

Path Planning for Cellular-connected UAV using Heuristic Algorithm and Reinforcement Learning

Junqi Bao *, Yunchu Yang*, Yapeng Wang*, Xu Yang*, Zhenyu Du*

*Faculty of Applied Sciences, Macao Polytechnic University, Macao SAR, China

{p2109422, p1507937, yapengwang, xuyang, p2108837}@mpu.edu.mo

Abstract— With the development of Unmanned Aerial Vehicle (UAV), a novel technology called cellular-connected UAV has been proposed to make UAV complete its mission more efficiently. We consider a scenario where UAV must take off from a random start point, travel over some specific points (e.g. collecting data from sparse sensors in large area) and reach a random end point while keep connected to the Ground Base Station. One of the major challenges is to plan the flying path of UAV while satisfies all constraints. We abstract the path planning problem into Travel Salesman Problem (TSP) and use A* combine with Genetic Algorithm, Simulated Annealing Algorithm and Reinforcement Learning Model to solve TSP to get the best path for cellular-connected UAV. In addition, we did experiments and recorded the results to analyze the advantages and disadvantages of these algorithms.

Keywords— UAV, Cellular-connected UAV, path planning, Heuristic Algorithm, Reinforcement Learning, Travel Salesman Problem

I. INTRODUCTION

Unmanned Aerial Vehicle (UAV) was first proposed in the early 20th century, until now the UAV industry is becoming completer and more advanced, and it is widely used in medical, transportation, rescue and disaster relief fields. The key for UAV to complete these missions is path planning which let UAV find the shortest and most safety path from a start state to a final state. Traditional UAV relies on simple point-to-point (P2P) wireless links to communicate with the ground base station (GBS), which may cause some problems such as low data rate transportation, system vulnerable and so on [1], [2]. With the research on cellular network, 4G and 5G has been proved useful and ultra-reliable in long range communication, especially for UAVs [3], [4]. To enhance the data transportation performance between UAV and GBS, a novel technology called cellular-connected UAV has been developed.

Compared to traditional path planning, cellular-connected UAV has faced some new challenges. First of all, the signal of cellular network is not uniformly distributed due to the influence of the environment (e.g., buildings, altitude and speed of UAV, etc.) on the cellular network. In this situation, shadowing and multi-path fading effects would happen because of the signal blockage and reflection and diffraction[2]. Zhang proposed a new radio map based on channel gain map and SINR map [2], then they solve an approximate Shortest Path Problem (SPP) over the SINR map to plan the path of UAV.

The second challenge is how to find a global optimal path under limited cellular network coverage. Previous research usually transforms the 3-Dimension path planning problem into a SPP in graph theory and use Heuristic Algorithm or some other traditional path planning algorithms to find the best path. Many studies have been applied in path planning. Behavior based methods, such as Artificial potential field (APF) method [5][6], vector field histogram (VFH) method [7], [8] are very useful in some situations. Grid based methods, including Dijkstra Algorithm [9], A Star (A*) algorithm [10] has a good performance in static environment.

Different from traditional path planning, cellular-connected UAV path planning has a more complex and dynamic situation, which may make the operation time longer. Therefore, Chi and et al in 2012 [11], proposed using 3G communication for civil UAV to extend the service range to unlimited by using A Star to plan the path of UAV. To design an algorithm that finds an optimal route between a given starting and a final location within a certain mission time for each UAV to gather data from a number of roadside units, Binol and et al used Genetic Algorithm and Harmony Search respectively to plan the path and compared the performance between these two algorithm [12]. In order to solve real-time motion planning problems in dynamic environments, Cao and et al proposed a cellular neural network called shortest-circuited CNN (SP-CNN) [13].

Our main contributions are summarized as follows.

We consider a scenario where a UAV take off from a start state and fly over several places covered by cellular networks to finish some mission such as data collection or search and rescue. Once the missions are completed, UAV must fly back to where it takes off. As the Figure 1 shows, a cellular-connected UAV will take off and visit Place 1 to Place 5 and back to the start point. In this paper, we use grid map to model the UAV flyable area and A* to calculate the distance between point and point. Then, we use three optimal algorithm, Genetic Algorithm (GA), Simulated Annealing Algorithm (SAA), Pointer Network (Reinforcement Learning) to calculate the best path between the star point and end point and we compared the performance of these four algorithms.

