Mathematical Biosciences and Engineering

MBE, 19(7): 7091–7121.
DOI: 10.3934/mbe.2022335
Received: 25 February 2022
Revised: 30 April 2022
Accepted: 05 May 2022
Published: 13 May 2022

http://www.aimspress.com/journal/MBE

Research article

Comparison of exact and approximate approaches to UAVs mission contingency planning in dynamic environments

Grzegorz Radzki[1,*], Grzegorz Bocewicz[1], Jarosław Wikarek[2], Peter Nielsen[3] and Zbigniew Banaszak[1]

[1] Faculty of Electronics and Computer Science, Koszalin University of Technology, Poland
[2] Department of Information Systems, Kielce University of Technology, Kielce, Poland
[3] Department of Materials and Production, Aalborg University, Denmark

* **Correspondence:** Email: radzki.grzegorz@gmail.com.

Abstract: This paper presents a novel approach to the joint proactive and reactive planning of deliveries by an unmanned aerial vehicle (UAV) fleet. We develop a receding horizon-based approach to contingency planning for the UAV fleet's mission. We considered the delivery of goods to spatially dispersed customers, over an assumed time horizon. In order to take into account forecasted weather changes that affect the energy consumption of UAVs and limit their range, we propose a set of reaction rules that can be encountered during delivery in a highly dynamic and unpredictable environment. These rules are used in the course of designing the contingency plans related to the need to implement an emergency return of the UAV to the base or handling of ad hoc ordered deliveries. Due to the nonlinearity of the environment's characteristics, both constraint programming and genetic algorithm paradigms have been implemented. Because of the NP-difficult nature of the considered planning problem, conditions have been developed that allow for the acceleration of calculations. The multiple computer experiments carried out allow for comparison representatives of the approximate and exact methods so as to judge which approach is faster for which size of the selected instance of the UAV mission planning problem.

Keywords: contingency planning; UAV fleet mission; weather changes; declarative modeling; genetic algorithm

1. Introduction

In this paper, we consider the planning of an unmanned aerial vehicle (UAV) fleet mission problem with highly dynamic and unpredictable environment constraints [1–5]. Typical disruptions in urban deliveries by UAVs (characterized by frequent stop-and-go movements, low consolidation, and frequent re-scheduling), may be caused by changes in the order by customers, or shifting weather conditions (e.g., a sharp drop in temperature, icing of propellers, turbulence), which affect the energy consumption of UAVs and cause their shorter range due to the depletion of batteries [6,7]. For that reason, the routing of a UAV fleet in a partially known and unpredictable environment should guarantee a reactive on-line determined contingency reaction.

The UAVs' routes can be determined in the course of proactive planning [8] and generated in offline mode or in reactive planning [9,10], then executed in the online mode. Routes determined in proactive planning guarantee the achievement of the planned mission's goal for the environment's parameters, which change at predetermined intervals. Scenarios corresponding to planned reactive rules do not guarantee the existence of reactive end-to-end paths employed in the routing process while adapting it to changes in the environment during mission execution [2,6,10–18]. In particular, the reactive routing strategies linking the "route discovery (proactive route planning)" and "route maintenance (reactive rules adopting)" concepts [19] are responsible for UAV robustness to the changes that appear in the urban distribution context. In this context, our study focuses on reactive planning [20,21] of deliveries by a UAV fleet, which are resistant to sudden changes in weather conditions and unforeseen changes in the delivery schedules.

The originality of the paper results from its merging of the proactive and reactive planning of missions of UAV fleet. The developed models allow for predictive (i.e., taking into account forecasted weather conditions changing) and reactive (i.e., enabling interruption of a drone's mission) planning of delivery missions in terms of the constraint satisfaction problem [6,22,23] and the easy implementation of the proposed genetic algorithm in a commercially available constraint programming environment, e.g., IBM ILOG.

The paper is structured as follows. Section 2 presents the state of art. The approach to UAV mission contingency planning is presented in Section 3. The declarative model for reactive planning of deliveries by UAV fleets according to the constraint satisfaction problem is described in Section 4. Sufficient conditions guaranteeing the existence of a non-empty space of admissible solutions and, consequently, a reduction in the time expenditure incurred on the assessment of the weather changes are outlined in Section 5. The structure and operation of the developed genetic algorithm are presented in Section 6. The results of the conducted experiments regarding implementation of an accurate and approximate approach to UAV fleet mission planning are discussed in Section 7. Final conclusions are stated in Section 8, followed by suggestions for future research.

2. Related work

UAV mission planning issues play a key role in supply management systems in distributed networks belonging to the VRP class of NP-hard computational complexity [2,24]. Consequently, the sub-problems of routing and scheduling occurring in them also encounter a combinatorial explosion barrier, forcing the use of approximate solution methods. The goal is to identify a set of routes for a fleet of vehicles that optimizes some underlying objective function while subject to problem-specific

constraints [25–29]. These efforts, according to the current taxonomy of available route planning methods, can be partitioned into two main categories: approximation and exact methods [6,26,30–34].

In general planning of the UAV path comes down to finding the path connecting the starting point with the destination that guarantees the fulfillment of the UAV operational requirements within the specified UAV flight constraints. Problems of this type undertaken in the area of UAV routing, i.e., VRP class problems [1,5], are formulated for various assumptions covering both the topography and the presence of obstacles. The topography, related to, e.g., mountainous and/or urbanized areas, influences the changes of the flight trajectory level, while the presence of stationary and/or dynamic obstacles makes it impossible to keep the straight shape of the trajectory [35]. Limitations of this type, force the inclusion of many new variables multiplying the complexity of UAVs trajectory planning problems, thus implying the need to use metaheuristic algorithms. The underlying nature-inspired algorithms such as the Bath algorithm, particle swarm optimization (PSO), artificial bee colony (ABC) algorithm, ant colony algorithm (ACO), grey wolf algorithm, etc. compete in terms of their convergence accuracy and proceeding efficiency [36–38].

Many works dealing with the problem of UAV routing, in particular those related to the use of UAVs in situations related to the monitoring of reservoirs, fields and forests, as well as flood victims support, assume UAV moves over flat terrain without obstacles that exclude the linear shape of their trajectory. In situations of this type, assuming that the UAV takes off vertically with the collected goods and also lands vertically in order to unload it, the problem of UAV routing boils down to determining the appropriate wind correction angle, correcting its drift [24]. Assuming these assumptions, this work addresses the problem of re-planning a UAV route in situations where sudden changes in wind direction and its intensity go beyond the fluctuation ranges adopted at the stage of proactive route planning.

An approximation method attempts to find the best possible solution while providing no guarantees as to the quality of that solution, i.e., it prefers quick solutions over optimal ones. In turn, an exact method is guaranteed to find the best solution to the problem given enough time, i.e., it is oriented toward the search for the optimal solution at the expense of the time incurred to obtain it. In the approximation approach, there are methods that implement heuristic algorithms (e.g., metaheuristics-driven ones, such as variable neighborhood search (VNS), simulated annealing (SA), and tabu search (TS)); population algorithms, such as PSO, ACO, ABC and evolutionary algorithms, such as memetic algorithms (MMA), and genetic algorithm (GA)) [26,30,32,34].

