 8, (1)a and (1)b illustrates the obtained path for test case 1 using PPA and WC respectively. Other figures in Fig. 8 are similar to that. The relevant simulation results for each test case are represented by Table 1. It shows the goal count, the DBA states count, the path length and the execution time (in seconds) for each test case when using PPA and WC.

According to Table 1, we can see the goal count is relatively smaller in PPA. The reason behind this is the proposed algorithm merging the successive safe areas into a single goal cluster at step 3(ii) in Section III. As a result that, the total number of DBA states count is also decreased. Now the address space of the updated transition matrix is also relatively smaller than WC's updated transition matrix.

As all the safe areas has to be visited in a single goal cluster, the Hamiltonian Cycle was used for PPA. However, in WC, there aren't any goal clusters and it is useless to apply the Hamiltonian Cycle while the Q learning process is ongoing. According to that, WC should speed up the training process, since there is no step 3(vii) in Section III with step 3(vi) in Section III. Nevertheless, the execution time of PPA is relatively trivial. The reason behind that is step 3(ii) in Section III of PPA enables the capability of covering the successive safe areas as a single goal cluster and it enables the Hamiltonian Cycle to visit inside it. Unfortunately, WC does not have such a mechanism. Initially, it considers all safe areas as separate goals. As a result that, it has a higher number of DBA states. The increasing DBA states count also increases the address space in the updated transition matrix. So the state space becomes relatively larger and step 3(ii) in Section III has to train and learn based on this larger transition matrix. Therefore, WC has a higher execution time than PPA. Since WC has to visit every goal separately, it cannot traverse to a certain safe area through the successive safe areas. This procedure is the reason to increase the path lengths in WC relatively.

When carefully checking Fig. 8 and Table 1, Figure (3)b and its relevant results were not available. At the testing time, it took more than 5 hours for the execution using WC. Since WC increases the goal count and the DBA states count relatively, the transition matrix becomes complex. Furthermore, comparatively test case 3 has an enclosed environment to achieve safe areas. These aforementioned reasons cause to take more time for training using WC. Therefore, there is no output for Fig. 8 3(b).

According to the obtained simulation results, it can be concluded that PPA outputs a shorter path with a minimum execution time since merging successive safe areas into a single goal cluster and utilizing the Hamiltonian Cycle. The results prove that the proposed path planning algorithm is more suitable for the given context.

Table 1. The simulation results with path lengths and execution times

Test Case Number	Goal Count		DBA States Count		Path Length		Execution Time (s)	
	PPA	WC	PPA	WC	PPA	WC	PPA	WC
(1)	1	3	1	3	4	7	4.35	5.89
(2)	1	4	1	4	6	12	3.84	28.59
(3)	2	5	2	5	8	-	6.72	-
(4)	3	5	3	5	9	12	5.51	57.34

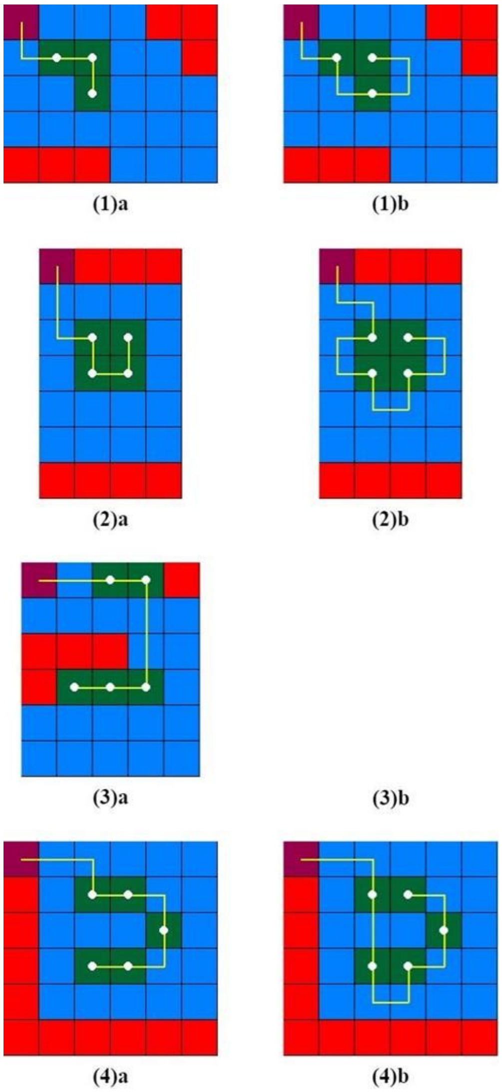

Fig. 8. The outcooms of the two evaluation approaches.