Instead of previous research, we transform the 3-Dimension path planning problem into a Traveling Salesman Problem (TSP). TSP define a question of given a list of cities and the distances between each pair of cities, which is the shortest possible route that a salesman can visit each city and returns to

the origin city. In other words, it is to find the shortest path to traveling each point from a start point to an end point. The way to find the answer of TSP is find all the possible paths first and choose a best path from the possible paths.

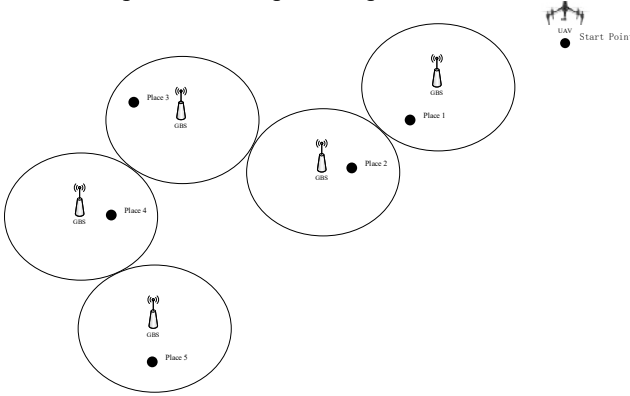


Figure 1. Planned Flight Path for Cellular-connected UAV

In this paper, we combine A* and Heuristic Algorithm to find the best optimal path from all paths to solve TSP. A* is a very popular path finding and graph traversal algorithm because it can search nodes in the map and setting a suitable heuristic function for guidance. By evaluating the generation value of each node, the next best node that needs to be expanded is obtained until the final target point location is reached. Genetic algorithm (GA) is an evolutionary algorithm for solving optimized search algorithms. GA were originally developed with reference to natural evolutionary processes, including genetic mutations, selection, and crossover. Its essence is an efficient, parallel global search method that automatically acquires and accumulates knowledge and finds the optimal solution through iteration. Simulated Annealing Algorithm (SAA) was first proposed by N.Metropolis in 1953 [14], which comes from the metallurgical term annealing. It is a stochastic optimization algorithm based on the Monte-Carlo iterative solution strategy and SAA has been proved that is useful to solve TSP [15] in 1993

Since the TSP is a NP-Hard problem and NP-Hard problem have been proved that can be solved by Reinforcement Learning, some researchers have proposed some novel models to solve TSP. Pointer network is a machine learning model first proposed in 2015 [16] by Google Brain's Vinyals. They designed the Pointer Network with encoder-decoder architecture and RNN and LSTM cells inspired by sequence-to-sequence model. In 2017, Irwan Bello et al. [17] combine reinforcement learning and Pointer Network in order to address some drawbacks of the original model. The result proved that Pointer Network with reinforcement learning can be useful to solve TSP.

This paper is organized as follows. The next section (Section II) introduces the recent of cellular-connected UAV path planning. Section III presents the system model and architecture. In Section IV, we will give some formulations. Section V presents the results of several experiments and comparison between other algorithms. Finally, we conclude and give insight on our future work in section VI.

II. RELATED WORK

Genetic algorithm proposed in 1960s by John Michigan is a global search algorithm which draws on some phenomena in biological evolution. GA first sets up an initial population, evaluates the fitness of the whole population in the process of generation iteration, and then selects the individuals with high fitness for crossover and mutation, and finally to the results. Each individual in population is usually abstracted as a set of coded sequences such as 0 and 1. Pop has successfully solved TSP problem in a grid environment by genetic algorithms in 2009 [18].

In 1983, S. Kirkpatrick et al. have succeeded introduce Simulated Annealing Algorithm to the field of combination optimization. Annealing is the process of heating a material and then cooling it at a specific rate, the atoms in the material originally stay in the position where the internal energy has a local minimum, and the heating makes the energy greater, and the atoms leave their original position and move randomly in other positions. The slower rate of annealing and cooling makes it more likely that the atoms will find positions with lower internal energies than before. N.Metropolis applies the theory of thermodynamics to statistics, imagining each point in the search space as a molecule in air; the energy of the molecule is its own kinetic energy; and each point in the search space, like a molecule in air, carries "energy" to indicate the suitability of the point for the proposition. The algorithm starts with an arbitrary point in the search space: at each step, a "neighbor" is chosen, and then the probability of reaching the "neighbor" from the existing position is calculated., the simulated annealing algorithm starts from a certain high initial temperature, and along with the decreasing temperature parameter, it combines the probabilistic jump property to find the global optimal solution of the objective function in the solution space randomly. In [19], authors using simulated annealing to deal with the TSP in grid environment and compare the result of TSP100 with genetic algorithm, which proved that SAA could solve TSP efficiently.