The exact approach leverages intelligent forms of enumerative search, such as dynamic programming (DP), mixed integer linear programming (MILP), branch-and-bound (BB), and constraint programming (CP) [6,31,33]. All of the above-mentioned approaches are supplemented by computer simulation tools, such as flight simulators [37,39,40].

There is a growing trend in research aimed at developing an interactive tool dedicated to online planning missions carried out by UAV teams, as reflected in numerous publications [12,13,16,22]. The aim of this research is to devise solutions that enable proactive and/or reactive planning of UAV missions in situations caused, for instance, by contingency planning in dynamic environments, i.e., decision support tools facilitating analysis and different scenarios comparison. Besides emergency situations, such tools can be helpful in various fields of application, e.g., reconnaissance and mapping, package delivery and delivery communication capabilities, healthcare and so on [3,4,12,13,18].

The UAV mission planning problems encountered in practice are characterized by a large variety of parameters, constraints, and specific evaluation criteria. Some examples of possible differences are listed in the following exemplary sets:

- parameters specifying UAVs (e.g., maximum loading capacity, maximum energy capacity, empty weight, payload weigh, ground speed, and aerodynamic drag coefficient), environmental conditions (e.g., weather forecasts including wind speed and direction, air density, temperature drizzle, or snowfall), and distribution network topology (e.g., number of delivery points, number of depots, and distances between delivery points and depots);
- constraints determining UAVs trajectories (e.g., to avoid collisions, a UAV is configured with at least one flight corridor and flight path), battery durability, delivery volumes, delivery periods, and changing weather conditions;
- evaluation criteria used to determine UAV fleet routings that guarantee the maximum benefits with the shortest total path, in particular, the maximum travel distance (limited by the capacity of the battery used), the minimum length of the sum of the flight path of each UAV, timely deliveries, resistance to changes in weather conditions, and/or service delivery deadlines.

As already mentioned, problems of this type belong to the category of NP-hard problems. This means that instances of such problems encountered in practice (due to their size as well as the number and variety of characteristics describing them) can be resolved within a reasonable time interval using approximate methods. In that context, the exact (i.e., optimal) solutions achieved in a similar time frame by exact methods refer, however, only to the small size instances out of practical significance. It is also easy to see that, in general, it is difficult to say which approach is best used for a given problem instance. The literature is rather scarce on this topic. Therefore, the research gap in this field is the motivation behind our research aimed at the implementation of selected methods representing both the approximate and exact approaches to compare them on using an arbitrarily selected example of a UAV fleet planning problem.

The comparison included CP as representative of the exact approach and GA as representative of the approximate approach. CP is a paradigm for solving combinatorial problems according to which users declaratively state the constraints on the feasible solutions for a set of decision variables. The acceptable solution sought is the set of decision variables values satisfying all the constraints [6,22,27]. It is worth emphasizing that CP is more general than MILP, allowing variable types beyond integer and continuous (e.g., interval and set variables), and dropping the restriction of linearity in the constraints and objective function. In turn, the GA, being the representative of evolutionary algorithms, is a search technique inspired by evolutionary biology such as inheritance, mutation, selection, and crossover used in computing to find true or approximate solutions to optimization and search problems [41–43]. Carrying out appropriate computer experiments allows to judge which approach is faster for which size of the selected instance of the UAV mission planning problem.

3. Approach to UAVs fleet online routing

We consider a distribution network, which is modeled by the graph $G = (N, E)$ where $N = \{N_1, \dots, N_\lambda, \dots, N_n\}$ signifies the set of $n = |N|$ nodes (distinguishing N_1 node representing a base and $\{N_2, \dots, N_n\}$ nodes representing delivery points), and $E = \{(N_i, N_j) |\ i, j \in \{1, \dots, n\}, i \neq j\}$ signifies the set of edges determining the possible connections between nodes. Given a fleet of UAVs $\mathcal{U} = \{U_1, \dots, U_k, \dots, U_K\}$ that delivers to the points $\{N_2, \dots, N_n\}$, to each delivery point N_λ an ordered quantity of goods $z_\lambda \in \mathbb{N}$ (taken from the base N_1) should be transported. Deliveries are made as part of the mission S, which consists of sub-missions ${}^lS$ (i.e., delivery plans that include a single course of UAVs: base-delivery points-base). Z denotes a sequence consisting of variables z_λ: $Z =$

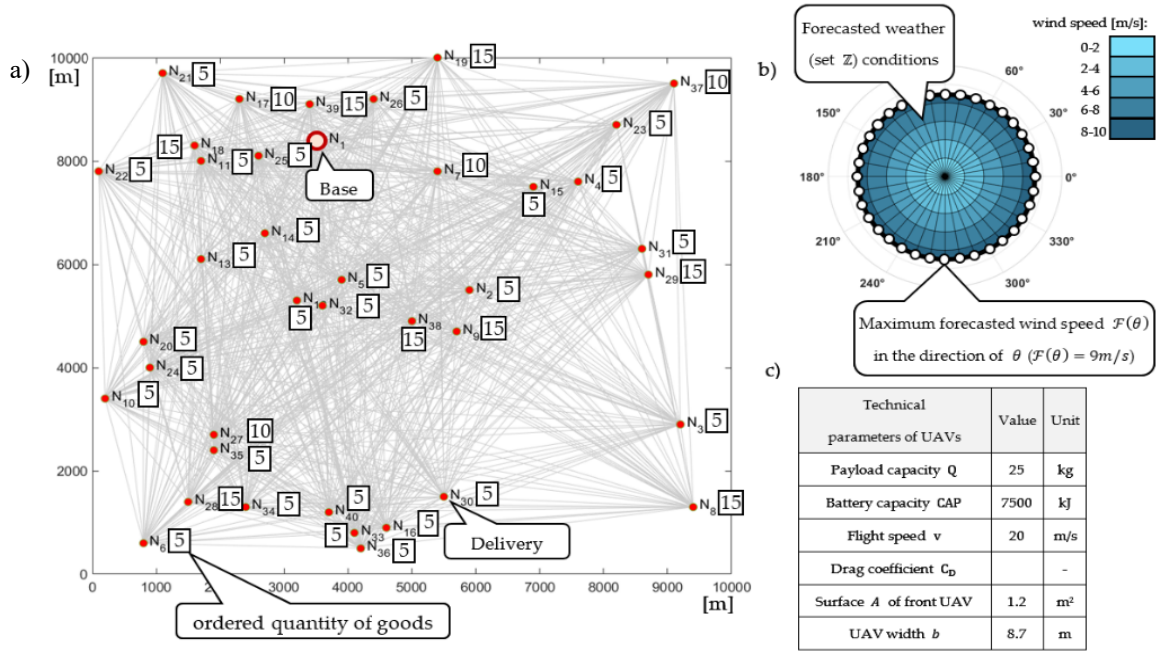
$(z_1, \ldots, z_n)$. It is assumed that all required goods should be delivered in the given horizon time H. The amount of goods delivered during one sub-mission ${}^lS$ by the U_k to the delivery point N_λ is determined by the variable ${}^lc_\lambda^k \in \mathbb{N}$. ${}^lC$ is a sequence: ${}^lC = ({}^lc_1^1, \ldots, {}^lc_1^K, \ldots, {}^lc_n^1, \ldots, {}^lc_n^K)$ determining the payload weight delivered by fleet $\mathcal{U}$. Variable Q_k denotes the payload capacity of U_k (amount of goods transported by U_k cannot exceed Q_k). Moreover, each U_k is described by the following technical parameters: battery capacity CAP, airspeed va, drag coefficient C_D, front surface A, and UAV width b. Time spent on take-off and landing U_k on delivery point N_λ is indicated by variable $w_\lambda \in \mathbb{N}$. Note that ${}^l\mathcal{U} \subseteq \mathcal{U}$ denotes a set of UAVs used during sub-mission ${}^lS$. The moment when the $U_k \in {}^l\mathcal{U}$ arrives to the delivery point N_λ during sub-mission ${}^lS$ is indicated by variable ${}^ly_\lambda^k \in \mathbb{N}$[s]. In that context, the sequence ${}^lY$ consisting of moments ${}^ly_\lambda^k$, is called the schedule of the fleet ${}^l\mathcal{U}$: ${}^lY = ({}^ly_1^1, \ldots, {}^ly_1^K, \ldots, {}^ly_n^1, \ldots, {}^ly_n^K)$.

