An Improved NSGA-II Algorithm for UAV Path Planning Problems

Haoyu Wang, Li Tan, Jiaqi Shi, Xinyue Lv, Xiaofeng Lian

College of Computer Science and Engineering, Beijing Technology and Business University, China
847136435@qq.com, tanli@th.btbu.edu.cn, 18811357269@163.com, 759910778@qq.com, lianxf@th.btbu.edu.cn

Abstract

The path planning problem is an important research in the field of UAV application. In practical applications, the path planning problem is usually multi-objective. This paper proposes an improved NSGA-II algorithm to achieve multi-objective optimization path planning. The algorithm introduces an improved directional mutation strategy by adaptively adjusting the crossover probability and the mutation probability, and searches for the optimal path of the UAV under the premise of considering the path length, threat, and concealment. Simulation experiments show that, compared with the NSGA and NSGA-II algorithm, the improved NSGA-II algorithm can reduce the risk of falling into a local optimum, increase the convergence speed, and better realize path planning in an obstacle environment.

Keywords: Path planning, Multi-objective optimization problem (MOOP), NSGA-II, Mutation strategy

1 Introduction

The main aim of the UAV path planning problem is to find a feasible path that allows the UAV to reach the destination safely without collisions. Geometric methods [1], spatial methods [2], and graph search methods [3] are all used to solve this problem. The improvement and mutual combination of the above methods have achieved good results in solving the shortest path length. However, in practical applications, multiple factors such as length, safety, and time need to be considered. Therefore, multiple optimization goals need to be considered at the same time to obtain an accurate solution. It can be regarded as a multi-objective optimization problem (MOOP).

There are two main methods for solving multi-objective optimization problems. The first is to use a weighted sum function to convert a multi-objective problem into a single-objective problem [4]. The characteristic of this method is that the weight of different optimization targets needs to be set in advance, and the weights are changed during the optimization process. Cheng et al. used the weighted sum of the transportation parameters as the input of the neural network model and the result was high in accuracy. But the impact of environmental factors on path planning was not considered, which is difficult to complete multi-objective optimization under an uncertain environment [5]. Zhang et al. used the key point selection strategy to improve the A* algorithm, removed the redundant inflection points and nodes in the path, but did not solve the problem of multi-objective optimization under actual conditions [6]. The second method is to provide a set of solutions based on the pareto optimal solution set. The pareto set means the improvement of a certain objective may cause a reduction in other objectives, and a set of optimal solutions of the objective function is called the pareto optimal solution set. Researchers can use the pareto optimal solution set to select the most suitable solution according to the actual situation of the application case. In recent years, many researchers have proposed the methods of using pareto, including ACO (ant colony optimization), SAA (simulated annealing algorithm), GA (genetic algorithm) and so on [7]. Among them, genetic algorithms based on meta-heuristic algorithms have become effective methods to solve MOOP, such as the New Approach to Multi-Objective A* (NAMOA*) [8], Multi-Objective Genetic Algorithms (MOGA) [9], Strength Pareto Evolutionary Algorithm 2 (SPEA2) [10], Non-dominated Sorting Genetic Algorithm-II (NSGA-II) [11].

NSGA-II is an improved algorithm based on the NSGA (Non-dominated Sorting Genetic Algorithm). The NSGA uses virtual fitness and shared radius technology to calculate the dominant relationship between individuals. By performing stratification before selection operations, individuals with good performance can have a higher probability of inheriting to the next generation. The NSGA-II inherits the non-dominated stratification idea of the NSGA and uses the crowding distance operator to replace the shared radius, introduces the elite strategy and the fast non-dominated sorting method, which shows excellent performance in path planning applications [12]. Davoodi et al. proposed to use NSGA-II as a framework for path length and a gap in discrete space, used a new path

*Corresponding Author: Li Tan; E-mail: tanli@th.btbu.edu.cn
DOI: 10.3966/160792642021052203008

optimization operator to solve the path planning problem in complex environments, but this method needed to change the crossover probability manually in different situations and might converge to the local optimum prematurely [13]. Lucas et al. considered the impact of ocean currents and used the NSGA-II to solve the four-dimensional multi-objective optimization path planning problem of underwater gliders, but its fitness function was only applicable to specific marine structural environments [14]. For the urban environment, Ren et al. proposed a distance-based CDNSGA-II method [15] by considering distance and safety. However, this method used the hierarchical structure data of octree and easily fell into the trap of local minima in an unstructured environment. Majumder et al. used the uncertainty theory to solve the multi-objective shortest path problem, but its optimization only aimed at the shortest path and did not consider other goals [16]. Figueras-Benitez et al. used NSGA-II with dual objective functions to solve the route optimization problem of FBS, but they put too much emphasis on the structured environment and did not consider the path planning problem in the complex and irregular environment [17].

Although NSGA-II algorithm has a fast running speed and strong convergence when solving the multi-objective path planning problem, and can find a set of paths with higher adaptability for multi-objective functions quickly with pareto, it also has limitations [18]. NSGA-II searches for the global optimum, it continues the basic process of genetic algorithm though the crowding distance operator is used, so there are still weak local search ability and the risk of premature convergence: the path found may be optimized for a single optimization goal in a complex environment especially. Moreover, NSGA-II failed to make full use of its feedback information in iterations, which makes its crossover and mutation proceed in the same direction and may evolve towards a non-optimal direction. At the same time, most of the current researches are aimed at path planning in structured environments or urban environments. Few researchers mention the impact of complex environments (such as post-disaster reconstruction or rescue search in mountain scenarios) on UAV path planning. To overcome the shortcomings and solve the path planning problem in complex 3D mountain scenarios, this paper dynamically adjusts the crossover and mutation probability through exponential distribution, and uses the directional mutation strategy to replace the random mutation mechanism of NSGA-II, and proposes an improved NSGA-II algorithm for multi-objective UAV path planning problem.