Challita and et al Proposed a disturbance perception path for cellular connectivity to unmanned machine (UAV) networks [20]. They use Deep Reinforcement Learning to let UAV map every observation of the state of the network to an action. Each UAV uses Echo State Network to learn the best path, transmission power, and small area correlation vectors at different locations on its path. The proposed solution achieves better wireless latency and rate per ground user equipment on each UAV.

Based on weighted sums that consider time costs and expected interruption durations, mapping navigation tasks to the Markov decision-making process, a Deep Reinforcement Learning (DRL) [1] assisted solution is proposed to help drones find the best flight direction within each time slot, thereby designing trajectories towards the destination. To help DRL agents make better trade-offs between sampling priorities and diversity, an innovative quantum-inspired empirical playback (QIER) framework is proposed that correlates the importance of empirical transformation with the qubits associated with it and applies groover-based amplification techniques. The

simulation results verify the effectiveness of the proposed DRL-QiER solution, and the performance comparison with several DRL-assisted baseline and straight line strategies demonstrates the superiority of the DRL-QiER method.

In 2020, Liu studied a mobile edge computing network installed on UAV [21], where the UAV performs computing tasks unloaded from the mobile terminal user (TU), and the movement of each TU follows the Gaussian-Markov random model. To ensure the quality of service (QoS) of each TU, energy-limited drones dynamically plan their trajectories according to the location of the moving TUs. Liu developed a QoS-based action selection strategy in an algorithm based on a dual-depth Q network. Output the Q value of each action and select the action according to the Q value. The simulation results show that the algorithm not only outperforms DQN, DQL, and QL in terms of productivity and total throughput, but also has a QoS guarantee rate of almost 99% per TU.

III. SYSTEM MODEL

We assume that each GBS generates a circular signal coverage area instead of perfect hexagonal cellular area because of the signal GBS generate is not uniform and a UAV can take off from a random start point. In our scenario, we construct a grid map base on the signal coverage area, which is tangent to the area to simulate the region where the UAV can pass or not to train our model as Figure 2 shows. The black region means UAV cannot fly over while the white region means opposite.

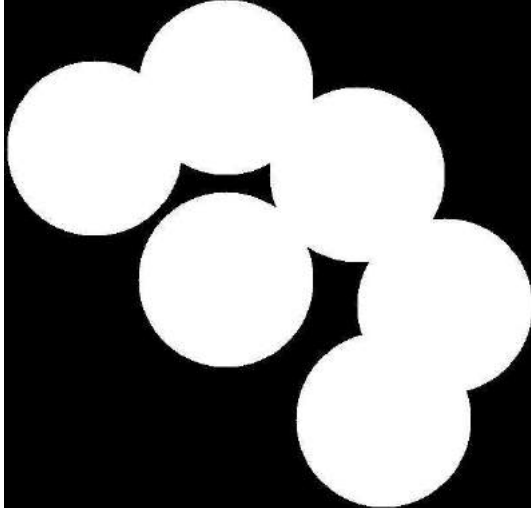


Figure 2. Grid Map shows the cellular signal coverage.

Consider the grid map with a shape of length X m and width Y m contains N cities that the UAV needs to pass through, and M GBS where $N = \{n_1, n_2, \dots, n_N\}$ and $M = \{m_1, m_2, \dots, m_M\}$. Each element $m \in M$ contains several elements $n \in N$. Besides, we assume that UAV flies at constant speed V m/s at a constant altitude H_U m while obstacles have the height H_O . So, we have the distance between the UAV and the m th GBS at any time t can be expressed as follows:

$$dist(t) = \sqrt{(H - H_G)^2 + \|u(t) - g_m\|^2} \quad m \in M, t \in [0, T]$$

where $u(t)$ represents the position of the UAV at a certain moment t , and g_m represents the position of the base station m and T means the total flight time of UAV.