We assume that the variable $t_{\beta,\lambda} \in \mathbb{N}$ determines traveling time between nodes N_β, N_λ, where: $(N_\beta, N_\lambda) \in E$, and routes of $U_k \in {}^l\mathcal{U}$ during sub-mission ${}^lS$ are represented by sequences: ${}^l\pi_k = \left(N_{k_1}, \ldots, N_{k_i}, N_{k_{i+1}}, \ldots, N_{k_\mu}\right)$, where: $k_i \in \{1, .., n\}$, $\left(N_{k_i}, N_{k_{i+1}}\right) \in E$. ${}^l\Pi$ denotes a sequence of UAVs routes executed during sub-mission ${}^lS$: ${}^l\Pi = ({}^l\pi_1, \ldots, {}^l\pi_k, \ldots, {}^l\pi_K)$ (in cases when $U_k \notin {}^l\mathcal{U}$ then ${}^l\pi_k = \triangle$). The delivery plan of one UAV sub-mission ${}^lS$ is defined as a sequence: ${}^lS = ({}^l\mathcal{U}, {}^l\Pi, {}^lY, {}^lC)$.

It is assumed that a plan of a UAV sub-mission ${}^lS$ is implemented under specific weather conditions, i.e., the weather forecast is known for each sub-mission ${}^lS$. The forecasted weather conditions are described by the set $\mathbb{F}$ of pairs composed of direction θ and wind speed vw $(\theta, vw) \in \mathcal{F}$, i.e., $\mathbb{F}$ is defined as follows: $\mathbb{F} = \{(\theta, vw) | \theta \in [0°, 360°), vw \in [0, \mathcal{F}(\theta)]\}$. Where $\mathcal{F}(\theta)$ is a function that's values determine the maximum forecasted wind speed for given direction θ. The weather conditions determine the admissibility of the adopted sub-mission's plan ${}^lS$, i.e., they determine whether during its implementation, the batteries of UAVs will not be prematurely discharged.

A function $\Upsilon_{k,l}(\theta)$ determines the borderline wind speed (for a given direction θ), which guarantees the successful completion of the delivery plan by the U_k during sub-mission ${}^lS$ in the distribution network G: $\Upsilon_{k,l}(\theta) = \max \Gamma_{k,l}(\theta)$, where: $\Gamma_{k,l}(\theta)$ – set of wind speed's values vw for a given direction θ, for which the batteries of U_k is not discharged. The sub-mission's plan ${}^lS$ is assumed [14] to be resistant to the forecast weather conditions $\mathbb{F}$ if the boundary wind $\Upsilon_{k,l}(\theta)$ of all $U_k \in {}^l\mathcal{U}$ in any direction θ does not exceed the forecasted value $Z(\theta)$: $\forall_{U_k \in {}^l\mathcal{U}} \forall_{\theta \in [0°, 360°)} \Upsilon_{k,l}(\theta) \geq \mathcal{F}(\theta)$.

The implementation of the designated mission S may be subject to various disturbances IS. Among them, there are sudden changes in the weather (beyond the expected $\mathcal{F}^*(\theta)$ range), and changes in orders Z), order changes (increase /decrease amount of ordered deliveries Z^*), changes in the number of delivery points served (changing the structure of the network G^*). The UAV fleet, when performing the mission plan S, meets a disturbance $IS(t^*)$ - covering one of the cases: the weather $\mathcal{F}^*(\theta)$, the network G^*, orders Z^*, at the time t^*. In such situations, it becomes necessary to answer the question: *Does a re-route plan exist for mission S^* that guarantees timely deliveries in a given time horizon H and at acceptable battery level?* To illustrate the motivation behind our approach, let us consider a distribution network from Figure 1a) covering an area of 100 km^2 and containing 39 delivery points (nodes $N_2, \ldots, N_{40}$).



Technical parameters of UAVs	Value	Unit
Payload capacity Q	25	kg
Battery capacity CAP	7500	kJ
Flight speed v	20	m/s
Drag coefficient C_D		-
Surface A of front UAV	1.2	m²
UAV width b	8.7	m

Figure 1. Graphical illustration of the distributed delivery problem.