2 Problem Description and Spatial Modeling

2.1 Problem Description

The planned path is a feasible non-collision path formed by the points passed by the UAV from the initial position to the target position in a preset obstacle space. The path must meet the constraints, such as the shortest path and the least time. The essence of the path planning algorithm is to find an optimal path or a collection of multiple optimized paths according to the conditions among all paths that meet the constraints.

2.2 Environmental Modeling

Environmental modeling is the prerequisite for the simulation of various UAV operations. In this paper, the grid graph method is used to model the environment, and the working environment of the UAV is divided into a series of grid areas of the same size. The schematic diagram of environmental modeling is shown in Figure 1.

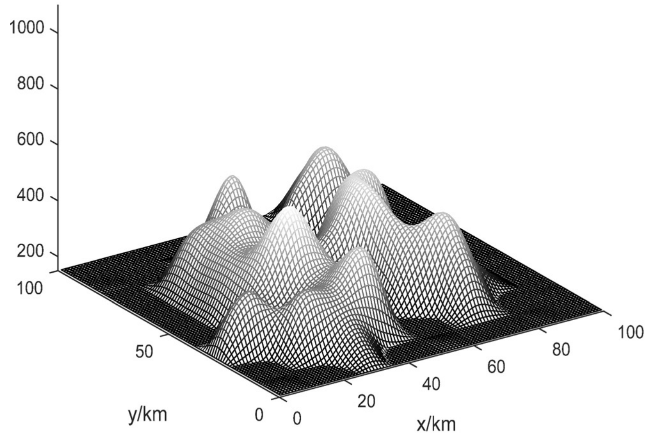


Figure 1. Schematic diagram of environment modeling

3 Improved NSGA-II Algorithm

3.1 Chromosome Coding

This paper uses integer representation and uses an entire chromosome as the solution. To improve efficiency, the chromosome is represented as a single-linked list in this algorithm. Figure 2 shows the data structure of the chromosome with length L.

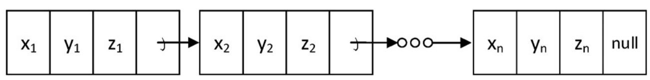


Figure 2. Data structure of chromosome

3.2 Initialization Population

Initializing each chromosome in the population means randomly generating multiple feasible paths. First, a discontinuous path is generated. Since there is at least one grid in each row in the feasible path, a barrier-free grid is randomly selected in each row in order during initialization to form a discontinuous path, the first and last grid are the start position and the target position. Then, the discontinuous path is connected into a continuous path by judging whether two adjacent grids are continuous from the first to the last grid. The path set obtained is the initial population.

3.3 Fitness Function

Considering the application scene of UAVs, time-related path length and safety are important optimization goals. Therefore, three factors, path length, threat, and concealment, are selected as the optimization goals of path planning in this paper. The fitness function is shown below.

3.3.1 Route Length

This function aims to make the path length as short as possible. The route length is the sum of the path lengths calculated from the ordered coordinate points, as is shown in formula (1).

$$f_1 = \sum_{i=1}^{n-1} d\left(P_i, P_{i+1}\right)$$
$$d\left(P_i, P_{i+1}\right) = \sqrt{(X_{i+1}-X_i)^2 + (Y_{i+1}-Y_i)^2 + (Z_{i+1}-Z_i)^2} \quad \textbf{(1)}$$

Where $d(P_i, P_{i+1})$ is the distance between point i and point i+1, and X_i, Y_i, Z_i represent the coordinates of point i.

3.3.2 Threat Index

This function is designed to calculate the flight smoothness on the decision path. When the smoothness exceeds a predetermined value, the threat index increases. The threat index can be calculated by two adjacent coordinate points, as is shown in formula (2).

$$\alpha_i = \operatorname{atan}\left[\frac{\left(X_{i-1}-X_i\right)\left(X_{i+1}-X_i\right)+\left(Y_{i-1}-Y_i\right)\left(Y_{i+1}-Y_i\right)}{d\left(P_{i-1}, P_i\right) * d\left(P_{i+1}, P_i\right)}\right]$$
$$f_2 = \sum_{i=2}^{n-1}\left(\pi - \alpha_i\right) \quad \textbf{(2)}$$

3.3.3 Concealment Index

This function is designed to calculate the safety degree of the decision path. The concealment is related to the flight height of the path and the distance between the path and obstacles. The concealment calculation formula is shown in formula (3).

$$S_i = (3 * (h_{\max} + d_{safe}) * \sqrt{(X_{i+1}-X_i)^2 + (Y_{i+1}-Y_i)^2}$$
$$S_j = (Z_j - d_{safe}) * \sqrt{(X_{i+1}-X_i)^2 + (Y_{i+1}-Y_i)^2} \quad \textbf{(3)}$$
$$f_3 = \begin{cases} \sum_{i=2}^{n-1} S_i \ , \ d\left(P_i, P_{i+1}\right) < d_{safe} \\ \sum_{i=2}^{n-1} S_j \ , \ d\left(P_i, P_{i+1}\right) > d_{safe} \end{cases}$$

Where h_{max} represents the height of the highest point in the environment, and d_{safe} is a preset constant, representing a safe distance.

