

# Simulation and Optimization of Multi-UAV Route Planning Based on Hybrid Particle Swarm Optimization



Xuezhao Peng, Feng Guan, Zhongpu Wang, and Shushida Gao

**Abstract** The multiple traveling salesman problem (mTSP) is an obviously NP-Hard problems in the optimization of complex planning. By adding much more conditions and meanings to the objective function of the mTSP, the vehicle routing problem (VRP) and trajectory optimization problem of the drones can be further evolved. This paper focuses on the multi-UAV route planning problem which is one of the typical mTSP. But traditional algorithms include genetic algorithm and particle swarm optimization are premature in varying degrees and easy to fall into local optimization. Therefore, this paper adds the crossover operator and mutate operator in the genetic algorithm to the particle swarm optimization, and proposes a hybrid particle swarm optimization to solve the multi-UAV route planning problem. The algorithm can make the particles cross the optimal value, and the particles can mutate themselves to strengthen the interchange between the particles. So it can enrich the diversity of the population. Meanwhile, it can improve the global optimality of the algorithm. Finally, it proves that the hybrid particle swarm optimization has better performance when compared with the simulation results of genetic algorithm.

**Keywords** Multi-UAV route planning problem · mTSP · Genetic algorithm · Hybrid particle swarm optimization

## 1 Introduction

The multi-unmanned aerial vehicle (multi-UAV) route planning problem is basically consistent with the description of the mTSP [1, 2]. Its application scenario can be described as: There are  $n$  locations and  $m$  drones.  $m$  drones start from the same location and return to the same departure location. In the process, each location is required to be visited and can only be visited one time. Each drone visits at least one location. Most importantly, it should ensure that the total distance of the entire route selection plan is the shortest.

---

X. Peng · F. Guan (✉) · Z. Wang · S. Gao  
School of Transportation Engineering, Shenyang Jianzhu University, Shenyang, China  
e-mail: [benguan@126.com](mailto:benguan@126.com)

© The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd. 2021  
Y. Li et al. (eds.), *Advances in Simulation and Process Modelling*,  
Advances in Intelligent Systems and Computing 1305,  
[https://doi.org/10.1007/978-981-33-4575-1\\_22](https://doi.org/10.1007/978-981-33-4575-1_22)

223

In recent years, a variety of heuristic algorithms and evolutionary algorithms proposed by scholars can effectively solve mTSP. Kencana [3] uses ant colony optimization to solve complex mTSP problems. Jiang et al. [4] put forward an ant colony partheno genetic algorithm (AC-PGA), which includes partheno genetic algorithm and ant colony algorithm, to solve the large-scale mTSP. Lu and Yue [5] gave an effective way for solving mTSP. They use a task-oriented form to assign ants to the population to optimize the ant colony algorithm. Zhou et al. [6] proposed two methods to solve mTSP. One is the PGA using the roulette model and elite strategy. The other is called IPGA, which combines selection and mutation operators to improve population diversity. Dhein et al. [7] gave a label to describe the difference in dispersion between different paths, and found a genetic algorithm that can realize local search as a new method to settle the matter. The simulation results obtained in the article are very satisfactory.

This article proposes a hybrid particle swarm optimization; it can solve the problems that may occur when the particle swarm optimization is applied to multi-UAV route planning. These problems include the shortcomings of particles easily falling into local optima and slow convergence speed. Finally, the result will be compared with the result of genetic algorithm.

## 2 Model of Multi-UAV Route Planning Problem

The development of drone technology has brought great convenience to the logistics of remote mountainous areas where vehicles cannot reach. The drone flight does not need to consider the factors of the road network. They can directly use the coordinates to calculate the distance between two points precisely [8, 9]. As a classic mTSP, the multi-unmanned aerial vehicle (multi-UAV) route planning problem is to find the shortest path traversing all locations. Its mathematical model can be described as [10–12].

Location set  $A = (1, 2, \dots, n)$  contains  $n$  locations, and location 1 is the start and end of all drones. Drone set  $B = (1, 2, \dots, m)$  contains  $m$  drones.  $S_{i,k}$  represents the collection of locations visited by the  $k$ th drone.

