UAV Route Planning using Multiobjective Ant Colony System

Wang Zhenhua, Zhang Weiguo, Shi Jingping, Han Ying
School of Automation
Northwestern Polytechnical University
Xi'an, Shaanxi Province, China
wzhua1983104@sina.com

***Abstract* – In this article we present an application of the Multiobjective Ant Colony System (MACS) algorithm to the Uninhabited Aerial Vehicles (UAVs) route planning problem based on VORONOI diagram. First, we construct the objective functions: minimize the route length and danger exposure. Then the MACS algorithm concept is introduced and modified to accommodate the route planning situation. The computational results show the efficiency of this method.**

***Keywords* – MACS, Pareto front UAV route planning, VORONOI diagram**

I. Introduction

The great advantages that we can take by using Uninhabited Aerial Vehicles (UAVs) to do various military tasks have attracted many researchers' attention. The UAVs' once reconnaissance only role is now shared with strike, force protection, and signals collection, and, in doing so, have helped reduce the complexity and time lag in the sensor-to-shooter chain for acting on "actionable intelligence." Just as its name suggests, the most significant characteristic, which is also its most considerable merit, of a UAV is there is no pilot in the aircraft, so no life risks for the pilots.

As there are no pilots, the mission of UAVs should be gracefully predesigned to make sure the UAVs can complete it with less fuel consumption, less exposure to the enemies and many other constrains. This is achieved mainly by preplanning the flying paths of the UAVs. MH Overmars and P Svestka recommend a probabilistic leaning approach to construct routes [1], in which the actions of randomly choose route points alternate with that of evaluate the route constructed by part or all of the chosen route points until a satisfying solution is found; in [2] YE Yuanyuan first uses VORONOI diagram to construct all the possible routes and then uses graph clipping method to find out the routes that satisfied the constrains.

Both of these methods contain the optimization phase. For example, the second part of the VORONOI diagram method, finding the optimum or feasible route, is a work that worth more consideration. Because there could be many objectives in doing this, such as, minimum route length minimum the exposure to threats. So this is a multiobjective optimization problem. In [2], this problem is converted into singleobjective one by summing up those objective functions which has been respectively multiplied the by their weight factors, which indicate the preference of different objectives. A major drawback to this approach is that it requires a priori preference information.

In this paper, we use a multiobjective ant colony system algorithm to solve this problem, and obtained a set of Pareto-optimal solutions, which gives the decision-maker a clearer picture of the solution space and more alternatives to choose.

The rest of this work is organized as follows: the problem is described in section II. Section III presents the ACS algorithm and its multiobjective version that used in the UAV route planning problem: MACS. Section IV describes the computational results; finally, conclusions in section V.

II. Problem Presentation

In Fig. 1, the stars represent the threats, say, the enemy's radars or antiaircraft missiles. The VORONOI diagram is constructed according to these threat points [2]. The lines are the edges of the VORONOI diagram, while in our route planning scenario, they are all the "possible routs". The dots are the vertices of the VORONOI diagram, and also named "route points". What we should do now is to find out a route, along the VORONOI diagram edges, starting from our airport (represented by a pentagon in Fig. 1) to the target (represented by a rectangle in Fig. 1), that minimizes the objective function F .

The mathematics descriptions are as follows:

$Path = \{p_1, p_2, \cdots, p_n\}$: represents a feasible route that contains n route points serially;

l_{ij} : the distance between p_i and p_j ;

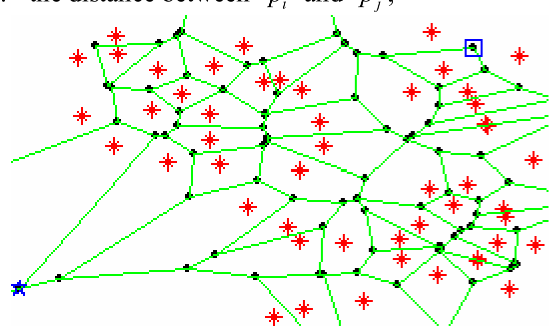

Figure 1. Routes based on NORONOI diagram

t_{ij} : the threat intensity between p_i and p_j.