3.4 Non-dominated Sorting

Using fitness functions to perform non-dominated sorting for the initial population and obtaining multiple levels of non-dominated layers. Each non-dominated layer contains multiple path individuals, a group of path individuals in multiple levels of non-dominated layers is set as population I_{fa}. In I_{fa}, starting from the first non-dominated layer, the paths in the layer will be passed on to the next generation firstly.

3.5 Fast Non-dominated Sorting

Performing genetic operations on I_{fa} can get I_{son}. I_{fa} and I_{son} are merged into the new path population P through fast non-dominated sorting, so that the first, the second, until the last non-dominated layers have priority order to pass on to the next generation, while reducing the computational complexity. The fast non-dominated sorting steps are as follows:

Calculating n_p and S_p for each individual in P, n_p is the number of dominating individuals p in the population, and S_p is the set of individuals dominated by p. The fast non-dominated sorting steps of the algorithm are as follows:

(1) Save all individuals with n_p=0 to set F1.

(2) For each individual i in the set F1, S_i is the set of individuals dominated by it, traverse each individual l in S_i, execute n_l=n_l-1, when nl=0, save the individual l to set H.

(3) Take the individuals obtained in F1 as the first non-dominated level individuals, and H as the current set

(4) Repeat operation (2)-(3) until the population stratification is completed.

3.6 Crowding Distance

Crowding distance refers to the density of surrounding individuals of a given individual. It is used to avoid the appearance of two similar paths, allow the population to evolve in the direction of better optimization goals, and increase the diversity of generation paths. The crowding distance of each individual is defined as n_d, which is initially 0. The calculation method is as follows:

(1) Sorting the population based on the objective

function value f.

(2) Setting the crowding distance of the two borders as infinite.

(3) Calculate $n_d=n_d+[f(i+1)-f(i-1)]$.

According to the non-dominated ranking n_{rank} and the crowding distance n_d, the dominance order of any two entities is compared. The comparison operator is defined as $\geq_n$, and the comparison basis is as follows:

$i \geq_n j$, then individual i is better than individual j if and only if $i_{rank}<j_{rank}$ or $i_{rank}=j_{rank}$ and $i_d>j_d$.

According to the sorting result, perform the selection.

3.7 Improved Mutation Strategy

Mutation strategy is an important means to cultivate outstanding individuals in evolutionary algorithms and get rid of local optima. The improved NSGA-II algorithm focuses on improving the mutation strategy. Generally speaking, there are two most commonly used differential mutation strategies in evolutionary algorithms, which are shown in formulas (4) and (5).

$$v_i^g = x_{r_1}^g + F(x_{r_2}^g - x_{r_3}^g) \quad \textbf{(4)}$$

$$v_i^g = x_{\text{best}}^g + F(x_{r_1}^g - x_{r_2}^g) \quad \textbf{(5)}$$

In formulas 4-5, $x_{r_1}^g$ represents a random individual and x_{best}^g represents the best individual in the current population, these two are called basic vectors. F is the scaling factor, which is a constant. $x_{r_2}^g - x_{r_3}^g$ and $x_{r_1}^g - x_{r_2}^g$ are called the difference vector. Formula 5 is called the DE/rand strategy, which focuses on the randomness of mutations, and formula 6 is called the DE/best strategy, which focuses on the characteristics of the best individuals in the population.

In recent years, many researchers have studied the mutation strategies of evolutionary algorithms, such as hybrid mutation strategies combined with PSO [19], mutation strategies using neighborhood direction infor -mation [20], model-based adaptive mutation strategies [21], adaptive ranking mutation strategy [22] and target mutation strategy [23-25], etc. Among them, Target Mutation Strategy (TM) is a new type of differential evolutionary mutation strategy proposed by Zheng et al. Its overall performance is better than the traditional differential evolution mutation strategy, and it has the advantages of simple operation and portability. The TM strategy is shown in formula (6).

$$v_i^g = x_{best}^g + F(x_r^g - x_i^g) \quad \textbf{(6)}$$

x_{best}^g is the best individual in current population, x_r^g is a random vector, x_{i}^g is a target vector, and F is a random number between (0, 1). To improve the convergence speed and enhance the search ability, this strategy uses the global optimal solution as the basic vector to search for a new solution in the optimal direction. Besides, the TM strategy uses x_r^g with random information and x_{i}^g with deterministic information to obtain the difference vector, which can well balance the relationship between the randomness and determinism of the difference vector, thereby avoiding the problems of low search efficiency and slow convergence caused by excessive randomness.

The TM strategy strengthens the search efficiency of the mutation operator and improves the search ability for the optimal individual, but it still needs to be strengthened in avoiding local optima. Therefore, based on the random mutation mechanism, the improved directional mutation strategy mainly includes two parts:

(1) The directional mutation strategy is based on the TM strategy, as is shown in formula (7).

$$x_i^{t+1} = x_i^t + F(x_j^t - x_i^t) \quad \textbf{(7)}$$

Where F is the scale factor between (0, 1). Compared with formula (6), formula (7) uses the target vector x_{i}^t instead of x_{best}^g as the basic vector to expand the search range of the mutation operator and reduce the local optimum risk.