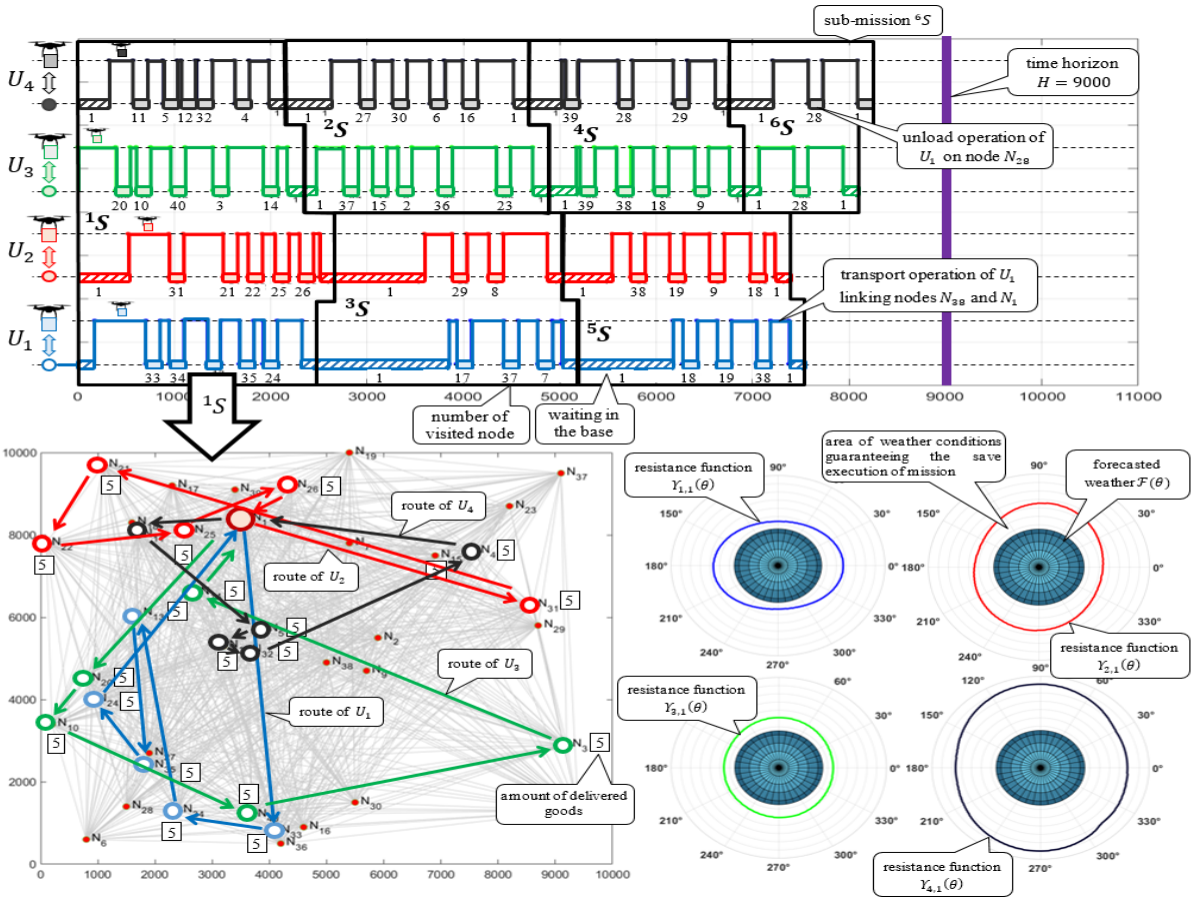


Figure 2. Example of "safe" execution of mission of fleet $\mathcal{U} = \{U_1, U_2, U_3, U_4\}$.

The goods are delivered by a fleet of UAVs, which is stationed in the base N_1–technical parameters of UAVs are collected in the table from Figure 1c). The weight of individual orders is: $z_2 - z_6 = 5$, $z_7 = 10$, $z_{18} = z_{19} = 15$, $z_{20} - z_{26} = 5$, $z_{27} = 10$, $z_{28} = 15, z_{30} - z_{36} = 5$, $z_{37} = 10$, $z_{38} = z_{39} = 15$, $z_{40} = 5$. In that context, we search for the minimum fleet size that guarantees timely deliveries of the required amount of goods. It is apparent that the smallest fleet that guarantees timely delivery (within the time horizon of 2.5 h) consists of $\mathcal{U} = \{U_1, U_2, U_3, U_4\}$ UAVs, see Figure 2. The mission S of Figure 2 consists of 6 sub-missions: $S = ({}^1S, {}^2S, \ldots, {}^6S)$ following conditions where the wind speed does not exceed 9 m/s.

An example illustrating the course of UAV fleet mission ${}^1S$ and the values of the resistance functions: $\Upsilon_{1,1}(\theta)$, $\Upsilon_{2,1}(\theta)$, $\Upsilon_{3,1}(\theta)$, $\Upsilon_{4,1}(\theta)$ is shown in Figure 2. It is evident that the UAVs' routes in the mission ${}^1S$ are weatherproof for the given forecasted weather (i.e., $\Upsilon_{k,l}(\theta) \geq \mathcal{F}(\theta)$):

$${}^1\pi_1 = (N_1, N_{33}, N_{34}, N_{13}, N_{35}, N_{24}, N_1); \quad {}^1\pi_2 = (N_1, N_{31}, N_{21}, N_{22}, N_{25}, N_{26}, N_1):$$
$${}^1\pi_3 = (N_1, N_{20}, N_{10}, N_{40}, N_3, N_{14}, N_1); \quad {}^1\pi_4 = (N_1, N_{11}, N_5, N_{12}, N_{32}, N_4, N_1)$$

and the robustness function UAVs: $\Upsilon_{1,1}(\theta)$, $\Upsilon_{2,1}(\theta)$, $\Upsilon_{3,1}(\theta)$, $\Upsilon_{4,1}(\theta)$.

All goods should be delivered within 2.5 hours ($T = 9000$ [s]). The deliveries take place in different forecasted weather conditions (set $\mathbb{F}$), which are illustrated in Figure 1b). According to the forecast, the wind speed does not exceed $vw = 9\frac{m}{s}$.

We consider a situation in which the weather conditions of the mission carried out rapidly changed at the time $t^* = 3000$[s], i.e., during the execution of the sub-mission ${}^2S$, the wind speed increased to $vw = 11\ m/s$ for direction $\theta = 210° - 230°$.

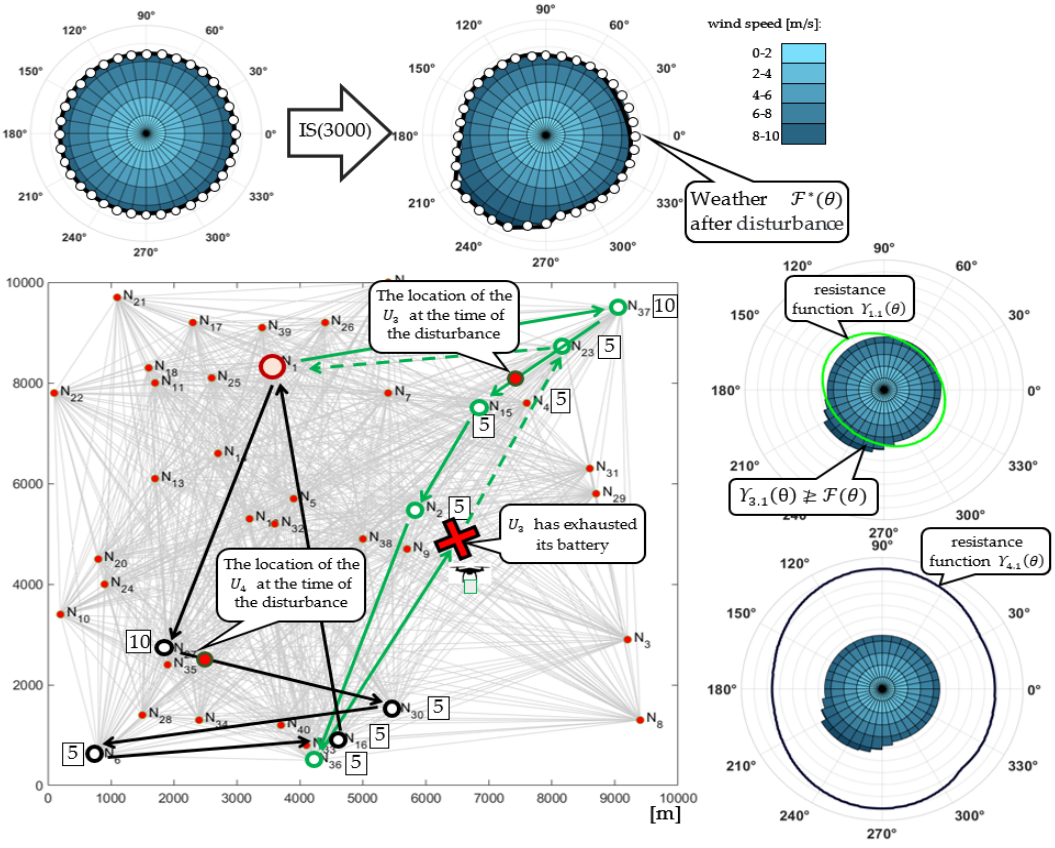


Figure 3. Sub-mission ${}^2S$ after changed weather conditions: the wind speed increased to $vw = 11\ m/s$ in the direction $\theta = 210° - 230°$.

