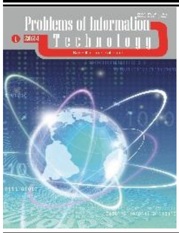


Available online at www.jpit.az

15 (1)
2024

Determination of the optimal trajectory of the movement of aircraft in areas with complex terrain under the control of the enemy

Nadir Aghayev [1,a,b], Namig Kalbiyev [a], Sabina Aghazade [a]

[a] Department of Computer systems and programming, National Aviation Academy, Mardakan Ave, 30, AZ1045 Baku, Azerbaijan

[b] Institute of Information Technology, B. Vahabzade str., 9A, AZ1141 Baku, Azerbaijan

[1] nadiraghayev@naa.edu.az

[1] https://orcid.org/0000-0001-9922-3885

ARTICLE INFO

Keywords:

Flight trajectory
Complex terrain
Digital terrain map
Evading enemy control
Ant swarm algorithm
Determining the optimal trajectory
Distance parameter
Height parameter
Smooth surface parameter

ABSTRACT

One of the main issues in the controlling of aircraft in difficult terrain during wartime is to ensure normal movement, but also to fulfill the requirements of evading enemy control. This paper proposes an improved ant swarm algorithm that makes it possible to pre-determine and optimize the trajectory of aircraft in such areas. When applying this method, a special parameter is included in the probability of choosing a movement trajectory – the height of the terrain above sea level, so that each ant does not enter territory controlled by the enemy. Using a 2D-H digital elevation map, the rectangular area under study is divided into 90 m × 90 m squares. To take into account the variability of the terrain, the heuristic function of the ant swarm algorithm takes into account the parameters of distance, height and smooth surface. Additionally, to reduce the number of iterations and computations, the ants are divided in half by number and released from the start and end points simultaneously. As a result, it allows you to choose the shortest and minimum trajectory among various calculated trajectories. To verify the effectiveness of the proposed scheme, a number of computational experiments were conducted. Experimental results on various simulated and real terrain maps show that this algorithm can be used to select an initial reference trajectory in difficult terrain.

1. Introduction

Currently, as a result of the IV industrial revolution, we are seeing the use of aircraft in various spheres of human activity. Thus, many aircraft, both manned and unmanned, have begun to increasingly penetrate our lives, like other means of transportation.

These devices are used (Kureychik et al., 2008; Aghazada et al., 2018; Aghazada et al., 2019) in military and civil affairs:

- for transportation of goods to hard-to-reach places using multiple modes of transport;
- to collect information for various purposes;
- in emergency situations and other similar operations

The main advantages of using unmanned aerial vehicles (UAVs) are their versatility, as well as low operating costs and low levels of human error (Merz et al., 2013).

One of the problems that arises when using UAVs is the issue of determining the most optimal route to avoid various natural and artificial obstacles. Various hardware and methods are used to detect obstacles: light detection, distance measurement (Carloni et al., 2013) or image processing, etc. (Dorigo, 1992). In any case, in order to avoid obstacles, you must know their location

Received 18 September 2023, Received in revised form 20 November 2023, Accepted 05 December 2023
http://doi.org/10.25045/jpit.v15.i1.04

and corresponding coordinates in advance.

The main purpose of route planning in the field of unmanned aerial vehicle research is to determine the optimal or suboptimal, safe and collision-free path from the starting point to the destination point in the presence of obstacles (Chen et al., 2010; Deepak et al., 2012). The process of planning the route of the UAV according to the history of development can be divided into:

- traditional route planning;
- intellectual route planning.

Traditional route planning algorithms include temperature simulation algorithm (Miao & Tian, 2013), algorithms created using potential function theory (Cetin et al., 2014; Nair et al., 2015), fuzzy logic algorithm (Li et al., 2013; Jiang et al., 2014; Bakdi et al., 2016), etc. However, these traditional methods cannot be improved in terms of route search efficiency and route optimization.

The intelligent path planning algorithm includes Ant Colony Optimization (ACO) (Jovanovic et al., 2016; Wang et al., 2016), Genetic Algorithm (Arantes et al., 2017, Lin et al., 2017), Neural Network (He et al., 2017, He et al., 2017a; He et al., 2017b) and Particle Swarm Algorithm (Das et al., 2016; Song et al., 2016).

The ant swarm algorithm has the advantages of high reliability, good global optimization ability, and internal parallelism. Moreover, it easily integrates with many heuristic algorithms to improve the performance of the algorithms.

The Ant Colony Optimization (ACO) algorithm was first proposed by Marco Dorigo in 1992 to solve the problem of finding an optimal path in a graph (Earl et al., 2005). The algorithm was created by simulating the feeding process of ants in nature. In search of food, ants first move randomly, and having found food, they return to their colony and mark the path they have traveled with a pheromone that has a special smell. Over time, the pheromone begins to evaporate, thereby reducing its attractiveness. Of course, pheromone evaporation varies depending on the time required to reach and return the bait. The shorter the distance, the more pheromone remains and the stronger its smell. Of course, other ants also prefer this, and as a result, the shortest path will be determined for all ants (Dorigo et al., 1997).

Currently, some metaheuristic models of ACO have been proposed. Among them, the most used are ACS (ant colony system) (Stützle et al., 2000), MMAS (Max-MİN ant system) (Schouwenaars et al., 20001), Ant Colony Optimization (ACO) and others (Lučić et al., 2003; Pham et al., 2009; Kureychik et al., 2010). Almost all of these algorithms boil down to the problem of finding the optimal path in a graph.

This feature of the ant algorithm allows it to be used to solve many practical problems – transport problems (Kochegurova et al., 2014), optimization of routes in urban transport (Levanov et al., 2010), production optimization problems and other problems.

Xiaolin Dai et al. also propose an improved ant swarm algorithm to achieve efficient search capabilities in path planning on complex terrain maps for UAVs (Agayev et al., 2023; Dai et al., 2019). The improved ant swarm algorithm uses features of Algorithm A and the MAX-MIN Ant system. So, in this work, a network environment model is first constructed. To improve the heuristic information of the ant swarm algorithm, the evaluation function and deformation prevention operator of Algorithm A are introduced, which can improve the convergence speed and smoothness of the global path. A backtracking mechanism is then applied to solve the deadlock problem. The pheromone is effectively restricted to identified unfavorable trajectories to prevent premature termination of the search. This gives ACO the ability to effectively address the challenges of tunnels, potholes and congestion encountered in challenging terrain and perform better than traditional ACO versions. The simulation results show that the improved ant colony algorithm is more efficient and faster.