(2) Based on the idea of random mutation, random variables are introduced into the mutation operator of NSGA-II to improve the ability of the algorithm to jump out of local extremes.

The improved directional mutation strategy is shown in Equation (8).

$$\begin{cases} x_i^{t+1} = x_{best}^t + F(x_i^t - x_i^t) & if\ rand < \text{w} \\ x_i^{t+1} = x_i^t + F(x_j^t - x_i^t) & if\ rand > \text{w} \end{cases} \quad \textbf{(8)}$$

Where rand is a random number between (0, 1), and w is the threshold.

3.8 Adaptive Adjustment of Crossover and Mutation Probability

The crossover operator and mutation operator in NSGA-II are the same as GA, so the crossover probability and the mutation probability are fixed constants, that's the reason why the NSGA-II falls into the local optimum. This paper adopts the method of adaptively adjusting the crossover probability and mutation probability. By dynamically adjusting the crossover probability and mutation probability using population information, the algorithm's ability to avoid local optimum is improved.

In the early stages of the iteration, to improve the initial global search capability of the algorithm and increase its convergence speed, the crossover probability and mutation probability should be as large as possible. In the later stage of the iteration, as the algorithm gradually approaches the optimal solution, the crossover probability and the mutation probability are relatively adjusted to smaller values to improve the local search ability of the algorithm and strengthen the search for the optimal solution set. The crossover probability and mutation probability used in the improved algorithm are shown in formula (9).

$$P_c = P_{cmax} \times e^{\frac{-3g(P_{cmax}-P_{cmin})}{G}}$$
$$P_m = P_{mmax} \times e^{\frac{-3g(P_{mmax}-P_{mmin})}{G}} \quad \textbf{(9)}$$

Where G is the maximum number of iterations, g represents the current number of iterations, P_c is the crossover probability, P_{cmax} is the maximum crossover probability, P_{cmin} is the minimum crossover probability, Pm is the mutation probability, P_{mmax} is the maximum mutation probability, and P_{mmin} is the minimum mutation probability.

3.9 The Flow of Improved NSGA-II Algorithm

The overall flow of the improved NSGA-II algorithm is as follows, and the flow chart is shown in Figure 3:

(1) Perform non-dominated sorting on I_{fa} to obtain the first generation filial population I_{son}.

(2) Merging I_{fa} with I_{son} to get population P. Using fast non-dominated sorting and crowding distance strategy to form a new parent population P_{fa}.

(3) Adaptively crossover operation, the improved mutation strategy are performed on P_{fa} to obtain the filial population P_{son}.

(4) Repeat (2)-(3) until the loop end condition is met.

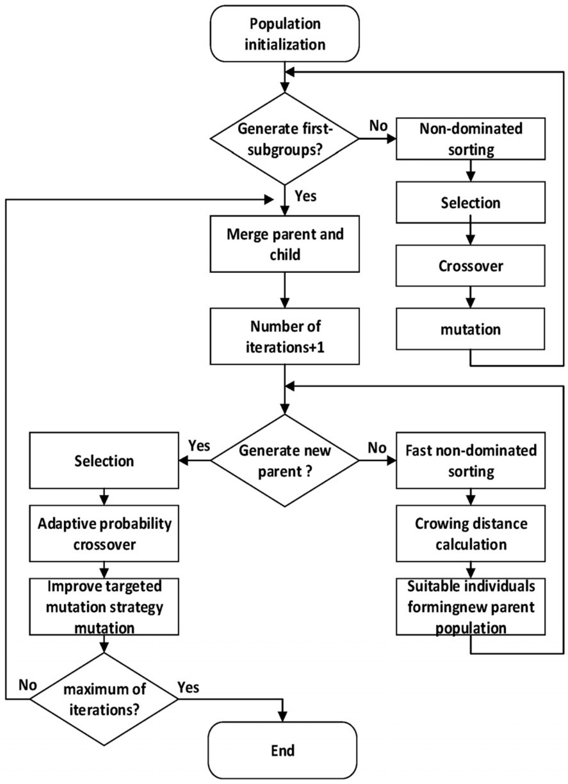


Figure 3. Improved NSGA-II algorithm flow chart

4 Experiment Results

To verify the performance of the improved NSGA-II algorithm in the UAV path planning problem, multiple sets of experiments in different environments are designed. The experimental parameters are shown in Table 1.

Table 1. Simulation experiments parameters

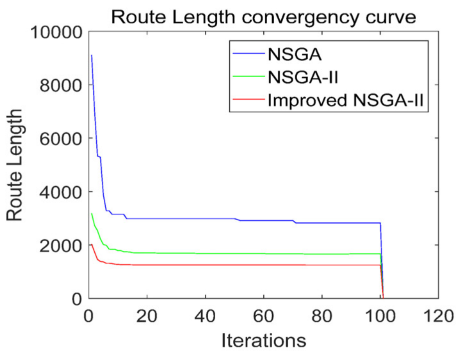
Simulation environment size/m	Scale factor F	Number of iterations	Cross probability boundary	Mutation probability boundary
100*100*1000	0.3	100	[0.2, 0.8]	[0.2, 0.8]