Such a change means that this mission cannot be continued due to too much energy consumption (the mission's resistance function $\Upsilon_{3,2}(\theta)$ values are below the level corresponding to speed $11\frac{m}{s}$ – see Figure 3). Figure 3 shows the location of the UAVs at time $t^* = 3000$ $[s]$, i.e., upon receipt of information about a change in weather, and marked the place where the battery U_3 will be discharged in the event of continuation of deliveries in accordance with the current plan ${}^2S$.

In this situation, it is necessary to correct the route of the sub-mission ${}^2S$ being carried out, which forces the search for an answer to the following question:

Given a UAV fleet $\mathcal{U} = \{U_1, U_2, U_3, U_4\}$ *providing deliveries to the delivery points allocated in the network* G *from Fig. 1, the UAV fleet realizes the delivery mission plan* S^* *from fig. 2 b). At the time* $t^* = 3000$ *a rapid weather* $\mathcal{F}^*(\theta)$ *change, i.e., disturbance* IS*, occurs and results in* $vw = 11\ m/s;\ \theta = 210° - 230°$*. Does a reroute plan exist for mission* S^* *that guarantees the timely delivery of the ordered supplies in a given time horizon* $H = 9000$ $[s]$ *and at an acceptable battery level?*

4. Approach to UAV fleet online routing

Planning drone missions is founded on setting the so-called proactive plans specifying the UAV flight routes (see Figure 4) to guarantee timely delivery to each recipient while taking into account the constraints of the assumed distribution network G and forecasted weather $\mathcal{F}(\theta)$ condition changes. This means that routes and schedules determining the flights of individual UAVs (obtained in the course of proactive fleet mission design) take into account the amount and magnitude of the anticipated disturbances.

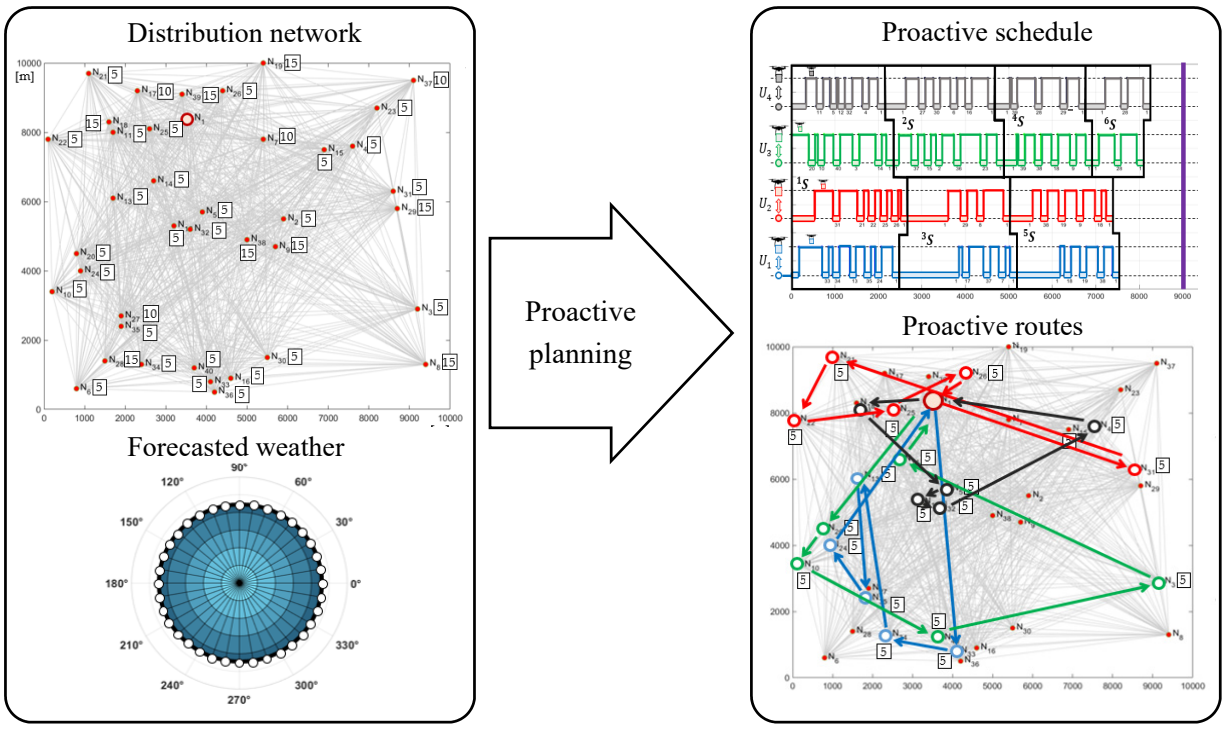


Figure 4. Proactive approach to UAV fleet mission planning.

In other words, proactive planning is carried out before the mission is carried out and boils down

to setting routes and schedules, guaranteeing timely deliveries in the given distribution network even in a situation of changing weather (within the set range specified by weather forecast). The implementation of such plans in a real environment may be disrupted as a result of a sudden unforeseen change in weather conditions (exceeding the range assumed at the proactive planning stage), which may lead to premature depletion of the UAV's battery. In such situations, attempts are made to replace proactive plans by implementing reactive planning principles. Reactive planning (Figure 5), implies changing a proactive plan already being implemented, e.g., due to the occurrence of a disturbance such as rapid weather and/or orders change. Reactive plans must therefore take into account the extent to which the planned deliveries were carried out and adjust the further actions of the UAVs to the prevailing weather conditions. The purpose of reactive planning is therefore an attempt to change mission plans to those that guarantee timely deliveries under new weather conditions. Due to the need to make decisions in the online mode, reactive planning methods must have a short time of determination of the solution.