Define variables:

$$x_{ijk} = \begin{cases} 1, & \text{The } k\text{th drone goes from location } i \text{ to } j \\ 0, & \text{Or} \end{cases} \quad (1)$$

$$y_{ki} = \begin{cases} 1, & \text{The } k\text{th drone visits location } i \\ 0, & \text{Or} \end{cases} \quad (2)$$

Objective function:

$$L = \min \left( \sum_{k=1}^m L_k \right) \quad (3)$$

$$L_k = \sum_{i=1}^n \sum_{j=1}^n d_{ij} \cdot x_{ijk}, \quad k = 1, 2, \dots, m \quad (4)$$

Subject to:

$$\sum_{k=1}^m y_{ki} = \begin{cases} m, & i = 1 \\ 1, & i = 2, \dots, n \end{cases} \quad (5)$$

$$\sum_{i=1}^n x(i, i+1, k) = y(k, i), \quad k = 1, \dots, m \quad (6)$$

$$\sum_{i=1}^n x(i, i+1, k) = y(k, i+1), \quad k = 1, \dots, m \quad (7)$$

$$S_{i,k} = \{x(i, i+1, k) = 1\} = \{s_{1k}, s_{2k}, \dots, s_{qk}\} \quad (8)$$

$$\sum_{k=1}^m \sum_{i=1}^q s_{ik} = n \quad (9)$$

$$a \leq q \leq b \quad (10)$$

where  $d_{ij}$  is the length of the drone from one point  $i$  to another point  $j$ . Equation (3) indicates the shortest flight distance of all drones; Eq. (4) represents the flight distance of a drone; Eq. (5) represents that all drones start at location 1, and each location has one and only one drone to visit; Eq. (6) indicates that the secondary destination city of any route has only one starting city connected to it; Eq. (7) indicates that the starting point city of any path has and only one secondary destination city is connected to it; Eqs. (6) and (7) work together to avoid sub-loops; Eq. (8) represents  $q$  locations visited by the  $k$ th drone; Eq. (9) represents the total number of locations visited by all drones; Based on Eqs. (8) and (9), Eq. (10) indicates that each drone

visits at least a location and at most b locations. According to the number of locations in this paper, Eq. (10) is determined as  $8 \leq q \leq 14$ .

### 3 Hybrid Particle Swarm Algorithm for Multi-UAV Route Planning

Genetic algorithm has strong robustness and global search capabilities. It is widely used to solve mTSP and VRP. The particle swarm optimization completes the optimization process of the entire algorithm by continuously seeking individual extreme values and group extreme values. Although the algorithm principle is relatively simple and it can complete convergence in a short time, with the accumulation of the number of particle swarm iterations, the population gradually converges and concentrates and the individual particles gradually become more and more similar which may cause the particle swarm to fall into local optimal solutions.

In our article, we propose a hybrid particle swarm algorithm to simulate and solve the multi-UAV route planning problem [13]. It changes the PSO that uses the method of tracking extreme values to give each particle a new position. The crossover and mutation operations are added to the particle swarm optimization, so that particles can cross individual extreme values and group extreme values. At the same time, the particles can mutate themselves. In this way, the shortest route can be calculated. Hybrid particle swarm optimization algorithm uses the crossover operator and mutation operator to strengthen the information exchange between particles. It improves the abundance of particle swarms, optimizes individual and group, and improves the old algorithm that is easier to fall into the dilemma of locally optimal solutions.

#### 3.1 Individual Code

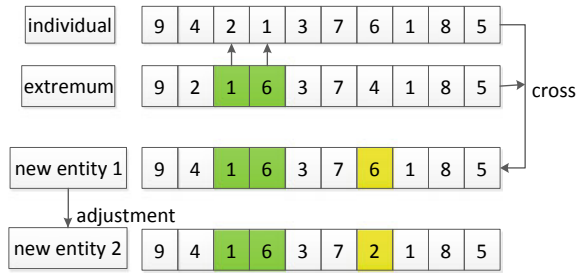
In this algorithm, the individual coding of the particles adopts an integer coding method, and the location is represented by numbers 1, 2, .... Mark all locations and assign a serial number to each location. There are 2 drones that traverse 9 locations, starting and ending at location 1, and the encoding method is shown in Fig. 1.