However, due to the randomness of the choice of a new path by ants and the inconsistency of updating the intensity of pheromones, the traditional ant swarm algorithm can easily fall into a local pit and is prone to poor convergence. With this goal, many researchers present various improved methods for solving problems of pheromone renewal and path search strategies (Stützle et al., 2000; Zeng et al., 2016; Zhao et al., 2016; Zhang, 2017). The authors proposed the Ant Colony System (ACS) algorithm to accelerate the convergence rate of ACO by updating pheromones on the path of ants of each generation (Stützle et al., 2000). Due to the adaptive change in the volatility of the pheromone in the ACS algorithms, the search capabilities of the ant colony and the aggregation speed of the algorithm were improved (Zhao et al., 2016). Some "smart algorithms" were proposed to determine the initial trajectory, which can be converted into the initial distribution of pheromones, in order to avoid the random search of an ant colony. Information about the path (for example, the density and weight of the path) is added to the initial matrix of pheromones, which can affect the efficiency of the algorithm (Zeng et al., 2016). The heuristic function was corrected to improve the speed of convergence of the algorithm to the target point (Stützle et al., 2000). The authors introduced the concept of an unlimited step distance to find the optimal path, so that the improved ACO could find a shorter path and converge better. In addition, many researchers also combined the ant swarm algorithm with other (intelligent) algorithms (He et al., 2016a; Liu et al., 2016; Yen et al., 2018; He et al., 2017c). It is also proposed to use the artificial potential field method to improve the ability of the ant colony algorithm to detect obstacles (Liu et al., 2016). To minimize the error of iterative training, a method of fuzzy optimization of an ant colony was proposed (Yen et al., 2018).

A hybrid scheme using mutual learning and adaptive ant swarm algorithm optimization (MuL-ACO) was proposed in the study (Cheng et al., 2010) aimed at determining and optimizing the trajectory of UAVs in complex terrains. A 2D-H map is presented in that paper to describe a non-uniform environment with various obstacles. Then, an adaptive ant swarm algorithm is proposed to determine the movement trajectories of UAs before flight. Based on the "temperature reduction function" of the simulated algorithm, the volatility coefficient of the pheromone is adjusted to adaptively accelerate the algorithm change. Additionally, distance, height factors, and smoothness factors are taken into account in the algorithm to adapt to uneven environments. A mutual learning algorithm is designed to further smooth and shorten the initial trajectories. In that work, the sequence of different trajectory nodes learn from each other to obtain the shortest trajectory sequence to optimize the trajectory. The experimental results of the authors' proposed scheme show that by applying MuL-ACO, the trajectory can be determined without colliding with obstacles with high comprehensive quality in uneven environments.

If the obstacles and route can be obtained prior to flight, a more convenient and relatively safe route can be determined using flight information recorded in the aircraft's on-board recorder. Thanks to this, you will be able to avoid existing permanent obstacles and conduct your flight as secretly as possible.

They are also known to use the protective properties of the terrain to ensure stealth flight. A feature of this type of flight is to ensure the movement of the aircraft in a horizontal plane at low altitudes by choosing an optimally specified trajectory. In this case, the initial trajectory of the aircraft must correspond to the coordinates with the lowest absolute terrain heights in the horizontal plane. The proposed method for calculating the optimal reference trajectory takes into account various restrictions on the trajectory and maneuverability of the aircraft, for example, bypassing a limited area of the earth's surface, the accuracy of on-board sensors, speed, etc.

Thus, the proposed problem comes down to the problem of choosing the optimal route between points A and B, the coordinates of which are known, which ensures the flight of the aircraft at the shortest distance, using the protective features of the terrain.

This type of question can be applied to military and similar special forces organizations. Considering the above, the question can be formulated as follows.

Finding the shortest desired flight route for an aircraft from a given point to another point by passing the lowest elevations of the terrain based on a digital map of the area.

2. Solving method

In the absence of obstacles, a suitable trajectory for the problem is a straight line segment connecting the start and target points. In this case, without taking into account the specific characteristics of the aircraft, the route can be found provided that the trajectory of movement is least different from a straight line by avoiding obstacles crossed by a straight line (Agayev et al., 2023). However, since the obstacles are randomly distributed, in special cases (for example, when they are located close to each other), the solution to the problem may not be optimal. In this case, in order to take into account the relative position of all obstacles, this work provided a solution to the problem by using the ant algorithm.

To solve the problem, a rectangular area is marked on the map containing points A and B and an obstacle. The rectangle is chosen so that segment AB is the midline along its length. Starting from point A, the rectangle is divided into M×N squares and a matrix L(M×N) is constructed, consisting of the coordinates of the centers of these squares.

It is assumed that each ant can travel from one center to another in one time (this can be any known fixed time). In other words, a rectangular area is divided into parts depending on the distance that the ant can travel per unit time. The ants are divided in half according to the number of K and one half is placed at the starting point (point A), and the other half at the end point (point B). Obstacles are areas of terrain of a certain height or territory controlled by the enemy. In both cases, we assume that it is possible to form a tabular (forbidden) set-TL(i,j), consisting of the coordinates of squares and linear boundaries of the location of obstacles on the map.