The first objective is to minimize the length of the *Path* :

$$\min\ F_1 = \sum_{\substack{i=1\cdots n-1 \\ j=i+1}} l_{ij} \tag{1}$$

The second objective is to minimize the threat intensity of the *Path* :

$$\min\ F_2 = \sum_{\substack{i=1\cdots n-1 \\ j=i+1}} t_{ij}\ . \tag{2}$$

While, t_{ij} is calculated as follows:

Evenly take (sample) k_0 points on the path segment between p_i and p_j, noted as p_{ikj} $k = 1, 2, \cdots, k_0 \in N$. Where, $k_0 = \left[\dfrac{l_{ij}}{interval}\right]$, *interval* is a predefined sampling step length, and, $[\ ]$ means take the integral part of a number.

$$t_{ij} = \frac{1}{d_j^4} + \sum_{k=1}^{k_0} \frac{1}{d_{ikj}^4} \tag{3}$$

where, d_j, d_{ikj} represent the distance between p_j, p_{ikj} and the nearest threat point from them respectively.

For the multiobjective optimization problem here, the objective function F is considered as a two-dimensional vector:

$$\min\ F = [F_1\ \ F_2] \tag{4}$$

As the objectives would conflict with each other, that is, one *Path* may be excellent to one objective, but not feasible to the other. So a set of *Path*es , called Pareto-optimal solutions, is presented, and also, unlike the method used in [2], each of these objectives is considered equally important, and no prior preference information is needed.

III. MACS Algorithm

ACS was first put forward by Dorigo and Gambardella to improve the performance of the Ant System (AS) algorithm in solving the TSP problem [3].

A. The Standard ACS algorithm

The main components of ACS are illustrated as follows:

In TSP problem, the translation rule, that is, how an ant now at node r chooses a next city s to move to, of ACS is as follows:

$$s = \begin{cases} \arg \max\limits_{u \in J_k(r)} \left\{ [\tau(r,u)] \bullet [\eta(r,u)]^\beta \right\} & \text{if } q \le q_0 \\ S & \text{otherwise} \end{cases} \tag{5}$$

Where τ is the pheromone;

$$\eta = 1/\delta \tag{6}$$

is the inverse of $\delta(r,u)$, which is the distance between the city r and the city u; q is a random number uniformly distributed in $[0\ldots1]$; q_0 is a parameter $(0 \le q_0 \le 1)$; β is a parameter that determine the relative importance of distance and pheromone; and S is a random variable selected according to the probability distribution given in (7).

$$p_k(r,s) = \begin{cases} \dfrac{[\tau(r,s)] \bullet [\eta(r,s)]^\beta}{\sum\limits_{u \in J_k(r)} [\tau(r,u)] \bullet [\eta(r,u)]^\beta} & \text{if } s \in J_k(r) \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

where $J_k(r)$ is the citied remain to be visited by ant k now positioned on city r.

The global pheromone updating rule in ACS is only applied by the globally best ant. The updating rule is:

$$\tau(r,s) \leftarrow (1-\alpha) \bullet \tau(r,s) + \Delta\tau(r,s) \tag{8}$$

where

$$\Delta\tau(r,s) = \begin{cases} (L_{gb})^{-1} & \text{if } (r,s) \in \text{global-best-tour} \\ 0 & \text{otherwise} \end{cases}; \tag{9}$$

$0 < \alpha < 1$ is the pheromone decay parameter; L_{gb} is the length of the globally best tour from the beginning of the trail.

The local pheromone updating rule in ACS is applied any time an ant visits an edge while building a solution. The updating rule is:

$$\tau(r,s) \leftarrow (1-\rho) \bullet \tau(r,s) + \rho\Delta\tau(r,s) \tag{10}$$

where $0 < \rho < 1$ is a parameter; $\Delta\tau(r,s) = \tau_0$, τ_0 is the initial pheromone level, and, in [3], is experimental optimally initialed as

$$\tau_0 = (n \bullet L_{nn})^{-1}, \tag{11}$$

where, n is the number of cities and L_{nn} is tour length produced by the nearest neighbor heuristic.

For the Ant Colony Optimization (ACO) algorithm family, the most fundamental element is the use of pheromone as a guidance of the search process. The improvements and modifications of these algorithms are mainly concentrated on the modification of the translation rules and the pheromone updating rules [6] [7]. While integrate ACO algorithms with other searching algorithms, typically local searching algorithms, is another way to improve their performance [5] [8]. The MACS belongs to the former category, as will be discussed bellow.

