Optimization of decisions when planning a UAV group mission with alternative depots

Leonid Hulianytskyi [1] and Oleh Rybalchenko[1]

[1] VM Glushkov Institute of Cybernetics of the National Academy of Sciences of Ukraine, Academician Glushkov Avenue, 40, Kyiv, 03187, Ukraine

Abstract

As new technologies develop, many optimization problems arise, generated by the problems of effective mission planning of individual UAVs and their groups (teams). The paper considers the problem of optimizing decisions when planning a UAV group mission to inspect or service a given set of customers (targets) in the presence of alternative depots. A substantive formulation and mathematical model of the problem of distributing targets by bases and UAVs and optimizing their routes when performing inspection and/or servicing a given set of targets with the condition of completing the route in certain reception areas (depots) and restrictions on UAV resources as a special combinatorial optimization problem are presented. To solve this problem, a max-min algorithm of ant systems has been developed, with the step-by-step interaction of ants to form solutions, as well as a special algorithm for deterministic local search. The results of a computational experiment are presented.

Keywords

Routing problem, UAV, alternative depot, ant colony optimization, local search

1. Introduction

Recently, developments in the field of posing and solving Vehicle Routing Problem (VRP) have intensified, as they arise in many areas of activity when optimizing costs in civil and military applications. The development of information technologies, the rapid spread of online trade, and special applications in risky and critical situations caused special attention to the use of drones, primarily unmanned aerial vehicles (UAVs), when servicing or inspecting a given set of customers or objects [1,2]. An effective option for the use of UAVs is the solution of assigned tasks by a group of UAVs acting as a team, which allows to reconsider approaches to solving the problems of surveying certain objects or delivering goods to customers.

In the literature, there are various names and abbreviations of routing problem using UAVs, such as UAV routing problem (UAVRP) [1] or with the specification electric vehicle routing problem (EVRP) [3]; when using hybrid transport systems – of a flying sidekick traveling salesman problem (FSTSP) [4, 5], routing problems with drones (VRP with drones, VRPD) [2, 6, 7, 8], drone routing problems together with a truck (VRP with truck, VRP - T) [9, 10]. Such problems are an extension and development of classic VRP, having, at that time, their own specificity [1, 10, 11, 12, 13].

One important type of routing problem with UAVs is mission planning problems with multiple UAVs that may use different locations (depots) for take-off and landing. Such depots can correspond to both base locations and UAV service points. In the case of fixing the location of the depot, some of which may be the starting point, and some of which may be the finishing point, we will call them alternative, as opposed to dynamic – cases when these base locations are located on the route of some moving vehicle [10, 14, 15].

This research examines the problems of mission planning based on the optimization of routes of a group of UAVs or other mobile robotic systems acting as a team, which is faced with the task of visiting a given set of targets (clients, objects), assuming the possibility of launching and landing UAVs in alternative depots. Attention is focused on situations where the carrying capacity of the UAV is not a limiting factor.

From the point of view of practical planning of operations involving UAVs, the three key problems that should be solved in the joint planning of missions of several UAVs are the distribution of targets by depots, optimization of routes, and the selection of platforms for basing (depots), which in many approaches proposed

International Scientific Symposium «Intelligent Solutions», September 27-28, 2023, Kyiv, Ukraine
EMAIL: leonhul.icyb@gmail.com (A. 1); rv.oleg.ua@gmail.com (A. 2);
ORCID: 0000-0002-1379-4132 (A. 1); 0000-0002-5716-030X (A. 2);

in the literature give rise to separate optimization problems [8, 16]. In contrast to this, an approach is proposed that allows combining all these three problems into one combinatorial optimization problem.

Chapter 2 considers the formulation and mathematical model of the UAV group mission planning problem with cost minimization in the presence of alternative depots, Chapter 3 contains a description of algorithms for solving the optimization problem, the results of the effectiveness study of which are presented in Chapter 4. At the end, brief conclusions are presented.

2. Mathematical model of the problem

The proposed mathematical model of the mission planning problem of a group of heterogeneous UAVs with cost minimization is a development of the formulations proposed in [14, 15, 17].

The formulated problem is solved under the following assumptions.

1. Each target is visited by only one UAV and only once.
2. UAVs have limitations on the flight resource.
3. Replenishment of the flight resource of the UAV is carried out in one of the available depots, which is determined during planning.
4. It is believed that there are enough means and supplies to replenish the flight resource of the UAV (batteries, fuel).
5. It is assumed that UAV energy consumption occurs according to a linear law, that is, possible cost overruns during take-off or landing of the UAV are not considered.
6. The route of a specific UAV can consist of sub-routes, each of which starts and ends at a given depot, and a specific UAV can start from one and return to another depot.
7. The selection of starting and finishing depots and targets for inclusion in sub-routes is carried out during the execution of the mission.
8. Tasks for UAVs in which their carrying capacity is not a limiting factor (monitoring, survey, delivery of light objects) are considered.
9. For reasons of expediency, some depots may be inactive.

The following notation will be used:

$B = \{1,\dots, b\}$ – a set of points (places of possible basing), which can potentially be used as a depot, b – the number of such places;

$N = \{1,\dots, n\}$ – the set of targets to be visited, n – their number;

$M = \{1,\dots, m\}$ – set of available UAVs, m – their number;

d_{st} – the distance between targets or targets and places of possible basing s,t, where $s,t \in B \cup N$;

c_i – the cost of placing a depot in point i, $i \in B$;

T_{ki} – is an estimate of the survey time of the k UAV of the target i, $i \in N$;

e_k – cost of resources of the k-th UAV per unit of path length;

v_k – the average speed of the k UAV;

R_k – is the resource of the k-th UAV (the cost of the entire fuel supply or battery charge).