4.1 Experiment A: In the Simple Environment

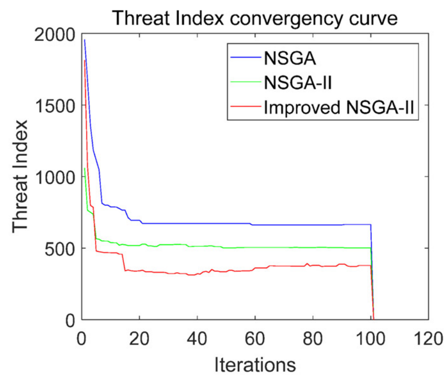
In a simple environment, the start point of the UAV is set to (0, 0, 0), the endpoint is set to (100, 100, 0), the improved NSGA-II algorithm is the experimental group, and the NSGA algorithm and NSGA-II algorithm are the control group. The simulation results are shown in Figure 4(a) to Figure 4(e). A comparison table of algorithm route data is obtained from 10 random experiments, which is shown in Table 2.

From Figure 4(a) to Figure 4(c), it can be seen that in terms of three fitness function, the improved NSGA-II approaches the optimal at the 10th iteration, 15th iteration and 8th iteration respectively, the NSGA are 45th, 39th and 60th iterations, the NSGA-II are 20th, 17th and 85th iterations. Therefore, the improved NSGA-II algorithm has a greater improvement in terms of route length and threat. However, in terms of concealment, the improved NSGA-II algorithm doesn't show much improvement in the final optimization parameters. This is because, in the simple environment experiments, the routes selected by the two algorithms are all along with the valley hollow, and the concealment index measures the height of the route from the ground, so the gap in concealment index is relatively small.

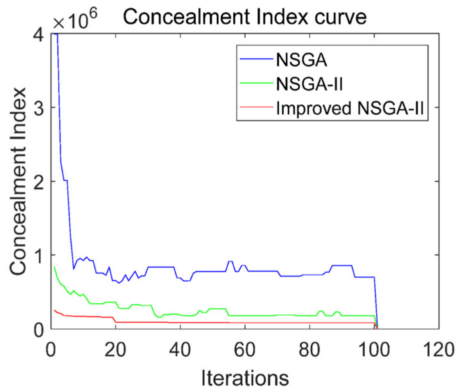
From Figure 4(d) to Figure 4(e), the path optimized by the improved NSGA-II has a lower height from the ground and a greater distance from obstacles than the paths obtained by NSGA and NSGA-II. It can reach the destination fast and smoothly, and the path conditions are more adapted to the needs of the real environments.



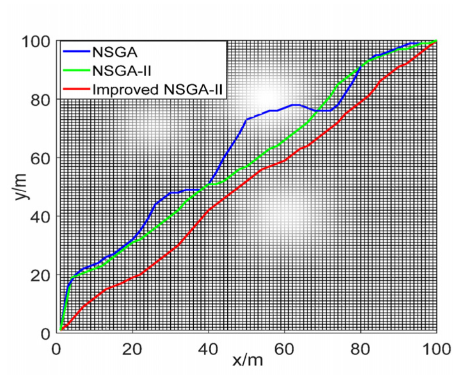
(a) Route length curve



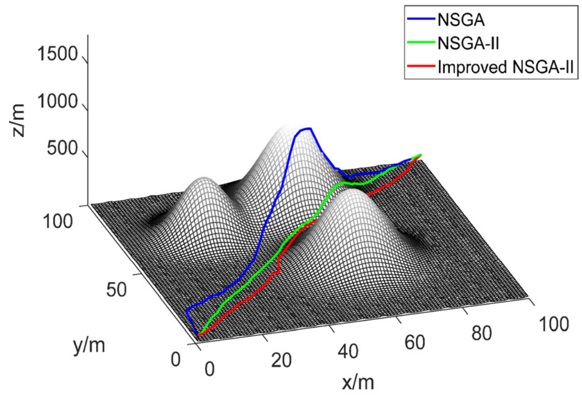
(b) Threat index curve



(c) Concealment curve



(d) Two-dimensional trajectory diagram



(e) Three-dimensional trajectory diagram

Figure 4. Algorithm simulation comparison chart

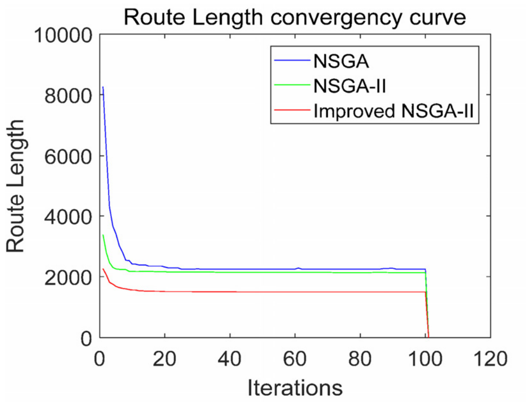
Table 2. Comparison of route data in experiment A