B. The MACS algorithm

In the TSP problem, the only one objective is to minimize the length of the route. And we can make out that both the heuristic information (from (6)) and the amount of the pheromone (from (9) and (11)), which is deposited each time the pheromone updating rules (both global and local) are applied, are related to "length". It is natural to come up with the idea that we use two sets of heuristic information working together (length and threat intensity) to solve our two objective route planning problem. Much like the algorithm in [4], which is used to solve the vehicle routing problem with time windows.

The modifications of the algorithm are:

For the heuristic information, (6) is replaced by

$$\eta_{ij} = 1/\left(l_{ij} \bullet t_{ij}\right). \tag{12}$$

For the local pheromone updating rule, (11) is redescribed as

$$\tau_0 = \left(n \bullet L_{p_0} \bullet T_{p_0}\right)^{-1} \tag{13}$$

where, n is the number of route points; L_{p_0} and T_{p_0} are the tour length and the tour threat intensity produced by the nearest neighbor heuristic respectively.

For the global pheromone updating rule, (9) is replaced by

$$\Delta\tau(r,s) = \sum_{k=1}^{k_0} \Delta\tau_k(r,s) \tag{14}$$

where, k_0 is the cardinality of the Pareto-optimal set and

$$\Delta\tau_k(r,s) = \begin{cases} \left(L_k T_k\right)^{-1} & \text{if } (r,s) \in \text{Pareto-optimal rout } k \\ 0 & \text{otherwise} \end{cases} \tag{15}$$

where L_k and T_k are the tour length and the tour threat intensity produced by Pareto-optimal route k.

Furthermore, unlike the TSP problem in [3], nor the VRPTW problem in [4], the connections among the route points are confined to the edge of VORONOI diagram. So, the symbol $J_k(r)$ in (7) should be redefined as all the route points that connected with point r by a VORONOI diagram edge.

We can see, until now, that the two objectives are treated equally importantly, the relative importance of them is left to the final decision-maker to decide.

C. PSEUDOCODE OF MACS

```
/* Initialization */
    p0 ← construct a feasible solution using...
                     only heuristic information;
    Pareto-optimal Set P = {p0};
    calculate τ0 using (13);
Repeat /* Main Loop */
    for each ant i ∈ {1,2,···m}
        pi ← construct_a_route(i);
    end for
    for each pi ∈ P
        perform global pheromone updating...
                     rule according to (8);
    end for
Until a stopping criterion is met.
/* End of MACS */
Procedure  construct_a_rout(i)
/* Initialization */
 put an ant on the start route point ps;
 pi = {ps}  /* All the route points are stored in  pi  */
Repeat
    calculate the heuristic information ηij using (12);
    calculate the possibilities to each of the next ...
                     possible rout points using (7);
    generate a random number q, and use (5)...
                     the next point is chosen, say, pij;
    pi = pi ∪ pij;
Until the destination is visited.
/* Pay attention to this next post-processing step. Unlike the
algorithms in [3] and [4], where there is a taboo table in their
methods, in this algorithm loops can appear in  pi . This step is
very important in improving the performance of the algorithm.
*/
    pi ← get rid of the route loops in the route pi;
   apply the local pheromone updating rule ...
                     on pi using (10);
    calculate F1 and F2;
    if pi is nondominated in P
        save P = P ∪ pi;
        remove dominated solution from P;
    end if
/* End of the Procedure. */
```

IV. Computational Results

In this section we conducted a digital simulation of route planning based on VORONOI diagram using MACS. 90 threats are considered in this experiment. The parameters of the MACS algorithm are as follows: $\rho = 0.6$, $q_0 = 0.7$, $\alpha = 0.1$, $\beta = 0.25$, and using 10 ants.

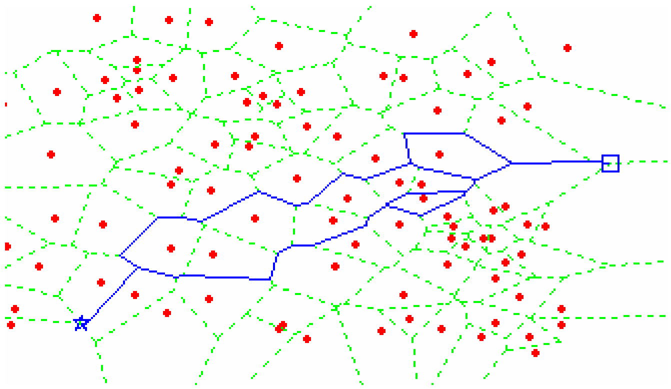

Figure2. The physical routs that are founded.