B. Accuracy

In this phase we have checked whether the proposed path planning algorithm can output the optimal path for different 2D grids as illustrated in Fig. 9. We executed the proposed solution 10 times for every 2D grid in Fig. 9 and obtained results as shown in Table 2.

According to Table 2, Grid (a) has the lowest accuracy. The main reason behind that is that the goal count and DBA states count are relatively higher than Grid (b), and there are comparatively a lot of obstacle-free areas between the initial distribution and the goals clusters. When there are a lot of obstacle-free areas between the initial distribution and the goals clusters, it is very hard to find an optimal path. Since Grid (c) has many obstacle-free areas between the initial distribution and the goal clusters, it also has lower accuracy. Grid (b) has the highest accuracy, as there are fewer obstacle-free areas between the initial distribution and the goal clusters and it is easy the find a path to cover all the goal clusters from the initial distribution. Furthermore, the other reason for the highest accuracy of Grid (b) is its goal count and DBA states count is relatively lower than the other 2D grids. However, the proposed algorithm outputs the shortest path for all the grids at least 7 times out of 10 executions. As an average, we got an 86.67% accuracy for all the 2D grids in Fig. 9. According to these results, we can state that the proposed path planning algorithm outputs the optimal path for a given 2D grid.

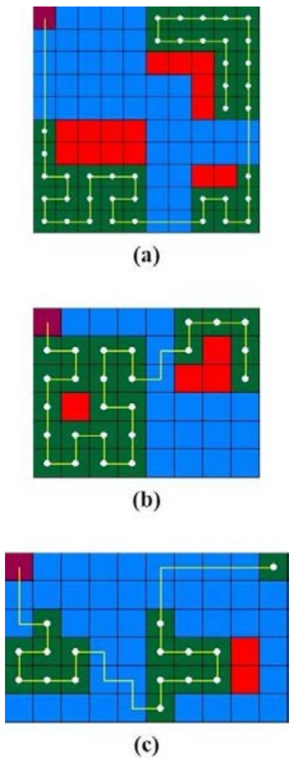

Fig. 9. The test results for different 2D grids

The obtained results from the 2 evaluation phases show the proposed path planning solution is capable to output the optimal path for a given farmland with a minimum execution time. Nevertheless, there can be some accuracy issues of the proposed path, if there are more obstacle-free areas among the initial distribution and the goal clusters. However, the proposed path planning algorithm has a few limitations: the goal cluster must have a Hamiltonian path and its shape must be a square or a rectangle.

Table 2. The path accuracy for different 2D grids

Grid Name	Goal Count	DBA States Count	Accuracy (%)
(a)	3	3	70.00
(b)	2	2	100.00
(c)	3	3	90.00
Overall Accuracy (%)			86.67

V. CONCLUSION

In this paper, we have proposed a drone simulator with an optimal path planning algorithm for an autonomous drone. The main purpose for developing this, is reducing the human intervention at the fertilizer spraying process in rice agriculture. The proposed solution has the 2D Grid Maker module to convert an aerial image of the farmland that we want to spray fertilizer to the relevant 2D grid image, the Path Planning Algorithm module to generate the optimal path to cover all safe areas while avoiding unsafe areas from the initial distribution, the Altitude Generation Service module to identify the current flying altitude of the drone without colliding on the ground and the Battery Monitoring Service module to monitor the real-time battery draining of the drone. Since the Altitude Generation Service module and the Battery Monitoring Service module are at the development stage, we do not have the path planning solution and altitude changes after the drone battery dies.