As already mentioned above, in the absence of obstacles, the shortest distance will be straight AB. During the application of the algorithm, deviations from the straight line connecting the points $L_{0,N/2}$ and $L_{M,N/2}$ are noted, and as a result, a broken line $[L_{i0,p}L_{imax,p}]$ is obtained. A matrix $L(M \times N)$consisting of the coordinates of the center of each (i,j) square and a matrix $H_{i,j}$ containing the corresponding heights are built on the basis of the digital map of the area. The distance from any point (i,j) of the horizontal plane to the projection of this straight line is called the smoothness parameter and denoted by $d_{i,j}$. In addition, the distance from this point to the target point will be called target desire and will be denoted by $z_{i,j}$. Within the mentioned marks, the desire of the ant to reach the goal or the attractiveness of the path of movement will be shown

$$\mu_{i,j} = \frac{1}{d_{i,j}} + \frac{1}{z_{i,j}} + \frac{1}{H_{i,j}} \quad (1)$$

At the beginning of the algorithm, K is the number of ants participating in the process, and T is the maximum number of iterations. State $j = (1, T)$ of each i-th ant $i = (1, K)$ at the j-th iteration

$$S_i(j) = \{X_{ik,jk}^{i,j}, \tau_{ik,jk}^{i,j}, \theta_{ik,jk}, ik = 1, N, jk = 1, M\}.$$

It will show with three parameters:

The first parameter $X_{ik,jk}^{i,j}$ indicates the number of ants that have entered the square, the second parameter $\tau_{ik,jk}^{i,j}$ indicates the current amount of pheromone in this square, and the third parameter $\theta_{ik,jk}$ indicates whether access to the square is allowed or prohibited:

$$\theta_{ik,jk} = \begin{cases} 1 & ik \neq 0 \cup M, jk \neq 0 \cup N \ (ik,jk) \in L(N \times M) \backslash TL(i,j) \\ 0 & ik \neq 0 \cup M, jk \neq 0 \cup N \ (ik,jk) \in TL(i,j) \\ \infty & ik = 0 \cup M, jk = 0 \cup N \end{cases} \quad (2)$$

It should be noted that $\theta_{ik,jk}$ does not change when the algorithm works. In the beginning, all cells, except $(0, N/2)$ and $(M, N/2)$, are empty

$$X_{i1,j1}^{i,1} = \begin{cases} 1 \ i1 = 0 \vee i1 = M, j1 = N/2 \\ 0 \ i1 \neq 0 \vee i1 = M, j1 \neq N/2 \end{cases} \quad i = 1, K \quad (3)$$

Number of ants in square $(i1, j1)$ at each j-th iteration

$$N_{i1,j1,j} = \begin{cases} \sum X_{i1,j1}^{i,j} + 1 \ if \ (i1,j1) & \text{is selected} \\ \sum X_{i1,j1}^{i,j} \ if \ (i1,j1) & \text{is not selected} \end{cases} \quad (4)$$

There can be no more than K ants in one square at a time.

$$N_{i1,j1,j} \leq K \quad (5)$$

Here, the summation is performed by the ants that chose the square $(i1, j1)$ at the j-th iteration. On the other hand, it is impossible to get one and the same situation in two consecutive iterations:

$$arg(X_{ik,jk}^{i,j}) \neq arg(X_{ik,jk}^{i,j+1}) \quad (6)$$

If the ith ant leaves the cell at the jth iteration, it will not be able to enter it at the $j + 1$ -th iteration. The initial amount of pheromone in all squares is equal to 0. Thus, for each i-th ant in the initial case.

$$S_i(1) = \{X_{i1,j1}^{i,1}, 0, \theta_{ik,jk} \ i = (1, K)\} \quad (7)$$

The probability of moving from square (ik, jk) to one of the squares (it can be 3, 5, etc.) directed to the target point B in the i-th iteration (and the ants released from point B are directed to point A) is calculated as

$$p_{ik,jk}^{l,k,i} = \frac{\alpha_{ik,jk}\tau_{l,k}^{i} + (1-\alpha_{ik,jk})\mu_{l,k}}{\sum_{(l,k) \notin TL(i,j)} \alpha_{ik,jk}\tau_{l,k}^{i} + (1-\alpha_{ik,jk})\mu_{l,k}} \theta_{l,k}. \quad (8)$$

In the initial case, it is possible to accept $p_{ik,jk}^{l,k,0} = 1/_5$. This means that out of 5 adjacent squares in the direction of point B $(i, j-1), (i, j+1), (i+1, j-1), (i+1, j), (i+1, j+1)$ is chosen randomly. According to formula (6), at step$(i-1)$ there is no chance to return to neighboring cells in order to reach the goal faster. Getting a probability of 0 or ∞ means that the square is not selected.

The parameter $\alpha_{ik,jk} = \frac{1}{\sum |[L_{i0,j0}L_{ik,jk}]|}$ is calculated as the inverse value of the length of the path from the origin of the coordinates to the square(ik, jk). Formula (1) regulates the movement of ants along the line AB.

Using formula (8), the ant calculates the probabilities of the remaining squares around each (ik, jk) square, taking into account condition (1) and selects the largest of them. Note that in formula (8) when moving from the square(ik, jk) to other squares under the accepted conditions, the highest probability value $p_{ik,jk}^{l,k,i}$ is obtained at the highest value $\mu_{l,k}$. According to formula (1), this value is obtained at the lowest values of all three parameters $d_{i,j}, z_{i,j}, H_{i,j}$.

The pheromone changes when each i-th ant falls into the (ik, jk) square.

The change of pheromone occurs when every i-th ant enters the square.

$$\tau^{i}_{ik,jk}(t+1) = (1-\alpha)\left(\tau^{i}_{ik,jk}(t) + \sum_{t1=1}^{t} \sum_{lk=1}^{X^{i,j}_{i1,j1}} \tau^{l,k}_{ik,jk}(t1)\right) \quad (9)$$

$$\alpha = \frac{\alpha_{ik,jk}}{\sum_{(l,k)\notin TL(i,j)} \alpha_{l,k}} \quad (10)$$

Here, α- is the evaporation rate of the pheromone.

Algorithm

Beginning

{

Input:

A: *Source station;*

B: *Target station;*

DRM: *Digital relief map* $(\boldsymbol{N} \times \boldsymbol{M})$ *size ;*

K: *Number of ants;*

T *s the number of iterations.*

Output:

Desired Path: *The trajectory along which the aircraft can move;*

{

A rectangle is drawn on the map with vertices A and B,

$\boldsymbol{L}(\boldsymbol{N} \times \boldsymbol{M})$ set of obstacles is constructed,

$\boldsymbol{TL}(\boldsymbol{i}, \boldsymbol{j})$ is a set of obstacles and a straight line AB.