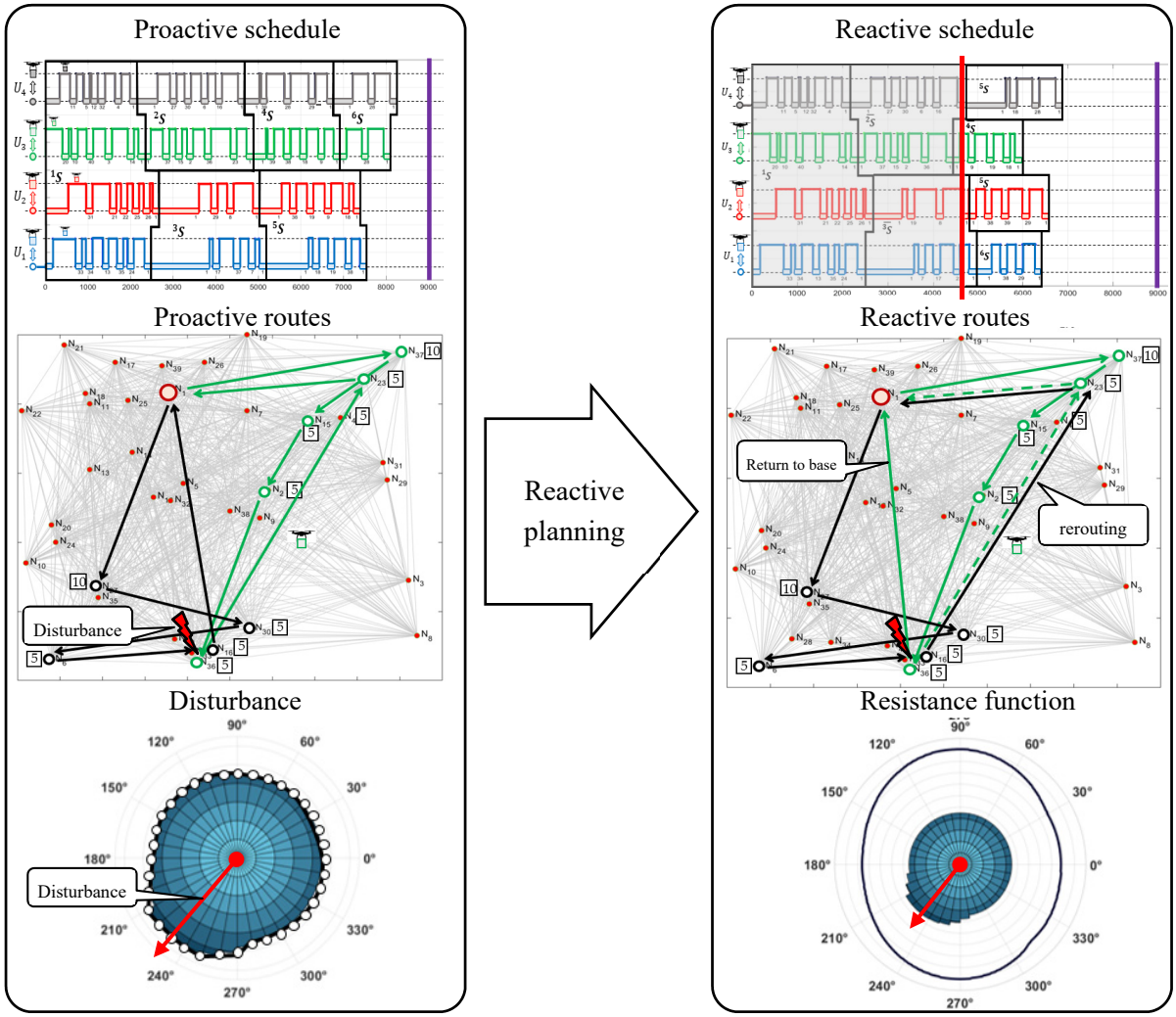


Figure 5. Reactive approach to UAV fleet mission planning.

The UAV mission planning process (Figure 6) therefore includes two stages. The first one

involves proactive planning, which results in a UAVs mission plan that guarantees timely deliveries in the forecasted weather conditions. The second stage is responsible for reactive planning undertaken in situations where during the implementation of the proactive plan there is a disruption exceeding previous predictions.

In that context, the proposed reaction (replanning) to the occurrence of a disturbance $IS(t^*)$ can be reduced to dynamic re-routing and rescheduling of previously adopted routes ${}^{l}\Pi$, schedules ${}^{l}Y$, and delivered goods ${}^{l}C$ is stated in the basic proactive plan for the mission by the UAV fleet. It is a feasible adjustment of the assumed ${}^{l}\Pi$, ${}^{l}Y$, and ${}^{l}C$ values to the changes in forecasted weather $\mathcal{F}^*(\theta)$, as well as corrections introduced to the network G^*or orders Z^*.

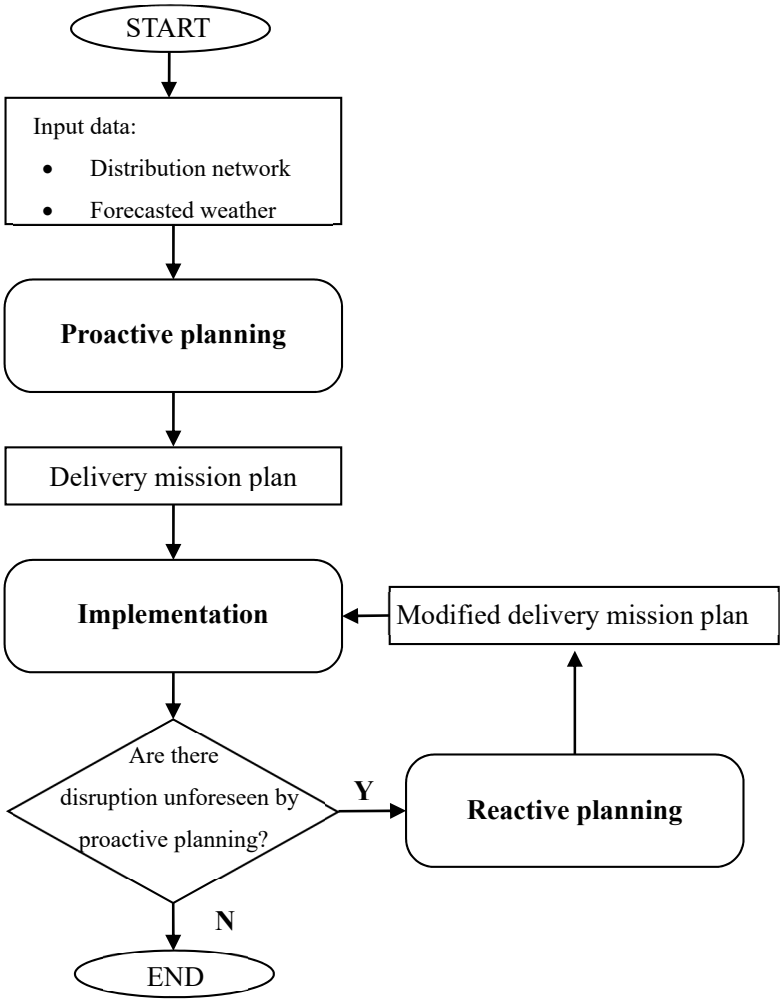


Figure 6. Proactive-reactive UAV fleet mission planning.

To formally define the concept of disturbance $IS(t^*)$, we will introduce the concept of the state of mission implementation S. The state of the mission S at the time t is defined as follows: $IS(t) = (M(t), \mathcal{F}^*(\theta, t), {}^*G(t), Z^*(t))$ where: $M(t)$ is an allocation of UAVs to nodes at the time t: $M(t) = \left(N_{a_1}, \ldots, N_{a_k}, \ldots, N_{a_K}\right)$, where: $a_k \in \{1, .., n\}$ determines the (delivery points) node N_{a_k} occupied by U_k (or the node the U_k is headed to). $\mathcal{F}^*(\theta, t)$ is the weather condition forecast at the time t. ${}^*G(t)$ is the graph model of the distribution network structure at time *t* (number and location of delivery points). $Z^*(t)$ is the sequence of goods requested at the time t. The state $IS(t^*)$

following condition $[\mathcal{F}^*(\theta,t^*)\neq\mathcal{F}^*(\theta)]\vee[^*G(t^*)\neq G]\vee[Z^*(t^*)\neq Z]$ is called the disturbance occurring at the time t^*.

