Application of the ACO algorithm for UAV path planning

Abstract. The ACO (Ant Colony Optimization) algorithm is a bio-inspired metaheuristic used to optimize problems or functions described by graphs, sequences of events, or queues of tasks. It is used, among a variety of other purposes, when routing Internet network packets, determining the shortest routes between designated points (traveling salesman’s problem), for the time and cost optimization of production, or setting public transport stops. In the article, the ACO algorithm was used to autonomously construct the optimal route for an unmanned aerial vehicle (UAV). The algorithm establishes the spatial orientation of the UAV, indicating the direction of its transition for each intermediate waypoint. The results of the simulations show the trajectory of the UAV depending on the selected weighting factors, determining the priority of avoiding detected hazards or choosing the shortest path. The quality of each variant is evaluated numerically by the calculated fitness function, the value of which is the sum of the costs of the transition to each intermediate route point. The effect of the algorithm is a set of executable trajectory variants, of which the one with the best fitness value is selected.


Keywords: path planning, UAV, ACO algorithm.
Słowa kluczowe: planowanie trasy, BSP, algorytm mrówkowy.

Introduction
The ACO (Ant Colony Optimization) algorithm is a bio-inspired metaheuristic algorithm developed in 1996 by Marco Dorigo [1]. His research on modeling the collective intelligence of ant swarms began in 1992 as part of his PhD thesis [2].

The ACO algorithm imitates the behavior of ants searching for the shortest path to a food source. Each ant chooses a path for itself, however, it is guided by the amount of perceptible pheromones left by other preceding individuals. When following this path, an ant leaves a certain amount of additional pheromones, making the intensification of the trail greater for the shortest route. Pheromones evaporate during the whole process, thanks to which longer paths with fainter traces are forgotten, and ants are more likely to choose those shorter, more perceptible ones as a consequence (Figure 1). In the initial phase of the search, when the amount of traces left are still small, the ants tend to spread, creating at the same time a set of possible variants, eventually leading to the most effective solution.

Fig. 1. The principle of choosing the shortest path by ants, where: 1 – random path search, 2 – variant construction, 3 – pheromone accumulation on the best solution and a – movement direction, b – pheromone distribution, N – nest (start), F – food source (target)

ACO algorithm
In the general case, the ACO procedure [7] consists of the selection by ants of subsequent nodes from the n-element set \( N = \{ u_1, u_2, ..., u_n \} \) and \( n(t) = \{ t_1, t_2, ..., t_n \} \) with the probability specified by the formula

\[
p_j(t) = \begin{cases} \frac{[\tau_{ji}(t)][\eta_j]}{\sum_{i} [\tau_{ji}(t)][\eta_j]}, & j \in K \\
0, & j \notin K \end{cases}
\]

where \( p_j(t) \) is the probability of transition to node \( j \) by the ant \( i \) at time \( t \), which in this case will mean the iteration number of the algorithm. The symbol \( \tau_{ji} \) represents (numerically) the amount of pheromones left in the \( u_i \) node, while \( \eta_j \) is the heuristic factor described by the expression

\[
\eta_j = \frac{J_1}{J_j},
\]

where \( J_i \) is the value of the fitness function (transition cost) to node \( j \). The coefficients \( \alpha \) and \( \beta \) determine the effect of the pheromone and the heuristic coefficient respectively on the process of selecting the next node. The probability is calculated only for nodes from the set \( K \), which contains all nodes from the set \( N \), for which the transition is not prohibited in any way, thus \( K \subseteq N \).
Pheromone trail on nodes
The principle of the ACO algorithm is based on the traces of pheromones left by each ant after the solution has been constructed (determining the road variant). The value of the pheromones left in node \( j \) by all ants after the completion of a given iteration is described by the equation

\[
\tau(t+1) = \tau(t) \cdot (1 - \rho) + \sum_{i=1}^{a} \Delta \tau_i(t)
\]

where \( \Delta \tau_i(t) \) is the amount of pheromones left in node \( j \) by the ant \( i \) in the given iteration \( t \), where \( a \) is the number of all ants. This value is described by

\[
\Delta \tau_i(t) = \frac{Q}{J_{ci,t}}
\]

where \( Q \) is a constant value, determining the amount of pheromones that each ant leaves, regardless of the length and cost of its route, while \( J_{ci,t} \) is the total cost of the route covered by the ant \( i \) in iteration \( t \).