The inclusion of a flight from point (depot, target) i to point (target, depot) j for UAV k will be set by variables x_{ijk}:

$$x_{ijk} = \begin{cases} 0, & \text{the flight is not performed,} \\ 1, & \text{the flight is performed,} \end{cases} \quad i, j \in B \cup N,\ k \in M. \tag{1}$$

The problem consists in minimizing the total cost of the mission and can be presented as follows: find

$$\min \sum_{k\in M} \sum_{i\in B\cup N} \sum_{j\in B\cup N} e_k d_{ij} x_{ijk} + \sum_{k\in M} \sum_{i\in N} \sum_{j\in B\cup N} e_k v_k T_{kj} x_{ijk} \tag{2}$$

for constraints

$$\sum_{k\in M} \sum_{i\in B\cup N} x_{ijk} \le 1, \quad j \in N; \tag{3}$$

$$\sum_{k\in M} \sum_{j\in B\cup N} x_{ijk} \le 1, \quad i \in N; \tag{4}$$

$$\sum_{i\in B} \sum_{j\in N} x_{ijk} \le 1, \quad k \in M; \tag{5}$$

$$\sum_{i\in B} \sum_{j\in N} x_{jik} \le 1, \quad k \in M; \tag{6}$$

$$u_i - u_j + x_{ijk} \sum_{s \in B \cup N} \sum_{t \in B \cup N} x_{stk} \le \sum_{s \in B \cup N} \sum_{t \in B \cup N} x_{stk} - 1, \; k \in M; \quad (7)$$

$$\sum_{i \in B} \sum_{j \in B} x_{jik} = 0, \; k \in M; \quad (8)$$

$$d_{ij} = \infty, \;\; i, j \in B; \quad (9)$$

$$\sum_{i \in B \cup N} \sum_{j \in B \cup N} e_k d_{ij} x_{ijk} + \sum_{i \in N} \sum_{j \in B \cup N} e_k v_k T_{kj} x_{ijk} \le R_k, \;\; k \in M; \quad (10)$$

$$x_{ijk} \in \{0,1\}, i, j \in B \cup N, \; k \in M; \quad (11)$$

$$y_i \in \{0,1\}, \; i \in B. \quad (12)$$

The objective function (2) determines the total costs of planning the choice of a depot for UAVs along with the construction of routes for flying over targets and the time spent on their maintenance.

Each UAV visits a certain number of targets, but each UAV must arrive and depart from each target only once. A value of one in formula (3) means that only one UAV leaves target *j*, and (4) that only one UAV arrives at the target – or zero if UAV k is not involved. Formula (5) sets the condition for UAV departure from one of the possible depots, and formula (6) - the condition for returning to one of these depots; again, if the UAV is involved in the solution variant, then we have equality. Formula (7) sets the condition of avoiding sub cycles in the route of each UAV, which makes the matrix of solutions asymmetric. The requirement not to fly from one possible depot directly to another is reflected in formulas (8)–(9). Formula (10) takes into account the limitations on UAV flight resources. Finally, formulas (11)–(12) specify the definition domains of the variables of the problem.

3. Solving algorithms

Two specialized algorithms based on the ant colony optimization (ACO) and the deterministic local search method (DLS) have been developed [17]. In ACO algorithms, a special model of the problem being solved is formed, therefore they belong to the class of model-oriented methods. The problem model is presented in the form of a weighted graph $G(V, E)$, where $v_i \in V, i = 1, \ldots, n + k$ vertices correspond to solution components, and $e_{ij} \in E, e_{ij} = (v_i, v_j), v_i, v_j \in V$ edges correspond to possible connections (transitions) between corresponding vertices (bridges). For each edge, a connection cost function is defined, which corresponds to the distance along the surface between the vertices connected by this edge.

At each step of the algorithm for any vertex $i \in V$ a set of neighboring vertices N_j can be constructed.

Heuristic information η_{ij} is a numerical value that does not depend on the solutions found in the previous steps and reflects the degree of desirability of including a particular new edge of the model graph in the constructed fragment of the solution $e_{ij} \in E$. Heuristic values are based on a priori information that reflects the conditions of a particular problem and is provided by a source other than the ants.

Pheromone level (pheromone trace) $- \tau_{ij}$, which corresponds to the edge $e_{ij} \in E$, is a positive number that shows how often this edge was used by ants in previous steps or when forming a complete solution. Pheromone trace serve as a long-term memory for ants regarding the entire search process.

So, the main components of the computing scheme of ant algorithms are as follows:

- a problem model presented by a special graph;
- pheromone values;
- heuristic information;
- memory (local and global).