The number of ants $\boldsymbol{K}$ and the maximum number of iterations $\boldsymbol{T}$ are set.

t=0

Half of the ants are placed in square **A** and the other half in square **B.**

The initial state is initialized by formula (1)-(3) and (7).

Repeat

{

t=t+1

for each $i = 1,2, K$ ants

{

in the initial state $(i0, j0)$, the length of the route is taken $l_i = 0$

repeat

{

Using formula (8), the probabilities for the squares around each (ik, jk) square are calculated taking into account conditions (1), (2) and (9)-(10) and the maximum of them is selected.

The coordinates of the selected square are added to the route.

The route length is calculated $l_i(t) = l_i(t) + 1$.

The pheromone is replaced by formulas (9) and (10).

}

until the selected square is not a border square

If ants $L_{0,N/2}$ or $L_{imax,N/2}$ it is squared

{

the ant is returned to the starting position and the length of the route is memorized

}

If ants is on the boundary

{

the ant is returned to the starting position and the route length is reset.

}

}

The shortest route from the saved routes is displayed

}

until number of iterations $t \leq T$

}

3. Experimental results

On the basis of the algorithm mentioned above, a program was compiled in the MATLAB MathWorks environment and reports were created for various variants. In the first iteration of the algorithm, the ants are chosen arbitrarily, and in the following iterations, taking into account the pheromone sensitivity of the ants, they are forced to follow the routes of the previous iteration. Based on this approach, the results obtained when the parameters were changed were analyzed. Experiments were conducted based on real digital relief maps. For this, sections were selected on the map, where the terrain is more complex. The allocated territory is divided into squares measuring 90 m x 90 m. When forming the table, the heights of a certain height were noted. The baseline is displayed as an orange line on all images. Various variants were considered in the calculations.

3.1. Variant I

In this variant, flight trajectories are determined only taking into account the influence of altitude.

In this version, the number of squares was taken as (50)×(40). The number of ants is 50, the maximum number of iterations is 100. All ants are released from point A. The choice of squares was limited to five squares of its surroundings, directed only towards the target point. The results are shown in Fig. 1.

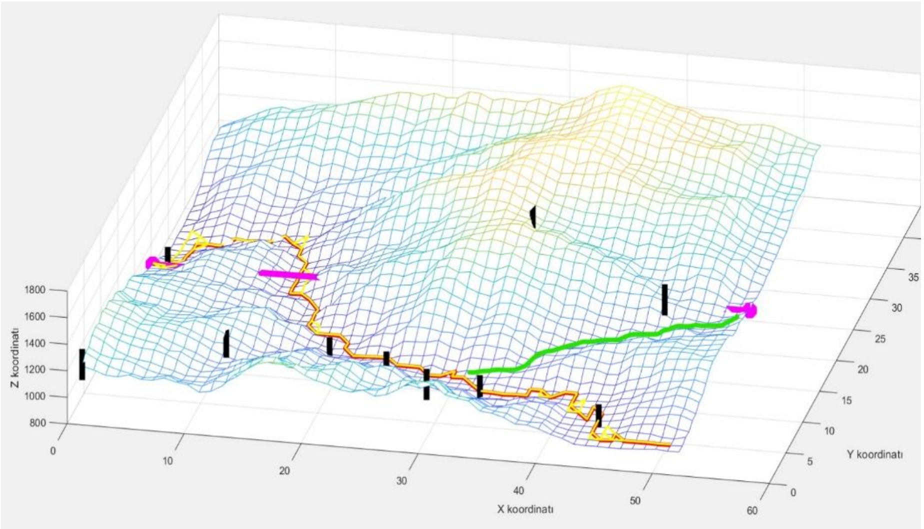


Fig. 1. Result of reports performed according to variant I

The most repeated trajectory among the possible trajectories is shown in the figure. Since the smoothness parameter and target aspiration are not taken into account during the calculation, the calculated routes are determined only at the lowest points of the terrain. At the same time, the lack of the route is not felt until a certain number of squares (in our case, up to the 30th square along the X coordinate), only after the 30-th square does the trajectory leave the target point. It should be noted that the ants, moving through the cells located above the base line, mostly remained unused (go faste) at the border point in the 20-th square along the X axis.

Note that the trajectory highlighted in green in the figure shows the case where the ants are released in two directions - from the starting point and the target point, and this will be explained in the following variants (see variant V).

3.2. *Variant II*

In this variant, the parameter values in the calculations are preserved, as in variant I, when moving forward, only three squares are taken. The purpose of carrying out the calculation variant is to determine the reason for not achieving the target point in the previous version. As a result of numerous calculations, it was found that the result can be obtained by weakening the conditions imposed on the parameter, in other words, by reducing the height limit (Fig. 2).

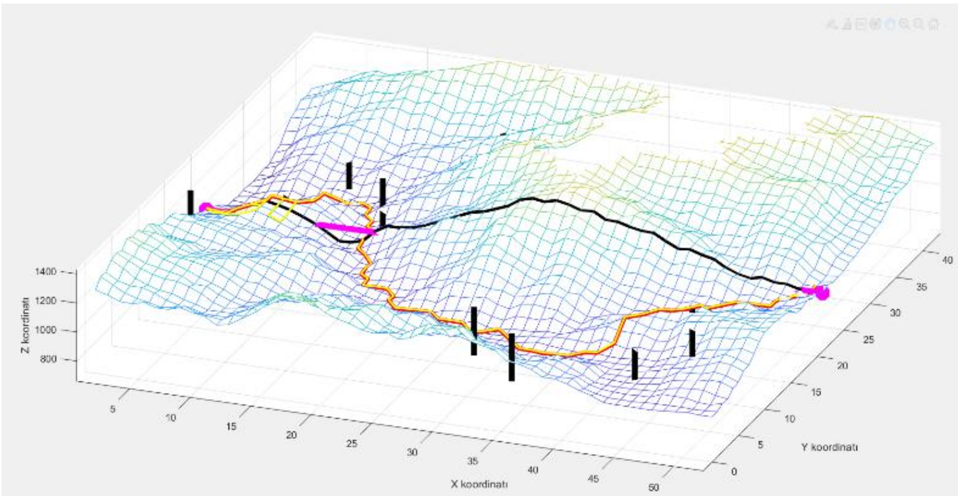


Fig. 2. Result of reports performed according to variant II

Note that when the height constraint was completely removed (meaning there were no obstacles), the ants reached the target point. (indicated by a black line). Although this is very close to the baseline, it is not a safe trajectory. When the height constraint is relatively relaxed, e.g. If in the previous version the obstacle was taken to be areas with a height of 500 m above sea level, then in this version taking the limit of 700 m or 800 m (orange trajectory) you can get different trajectories.