Decoding means that the first drone departs from location 1. It follows the path 1-2-3-4-5-1 to location 5 and then returns to location 1; the second drone departs from location 1. It follows path 1-6-7-8-9-1 to location 9 and returns to location 1.

Fig. 1 Individual code

1	2	3	4	5	1	6	7	8	9	1
---	---	---	---	---	---	---	---	---	---	---

**Fig. 2** Cross operation



### 3.2 Fitness Value

In this algorithm, the fitness value is expressed as the total length of the route that all drones traverse all locations. The calculation formula is:

$$\text{fitness} = \sum_{i=1}^m L_i \tag{11}$$

where  $m$  represents the number of drones,  $i$  is the route of the  $i$ th drone, and  $L_i$  is the length of the  $i$ th route. The smaller the fitness, the better the individual fitness.

### 3.3 Cross Operation

The update of the individual is realized by the crossover between the individual and the individual extreme value or the group extreme value [14]. The crossover method in this algorithm adopts the integer crossover method. We cross the individual with the individual extreme value or the group extreme value and randomly select two intersection positions. If there is a duplicate position in the new individual, we need to replace the duplicate place with a place not included in the individual. The operation method is shown in Fig. 2.

### 3.4 Mutate Operation

#### Flip mutation operator

Any two pieces of code are interchanged [15]. The mutation is only carried out within one individual code such as the inversion of the individual code.

#### Swap mutation operator

Part of any two individual codes is exchanged to form a simple “swap” mutation method, as shown in Fig. 3.

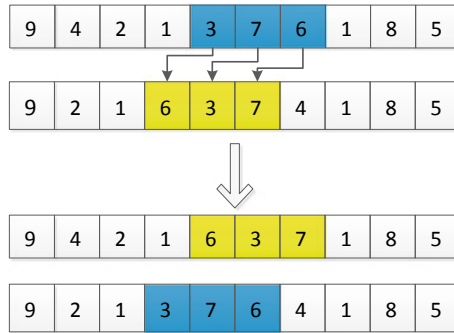


Fig. 3 Swap mutation operator

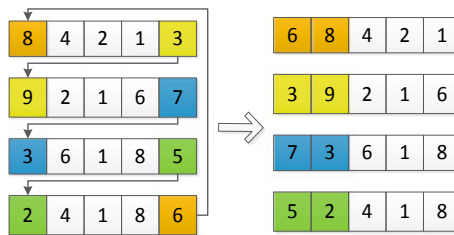


Fig. 4 Slide mutations operator

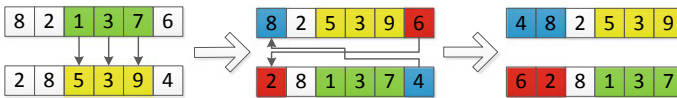


Fig. 5 “Swap + Slide” mutation operator

**Slide mutations operator**

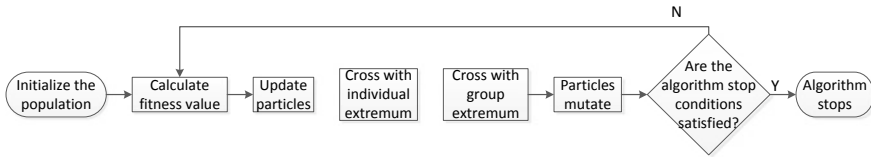
The last bit of code is transferred from one individual to another to constitute a “sliding” mutation, as shown in Fig. 4.

**“Swap + Slide” mutation operator**

It is a hybrid mutation operator that combines exchange mutation and sliding mutation which can significantly improve the performance, as shown in Fig. 5.

**3.5 Procedure of Algorithm**

The algorithm flowchart based on the optimization we proposed in the article to solve the problem is indicated in Fig. 6.