	Route length			Threat index			Concealment (*10^5)		
	NSGA	NSGA-II	ImprovedNSGA-II	NSGA	NSGA-II	ImprovedNSGA-II	NSGA	NSGA-II	ImprovedNSGA-II
1	2743	1676	1446	679.2	457.6	341.8	7.841	2.934	2.884
2	2537	1698	1637	746.4	412.1	358	7.388	3.864	3.157
3	2641	1701	1435	839.5	442.4	367.7	7.114	3.539	2.853
4	2624	1543	1437	631.2	484.2	315.5	6.961	3.183	2.82
5	2663	1579	1393	722.9	444.1	365.5	6.358	3.185	2.666
6	2564	1602	1599	716.1	335.4	328.9	7.417	3.499	3.126
7	2697	1626	1513	764.7	427.9	361.2	6.915	2.942	2.781
8	2516	1634	1633	703.3	538.4	329.4	6.422	3.597	2.535
9	2713	1691	1478	621.5	528.4	304	7.217	2.993	2.983
10	2612	1581	1398	665.2	499.8	388	7.372	3.895	3.263
Ave	2631	1633.1	1496.9	709	457.03	346	7.1005	3.3631	2.9068

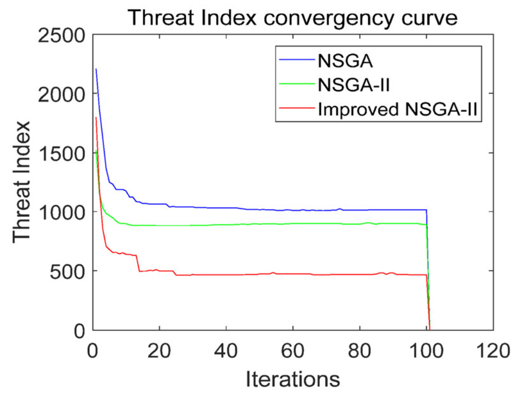
It can be concluded from Table 2 that, in terms of shortening the length of the path, the improved NSGA-II algorithm has an average increase of 49.15% and 18.08% compared with NSGA and NSGA-II. In terms of reducing the threat, compared with NSGA and NSGA-II, the improved NSGA-II algorithm has an average increase of 51.2% and 24.29% respectively. In terms of improving concealment, the average increase is 59.06% and 13.57% respectively, compared with NSGA and NSGA-II.

4.2 Experiment B: In the Complex Environment

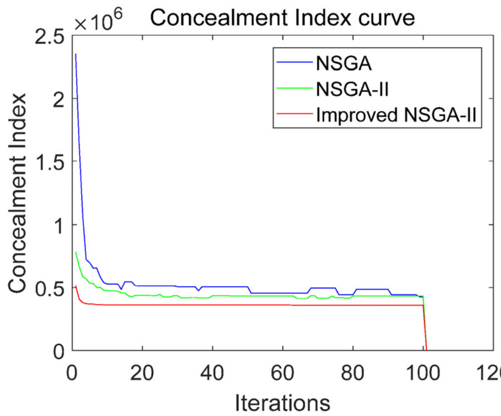
In a complex environment, the start point is set to (0, 0, 0), the endpoint is set to (100, 100, 100), the improved NSGA-II algorithm is the experimental group, and the NSGA algorithm and NSGA-II algorithm are the control group. The simulation results are shown in Figure 5(a) to Figure 5(e). Through 10 random experiments, the algorithm route data comparison table is obtained, which is shown in Table 3.



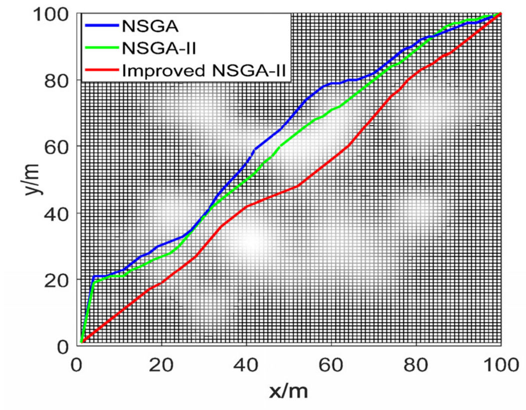
(a) Route length curve



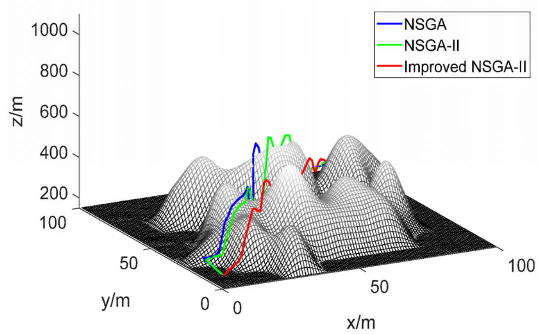
(b) Threat index curve



(c) Concealment curve



(d) Two-dimensional trajectory diagram



(e) Three-dimensional trajectory diagram

Figure 5. Algorithm simulation comparison chart

Table 3. Comparison of route data in experiment B

	Route length			Threat index			Concealment (*10^5)		
	NSGA	NSGA-II	Improved NSGA-II	NSGA	NSGA-II	Improved NSGA-II	NSGA	NSGA-II	Improved NSGA-II
1	2315	2128	1446	994.3	913.6	258.9	5.148	4.372	3.431
2	2307	2106	1637	1052.4	896.8	238.8	4.196	4.385	2.987
3	2290	2085	1435	1051	891.5	350.8	5.075	4.654	3.183
4	2287	2141	1437	1037	879.2	358.3	5.541	3.898	3.246
5	2167	2104	1393	1051	910.5	344.4	4.95	4.565	3.186
6	2345	1613	1599	1039.3	321.4	333	4.596	2.979	3.084
7	2251	2113	1513	1053	911.3	361.4	4.275	4.119	2.551
8	2298	2051	1633	1057	898.4	339.2	4.782	3.98	3.161
9	2237	2116	1478	1042.7	891.7	254.3	4.995	4.508	3.312
10	2255	2087	1398	1045.3	907.2	293.9	5.217	4.513	3.426
Ave	2275.2	2101	1496.9	1042.3	900.58	313.3	4.8775	4.3235	3.1567