Occurrence of $IS(t^*)$ disturbance should be assessed in terms of its impact on the further course of the mission of S (that is, whether the value of the resistance function $Y_{k,l}(\theta)$ is greater than $\mathcal{F}(\theta)$). If the implementation of the mission is at risk ($Y_{k,l}(\theta)\ngeq\mathcal{F}(\theta)$), an attempt should be made to reschedule it. The following condition action (if-then) rules are used for this purpose:

1) If the adopted mission plan S is not resistant to disturbance $IS(t^*)$ $(\exists_{k\in\{1,..,K\},l\in\{1,..,L\}}Y_{k,l}(\theta)\ngeq\mathcal{F}(\theta))$, then it should be checked whether it is possible to adapt (re-plan) it to adjust to new conditions, i.e., decide whether all UAVs in the air should continue their current missions or make the appropriate corrections.

2) If there are UAVs (in the set $\mathcal{U}$R) that cannot continue to fly due to disturbance $IS(t^*)$, then they should be returned to the base and allow, if possible airborne UAVs (the set $\mathcal{U}\backslash\mathcal{U}$R) to take over their tasks.

3) If the tasks of the UAVs returning to the base (the set $\mathcal{U}$R) cannot be taken over by UAVs still performing their missions, then it should be checked whether the reserve UAVs available at the base (the set $\mathcal{U}$B) can take over their responsibilities. This means the UAVs in the air continue their existing missions, while the reserve UAVs take over the liabilities of the UAVs returned to the base.

4) If the reserve UAVs (the set $\mathcal{U}$B) are unable to take over the responsibilities of those returned to the base (the set $\mathcal{U}$R), then their activity should be suspended until the disturbance is resolved.

The above rules have been used in the reactive mission planning method S. The idea behind this method is as follows. During the implementation of the mission S, there is a continuous monitoring of the state of the $IS(t)$ (for $t\in\{0\ldots H\}$). If at the state $IS(t)$ the following condition holds $[\mathcal{F}^*(\theta,t^*)\neq\mathcal{F}^*(\theta)]\vee[^*G(t^*)\neq G]\vee[Z^*(t^*)\neq Z]$ (i.e., there is a change in the weather forecast or in the structure of the distribution network as well as in the size of the requests, etc.) and the mission S is under threat (i.e., at least one of the UAVs will not return to base due to low battery), then an attempt is made to replan it. The purpose of replanning is to select a mission $^*S$ adapted to the new conditions caused by the disturbance $IS(t)$. In practice, it comes down to solving the relevant constraints optimization problem (COP) denoted as $CS\left({}^{\circ}\mathcal{U},S,IS(t)\right)$, where:${}^{\circ}\mathcal{U}$—defines the fleet designated by condition action rules 1–4. The relevant constraints distribution process follows the sequence where, firstly, an attempt is made to designate the mission $^*S$ for the fleet ${}^{①}\mathcal{U}=\mathcal{U}$ (according to rule 1). In the event of failure, an attempt is made to designate it for the fleet ${}^{②}\mathcal{U}=\mathcal{U}\backslash\mathcal{U}R$ (according to rule 2) and then for the fleet ${}^{③}\mathcal{U}=(\mathcal{U}\backslash\mathcal{U}R)\cup\mathcal{U}B$ (according to rule 3). If an admissible solution$^*S$ sitll does not exist, then the currently used mission plan should be modified (due to the $reduce$ function) in such a way that it removes the UAVs sub-missions which are not resistant to disturbance $IS(t)$ (according to rule 4).

It should be noted that the designation of mission $^*S$ is associated with the designation of routings ${}^l\Pi$, schedules ${}^lY$, and delivery sequences ${}^lC$ throughout the remaining time horizon $\{t\ldots H\}$. Due to the disturbance $IS(t^*)$ occurrence, the proposed reactive planning algorithm (implemented in the IBM ILOG environment) generates the end-to-end paths that modify previously planned routes, restoring the ability to implement the designated mission delivery plan.

The mathematical formulation of the COP $CS\left({}^{\circ}\mathcal{U},S,IS(t)\right)$ aimed at contingency planning of UAV mission design employs the following parameters, variables, sets, and constraints.

Parameters:

${}^lG$: graph of a distribution network for sub-mission ${}^lS$;

${}^l\mathcal{U}$: the subset of UAVs ${}^l\mathcal{U}\subseteq\mathcal{U}$ carrying out the sub-mission ${}^lS$;

z_λ: demand at node N_λ, $z_1 = 0$;
K: the size of the fleet of UAVs;
np_λ: consumer priority at the point N_λ, $np_1 = 0$;
$d_{\beta,\lambda}$: distance between N_β, N_λ;
A: the front-facing area of a UAV;
$t_{\beta,\lambda}$: travel time between N_β, N_λ;
C_D: the aerodynamic drag coefficient;
w: time spent on take-off and landing of a UAV;
ts: the time interval at which UAVs can take off from the base;
Q: maximum loading capacity;
ep: the empty weight of a UAV;
$IS(t)$: state of UAV mission;
D: an air density;
H: time horizon;
$\Upsilon_{k,l}(\theta)$: weather resistance function;
g: the gravitational acceleration;
$\mathcal{F}(\theta)$: forecasted wind speed;
b: the width of an UAV;
CAP: the energy capacity of an UAV;
$va_{\beta,\lambda}$: air speed between nodes N_β, N_λ;
$vg_{\beta,\lambda}$: ground speed between N_β, N_λ;
$\varphi_{\beta,\lambda}$: heading angle of vector $va_{\beta,\lambda}$;
$\vartheta_{\beta,\lambda}$: the course angle of vector $vg_{\beta,\lambda}$.

Decision variables:

$\overline{{}^{l}x^{k}_{\beta,\lambda}}$: the binary variable used to indicate if U_k travels between nodes N_β, N_λ, after the disturbance $IS(t^*)$ occurrence (during sub-mission ${}^{l}S$);

$\overline{{}^{l}x^{k}_{\beta,\lambda}} = \begin{cases} 1 & \text{if } U_k \text{ travels between nodes } \mathrm{N_\beta}, \mathrm{N_\lambda} \\ 0 & \text{otherwise.} \end{cases}$;

$\overline{{}^{l}y^{k}_{\lambda}}$: the time at which U_k arrives at node N_λ, after the disturbance $IS(t^*)$ occurrence (during sub-mission ${}^{l}S$);

$\overline{{}^{l}c^{k}_{\lambda}}$: the weight of freight delivered to node N_λ by U_k, after the disturbance $IS(t^*)$ occurrence (during sub-mission ${}^{l}S$);

$\overline{{}^{l}f^{k}_{\beta,\lambda}}$: the weight of freight carried between nodes N_β, N_λ by U_k, after the disturbance $IS(t^*)$ occurrence (during sub-mission ${}^{l}S$);

$\overline{{}^{l}P^{k}_{\beta,\lambda}}$: the energy per unit of time consumed by U_k during the flight between nodes N_β, N_λ (after the disturbance $IS(t^*)$ occurrence);

$\overline{{}^{l}bat^{k}}$: the total energy consumed by U_k, after the disturbance $IS(t^*)$ occurrence (during sub-mission ${}^{l}S$);

$\overline{{}^{l}s^{k}}$: the take-off time of U_k, after the disturbance $IS(t^*)$ occurrence;

$\overline{{}^{l}cp_{\lambda}}$: the total weight of freight delivered to node N_λ, after the disturbance $IS(t^*)$ occurrence (during sub-mission ${}^{l}S$);

$\overline{{}^{l}\pi_k}$: the route of U_k after the disturbance $IS(t^*)$ occurrence (during sub-mission ${}^{l}S$), $\overline{{}^{l}\pi_k} = (N_{k_1}, \dots, N_{k_i}, N_{k_{i+1}}, \dots, N_{k_\mu})$, $k_i \in \{1, .., n\}$, $\left(N_{k_i}, N_{k_{i+1}}\right) \in E$.