**Fig. 6** Algorithm flowchart

The procedure of algorithm is as follows:

**Step1:** Initialize the particle swarm population, encode the individual particles, set the particle swarm parameters, and initialize the particle position;

**Step2:** Use the total length of the route as the fitness function to get the fitness value of the algorithm;

**Step3:** Update the individual optimal particle and the group optimal particle according to the particle fitness value and compare the obtained fitness value with the previous best fitness. It will be accepted instead of the best fitness value previously accumulated if the latest fitness is more suitable;

**Step4:** Use the crossover operator to cross the individual and individual optimal particles to obtain new particles. Cross the individual and the group optimal particles to obtain new particles;

**Step5:** Make the particles themselves mutate through the mutation operator to get better particles; and

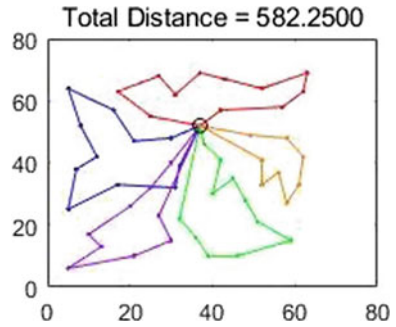
**Step6:** Determine whether the optimal solution is obtained or the algorithm has reached the specified maximum number of iterations. If the termination conditions are met, the algorithm terminates; otherwise, go to Step 2.

## 4 Result and Analysis of the Simulation

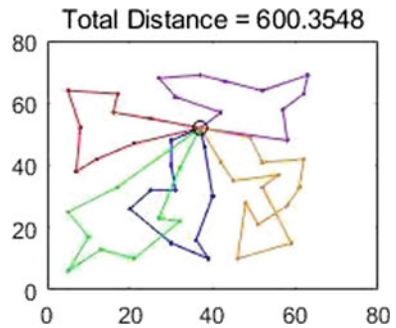
### 4.1 Simulation Result

The coordinate data of 51 locations is used as the simulation data to solve the problem in our article. The genetic algorithm and the hybrid particle swarm optimization are, respectively, used for simulation optimization [16]. This paper uses MATLAB-R2017b to make a program to implement the algorithm. It sets the number of drones to 5 and the number of detection locations to 51. The genetic algorithm sets the total group number to 100, and the maximum iteration round is 1000. We set the crossover and mutation probabilities to 0.9 and 0.1, respectively. The particle swarm size of the optimization is 200, and the maximum iteration round is 1000. Figures 7 and 8 show the results of the two algorithms.

**Fig. 7** Hybrid particle swarm optimization



**Fig. 8** Genetic algorithm



**Table 1** Comparison of simulation results

Algorithm	Optimal value	Worst value	Average value	Average number of iterations
Genetic algorithm	570.5684	627.3180	596.2746	751.24
Hybrid particle swarm optimization	581.8723	649.2876	624.2186	862.17

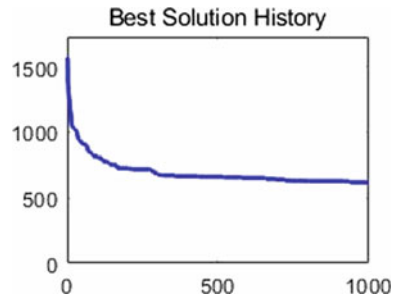
According to the simulation results, we can know that the calculation result obtained by the hybrid particle swarm optimization is 582.25. The calculation result obtained by the genetic algorithm is 600.3548. The two algorithms were run 30 times. We can see the comparison results in Table 1.

### 4.2 Result Analysis

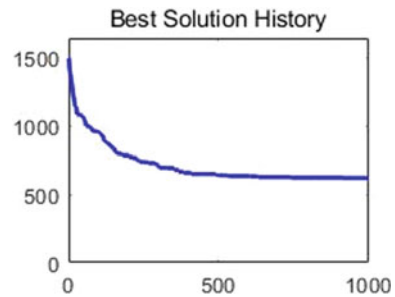
According to the simulation calculation results, it can be seen that, compared with genetic algorithm, hybrid particle swarm algorithm can greatly speed up the iteration speed of the algorithm. The crossover and mutation operations are added on the basis of the particle swarm algorithm, so that the optimal solution can be obtained



**Fig. 9** Hybrid particle swarm optimization



**Fig. 10** Genetic algorithm



faster. From the perspective of the searched route length, the calculation result of hybrid particle swarm optimization is better than genetic algorithm. It is found that increasing the population size of the genetic algorithm will significantly increase the calculation amount and computing time of the algorithm in the process of parameter adjustment. However, increasing the number of particles has a limited impact on the computing time of the algorithm in the optimization. The iterative diagrams of the two algorithms are as follows. The iterative diagrams of the two algorithms are shown in Figs. 9 and 10.