In Figure 1 the computational scheme of ACO algorithms is described in pseudocode.

```
procedure ACO ( x )
    initialize_algorithm;
    while termination_criterion_not_satisfied do
        form_of_ant_population;                          {current generation}
        foreach ant_from_population do                   {ant life cycle}
            initialize_ant;
            M = update_ant_memory;
            while current_state ≠ complete_solution do
                A = local_matrix_ of_ ant_ routes;
                form_set_of_allowed_vertices;
                p = compute_ transition_probabilities(A, M, Π);
                next_state = decision_rule(p , Π);
                move_to_next_state(next_state);
                if online_pheromone_update then
                    deposit_pheromone_on_visited_arc;
                    update_matrix_of_ant_routes_A;
                endif
                M = update_internal_state;
            endwhile
            if online_delayed_pheromone_update then
                foreach visited_arc_of_constructed_junction do
                    deposit_pheromone_on_visited_arc;
                    A = update_the_matrix_of_ant_routes;
                endforeach
            endif
            finish;
        endforeach
        pheromone_evaporation;
        update_best_known( x );
        Demon_actions;                                   {optional}
    endwhile
end
```

Figure 1: Pseudocode of the ACO algorithm.

Let's take a closer look at the rules for moving to the next vertex and the process of calculating transition probabilities. The states of the problem are defined in terms of finite sequences $y = (v_{s_1}, v_{s_2}, \dots), v_{s_r} \in V$ of elements V (or, equivalently, E), which at all intermediate steps of the ant are fragments of the solution of the optimization problem. If Y is the set of all possible sequences, then the set of Y^{∞}all (sub)sequences that satisfy the constraint $\Pi = \Pi(V, E, t)$ is a subset Y: $Y^{\infty} \subseteq Y$, and its elements determine the permissible states of the problem. Suppose that at a certain step, an ant k constructed a fragment of the solution y , the last component of which is a vertex, $i \in V$, that is, it is in this vertex: $y = (\dots, i)$. Then the ant can move to any vertex j from the set of possible neighboring vertices N_i^k, defined as $N_i^k = \{j: j \in N_i \wedge (y, i) \in Y^{\infty}\}$, where N_i is the set of all adjacent to i vertices of the graph of the problem [17]. The selection of the next vertex is based on a pseudo-random proportional rule. Let's enter a new parameter $p_0 \in [0,1]$, each ant moves from vertex $i \in V$ to vertex $j \in N_i^k$ with probability p_0; j is determined as follows: $j = arg\ max\left\{\tau_{ir}^{\alpha}(t)\eta_{ir}^{\beta}, r \in N_i^k\right\}$, and $1 - p_0$ the vertex are chosen with a probability according to the rule of the roulette wheel using the probability p_{ij}^k:

$$p_{ij}{}^{k} = \frac{\tau_{ij}{}^{\alpha}(t)\eta_{ij}{}^{\beta}(t)}{\sum_{r \in N_i^k} \tau_{ir}{}^{\alpha}(t)\eta_{ir}{}^{\beta}(t)} \tag{13}$$

The deposition and evaporation of pheromones occurs according to the following formula:

$$\tau_{ij}(t+1) = \rho\tau_{ir}(t) + \frac{(1-\rho)}{f_{\min}^{0}}\}$$

where ρ – the evaporation coefficient, which lies in the range from 0 to 1, and f_{min}^0 is the best value of the objective function on the initial population of ants.

The lower and upper bounds of pheromones are determined by the following formulas [17]:

$$\tau_{\max} = 1/(\rho f_{\min}^{0})$$

$$\tau_{\min} = [\tau_{\max}(1-\sqrt[n]{0.05})]/[(\frac{n}{2}-1)\sqrt[n]{0.05})]$$

where $n = |V|$.

The pheromone matrix is adjusted as follows: $a' = \min\{a, \tau_{max}\}$, $a' = \min\{a', \tau_{min}\}$, where a is an element of the pheromone matrix.

Stagnation is combated by resetting the values of the pheromone matrix to the initial state if the best solution has not been improved after a certain number of iterations. To solve the given problem, an algorithm was developed that takes into account the following aspects:

- selection of a subset of UAVs to be involved;
- selection of initial and final bases for each involved UAV;
- formation of an operation plan that minimizes the total costs necessary for surveying all available targets.

The initial placement of UAVs occurs thanks to the introduction of the concept of a zero base - the point at which all available UAVs are placed before the algorithm starts. Only one such base is needed, let's define the set D consisting exclusively of it. This base is part of the graph of the problem consisting of the following components:

- initial and zero bases;
- all targets;
- edges connecting all targets and initial bases in pairs;
- edges connecting the zero base with the initial ones.

Returning to the zero base and moving from the zero base directly to targets are prohibited. All edges emanating from the zero base have zero length

$$d_{st} = 0 \text{ where } s \in D, t \in B.$$

The movement from the zero base to the target is considered to be the consecutive movement to the initial base closest to the target (the length of such movement is zero) and the movement from this base to the target. Thus, the length of the movement is determined by the formula:

$$d_{st} = \min\{d_{kt}\} \text{ where } s \in D,\ k \in B,\ t \in N.$$

The initial placement of the UAV is part of the obtained solution, since the flights between the zero base and the target (respectively, the initial placement of each UAV) are the subject of optimization for the general algorithm on the given problem graph.

To solve the formulated problem, a modified max-min algorithm of ant systems with a step-by-step construction of the solution was used. Each ant choses the next vertex of the problem graph makes it abandoned for further visits after the transition. The following sequence of actions occurs for all ants at each step:

- formation of a subset of admissible vertices that can be visited by an ant;
- calculation of the probability of transitions from the current vertex i to all admissible vertices (13);
- choice of an admissible vertex and transition to it.