The expression \((1-\rho)\) reflects the process of pheromone evaporation. After each iteration, the value of pheromones found in each node is reduced by the value of \( \rho \) (expressed as a percentage), thanks to which weaker solutions - those burdened with higher costs - are forgotten.

Single ant model
Each ant has the ability to remember constructed variants, memorizing the way traveled step by step. Thanks to this, it is possible to reject nodes that have already been visited. This approach prevents the ant from visiting the same node, which can cause the algorithm to loop. The vector describing a single ant in a given iteration \( t \) is presented as follows

\[
ant_i(t) = \{ s_i(t), F_i(t), \}
\]

where \( s_i(t) \) is the node selected by the ant in step \( p \), \( F_i(t) \) is the cost of transition to this node, and \( s_i(t) \) is the number of steps made by the ant \( i \) in the given iteration \( t \). Every ant is associated with a traveled path saving set

\[
\text{path} = \{ u_1, u_2, ..., u_s \}
\]

Path planning as an optimization problem for ACO
The problem of path planning can be described as selecting the appropriate sequence of steps leading to reaching the destination point [10]. The process begins when the UAV locates obstacles or hazards. In this situation it is necessary to re-plan the route, considering the amount of fuel remaining (energy) and possible approach to hazardous areas. This ensures shaping the trajectory in such a way that the UAV moves away from the target point when the amount of fuel is insufficient and the level of the given threat is low enough to affect its zone.

Environment and simulation object
The simulated object is the UAV platform presented as a rigid solid with six degrees of freedom 6DoF [8,10,14]. The state vector, characterizing the state of the UAV in space, is presented in the form

\[
x(p) = [x, y, z, F_{ax}, F_{ay}, F_{az}]
\]

where \( p \) is the number of UAV steps, \( x, y, z \) are the Cartesian coordinates of the object, \( F_{ax}, F_{ay}, F_{az} \) are \( F_a \)-coordinate system coordinates indicating the direction of UAV rotation (its spatial orientation to the Earth's inertial reference system) by Euler angles [8] \( \phi, \theta, \psi \). These are the angles respectively: roll describing the rotation relative to the axis \( \theta \), pitch describing the rotation with respect to the axis \( \phi \) and yaw describing the rotation with respect to the axis \( \psi \).

The element indicating the direction of UAV movement is the transition matrix \( V(p) \) in the form

\[
V(p) = \begin{bmatrix} V_x & 0 & 0 \\ 0 & V_y & 0 \\ 0 & 0 & V_z \end{bmatrix}
\]

with

\[
V_x(p) = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad V_y(p) = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad V_z(p) = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}
\]

These vectors are permanently attached to the center of the UAV coordinate system and rotate with it, and their direction is consistent with the \( ax \) axis.

In the ACO procedure a single ant takes on the role of the UAV, so according to the formula (7), the expression (5) should be extended to

\[
ant_i(t) = \{ s_i(t), F_i(t), F_{dax}, F_{day}, F_{da\psi} \}, \quad p \in \{ 1, 2, ..., s_i(t) \}
\]

and associate an additional speed matrix with the ant

\[
V^p = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}
\]

The spatial orientation of the unmanned aerial vehicle in relation to the inertial Earth reference system is determined by the rotation of the coordinate system permanently associated with the UAV (Figure 2) during the maneuver.

Fig. 2. Coordinate system fixed to an aircraft

Coordinate system transformation is carried out by making three rotations in accordance with the Euler-Rodrigues formula [9]. Each subsequent rotation is made relative to the transformed coordinate system from the previous rotation. Initially, the \( Fa \) plane coordinate system is parallel to the inertial system and its orientation can be described by the matrix of three column versors

\[
F_{ax}(p) = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}
\]

and

\[
F_{ay}(p) = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}
\]

which are versors of the \( Fa \) coordinate system.