From Figure 5(a) to Figure 5(c), it can be seen that in terms of three fitness function, the improved NSGA-II approaches the optimal at the 10th iteration, 4th iteration and 3th iteration respectively, the NSGA are 17th, 12th and 42th iterations, the NSGA-II are 11th, 25th and 61th iterations. Because the experimental terrain is relatively complex, the path planning capabilities of the three algorithms are different, there is no such phenomenon that the concealment index is nearly the same caused by the three algorithms choosing the same path. It can be seen that, compared to simple environments, the improved NSGA-II algorithm is more suitable for path planning in complex environments.

From Figure 5(d) to Figure 5(e), it can be seen that the path optimized by the improved NSGA-II can fly along the valley recesses compared with the paths obtained by NSGA and NSGA-II, which takes account of threat and concealment. It also ensures that the length of the route is relatively low and the path conditions are more suitable for the needs of the actual environments.

The experimental data in Table 3 shows that in terms of shortening the length of the route, the improved NSGA-II algorithm has an average increase of 34.21% and 28.75% compared with NSGA and NSGA-II. In terms of reducing the threat, compared with the NSGA and NSGA-II algorithms, the improved NSGA-II algorithm has an average increase of 69.94% and 65.21% respectively. In terms of improving concealment, the improved NSGA-II algorithm has increased by 35.28% and 26.99% compared with NSGA and NSGA-II, respectively. Therefore, the improved NSGA-II has a significant improvement in reducing the route length, reducing the threat, increasing the concealment, and obtaining better path performs in complex environments.

It is worth noting that the improved NSGA-II algorithm in the sixth set of experiments does not improve the threat and concealment index compared with the NSGA-II algorithm. This is because the NSGA-II algorithm does not follow mountain obstacles when generating paths. Instead, it vertically rises to the highest point near the start point and crosses the entire obstacle environment horizontally, so its path length is less different from the improved NSGA-II algorithm, and the threat and concealment are even lower than the improved NSGA-II algorithm. In actual situations, the main task of the UAV is operations near the ground, so the UAV is not allowed to pass through the target area in a straight line at the maximum height. The results of this group of experiments should be regarded as a special case.

In summary, the improved NSGA-II algorithm can accurately complete the path planning, especially in the complex mountains obstacle environment. In the middle and late stages of the iteration, it can better get rid of the local optimum and suppress the premature phenomenon. Compared with the NSGA and NSGA-II algorithm, the improved NSGA-II algorithm has greatly improved in terms of route length, threat, and concealment

5 Conclusion

Aiming at the problem of UAV path planning, this paper proposes an improved NSGA-II algorithm. Through dynamic adaptive adjustment of crossover probability and mutation probability, combined with an improved directional mutation strategy, it achieves accurate path planning in obstacles environment, and the search for the optimal path avoids the premature phenomenon of falling into the local optimum. The simulation experiments show that the improved NSGA-II algorithm can effectively complete the path planning in the obstacle environment.

Acknowledgments

This work was supported by Beijing Natural Science

Foundation-Haidian Primitive Innovation Joint Fund grant number (L182007); the National Natural Science Foundation of China grant number (61702020); and Beijing Natural Science Foundation grant number (4172013).

References

[1] N. S. V. Rao, N. Stoltzfus, S. S. Iyengar, A ‘retraction’ Method for Learned Navigation in Unknown Terrains for a Circular Robot, *IEEE Transactions on Robotics and Automation*, Vol. 7, No. 5, pp. 699-707, October, 1991.

[2] J. Minguez, L. Montano, Extending Collision Avoid Ance Methods to Consider the Vehicle Shape, kinematics, and Dynamics of a Mobile Robot, *IEEE Transactions on Robotics*, Vol. 25, No. 2, pp. 367-381, April, 2009.

[3] P. P. Wu, D. Campbell, T. Merz, Multi-objective Four-Dimensional Vehicle Motion Planning in Large Dynamic Environments, *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, Vol. 41, No. 3, pp. 621-634, June, 2011.

[4] X. L. Fan, X. Y. Zhang, F. F. Du, An Improved Routing Algorithm Based on Energy Efficient Ant Colony, *Journal of Internet Technology*, Vol. 20, No. 2, pp. 581-587, March, 2019.

[5] J. R. Cheng, B. Y. Liu, K. Q. Cai, X. Y. Tang, B. Y. Zhang, ETC Intelligent Navigation Path Planning Method, *Journal of Internet Technology*, Vol. 19, No. 2, pp. 619-631, March, 2018.

[6] Y. Zhang, L. L. Li, H. C. Lin, Z. W. Ma, J. Zhao, Development of Path Planning Approach Using Improved A-star Algorithm in AGV System, *Journal of Internet Technology*, Vol. 20, No. 3, pp. 915-924, May, 2019.