Each ant forms only a partial solution to the problem, and their totality gives a complete solution. Due to this, a set of partial solutions is formed at each iteration, from which the best one is selected, and the pheromone is deposited on it. At the following stages, the following actions take place:

- evaporation of pheromones;
- update of permissible lower and upper bounds of pheromone;
- updating the pheromone matrix of the lower and upper bounds, respectively.

Figure 2 shows the computational scheme of the developed algorithm for solving the given problem. The general procedure governing the solution process consists of:

- starting the greedy algorithm to determine Q - the initial approximation of the solution for use in the formula for calculating the pheromone traces;
- initial placement of agents;
- selection of parameters;
- launch with the selected parameters.

```
procedure UAVRP (x)
 run_greedy_algorithm;
 set_agents_initial_placement_to_zero_depot;
 while not all_targets_visited do
     form_set_of_allowed_vertices_for_all_agents_considering_resources;
     if not valid then
         form_set_of_allowed_vertices_visiting_depot;
     endif
     move_agent_using_shortest_path_to_selected_target;
     foreach parameter_set do
         foreach randomization_source do
             run_aco_with_strict_time_limit;
         endforeach
     endforeach
     select_parameters_according_to_smallest_objective_function_value;
     launch_with_selected_parameters;
 endwhile
end
```

Figure 2: Pseudocode of the algorithm for solving the problem.

The main part of the ant algorithm is presented in Figure 3.

```
procedure UAVRP_ ACO ( x )
 while not time_or_iterations_limit_exceeded do
     initialize_new_iteration;
     set_agents_initial_placement_to_zero_depot;
     while not all_targets_visited do
         find_allowed_flights_to_targets;
         foreach unvisited_target do
             check_resource_sufficiency_to_visit_target;
             check_resource_sufficiency_to_return_to_depot_after_visiting;
         endforeach
         if valid_options then
             calculate_probability_of_making_each_flight;
             perform_flight_according_to_ probabilities _distribution;
         else
             move_all_agents_to_respective_nearest_depots;
         endif
         update_agent_resource;
     if with_fixed_probability then
         run_local_search (Daemon actions);
     endif
     if error_occurred then
         initialize_new_iteration;
         continue_from_beginning_of_cycle;
     endif
     if better_result then
         record_result_and_routes;
         Q = update_Q                    {numerator for calculating pheromone};
     endif
     evaporation_with_min_max_constraints;
     foreach flight_from_result do
         update_traces_considering_Q;
     endforeach
 endwhile
end
```

Figure 3: Pseudocode of the developed ACO algorithm.

The modification of the DLS is based on the algorithm of the decay vector method using the 2-opt replacement operator [17] and is carried out for each received fragment of the UAV route that runs between two bases. The main idea of the modification is adaptation to the given mathematical model considering limited flight resource, multiple vehicles and depots, and starting base selection.

The procedure based on the DLS algorithm, which describes the Demon's actions in the algorithm from Figure 3, is shown in Figure 4.

```
procedure Execute_LS (x)
  foreach subroute(between_departure_and_arrival_from_or_to_depot) as route do
        :start_again:
        foreach target_in_subroute as i do
              foreach target_in_subroute_after_i as j do
                    take_subroute_between(route.start, i);
                    take_reversed_subroute_between(i +1, j);
                    take_subroute_between(j +1, route.end);
                    if new_route_length < route_length then
                          route = new_route;
                          goto start_again;
                    endif
              endforeach
        endforeach
  endforeach
end
```

Figure 4: General pseudocode of modification of the results obtained by ACO.

Each agent (the ant corresponding to the UAV) has a probability of making a move during each iteration of the algorithm, so the agents interact directly throughout the algorithm. The presented procedure is called both within the framework of the modified ACO and in the modified DLS as the main part of the algorithm.

4. Study of the effectiveness of the algorithm

To assess the applicability of the proposed approach to planning in real-time, a computational experiment was conducted to solve a number of problems formed on the basis of using data on traveling salesman problems from the well-known library TSPLIB [18], some of the points in which were selected as bases.

Three problems were formed on real geodata, and four – by using known problems from TSPLIB:

- Problem 1 with 48 targets and 4 bases, topologically based on berlin52 problem from TSPLIB;
- Problem 2 with 15 targets and 5 bases;
- Problem 3 with 24 targets and 5 bases;
- Problem 4 with 19 targets and 3 bases;
- Problem 5 with 12 targets and 3 bases;
- Problem 6 with 11 targets and 3 bases, topologically based on burma14 problem from TSPLIB;
- Problem 7 with 15 targets and 4 bases;
- Problem 8 with 18 targets and 4 bases;
- Problem 9 with 36 targets and 6 bases, topologically based on danzig42 problem from TSPLIB;
- Problem 10, with 19 targets and 3 bases, is topologically based on ulysses22 problem from TSPLIB.

For each problem, a preliminary selection of the parameters of the ant algorithm was performed using accelerated runs with fewer iterations. With each set of parameters, 3 runs were performed with different initializers of the pseudorandom number generator. The running time of the algorithm in all runs for selecting parameters is limited to 20 seconds.

Calculations were performed on a PC with the following parameters:

- MacBook Pro 16-inch 2019;
- Processor: 2.6 GHz 6-Core Intel Core i7;
- Graphics:
    a. AMD Radeon Pro 5300M 4 GB;
    b. Intel UHD Graphics 630 1536 MB;
- RAM: 16 GB 2667 MHz DDR4.