Rotation of the system by an angle \( \gamma \) around the axis defined by the vector \( w \) describes the resultant rotation matrix in the form

\[
w = [w_x, w_y, w_z]^T
\]

116 PRZEGŁAD ELEKTROTECHNICZNY, ISSN 0033-2097, R. 95 NR 7/2019
The limitation of the movement and maneuverability of the simulation object \( s(j) \) is accomplished by narrowing the range of changes in the angles \( \varphi, \psi, \theta \) to the value
\[
\begin{align*}
\varphi < \varphi < \varphi & < \varphi, \\
\psi < \psi < \psi & < \psi, \\
\theta < \theta < \theta & < \theta,
\end{align*}
\]
with \( \varphi_{\text{min}} = \varphi_{\text{min}} = \theta_{\text{min}} = -15 \, \text{deg} \), \( \varphi_{\text{max}} = \varphi_{\text{max}} = \theta_{\text{max}} = 15 \, \text{deg} \).

Next node selection
In the proposed solution, the ACO algorithm was used to determine the sequence of subsequent route points (nodes). However, in order to approximate the problem of route planning to the problem of UAV control, these points are not directly determined. Each ant constructing its solution does not select target coordinates, but selects a set of three angles about which a UAV should be rotated to move in the appropriate direction.

Due to the fact that ants do not choose the coordinates of the space and the angles of rotation of the object, it is not possible to take angle sets as nodes. This is because a sequence of angles in the form of nodes as \( \text{path}_{\text{ant}} = \{u^1, u^2, u^3, u^4\} \) and \( \text{path}_{\text{node}} = \{u^1, u^2, u^3, u^4\} \) will cause the same distribution of pheromones on the nodes in both cases while the actual \( \text{path}_{\text{ant}} \) and \( \text{path}_{\text{node}} \) trajectories will be different, which means a critical malfunction in the logic of the algorithm's operation. Therefore, the set of \( N \) nodes will be the coordinates in space corresponding to the points visited by the ants during the selection of the sets of angles of rotation and movement in accordance with the directions set by them in all iterations. So let
\[
D^t = \text{path}^t \cup (\text{path}^{t-1}),
\]
then \( D^t \) will mean a set of all different nodes visited in iteration \( t \) by all ants and
\[
N^t = D^t \cup \left( D^t \cup \text{path}_{\text{initial}} \right),
\]
then \( N \) will mean the set of all nodes in the iteration \( t \), but the traveled path (6) should be modified to the expression
\[
\begin{align*}
\text{path}^t = \{ u_1^1, u_1^2, ..., u_1^{s_j} \}, \\
\sigma^t = \{ \varphi^t, \psi^t, \theta^t \}, \\
p \in \{ 1, 2, ..., s_j \},
\end{align*}
\]
and also
\[
\begin{align*}
\text{path}^t &= \{ u_1^1, u_1^2, ..., u_1^{s_j} \}, \\
\sigma^t &= \{ \varphi^t, \psi^t, \theta^t \}, \\
p \in \{ 1, 2, ..., s_j \}.
\end{align*}
\]

During the process of calculating the probability of going to the node \( u_i \in N \), the value of the cost of transition \( J_j \) is determined by the fitness function with the form
\[
\begin{align*}
F_{\text{cost}}(u_i, u_j, tw + fw) &= f_{\text{cost}}(u_i) + tw \text{ threat}(u_j),
\end{align*}
\]
where \( f_{\text{cost}} \) is a fuel energy cost function for a transition from node \( u_i \) to node \( u_j \) and \( f_{\text{threat}} \) is a fuel energy cost function for threat (threat). The values of \( tw \) and \( fw \) are parameters regulating the impact of the level of threats and the length of the route on the total cost value, respectively, for the transition to the node \( u_i \).
(34) \[ \text{thread}(\mathbf{u}) = \begin{cases} \sum_{q=1}^{h} \mathbf{l}_q(r_q-d_{jq}), & d_{jq} \leq r_q, \\ 0, & d_{jq} > r_q. \end{cases} \]

where \( l_q \) is the threat level of hazard \( q \) and \( d_{jq} \) is the distance between the hazard center \( j \) from the center of \( q \), while \( h \) is the number of threat areas. This approach means that the cost of the threat will be the greater, which is the ant closer to the center of the given threat.