We have evaluated the proposed path planning algorithm under 2 phases in Section IV: suitability-wise and accuracy- wise. According to the obtained results, it proves the algorithm outputs the optimal path within a minimum execution time while avoiding unsafe areas. The proposed path planning algorithm produces the optimal path from the initial distribution to cover all the safe areas in a single visit. However, there are several limitations in the proposed solution: there could be accuracy issues when there are more obstacle-free areas among the initial distribution and the goal clusters, there must be at least a single Hamiltonian path to visit inside the goal cluster and the goal cluster's shape should be a square or a rectangle.

Through the proposed drone simulator, we can cover large farmland within a minimum time and minimum human intervention. This helps to reduce the fertilizer wastage during the spraying process and speed up the spraying process. Also, through reducing the overuse of fertilizer the farmer can save money without spending extravagantly on fertilizer. Using drone technology for rice agriculture can save the farmer's money from expensive labourers as well.

In the future, we expect to complete the development of the ongoing modules and tackle the problems while the drone battery dies. Furthermore, we hope to test the drone simulator with an autonomous drone. We anticipate improving the proposed drone simulator not only for rice but also for other agricultural crops. Moreover, we look forward to selling or renting out the developed drone solution to the farmers as a package with a concessional price.

This research helps to promote the drone technology in developing countries and regulate the fertilizer usage. Through the regulation of fertilizer, the farmer can increase not only the quantity of the harvest, but also the quality. It will provide a great support to the economical development of the country.

REFERENCES

[1] "Six Ways Drones Are Revolutionizing Agriculture - MIT Technology Review." https://www.technologyreview.com/s/601935/six-ways-drones-are- revolutionizing-agriculture/ (accessed Apr. 01, 2018).

[2] M. Jun and R. D'Andrea, "Path Planning for Unmanned Aerial Vehicles in Uncertain and Adversarial Environments," Springer, Boston, MA, 2003, pp. 95–110.

[3] M. Hasanbeig, A. Abate, and D. Kroening, "Logically-constrained neural fitted Q-iteration," in Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS, 2019, vol. 4, pp. 2012–2014.

[4] L. V. Nguyen, B. Hoxha, T. T. Johnson, and G. Fainekos, "Mission Planning for Multiple Vehicles with Temporal Specifications using UxAS," in IFAC-PapersOnLine, Jan. 2018, vol. 51, no. 16, pp. 67– 72, doi: 10.1016/j.ifacol.2018.08.012.

[5] C. Baier and J. Katoen, Principles of Model Checking (Representation and Mind Series). The MIT Press, 2008.

[6] P. R. Montague, "Reinforcement Learning: An Introduction, by Sutton, R.S. and Barto, A.G.," Trends Cogn. Sci., vol. 3, no. 9, p. 360, 1999, doi: 10.1016/s1364-6613(99)01331-5.

[7] R. J. Gould, "Advances on the Hamiltonian Problem-A Survey," 2002.

[8] "Google Earth." https://earth.google.com/web (accessed Jun. 23, 2021).

[9] Y.-W. Chang, C.-J. Hsieh, K.-W. Chang, M. Ringgaard, and C.-J. Lin, "Training and Testing Low-degree Polynomial Data Mappings via Linear SVM," 2010. Accessed: Jun. 23, 2021. [Online]. Available: http://jmlr.org/papers/v11/chang10a.html.

[10] R. Kress, Numerical Analysis, vol. 181. New York, NY: Springer New York, 1998.

[11] A. Amarasinghe, V. B. Wijesuriya, D. Ganepola, and L. Jayaratne, "A swarm of crop spraying drones solution for optimising safe pesticide usage in arable lands: Poster abstract," in SenSys 2019 - Proceedings of the 17th Conference on Embedded Networked Sensor Systems, Nov. 2019, pp. 410–411, doi: 10.1145/3356250.3361948.

[12] A. Amarasinghe, V. B. Wijesuriya, and L. Jayaratne, "A Path Planning Algorithm for an Autonomous Drone Against the Overuse of Pesticides," in 2021 10th International Conference on Information and Automation for Sustainability (ICIAfS), Aug. 2021, pp. 446–451, doi: 10.1109/ICIAFS52090.2021.9606118.