The parameters of the ACO algorithm ρ, α, and β were defined as those corresponding to the best solution obtained at the parameter selection stage.

For the experiment, the estimated range of each UAV is 600 km. The number of available UAVs is 2.

Each problem was solved by three runs of the proposed algorithm with different initializers of the pseudorandom number generator. The main results of the conducted experiment are given in Table 1.

Table 1
Problem specifications

Problem number	n	b	S_{rec}, km	S_{loc}, km
1	52	4	186.33	193.86
2	20	5	296.89	316.66
3	29	5	202.89	223.58
4	22	3	318.65	326.63
5	15	3	313.64	315.55
6	14	3	102.75	102.99
7	19	4	176.37	209.36
8	22	4	129.76	137.84
9	42	6	154.58	182.29
10	22	3	460.86	469.48

Here, S_{rec} is the total flight length of each UAV corresponding to the best found operation plan, S_{loc} is the total flight length of each UAV corresponding to the operation plan obtained using DLS, *n* is the dimension of the problem (number of targets and bases), *b* is the number of bases.

Figure 5: Visualization of the obtained plan of operations for problem 5 using the modified ACO.

Figure 5 shows the operation plan obtained for problem 5 (15 points, 3 bases). We number the bases from 1 to 3 according to the captions in the figure. Then the operation plan can be interpreted as follows:

- UAV No. 1 took off from base No. 1, visited 7 targets, updated the power reserve at base No. 2, visited 5 targets, and completed the flight at base No. 3;
- UAV No. 2 was not involved.

One of the alternative interpretations:

- UAV #1 took off from base #1, visited 7 targets, and completed the flight at base #2;
- UAV No. 2 took off from base No. 2, visited 5 targets, and ended the flight at base No. 3.

Since the total distance is the same for both cases, the choice of a specific interpretation does not affect the objective function, however, in the case of time optimization of the operation, it may affect the total time due to the possibility of parallel operation of UAVs. Similarly, in this case, the objective function is not affected by the change in the direction of UAV movement.

For comparison, consider the plan obtained using the greedy algorithm and local search.

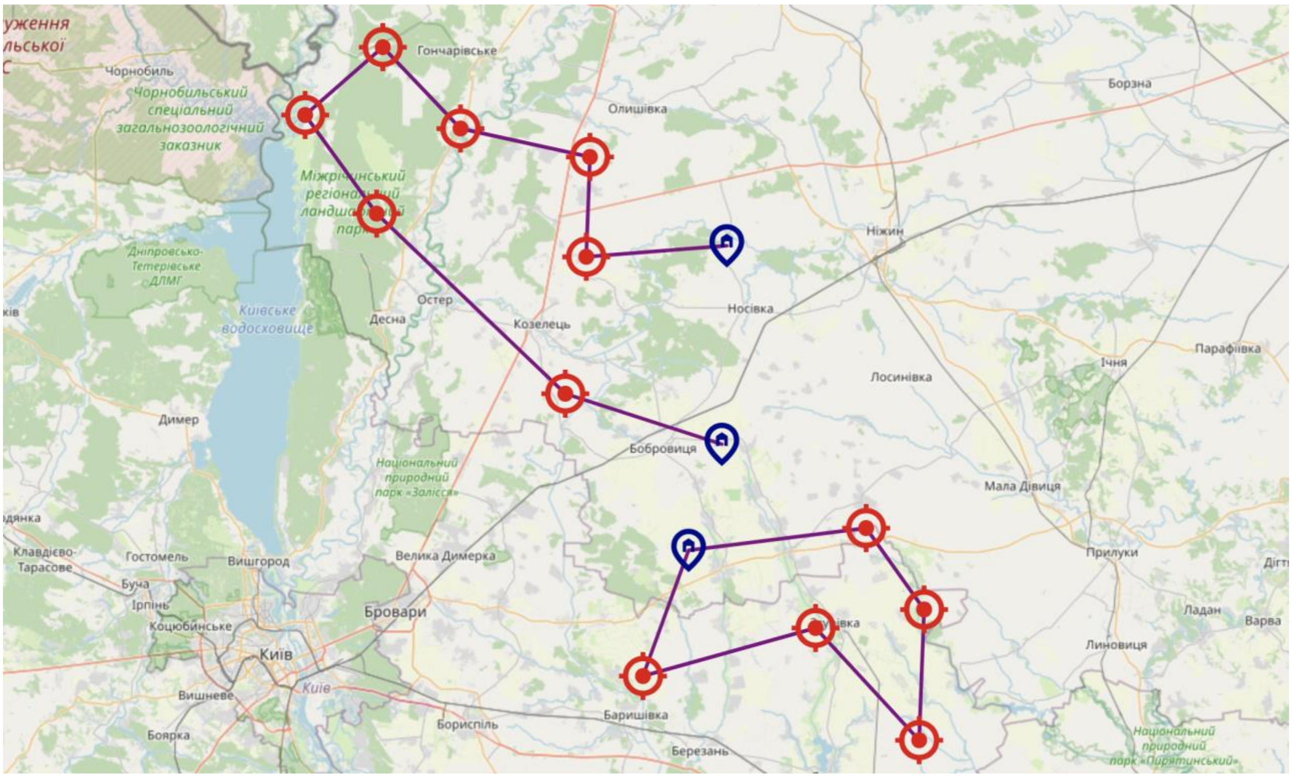

Figure 6: Visualization of the obtained plan of operations for problem 5 using a modified DLS.