The channel gain between the m th GBS and the UAV can be expressed as follows:

$$|h(t)|^2 = \frac{\beta_0}{(H - H_G)^2 + \|u(t) - g_m\|^2} \quad m \in M, t \in [0, T]$$

Where β_0 is the channel power gain at reference distance $d_0 = 1$ m.

In the total flight time T of UAV, the SNR from UAV to GBS at any time $t \in T$ is expressed as:

$$\rho(t) = \frac{\gamma_0}{(H - H_G)^2 + \|u(t) - g_m\|^2} \quad m \in M, t \in [0, T]$$

$$\gamma_0 = \frac{P\beta_0}{\sigma^2}$$

where γ_0 is the reference signal-to-noise ratio, P , σ^2 is the transmit power of each GBS and the noise power at the UAV receiver.

In order to ensure that the UAV is keep connected with the GBS, UAV must fly within the signal radiation range R of GBS. The signal radiation radius R of GBS can be expressed as:

$$R = \sqrt{\frac{\gamma_0}{T_{SNR}} - (H - H_G)^2}$$

where T_{SNR} means the minimum SNR received by the UAV and H_G is the height of GBS.

IV. METHOD

In this section, we introduce how we work out TSP in detail. First, we use A^* to calculate the distance between each point, then a Heuristic Algorithm will be applied to extract the best path.

A. Search for Point-to-Point Distance(A^*)

A^* search is an optimal path search algorithm using Grid Map. According to the form $f(n) = g(n) + h(n)$, A^* search will operate on 8 neighbors of the center at each step where $f(n)$ is the evaluation function, $g(n)$ is the cost of moving from the starting point to the specified point. $H(n)$ is the estimated distance from the current node to the end point, which is an estimate of the remaining distance, not the actual value. It is an ideal value that ignores the effect of obstacles. In order to estimate $h(n)$, there are three kinds of distance metrics can be used in path planning which are Euclidean Distance, Manhattan Distance, Chebyshev Distance.

If a point can be reached by four directions to another point, then the Manhattan Distance can be used to measure it while Chebyshev Distance means eight directions and Euclidean Distance means any directions. In our scenario, we choose Manhattan Distance to metric the distance between two points, which can be formulated by:

$$D_{12} = |x_1 - x_2| + |y_1 - y_2|$$

As described in [22] the steps of A^* search are as follows:

1. First, we define OPEN list which contains unexplored blocks and CLOSE list which maintain explored blocks.

2. Iterate through the OPEN list, find the node with the smallest F value, and treat it as the node to be processed.
3. Move the node into CLOSE list.
4. For each of the 8 adjacent squares of the current square:
 - a) If it is unreachable or is in the CLOSE list, ignore it.
 - b) If it is not in the OPEN list, add it to the open list and set the current square as its parent node, and calculate the F, G and H values of that square.
 - c) If it is already in the OPEN list, check if it costs less to reach that square through the current square, i.e., the G value is smaller. If so, set its father to the current square and recalculate its G and F values.
5. Repeat Step 2 until the endpoint has been added into the OPEN list.

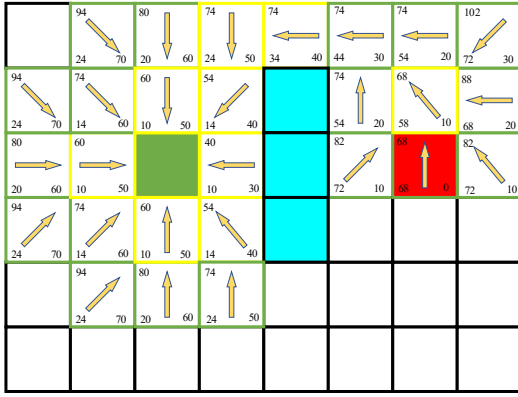


Figure 3. Result of A Star Algorithm

The Figure 3 shows the results of path planning using A* while green square is the start point, red square is the end point and blue point is the obstacles. Also, the green border squares are points in the open list, yellow border squares are points in the close list, the yellow arrow in the square points to the father node, and the number in the upper left corner of the square represents the F value, the G value in the lower left corner, and the H value in the lower right corner.