3.3. Variant III

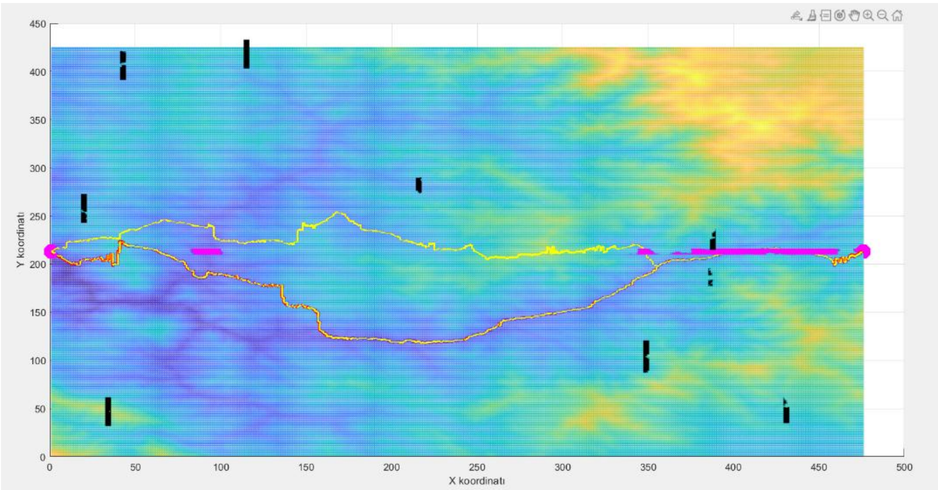
In this variant, the influence of terrain size on search results was studied. Firstly, the area has decreased.

It is known that height restrictions on small areas sometimes do not give the desired solution. This is due to the fact that in the selected area there are no areas that satisfy the constraint condition, or such areas are located far from each other.

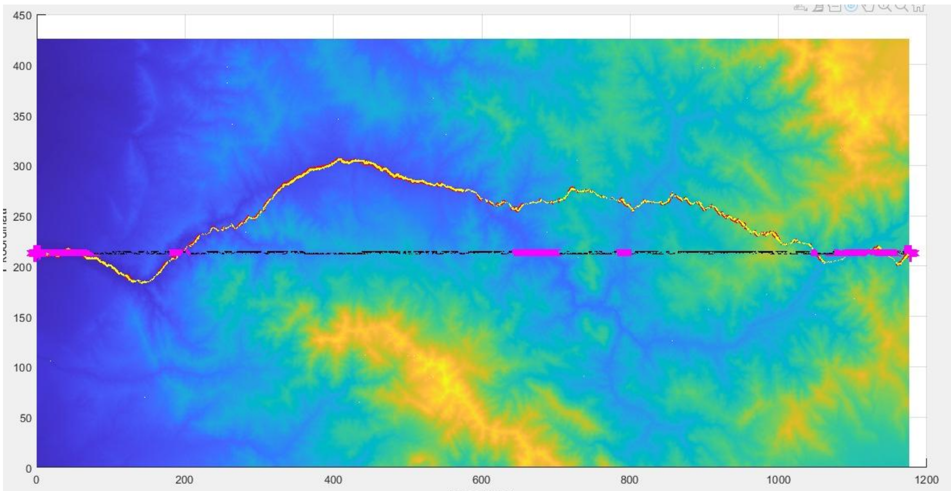
In other cases, the smallness of the territory leads to the fact that the border is not used (moves quickly) at the border point. Increasing the size of the area increases the choice.

By increasing the area size by (470)×(423) squares, it was possible to reach the target point with the parameter values adopted in variant I (Fig. 3a). Different results were obtained even when the height restrictions in this variant were taken as in variant I (Fig. 3a).

With an increase in the terrain size (1276)×(423), it was possible to determine the desired trajectory even in the presence of a high-mountain massif in the territory (yellow spotted areas, Fig. 3b).



(a)



(b)