It should be noted that on small-dimensional problems, the difference between the results of modified DLS and ACO turned out to be significantly smaller than on larger-dimensional problems with complex sub-routes.

Let's also consider the plans built for problem 3.

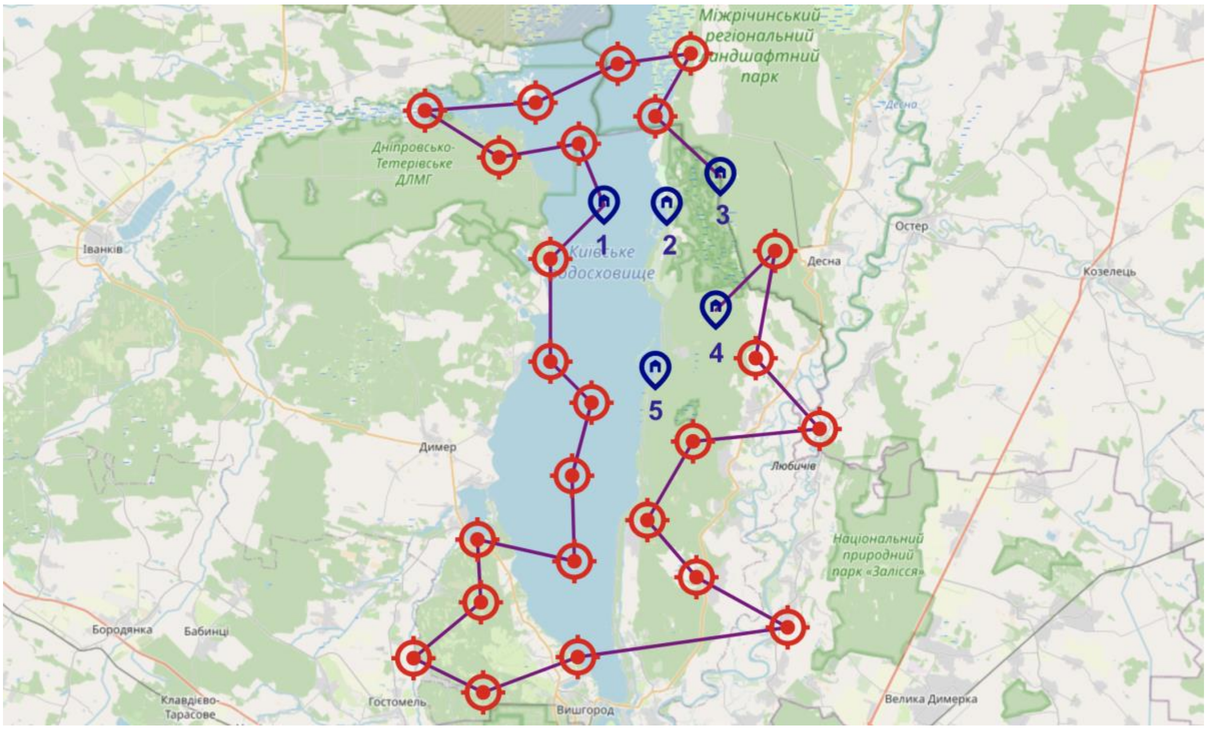

Figure 7: Visualization of the resulting plan of operations for problem 3, obtained using the modified ACO.

Figure 7 shows the operation plan obtained for problem 3 (29 points, 5 bases). We number the bases from 1 to 5 according to the captions in the figure. Then the operation plan can be interpreted as follows:

- UAV #1 took off from base #1, visited 7 targets, completed the flight at base #3;
- UAV #2 took off from base #1, visited 17 targets, completed the flight at base #4;
- Bases #2 and #5 remained inactive.

In the Table 2 shows the results of running the algorithm obtained with different combinations of parameters ρ, α and β, the time limit is 5 seconds for each run of the algorithm. The deviation is indicated in comparison with the best result found (in this case it corresponds to the combinations ρ=0.8, α=0.4, β=4 and ρ=0.4, α=0.8, β=4).

Table 2
Results obtained with selected sets of parameters

ρ	a	β	Length, km	Deviation, %
0.1	0.1	0.5	285	28.93
0.1	0.1	4	226.14	10,28
0.1	0.1	7	215.78	5.97
0.1	0.4	0.5	278.26	27.08
0.1	0.4	4	210.96	3.82
0.1	0.4	7	210.23	3.49
0.1	0.8	0.5	281.48	27.92
0.1	0.8	4	204.06	0.57
0.1	0.8	7	206.67	1.83
0.4	0.1	0.5	265.68	23.63
0.4	0.1	4	217.83	6.86
0.4	0.1	7	211.01	3.85
0.4	0.4	0.5	299.25	32.20
0.4	0.4	4	206.64	1.81
0.4	0.4	7	205.27	1.16
0.4	0.8	0.5	271.37	25,23
0.4	0.8	4	202.89	0
0.4	0.8	7	206.68	1.83
0.8	0.1	0.5	276.07	26.51
0.8	0.1	4	212.93	4.71
0.8	0.1	7	205.27	1.16
0.8	0.4	0.5	250.11	18.88
0.8	0.4	4	202.89	0
0.8	0.4	7	206.68	1.83
0.8	0.8	0.5	266.83	23.96
0.8	0.8	4	209.49	3.15
0.8	0.8	7	208.83	2.84

The deviation is calculated as $(1 - S_{rec} / S) \cdot 100$, where S_{rec} is the best result obtained, S is the result for the corresponding combination of parameters. The selected combination is the same as the combination selected during the automatic tuning phase of the algorithm. When running the algorithm with an increased time limit, certain combinations also lead to the best-known solution. For example, when running the algorithm with a limit of 30 seconds and combinations of ρ=0.4, α=0.4, β=7 and ρ=0.8, α=0.4 and β=4, the length of the constructed operation plan is 202.89 km.