B. Find the best optimal path (Heuristic Algorithm and RL)

1) Genetic Algorithm: Genetic Algorithm (GA) is an intelligent optimization algorithm, mainly used to solve optimization problems. The steps of GA can be concluded as follows:

Genetic Algorithm (GA) is an intelligent optimization algorithm, mainly used to solve optimization problems. The steps of GA can be concluded as follows:

The process begins with a set of the individual called population. An individual consists of a set of character which is gene. In TSP, an individual represented by a sequence of number means the order that UAV visits each city, a gene means a city number. A fitness function determines how fit an individual is, it will give a fitness score for every individual, a

sequence with a shorter tour length means higher fitness score. And then the selection phase will select several individual for the reproduction, individuals which have higher fitness score will have a higher probability to be selected. After selection, and crossover phase with process crossover for each pair of parents, which means exchange genes based on crossover points between two sequences be selected in the previous step. Those parents and offspring formed a new generation. In certain offspring, a mutation occurs, some positions of gene changes randomly, in TSP mean gene swapping at two positions. This process will repeat until the set of iteration is reached.

2) Simulated Annealing Algorithm: The Simulated Annealing Algorithm is a general probabilistic algorithm for finding the optimal solution in a large search space [19]. It is a stochastic optimality-seeking algorithm based on the Monte-Carlo iterative solution strategy. Simulated annealing algorithm is one of the effective methods to solve TSP problems. The steps of SA are:

1. Start with an initial solution s_0 which is calculated by A* and an initial temperature t_0 .
2. Choose a temperature reduction function:
 - Linear Reductino Rule: $t = t - a$
 - Geometric Reductino Rule: $t = t * a$
 - Slow - Decrease Rule: $t = \frac{t}{1 + \beta t}$
3. Given a new solution s_1 , which is a neighbor of s_0 , calculate the difference in cost between s_0 and s_1 .
4. If s_1 is better than s_0 , then accept s_1 . If s_0 is better, than generate a random number r from 0 and 1. Then we calculate the probability of accepting s_1 :

$$p = e^{-|s_0 - s_1|/t_0}$$

If $r < p$, then accept s_1 . If not, go back to Step 3.

5. Output the best optimal solution.

3) Reinforcement Learning: The original pointer network is using supervised loss function comprising conditional log-likelihood to train the network, but it is undesirable in NP-hard problem [18], Irwan Bello at el. proposed a reinforcement learning to optimize the parameters of a pointer network. The train objectives are to expected tour length, given an input graph Q, we have:

$$J(\theta|Q) = E_{C \sim p_{\theta}(\cdot|Q)} \text{reward}(C^Q|Q)$$

The gradient of above formular using the well-known reinforce algorithm.

$$\nabla_{\theta} J(\theta|Q) = E_{C \sim p_{\theta}(\cdot|Q)} [\text{reward}(C|Q) - b(Q)] \nabla_{\theta} \log_{p_{\theta}}(C|Q)$$

Where $b(Q)$ is the baseline function, it reduces the overall gradient variance by estimating value of reward, while the reward means the tour length in this model. For improves the learning, an actor-critic model is used here to expected $b(Q)$. Actor-critic: The actor-critic first proposed in 2016. Briefly, critic network can predict the value of reward from actor network with known input Q. In this model, pointer network is the actor, we add a critic network into the final model which is

also a RNN consisted by LSTM cells. The critic network takes the mean square error of the predicted reward and the actual reward value as the optimization objective and uses stochastic gradient descent for training, also critic network training itself by using:

$$l(\theta_v) = \frac{1}{B} \sum_{i=1}^B \|b_{\theta_v}(Q_i) - \text{reward}(C_i^Q|Q_i)\|_2^2$$

V. EXPERIMENT RESULT

In this section, we test the result of the fusion of A* with GA SAA and RL. We first generate a TSP20 dataset containing 12,800 location coordinates. We use A* to calculate the distance between each of the 20 points. The result of the A* should be like:

$$\text{dist} = \begin{pmatrix} d_0^0 & d_0^1 & \dots & d_0^{18} & d_0^{19} \\ d_1^0 & d_1^1 & \dots & d_1^{18} & d_1^{19} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ d_{18}^0 & d_{18}^1 & \dots & d_{18}^{18} & d_{18}^{19} \\ d_{19}^0 & d_{19}^1 & \dots & d_{19}^{18} & d_{19}^{19} \end{pmatrix}$$

According to the distance A* calculated, a Heuristic Algorithm will be applied to calculate the best path. For validation data, we generate 1000 data of cities for TSP5, TSP10, TSP20 and TSP50 respectively and we calculate the theoretically optimal path by enumeration. Each validation data includes coordinate points, matrix of rout length, optimal distance, optimal travel permutation.