After many tests, the results show that the optimal value of hybrid particle swarm algorithm is better than the result of GA. The GA usually converge in advance, while hybrid particle swarm algorithm has a wider search range and better diversity. It can also effectively overcome premature convergence and has faster convergence speed and better results.

## 5 Conclusion

In this paper, it gives a hybrid particle swarm algorithm that adds genetic algorithm ideas to the standard PSO. It is applied to solve the problem of multi-UAV route planning. Its results are compared with the simulation results of genetic algorithm. We can know from the results given by the algorithm that the particles in the algorithm cannot be restricted by local optimal solutions. It can overcome the defect of early

convergence to a certain extent. So it has better performance. However, the individual coding method of this algorithm has room for further optimization and adjustment. So subsequent research can further improve the performance of the algorithm in this respect.

**Acknowledgements** This research was supported in Liaoning Provincial Nature Fund (2019-ZD-0661), Project of Liaoning Province Social Science Circle (2020lslktqn-058).

## References

1. Koerkamp, N.W., Borst, C., Mulder, M.: Supporting humans in solving multi-UAV dynamic vehicle routing problems. *IFAC-PapersOnLine* **52**(19), 359–364 (2019)
2. Coutinho, W.P., Battarra, M., Fliege, M.: The unmanned aerial vehicle routing and trajectory optimisation problem, a taxonomic review. *Comput. Ind. Eng.* **120**, 116–128 (2018)
3. Kencana, E.: The performance of ant system in solving multi traveling salesmen problem. *Procedia Comput. Sci.* **124**, 46–52(2017)
4. Jiang C., Wan Z., Peng Z.: A new efficient hybrid algorithm for large scale multiple traveling salesmen problems. *Expert Syst. Appl.* **139**, 112867(2020)
5. Lu, L.C., Yue, T.Y.: Mission-oriented ant-team ACO for min-max MTSP. *Appl. Soft Comput.* **76**, 436–444 (2019)
6. Zhou, H.L., Song, M.L., Pedrycz, W.: A comparative study of improved GA and PSO in solving multiple traveling salesmen problem. *Appl. Soft Comput.* **64**, 564–580 (2018)
7. Dhein, G., Serrano, M.: Olinto César Basside. minimizing dispersion in multiple drone routing. *Comput. Oper. Res.* **109**, 28–42 (2019)
8. Müller, S., Rudolph, C., Janke, C.: Drones for last mile logistics: Baloney or part of the solution? *Transp. Res. Procedia* **41**, 73–87 (2019)
9. Pugliese, L.D.P., Guerriero, F., Macrina, G.: Using drones for parcels delivery process. *Procedia Manuf.* **42**, 488–497 (2019)
10. Coelho, B.N., Coelho, V.N., Coelho, I.M.: A multi-objective green UAV routing problem. *Comput. Oper. Res.* **88**, 306–315 (2017)
11. Dasdemir, E., Ksalan, M.E., Öztürk, D.T.: A flexible reference point-based multi-objective evolutionary algorithm: an application to the UAV route planning problem. *Comput. Oper. Res.* **114**, 104811 (2020)
12. Ning, Q., Tao, G.P., Chen, B.C.: Multi-UAVs trajectory and mission cooperative planning based on the Markov model. *Phys. Commun.* **35**, 100717 (2019)
13. Yang, X.D., Li, G.Y.: GA-PSO based decomposition of distributed model predictive control systems. *Sci. Technol. Eng.* **19**(25), 262–267 (2019). (in Chinese)
14. Xu, J., Pei, L., Zhu, R.Z.: Application of a genetic algorithm with random crossover and dynamic mutation on the travelling salesman problem. *Procedia Comput. Sci.* **131**, 937–945 (2018)
15. Hacizade, U., Kaya, I.: GA based traveling salesman problem solution and its application to transport routes optimization. *IFAC-PapersOnLine* **51**(30), 620–625 (2018)
16. Mu, X.C., Xie, D.L., Yan, W., Nie, J., Li, X.: Research based on genetic algorithm traveling sealer problem of trajectory optimization. *J. Syst. Simul.* **25**, 2013 (2013). (in Chinese)