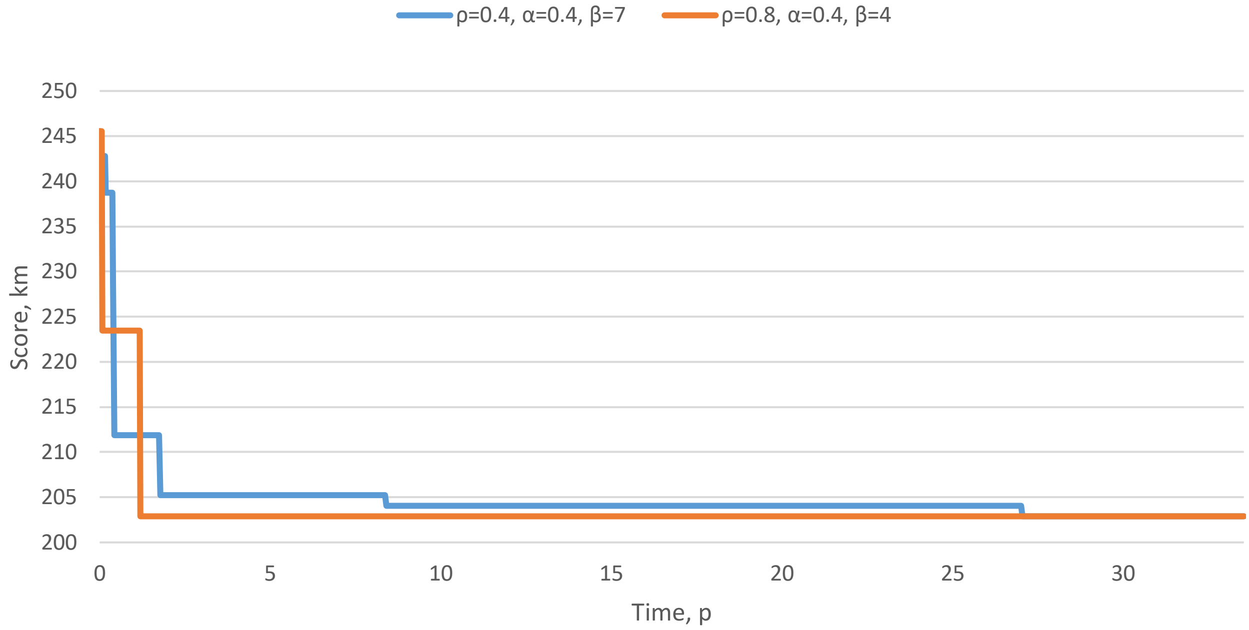

Figure 8: Visualization of the improvement of the result over the time of operation of the algorithm.

Figure 8 illustrates the dynamics of finding the best result over the operation time for the two groups of parameters indicated above. For the first combination, the best known value is obtained at 27 seconds of operation, while for the second - 1.2 seconds after launch.

For comparison, the result of the operation of the modified DLS for problem 3 is presented.

Figure 9: Visualization of the plan of operations for problem 3 obtained using the modified DLS.

The length of the best route obtained using DLS is 223.58 km against 202.89 for ACO. It should be emphasized that DLS runs are also part of the modified ACO, however, better results are provided by the variability of the initial solutions.

5. Conclusions

The paper considers the problem of optimizing decisions when planning a UAV group mission to inspect or service a given set of customers (targets) in the presence of alternative depots. A substantive formulation and mathematical model of the problem of distributing targets by bases and UAVs and optimizing their routes when performing inspection and/or servicing a given set of targets with the condition of completing the route in certain reception areas (depots) and restrictions on UAV resources as a special combinatorial optimization problem are presented.

The max-min algorithm of ant systems has been developed, the feature of which is the step-by-step interaction of ants for the formation of solutions, and deterministic local search algorithm – the decline vector method.

The developed algorithms have been tested both on known instances of a traveling salesman problem and on problems specially formed in the area with many depots and existing restrictions. The proposed algorithm based on ACO has shown better results in terms of accuracy, although the calculation time increased.

Given a certain similarity of problems, the developed algorithms can be developed for the purpose of application in the creation of information technologies for planning missions of hybrid transportation systems, which include UAV or other drone and a vehicle.

Also, the statement of the problem given allows for the development of a UAV mission plan, which provides for operation outside the reach of communication with eventual return for the synchronization of accumulated data.

The direction of further research may be to consider in the mathematical model the characteristics of the battery discharge process, considering prohibited flight areas, weather conditions (wind direction). Another promising direction is improving the ACO algorithm involving diversified algorithms for finding solutions [19], and parallelized implementation of the island model of the ACO algorithm [20, 21].