[7] D. Y. Zhou, P. F. Wang, X. Y. Li, K. Zhang, Cooperative Path Planning of multi-UAV Based on multi-objective optimization algorithm, *Systems Engineering and Electronics*, Vol. 39, No. 4, pp. 782-787, April, 2017.

[8] L. Hernández-Hernández, A. Tsourdos, H. S. Shin, A. Waldock, Multi-objective Uav Routing, *International Conference on Unmanned Aircraft Systems*, Orlando, FL, USA, 2014, pp. 534-542.

[9] M. Niendorf, P. T. Kabamba, A. R. Girard, Stability Analysis of Multi-objective Planning Problems for Unmanned Aircraft, *IEEE Conference on Decision and Control*, Osaka, Japan, 2015, pp. 7238-7243.

[10] E. Zaenudin, R. Sustika, H. Pardede, Using Cluster for Fixing Kernel Transpose to Improve Parallel of Strength Pareto Evolutionary Algorithm 2 (pSPEA2), *International Conference on Computer, Control, Informatics and its Applications*, Jakarta, Indonesia, 2017, pp. 63-68.

[11] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II, *IEEE Transactions on Evolutionary Computation*, Vol. 6, No. 2, pp. 182-197, April, 2002.

[12] K. Kesireddy, W. Shan, H. Xu, Global Optimal Path Planning for Multi-agent Flocking: A Multi-Objective Optimization Approach with NSGA-III, *2019 IEEE Symposium Series on Computational Intelligence*, Xiamen, China, 2019, pp. 64-71.

[13] M. Davoodi, F. Panahi, A. Mohades, S. N. Hashemi, Multi-Objective Path Planning in Discrete Space, *Applied Soft Computing Journal*, Vol. 13, No. 1, pp. 709-720, January, 2013.

[14] C. Lucas, D. Hernadez-Sosa, R. Caldeira, Multi- Objective Four-Dimensional Glider Path Planning Using NSGA-II, *2018 IEEE/OES Autonomous Underwater Vehicle Workshop*, Porto, Portugal, 2018, pp. 1-5.

[15] Q. Ren, Y. Yao, G. Yang, X. Zhou, Multi-objective Path Planning for UAV in the Urban Environment Based on CDNSGA-II, *2019 IEEE International Conference on Service-Oriented System Engineering*, San Francisco East Bay, CA, USA, 2019, pp. 350-355.

[16] S. Majumder, M. Kar, S. Kar, T. Pal, Uncertain Programming Models for Multi-objective Shortest path Problem with Uncertain Parameters, *Soft Computing*, Vol. 24, No. 12, pp. 8975-8996, June, 2020.

[17] G. Figueras-Benitez, R. Badra, Genetic Algorithm for Biobjective Optimization of Indoor LTE Femtocell Deployment, *International Journal of Communication systems*, Vol. 33, No. 16, pp. e4564, November, 2020.

[18] P. Duan, J. Li, H. Sang, Y. Han, Q. Sun, A Developed Firefly Algorithm for Multi-Objective Path Planning Optimization Problem, *2018 IEEE 8th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems*, Tianjin, China, 2018, pp. 1393-1397.

[19] Y, Shi, H. Gao, D. Wu, An Improved Differential Evolution Algorithm with Novel Mutation Strategy, *2014 IEEE Symposium on Differential Evolution*, Orlando, FL, USA, 2014, pp. 1-8.

[20] J. Liao, Y. Cai, Y. Chen, T. Wang, H. Tian, Improving Differential Evolution with Ring Topology-Based Mutation Operators, *2014 Ninth International Conference on P2P, Parallel, Grid, Cloud and Internet Computing*, Guangdong, China, 2014, pp. 103-109.

[21] M. Mayo, M. Daoud, An Adaptive Model-based Mutation Operator for the Wind Farm Layout Optimisation Problem, *2015 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Kowloon, Hong Kong, China, 2015, pp. 671-676.

[22] W. Gong, Z. Cai, D. Liang, Adaptive Ranking Mutation Operator Based Differential Evolution for Constrained optimization, *IEEE Transactions on Cybernetics*, Vol. 45, No. 4, pp. 716-727, April, 2015.

[23] Y. Wang, D. Li, Y. Gao, Improved Flower Pollination Algorithm Based on Mutation Strategy, *Application Research of Computers*, Vol. 34, No. 12, pp. 3594-3598, December, 2017.

[24] Y. Wang, D. Li, Y. Lu, Z. Cheng, Y. Gao, Improved Flower Pollination Algorithm Based on Mutation Strategy, *2017 9th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, Hangzhou, China, 2017, pp. 337-342.

[25] W. Zheng, H. Fu, G. Yang, Targeted Mutation: A Novel Mutation Strategy for Differential Evolution, *2015 IEEE 27th*

International Conference on Tools with Artificial Intelligence, Vietri sul Mare, Italy, 2015, pp. 286-293.

Biographies

Haoyu Wang, born in 1995, M.S. candidate. His research interests include Wireless Sensor Network.

Li Tan, born in 1980, Ph.D., professor. Her research interests include Wireless Sensor Network, Intelligent information network and machine learning.

Jiaqi Shi, born in 1996, M.S. candidate. Her research interests include Wireless Sensor Network.

Xinyue Lv, born in 1998, M.S. candidate. Her research interests include Wireless Sensor Network.

XiaoFeng Lian, born in 1977, Ph.D., assistant professor. His research interests include Mobile robot, machine vision, 3D reconstruction.