Fig. 3. Result of reports performed according to variant III

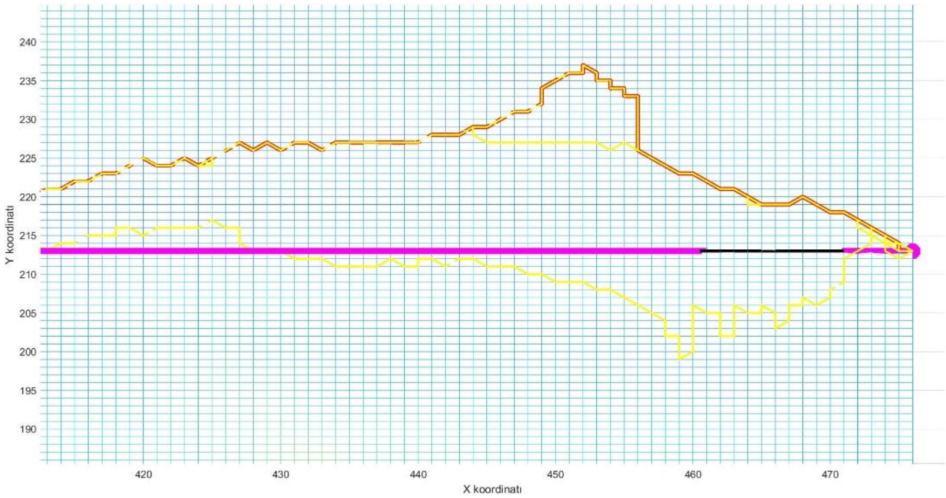


Fig. 4. Result of reports performed according to variant IV

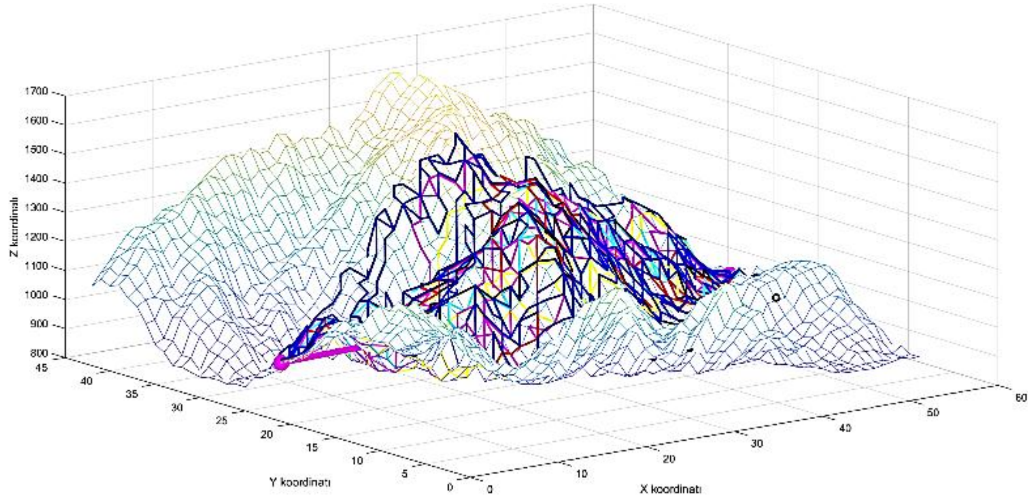


Fig. 5. Selecting the desired route for a (50)*(50) sized area considering the height, smoothness parameter and target desire

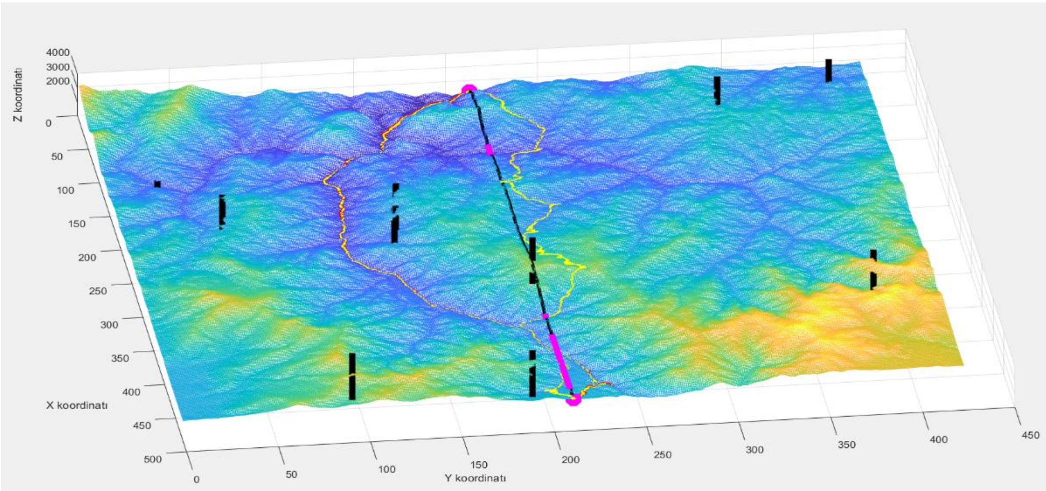


Fig. 6. Selecting the desired route for a (450) * (423 sized area considering the height, smoothness parameter and target desire

3.4. Variant IV

In this variant, the selection restrictions imposed on neighboring cells are removed, that is, when choosing the next cell, not only forward movement is taken into account, but also movement in other directions. Throughout all iterations of the algorithm, the ants continue to move mainly due to changes in height restrictions.

Cases of reaching the target point were observed despite the small size of the area. Based on this approach, the results obtained by changing the parameters are shown in several versions in Fig. 4. As can be seen in the picture, when there is no forward movement restriction, there are more possibilities than in the presence of this restriction. The effect of the altitude limit remains the same as in previous variants. When the constraint is completely removed (shown by the black line), the trajectory is close to a straight line, and as the constraint increases, the calculated trajectory will deviate more and more from this line.

3.5. Variant V

All previous variants take into account the restrictions imposed by altitude above sea level. From numerous calculation variants it is known that a positive solution to the problem directly depends on the relief parameters. In this variant, in addition to height, the smoothness parameter and target aspiration are taken into account, as in formula (1).

To ensure that optimality conditions are met, the number of ants is halved, labeled as target ants and source ants, and launched oppositely from the source and target points, respectively. At each iteration, the trajectories of each ant species are combined into one trajectory, noting when the target and source ants meet, and the process continues. The shortest distances along the obtained general trajectories are marked. Calculations were carried out for areas with the number of squares (50) * (50) (Fig. 5) and (450) * (423) (Fig. 6). Thus, at each subsequent iteration, route trajectories are determined in compliance with this principle. Finally, among the mentioned trajectories, the required routes are selected according to the minimum length and height (Fig. 6).

4. Conclusions

An improved ant swarm algorithm is proposed, aimed at determining the desired UAV trajectory in difficult terrain. In difficult terrain conditions, a new type of terrain condition function is proposed that includes height, as well as smoothness and target desired value parameters.

The analysis and experiment results show that the improved ant swarm algorithm can effectively solve the global problem of optimizing the trajectory of aircraft in complex terrain. In this article, multivariate computational experiments were carried out to determine and optimize the desired flight path of an aircraft in difficult terrain conditions.

Experiments conducted in complex environments of varying terrain and size, simulated by 2D-H terrain maps, have shown that the trajectory planned by the algorithm produces different results in terms of smoothness, elevation change, length, and target aspiration parameters. This algorithm design is more adaptable and allows for better trajectory planning, especially on more complex and larger maps.

It has been determined that taking into account the features of the relief cannot solve the problem in small areas. Relaxing the height requirements or increasing the physical size of the area may result in finding the desired path between source and target, but in practical situations relaxing these requirements is not always possible.