6. References

[1] Thibbotuwawa A., Bocewicz G., Nielsen P., Banaszak, Z. (2020). Unmanned aerial vehicle routing problems: a literature review. *Applied sciences, 10(13)*, 4504 . https://doi.org/10.1111/itor.12783

[2] Ahn, N., & Kim, S. (2022). Optimal and heuristic algorithms for the multi-objective vehicle routing problem with drones for military surveillance operations. *Journal of Industrial and Management Optimization, 18* (3), 1651-1663. https://doi:10.3934/jimo.2021037

[3] Kucukoglu, I., Dewil, R., & Cattrysse, D. (2021). The electric vehicle routing problem and its variations: A literature review. *Computers & Industrial Engineering, 161,* 107650. https://doi.org/10.1016/j.cie.2021.107650

[4] Murray CC, Chu AG (2015). The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery. *Transportation Research Part C: Emerging Technologies, 54, May 2015*, 86-109. https://doi.org/10.1016/j.trc.2015.03.005

[5] Murray CC, Raj R. (2020). The multiple flying sidekicks traveling salesman problem: Parcel delivery with multiple drones. *Transportation Research Part C: Emerging Technologies, Vol. 110*. 368-398. https://doi.org/10.1016/j.trc.2019.11.003

[6] Wang, X., Poikonen, S., & Golden, B. (2017). The vehicle routing problem with drones: several worst-case results. *Optimization Letters, 11*, 679-697.

[7] Wang, Z., & Sheu , JB (2019). Vehicle routing problem with drones. *Transportation research part B: methodologica* , *122* , 350-364. https://doi.org/10.1016/j.trb.2019.03.005

[8] Kuo , RJ, Lu, SH, Lai, PY, & Mara, STW (2022). Vehicle routing problem with drones considering time windows. *Expert Systems with Applications, 191,* 116264. https://doi.org/10.1016/j.eswa.2021.116264

[9] Jeong , HY, & Lee, S. (2021). Collaborative hybrid delivery system: Drone routing problem assisted by truck. In Advances in Production Management Systems. Artificial Intelligence for Sustainable and Resilient Production Systems: IFIP WG 5.7 International Conference, APMS 2021, Nantes, France, September 5–9, 2021, Proceedings, Part III. Springer Int. Publ. 33-42.

[10] Jeong , HY, & Lee, S. (2023). Drone routing problem with truck: Optimization and quantitative analysis. *Expert Systems with Applications, 227* , 120260. https://doi.org/10.1016/j.eswa.2023.120260

[11] Golden, B., Wang, X., & Wasil , E. (2023). The Evolution of the Vehicle Routing Problem – A Survey of VRP Research and Practice from 2005 to 2022. In The Evolution of the Vehicle Routing Problem: A Survey

of VRP Research and Practice from 2005 to 2022 (pp. 1-64). Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-18716-2

[12] Gonzalez-R, PL, Canc , D., Andrade-Pineda, JL, Calle, M., & Leon-Blanco, JM (2020). Truck-drone team logistics: A heuristic approach to multi-drop route planning. *Transportation Research Part C: Emerging Technologies*, *114*, 657-680. https://doi.org/10.1016/j.trc.2020.02.030

[13] Zheng C. (2023). Review on Combined Vehicle Routing Problem of Drones and Vehicles. *Journal of Innovation and Development*, *2(1)*, 55–57. https://doi.org/10.54097/jid.v2i1.5420

[14] Horbulin V.P., Hulianytskyi L.F., Sergienko I.V. (2019). Formulations and mathematical models of problems of optimization of aircraft routes with dynamic depots. *Managers systems and machines*, *1*, 3-10. https://doi.org/10.15407/usim.2019.01.003

[15] Horbulin , V. P. , Hulianytskyi , L. F., Sergienko , I.V. (2020). Optimization of UAV Team Routes in the Presence of Alternative and Dynamic Depots. *Cybern Syst Anal 56*, *2*, 195–203. https://doi.org/10.1007/s10559-020-00235-8

[16] Li, J., Xiong, Y., & She, J. (2023). UAV path planning for target coverage task in dynamic environment. *IEEE Internet of Things Journal* . https://doi.org/10.1109/jiot.2023.3277850

[17] Hulianytskyi L.F., Rybalchenko O.V. (2021) Formalization of the problem of optimizing the base locations and routes of the UAV group. *Cybernetics and Computer Technologies*, *4*, 12–26. https://doi.org/10.34229/2707-451X.21.4.2

[18] Reinelt, G. (1991). TSPLIB — A traveling salesman problem library. *ORSA journal on computing*, *3(4)*, 376-384.

[19] Hulianytskyi LF (2019). ACO algorithms with diversified search. Int. Symposium "Intelligent Solutions". Decision Theory: *Proceedings of the IX International School-Seminar (Uzhhorod , April 15-20, 2019)*. Uzhhorod: UzhNU , 17–21. (in Ukrainian) .

[20] Stützle , T. (1998). Parallelization strategies for ant colony optimization. In *International Conference on Parallel Problem Solving from Nature*. Berlin, Heidelberg: Springer Berlin Heidelberg. 722-731.

[21] Mora AM, García-Sánchez P., Merelo JJ (2013), Pareto-based multi-colony multi-objective ant colony optimization algorithms: an island model proposal. *Soft Computing*, *17*. 1175. https://doi.org/10.1007/s00500-013-0993-y