At each step, the transition probability will be calculated for points indicated by all combinations of angles sets. In order to reduce the computational complexity of the algorithm, all ranges of angles will be divided with a 7.5 deg resolution, then \( \phi = \psi = \theta = \{-15, -7.5, 0, 7.5, 15\} \) deg, creating a set of 125 possible combinations of angles in each step.

For each set \( \alpha_{e,f,g} = [\phi, \psi, \theta] \), where \( e \in \phi, f \in \psi, g \in \theta \) in step \( p \), locate the node \( \mathbf{u}_i \) which it indicates according to (14-24) and read the corresponding node value \( \tau \), where

(35) \[ \tau = \begin{cases} \tau_j, & \mathbf{u}_i \in N, \\ 1, & \mathbf{u}_i \notin N, \end{cases} \]

then the probability of transition to node \( \mathbf{u}_i \) should be calculated using the expression

(36) \[ p_{x,y}^{(j)}(\mathbf{u}) = \frac{(\text{edge}[\mathbf{u}])^{\rho}}{\sum_{k \in K} (\text{edge}[k])^{\rho}}, \quad j \in K, \]

and select the node and carry out the transition, saving the node to \( \text{path}^i \) and \( \text{path}_i \), and adding the newly visited nodes to the set \( N \). This scheme is repeated until the moment the ant reaches the target.

After completion of the routes by all ants in the given iteration, the pheromone values should be updated on the nodes visited by the ants in accordance with (3-4) and

(37) \[ J_{x,y} = \sum_{j=1}^{N} J_{x,y}^{(j)}. \]

Then the best ant is chosen

(38) \[ \text{ant}^i_{\min} \rightarrow \text{ant}_{\text{best}}. \]

Simulation results

The simulations were carried out with the following parameters: \( \rho = 0.8, \alpha = 1, \beta = 3, Q = 50 \), iteration limit \( itim = 180 \), starting point \( s_0 = [7 1 5] \), destination point \( s_5 = [20 20 10] \) and the initial transition vector

(39) \[ V(p_0) = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \]

and also, initial orientation

(40) \[ F_{\text{at}}(p_0) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \]

The flight direction is determined by the UAV’s \( \omega_3 \) axis, which is initially compatible with the \( \omega_1 \) axis in the MATLAB simulation environment. The transition vector has a length of 2 units, which is the length of every step in the UAV’s movement. The level of all threats has been set to 0.5. The simulations were carried out for three variants of weights:

1. \( f_{w1} = 0.2, f_{w2} = 0.8 \) (Figures 4, 7),
2. \( f_{w2} = 0.5, f_{w3} = 0.5 \) (Figures 5, 8),
3. \( f_{w3} = 0.8, f_{w3} = 0.2 \) (Figures 6, 9),

which allows for the observation of how different weights affect the ACO path building process.

Fig. 4. UAV trajectory for parameters: \( f_w = 0.2, tw = 0.8 \)

Fig. 5. UAV trajectory for parameters: \( f_w = 0.5, tw = 0.5 \)

Fig. 6. UAV trajectory for parameters: \( f_w = 0.8, tw = 0.2 \)

Fig. 7. Path cost optimization plot for parameters: \( f_w = 0.2, tw = 0.8 \)
distance from the hazard center, where according to the energy weight, while still keeping the largest possible extent than in the first attempt. This means that the base route already largely met the weight requirements of the second and third variants and did not require major modifications. The difference between the base shape and the resulting shape of the trajectory is the highest for the first variant, which results in the largest percentage change.

The correct operation of the algorithm allows for further development of the proposed solution. The next step may be the application of known algorithm modifications to improve its performance such as Ant System Rank (ASR) or Max Min Ant System (MMAS) [15].

Authors: dr inż. S. Konatowski, e-mail: skonatowski@wat.edu.pl; mgr inż. Piotr Pawłowski, e-mail: piotr_pawlowski@icloud.com, Wojskowa Akademia Techniczna, Instytut Radioelektroniki, ul. Gen. Witolda Urbanowicza 2, 00-908 Warszawa.

REFERENCES