TABLE 1. MEAN TIME CONSUMPTION

	TSP5	TSP10	TSP20	TSP50
RL	0.05	0.092	0.176	0.572
GA	0.126	1.226	1.831	8.555
SA	0.207	0.414	1.136	5.492

Table 1 shows the mean time consumption of each algorithm in different condition. We calculated the mean time taken by these three algorithms to find the optimal path on 1000 validation sets. Reinforcement Learning-based algorithm clearly take the least amount of time.

Table 2 shows the result of the best path find by these three algorithms. We define y_i as the distance of the best path calculated by algorithm and x_i represents for the ground truth of the best path. Then we calculate MAE, RMSE, MAPE to evaluate the performance of these three algorithms in different scenarios.

MAE (Mean Absolute Error) is a measure of errors between paired observations expressing the same phenomenon. MAE represent the actual situation of the error of calculated distance.

RMSE (Root Mean Square Error) is the square of the difference between the true value and the predicted value, and then the average of the sum, RMSE reflect the sample standard deviation of the difference between the calculated optimal path and the true optimal path.

MAPE (Mean Absolute Percentage Error) represent the precision of the path planning algorithm.

$$MAE = \frac{\sum_{i=1}^n |y_i - x_i|}{n}, \quad n = 1000$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (y_i - x_i)^2}{n}}, \quad n = 1000$$

$$MAPE = \frac{\sum_{i=1}^n \left| \frac{y_i - x_i}{x_i} \right|}{n} \times 100\%, \quad n = 1000$$

It can be seen from Table 2 that SA performance best in all scenarios while Reinforcement Learning-based algorithm performance worst. Combine Table 1 and Table 2, SA spends the most time to achieve the best results while Reinforcement Learning-based algorithm cost the least amount of time but perform worst, especially in TSP50. Considering we are using a dataset of TSP20 to train the reinforcement learning model, we think if we can train the model on a dataset of TSP50, it may perform better. If the accuracy requirement is relatively high, then undoubtedly SA is the most suitable, but if in a real-time scenario, Reinforcement Learning-based algorithm although sacrifices some accuracy but can also calculate a relatively good path.

Since we train these three models on the dataset of TSP20, SA and GA get the lowest RMSE, MAE and MAPE which equals to 0 on TSP5, while the RL get 66.57, 27.83, 2%. With the problem get more complexity, on TSP10, SA keeps the 0% error rate, while the results of GA are not always optimal. On TSP20, RL and GA almost performance at the same level, but the RL costs the most less time while GA cost nearly 2 second to find one best path. At the same time, SA keeps its best performance of MAE equals to 0.06%.

On TSP50, SA still performance best but cost too much time. Consider the SA is trained on TSP20, we think SA still has room for improvement. GA performance worst in TSP50, the MAPE of GA up to 98% which has an 86% increase in MAPE compared to TSP20. Despite RL's previous poor performance, on TSP50, RL's MAPE was only 48%. Compared to TSP20, the MAPE increased by only 32, while GA increased by 50% in the same situation. However, RL only costs 0.572 second in TSP50, while GA costs 8.555, SA costs 5.492. We consider if we use a dataset of TSP50, we can get a better performance RL.