In difficult terrain conditions, by including the parameters of height, ride and target aspiration, the desired trajectory can be determined, which allows a positive solution to the problem, but in most cases it was necessary to stop the iterations before reaching the target point. To achieve a normal solution to the problem under any terrain conditions, it was proposed to divide the ants into two parts and release them from the source and target points. Analysis of the results of the solutions obtained showed that, despite the lack of a solution in cases where the ants were released only from the source, it was possible to determine the desired trajectory when released from two points.

References

Agayev, N. B., Orudzhov, G. G., & Kalbiyev, N. N. (2023). Planirovaniye optimal'noy obespechivayushchey trayektorii poleta samoleta s ispol'zovaniyem karty mestnosti. //Mekhatronika, avtomatizatsiya, upravleniye, 24(9), 496-502. https://doi.org/10.17587/mau.24.496-502 (in Russian)

Aghazada, S. & Ciylan, B. (2018). Determining the Optimal Route of Drones in Restricted Areas. Materials İnternational Scientific Conference for the Information Systems and Technologies Achievements and Perspectives (pp. 258-260). (in Azerbijani)

Aghazada, S. & Ciylan, B. (2019). Engel Olarak Belirlenen Araziler Olduğunda İnsansız Hava Araçlarının Karınca Kolonisi Algoritması ile Optimal Rotanın Bulunması. İCEST'19 I Uluslararası Bilimsel-Uygulamalı Konferansı (pp. 220-225). (in Turkish)

Arantes, J. D., Arantes, M. D., Toledo, C. F., Júnior, O. T., & Williams, B. C. (2017). Heuristic and genetic algorithm approaches for UAV path planning under critical situation. International Journal on Artificial Intelligence Tools, 26, 1–30. https://doi.org/10.1142/S0218213017600089

Bakdi, A., Hentout, A., Boutami, H., Maoudj, A., Hachour, O., & Bouzouia, B. (2017). Optimal path planning and execution for mobile robots using genetic algorithm and adaptive fuzzy-logic control. Robotics and Autonomous Systems, 89, 95-109. https://doi.org/10.1016/j.robot.2016.12.008

Carloni, R., Lippiello, V., D'auria, M., Fumagalli, M., Mersha, A. Y., Stramigioli, S., & Siciliano, B. (2013). Robot vision: Obstacle-avoidance techniques for unmanned aerial vehicles. IEEE Robotics & Automation Magazine, 20(4), 22-31. https://doi.org/10.1109/MRA.2013.2283632

Cetin, O. & Yilmaz, G. (2014). Sigmoid limiting functions and potential field based autonomous air refueling path planning for UAVs. Journal of Intelligent & Robotic Systems, 73, 797–810. https://doi.org/10.1007/s10846-013-9902-y

Cheng, C. T., Fallahi, K., Leung, H., & Tse, C. K. (2010). An AUVs path planner using genetic algorithms with a deterministic crossover operator. 2010 IEEE International Conference on Robotics and Automation, Anchorage, USA, May 2010 (pp. 2995-3000). https://doi.org/10.1109/ROBOT.2010.5509335

Dai, X., Long, S., Zhang, Z., & Gong, D. (2019). Mobile Robot Path Planning Based on Ant Colony Algorithm With A* Heuristic Method. Frontiers in Neurorobotics, 13, 444376. https://doi.org/10.3389/fnbot.2019.00015

Das, P., Behera, H., & Panigrahi, B. (2016). A hybridization of an improved particle swarm optimization and gravitational search algorithm for multi-robot path planning. Swarm and Evolutionary Computation, 28, 14-28. https://doi.org/10.1016/j.swevo.2015.10.011

Deepak, B. B. V. L., Parhi, D. R., & Kundu, S. (2012). Innate immune based path planner of an autonomous mobile robot. Procedia Engineering, 38, 2663-2671. https://doi.org/10.1016/j.proeng.2012.06.313

Dorigo, M. & Gambardella, L.M. (1997). Ant colony system: A cooperative learning approach to the traveling salesman problem. IEEE Transactions on Evolutionary Computation, 1(1), 53-66. https://doi.org/10.1109/4235.585892

Dorigo, M. (1992). Optimization, Learning and Natural Algorithms, PhD thesis, Politecnico di Milano, Italie.

Earl, M. G. & D'Andrea, R. (2005). Iterative MILP methods for vehicle-control problems. IEEE Transactions on Robotics, 21(6), 1158-1167. https://doi.org/10.1109/TRO.2005.853499

Fahmi, H., Zarlis, M. Nababan, E., & Sihombing, P. (2019). Ant Colony Optimization (ACO) algorithm for determining the nearest route search in distribution of light food production. 4th International Conference on Computing and Applied Informatics 2019 (ICCAI 2019), Medan, Indonesia, November 2019 (pp. 1-7). https://doi.org/10.1088/1742-6596/1566/1/012045

He, W. & Zhang, S. (2017c). Control design for nonlinear flexible wings of a robotic aircraft. IEEE Transactions on Control Systems Technology, 25(1), 351–357. https://doi.org/10.1109/TCST.2016.2536708

He, W., Chen, Y., & Yin, Z. (2016). Adaptive neural network control of an uncertain robot with full-state constraints. EEE Transactions on Cybernetics, 46(3), 620–629. https://doi.org/10.1109/TCYB.2015.2411285

He, W., Dong, Y., & Sun, C. (2016a). Adaptive neural impedance control of a robotic manipulator with input saturation. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 46(3), 334–344. https://doi.org/10.1109/TSMC.2015.2429555

He, W., Yan, Z., Sun, C., & Chen, Y. (2017a). Adaptive neural network control of a flapping wing micro aerial vehicle with disturbance observer. IEEE Transactions on Cybernetics, 47(10), 3452–3465. https://doi.org/10.1109/TCYB.2017.2720801

He, W., Yin, Z., & Sun, C. (2017b). Adaptive neural network control of a marine vessel with constraints using the asymmetric barrier lyapunov function. IEEE Transactions on Cybernetics, 47, 1641–1651. https://doi.org/10.1109/TCYB.2016.2554621