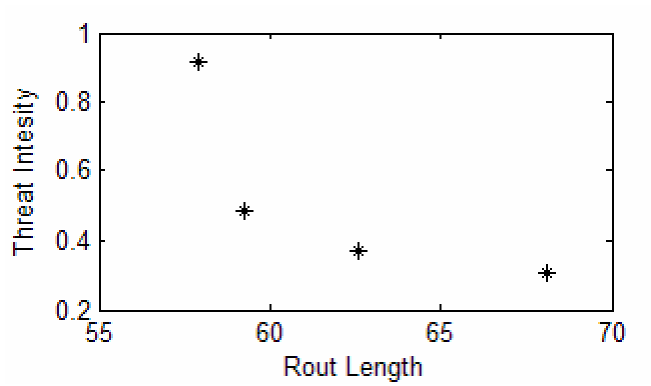

Figure 3. The pareto front.

The results are illustrated in Fig.2 and Fig.3. From the point that is represented by a pentagon to the point that is represented by a rectangle, five routes are found, which is much better than that in [2], where only one route is found between one pair of start-target points. So the final decision-maker can use his/her preferences to choose one or more routes from the five candidates.

V. Conclusion

In this paper, we first described the MACS algorithm, and then applied it to the UAV route planning problem. A set of pareto optimal solutions are established, and the results are satisfying. But there still a lot can be done to improve the algorithm. For example, how to decide the values of the parameters in section 4 would be a great problem to be discussed [9]. Moreover, the resulted routes in this paper are primary ones, other processes (such as smoothing the routs) should be added in, so that they can be accommodated to other constrains (such as the minimum turning radius of UAVs) [10].

References

[1] L. E. Kavraki, J-C Latombe, "Randomized preprocessing of configuration space for fast planning, " Proc. IEEE Conf. Roboticsand Automation, San Diego, CA, pp. 2138-2145, May 1994.

[2] YE Yuan-yuan, MIN Chun-ping, SHEN Lin-cheng, CHANG Wen-sen1, "VORONOI diagram based spatial mission planning for UAVs," Journal of System Simulation, vol. 17, No. 6, pp. 1353-1359, Jun. 2005.

[3] Marco Dorigo, Luca Maria Gambardella, "Ant colony system: a cooperative learing approach to the traveling salesman problem," IEEE Transacted, vol.1, No.1, 1997.

[4] Benjamín Barán, Matilde Schaerer, "A multiobjective ant colony system for vehicle routing problem with time window, " Proceedings of the 21st IASTED International Conference, pp.97-102, Innsbruck, Austria, February 2003.

[5] CHEN Ye, "Scaleable chaotic ant colony optimization," Computer Engineering and Applications, vol.43, No.3, pp.68-70, 2007.

[6] V. Maniezzo, M. Dorigo, A. Colorni, "The ant system applied to the quadratic assignment problem," Technical Report IRIDIA/94-28, University de Bruxelles, Belgium, 1994.

[7] Z. J. Lee, C. Y. Lee, S. F. Su, "An immunity-based ant colony optimization algorithm for solving weapon-target assignment problem," Applied Soft Computing, vol.2, pp.39-47, 2002.

[8] ZHAO Bao-jiang, LI Shi-yong, JIN Jun, "Ant colony algorithm based on adaptive selection of paths and pheromone updating," Computer Engineering and Applications, vol.43, No.3, pp.12-15, 2007.

[9] HUANG Yong-qing, LIANG Chang-yong, Zhang Xiang-de, "Parameter establishment of an ant system based on uniform design," Control and Decision, vol.21, No.1, pp.93-96, January 2006.

[10] Liu Changan, Wang Heping, Li Weiji, "Coordinated path planning of attacking unmanned aerial vehicles (UAVs)," Journal of Northwestern Polytechnical University, vol.21, No.6, pp.707-710, December