TABLE 2. RL VS. GA VS. SA ON TSP5, TSP10, TSP20 AND TSP50

		TSP5	TSP10	TSP20	TSP50
RL	RMSE	66.57	234.48	419.70	1722.32
	MAE	27.83	189.82	384.5	1557.76
	MAPE	2%	10%	16%	48%
GA	RMSE	0	15.96	326.37	3233.91
	MAE	0	4.04	301.71	3169.57
	MAPE	0%	0.2%	12%	98%
SA	RMSE	0	0	12.26	248.24
	MAE	0	0	1.34	197.82
	MAPE	0%	0%	0.06%	3%

VI. CONCLUSIONS

In conclusion, this paper introduces Heuristic Algorithm and Reinforcement Learning-based algorithm to work out the TSP under the scenario of Cellular-connected UAV. We have compared and discussed the performance of these three algorithms under different TSP scenario and we concluded that

SA is suitable for tasks that require high accuracy and Reinforcement Learning-based algorithm can be useful in real-time tasks. Although RL cannot get a high accuracy, we think there still are some room for it to develop. In the future, we will focus on the RL-based method to handle TSP.

ACKNOWLEDGEMENT

This work is funded by Macao Polytechnic University under grant number RP/ESCA-08/2021.

REFERENCES

- [1] Y. Li, A. Hamid Aghvami, and D. Dong, "Path Planning for Cellular-Connected UAV: A DRL Solution with Quantum-Inspired Experience Replay," *IEEE Trans. Wirel. Commun.*, pp. 1–1, 2022, doi: 10.1109/TWC.2022.3162749.
- [2] S. Zhang and R. Zhang, "Radio Map-Based 3D Path Planning for Cellular-Connected UAV," *IEEE Trans. Wirel. Commun.*, vol. 20, no. 3, pp. 1975–1989, Mar. 2021, doi: 10.1109/TWC.2020.3037916.
- [3] X. Lin *et al.*, "The Sky Is Not the Limit: LTE for Unmanned Aerial Vehicles," *IEEE Commun. Mag.*, vol. 56, no. 4, pp. 204–210, Apr. 2018, doi: 10.1109/MCOM.2018.1700643.
- [4] B. Van Der Bergh, A. Chiumento, and S. Pollin, "LTE in the sky: trading off propagation benefits with interference costs for aerial nodes," *IEEE Commun. Mag.*, vol. 54, no. 5, pp. 44–50, 2016, doi: 10.1109/MCOM.2016.7470934.
- [5] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *1985 IEEE International Conference on Robotics and Automation Proceedings*, Mar. 1985, vol. 2, pp. 500–505. doi: 10.1109/ROBOT.1985.1087247.
- [6] Y. Zhang, "Improved Artificial Potential Field Method for Mobile Robots Path Planning in a Corridor Environment," in *2022 IEEE International Conference on Mechatronics and Automation (ICMA)*, 2022, pp. 185–190. doi: 10.1109/ICMA54519.2022.9856300.
- [7] C.-F. Wu, Y.-L. Wang, L. Ma, and A. Rakić, "VFH+ Based Local Path Planning for Unmanned Surface Vehicles," in *2021 IEEE International Conference on Recent Advances in Systems Science and Engineering (RASSE)*, 2021, pp. 1–6. doi: 10.1109/RASSE53195.2021.9686856.
- [8] H. Zhang *et al.*, "Path planning for fixed-wing UAVs based on expert knowledge and improved VFH in cluttered environments," in *2022 IEEE 17th International Conference on Control & Automation (ICCA)*, Jun. 2022, pp. 255–260. doi: 10.1109/ICCA54724.2022.9831848.
- [9] S. A. Fadzli, S. I. Abdulkadir, M. Makhtar, and A. A. Jamal, "Robotic Indoor Path Planning Using Dijkstra's Algorithm with Multi-Layer Dictionaries," in *2015 2nd International Conference on Information Science and Security (ICISS)*, 2015, pp. 1–4. doi: 10.1109/ICISSEC.2015.7371031.
- [10] C. W. Warren, "Fast path planning using modified A* method," in *[1993] Proceedings IEEE International Conference on Robotics and Automation*, 1993, pp. 662–667 vol.2. doi: 10.1109/ROBOT.1993.291883.
- [11] T.-Y. Chi, Y. Ming, Tseng, S.-Y. Kuo, and C.-C. Liao, "Civil UAV Path Planning Algorithm for Considering Connection with Cellular Data Network," in *2012 IEEE 12th International Conference on Computer and Information Technology*, 2012, pp. 327–331. doi: 10.1109/CIT.2012.83.
- [12] H. Binol, E. Bulut, K. Akkaya, and I. Guvenc, "Time Optimal Multi-UAV Path Planning for Gathering its Data from Roadside Units," in *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, 2018, pp. 1–5. doi: 10.1109/VTCFall.2018.8690730.
- [13] Y. Cao *et al.*, "Design of path planning based Cellular Neural Network," in *2010 8th World Congress on Intelligent Control and Automation*, Jul. 2010, pp. 6539–6544. doi: 10.1109/WCICA.2010.5554403.
- [14] M. Steinbrunn, G. Moerkotte, and A. Kemper, "Heuristic and randomized optimization for the join ordering problem," *VLDB J.*, vol. 6, no. 3, pp. 191–208, 1997.
- [15] F.-T. Lin, C.-Y. Kao, and C.-C. Hsu, "Applying the genetic approach to simulated annealing in solving some NP-hard problems," *IEEE Trans. Syst. Man Cybern.*, vol. 23, no. 6, pp. 1752–1767, Nov. 1993, doi: 10.1109/21.257766.
- [16] O. Vinyals, M. Fortunato, and N. Jaitly, "Pointer networks," *Adv. Neural Inf. Process. Syst.*, vol. 28, 2015.