Jiang, K., & Li, C. G. (2014). Path Planning Based on Fuzzy Logic Algorithm for Robots in Hierarchical Control. Applied Mechanics and Materials, 644–650, 701–704. https://doi.org/10.4028/www.scientific.net/amm.644-650.701

Jovanovic, R., Tuba, M., & Voß, S. (2016). An ant colony optimization algorithm for partitioning graphs with supply and demand. Applied Soft Computing, 41, 317-330. https://doi.org/10.1016/j.asoc.2016.01.013

Kochegurova, Ye. A., Martynov, YU. A., Martynova, YU. A., & Tsapko, S. G. (2014). Algoritm murav'inykh koloniy dlya zadachi proyektirovaniya ratsional'nykh marshrutnykh setey gorodskogo passazhirskogo transporta. Vestnik SibGUTI, 3, 89–100. (in Russian)

Kureychik, V. M. & Kazharov, A. A. (2008). O nekotorykh modifikatsiyakh murav'inogo algoritma. Izvestiya YUFU. Tekhnicheskiye nauki, 4(81), 7–12. (in Russian)

Kureychik, V. M. & Kazharov, A. A. (2010). Murav'inyye algoritmy resheniya transportnyye zadachi. Izvestiya RAN. Teoriya i sistemy upravleniya, 1, 32–45. (in Russian)

Levanov, T. V. & Usko, O. V. (2010). Algoritm raboty murav'inoy kolonii dlya odnoy zadachi razmeshcheniya s ogranicheniyami po proizvodstvennoy moshchnosti. Vestnik UGATU, 2(37), 202–208. (in Russian)

Li, Q., Zhang, C., Han, C., Xu, Y., Yin, Y., & Zhang, W. (2013). Path planning based on fuzzy logic algorithm for mobile robots in static environment, 2013 25th Chinese Control and Decision Conference (CCDC), Guiyang, China, May 2013 (pp. 2866-2871). https://doi.org/10.1109/CCDC.2013.6561434

Li, X., Li, Q., & Zhang, J. (2022). Research on global path planning of unmanned vehicles based on improved ant colony algorithm in the complex road environment. Measurement and Control, 55, 9-10. https://doi.org/10.1177/00202940221118132

Lin, D., Shen, B., Liu, Y., Alsaadi, F. E., Alsaedi, A., & Cheng, H. (2017). Genetic algorithm-based compliant robot path planning: an improved Bi-RRT-based initialization method. Assembly Automation, 37(3), 261–270. https://doi.org/10.1108/AA-12-2016-173

Liu, J., Yang, J., Liu, H., Tian, X., & Gao, M. (2016). An improved ant colony algorithm for robot path planning. Soft Computing, 1, 1–11. https://doi.org/10.1007/s00500-016-2161-7

Lučić, P. & Teodorović, D. (2003). Computing with bees: attacking complex transportation engineering problems. International Journal on Artificial Intelligence Tools, 12(3), 375–394. https://doi.org/10.1142/S0218213003001289

Merz, T. & Kendoul, F. (2013). Dependable low-altitude obstacle avoidance for robotic helicopters operating in rural areas. Journal of Field Robotics, 30(3), 439-471. https://doi.org/10.1002/rob.21455

Miao, H. & Tian, Y. (2013). Dynamic robot path planning using an enhanced simulated annealing approach. Applied Mathematics and Computation, 222, 420-437. https://doi.org/10.1016/j.amc.2013.07.022

Nair, R. R., Behera, L., Kumar, V., & Jamshidi, M. M. (2015). Multisatellite formation control for remote sensing applications using artificial potential field and adaptive fuzzy sliding mode control. IEEE Systems Journal, 9(2), 508–518. https://doi.org/10.1109/JSYST.2014.2335442

Pham, D. T., & Castellani, M. (2009). The Bees Algorithm: Modelling foraging behaviour to solve continuous optimization problems. Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, 223(12), 2919-2938. https://doi.org/10.1243/09544062JMES1494

Schouwenaars, T., De Moor, B., Feron, E., & How, J. (2001). Mixed integer programming for multivehicle path planning. 2001 European control conference (ECC), Porto, Portugal, September 2001 (pp. 2603–2608). https://doi.org/10.23919/ECC.2001.7076321

Song, B., Wang, Z., & Zou, L. (2016). On global smooth path planning for mobile robots using a novel multimodal delayed PSO algorithm. Cognitive Computation, 9, 5–17. https://doi.org/10.1007/s12559-016-9442-4

Stützle, T. & Hoos, H. H. (2000). MAX MIN Ant System. Future Generation Computer Systems, 16(8), 889-914. https://doi.org/10.1016/S0167-739X(00)00043-1

Wang, P., Lin, H., & Wang, T. (2015). An improved ant colony system algorithm for solving the IP traceback problem. Information Sciences, 326, 172-187. https://doi.org/10.1016/j.ins.2015.07.006

Yen, C. T. & Cheng, M. F. (2018). A study of fuzzy control with ant colony algorithm used in mobile robot for shortest path planning and obstacle avoidance. Microsystem Technologies, 24, 125-135. https://doi.org/10.1007/s00542-016-3192-9

Zeng, M., Xi, L., & Xiao, A. (2016). The free step length ant colony algorithm in mobile robot path planning. Advanced Robotics, 30(23), 1509–1514. https://doi.org/10.1080/01691864.2016.1240627

Zhang, W., Gong, X., Han, G., & Zhao, Y. (2017). An improved ant colony algorithm for path planning in one scenic area with many spots. IEEE Access, 5, 13260–13269. https://doi.org/10.1109/ACCESS.2017.2723892

Zhao, J., Cheng, D., & Hao, C. (2016). An improved ant colony algorithm for solving the path planning problem of the omnidirectional mobile vehicle. Mathematical Problems in Engineering, 2016, 7672839, 1–10. https://doi.org/10.1155/2016/7672839

Zhou, Z., Nie, Y., & Min, G. (2013). Enhanced ant colony optimization algorithm for global path planning of mobile robots. 2013 International Conference on Computational and Information Sciences, Shiyang, China, June 2013 (pp. 698–701). https://doi.org/10.1109/ICCIS.2013.189