- [17] I. Bello, H. Pham, Q. V. Le, M. Norouzi, and S. Bengio, "Neural combinatorial optimization with reinforcement learning," *ArXiv Prepr. ArXiv161109940*, 2016.
- [18] F. Pop, C. Dobre, and V. Cristea, "Genetic algorithm for DAG scheduling in Grid environments," in *2009 IEEE 5th International Conference on Intelligent Computer Communication and Processing*, Aug. 2009, pp. 299–305. doi: 10.1109/ICCP.2009.5284747.
- [19] I. Iryanto and P. H. Gunawan, "Numerical Approach of Symmetric Traveling Salesman Problem Using Simulated Annealing," *J. RESTI Rekayasa Sist. Dan Teknol. Inf.*, vol. 5, no. 6, pp. 1090–1098, 2021.
- [20] U. Challita, W. Saad, and C. Bettstetter, "Deep Reinforcement Learning for Interference-Aware Path Planning of Cellular-Connected UAVs," in *2018 IEEE International Conference on Communications (ICC)*, Kansas City, MO, May 2018, pp. 1–7. doi: 10.1109/ICC.2018.8422706.
- [21] Q. Liu, L. Shi, L. Sun, J. Li, M. Ding, and F. S. Shu, "Path Planning for UAV-Mounted Mobile Edge Computing With Deep Reinforcement Learning," *IEEE Trans. Veh. Technol.*, vol. 69, no. 5, pp. 5723–5728, May 2020, doi: 10.1109/TVT.2020.2982508.
- [22] J. Pearl, *Heuristics: intelligent search strategies for computer problem solving*. USA: Addison-Wesley Longman Publishing Co., Inc., 1984.



Junqi Bao received the B.Sc. in Nanjing University of Science and Technology ZhiJin College in 2021. He is currently pursuing the M.Sc. degree in computer applied technology with the Faculty of Applied Sciences, Macao Polytechnic University. His research interests include image matching, computer vision, multiple view geometry, robotics, SLAM.



Yunchu Yang received the B.Sc. in Macao Polytechnic University in 2019. He is currently pursuing the M.Sc. degree in computer applied technology with the Faculty of Applied Sciences, Macao Polytechnic University. His research interest include path planning for UAVs, speech recognition, natural language processing, matching learning, text translation.



Xu Yang received the B.Sc. in University of Electronic Science and Technology of China in 1997, M.Sc. and Ph.D. degrees in Queen Mary University of London, UK in 2003 and 2009. She joined the Faculty of Applied Sciences, Macao Polytechnic University in 2013 as an lecturer. Her current research interests include wireless communications, medical image analyses, machine learning.



Yapeng Wang received the B.Sc. in North China Electric Power University, China in 1998, M.Sc. and Ph.D. degrees in Queen Mary University of London, UK in 2002 and 2007. He joined the Faculty of Applied Sciences, Macao Polytechnic University in 2021 as an associate professor. His current research interests include wireless communications, automatic speech recognition, machine learning.



Zhenyu Du received the B.Sc. in Zhengzhou University in 2020. He is currently pursuing the M.Sc. degree in Faculty of Applied Sciences, Macao Polytechnic University. His research areas include wireless communications and reinforcement learning for radio resource allocation.