Sets:

$\overline{{}^{l}Y}$: is a sequence of moments $\overline{{}^{l}y_{\lambda}^{k}}$, schedule of the fleet ${}^{l}\mathcal{U}$ after the disturbance $IS(t^{*})$ occurrence;

$\overline{{}^{l}C}$: is a sequence of weights of delivered goods $\overline{{}^{l}c_{\lambda}^{k}}$;

$\overline{{}^{l}\Pi}$: the set of UAV routes $\overline{{}^{l}\pi_{k}}$;

$\overline{{}^{l}S}$: the plan of sub-mission after the disturbance $IS(t^{*})$ occurrence: $\overline{{}^{l}S} = \left({}^{l}\mathcal{U}, \overline{{}^{l}\Pi}, \overline{{}^{l}Y}, \overline{{}^{l}C}\right)$;

${}^{*}S$: the re-route plan of the mission ${}^{*}S = \left(\overline{{}^{1}S}, \ldots, \overline{{}^{l}S}, \ldots, \overline{{}^{L}S}\right)$.

Constraints limiting: routes, delivery of freight, and energy consumption:

1) **Routes.** Relationships between the variables describing UAV take-off times/mission start times and task order:

$$\overline{{}^{l}s^{k}} \geq 0; \;\; k = 1 \ldots K, \; l = 1 \ldots L, \tag{1}$$

$$\overline{{}^{l}y_{i}^{k}} \geq 0; \;\; i = 1 \ldots n; \;\; k = 1 \ldots K, \tag{2}$$

$$\overline{{}^{l}x_{i,i}^{k}} = 0; \;\; i = 1 \ldots n; \;\; k = 1 \ldots K. \tag{3}$$

$$\sum_{j=1}^{n} \overline{{}^{l}x_{i,j}^{k}} = 1 \;\; ; \;\; k = 1 \ldots K, \; l = 1 \ldots L, \tag{4}$$

$$\left({}^{l}s^{k} \leq t^{*}\right) \Rightarrow \left(\overline{{}^{l}s^{k}} = {}^{l}s^{k}\right); \;\; k = 1 \ldots K, \; l = 1 \ldots L, \tag{5}$$

$$\left(\;\; {}^{l}y_{j}^{k} \leq t^{*}\right) \Rightarrow \left(\overline{{}^{l}x_{i,j}^{k}} = {}^{l}x_{i,j}^{k}\right); \; j = 1 \ldots n; \;\; i = 2 \ldots n; \;\; k = 1 \ldots K, \; l = 1 \ldots L, \tag{6}$$

$$\left(\;\; {}^{l}y_{j}^{k} \leq t^{*}\right) \Rightarrow \left(\overline{{}^{l}y_{j}^{k}} = {}^{l}y_{j}^{k}\right); \; j = 1 \ldots n; \;\; i = 2 \ldots n; \;\; k = 1 \ldots K, \; l = 1 \ldots L, \tag{7}$$

$$\left(\left|\overline{{}^{l}s^{k}} - \overline{{}^{l}s^{q}}\right| \geq ts\right) \;\; ; k, q = 1 \ldots K; \;\; k \neq q, \; l = 1 \ldots L, \tag{8}$$

$$\left(\overline{{}^{l}y_{i}^{k}} \neq 0 \;\; \wedge \overline{{}^{l}y_{i}^{q}} \neq 0 \;\; \right) \Rightarrow \left(\left|\overline{{}^{l}y_{i}^{k}} - \overline{{}^{l}y_{i}^{q}}\right| \geq w\right) \;\; ; \;\; k, q = 1 \ldots K; \; k \neq q, \tag{9}$$

$$\left(\overline{{}^{l}x_{1,j}^{k}} = 1\right) \Rightarrow \left(\overline{{}^{l}y_{j}^{k}} = \overline{{}^{l}s^{k}} + t_{1,j}\right) \;\; ; \;\; j = 1 \ldots n; \;\; k = 1 \ldots K, \tag{10}$$

$$\left(\overline{{}^{l}x_{i,j}^{k}} = 1\right) \Rightarrow \left(\overline{{}^{l}y_{j}^{k}} = \overline{{}^{l}y_{i}^{k}} + t_{i,j} + w\right); \;\; j = 1 \ldots n; \;\; i = 2 \ldots n; \;\; k = 1 \ldots K, \tag{11}$$

$$\overline{{}^{l}y_{i}^{k}} \leq H \times \sum_{j=1}^{n} \overline{{}^{l}x_{i,j}^{k}}, i = 1 \ldots n; \;\; k = 1 \ldots K, \tag{12}$$

$$\sum_{j=1}^{n} \overline{{}^{l}x_{i,j}^{k}} = \;\; \sum_{j=1}^{n} \overline{{}^{l}x_{j,i}^{k}}; \;\; i = 1 \ldots n; \;\; k = 1 \ldots K, \tag{13}$$

Delivery of freight. Relationships between variables describing already delivered and requested amount of freight:

$$\left(\;\; \overline{{}^{l}y_{j}^{k}} \leq t^{*}\right) \Rightarrow \left(\overline{{}^{l}c_{j}^{k}} = {}^{l}c_{j}^{k}\right); \; j = 1 \ldots n; \;\; i = 2 \ldots n; \;\; k = 1 \ldots K; \;\; l = 1 \ldots L, \tag{14}$$

$$\overline{{}^{l}c_{i}^{k}} \geq 0; \;\; i = 1 \ldots n; \;\; k = 1 \ldots K; \; l = 1 \ldots L, \tag{15}$$

$$\overline{{}^{l}c_{i}^{k}} \leq Q \times \sum_{j=1}^{n} x_{i,j}^{k}; \;\; i = 1 \ldots n; \;\; k = 1 \ldots K, \; l = 1 \ldots L, \tag{16}$$

$$\sum_{i=1}^{n} \overline{{}^{l}c_i^k} \le Q;\ \ k = 1 \ldots K;\ \ l = 1 \ldots L, \tag{17}$$

$$\left(\overline{{}^{l}x_{i,j}^k} = 1\ \right) \Rightarrow \left(\overline{{}^{l}c_i^k} \ge 1\right);\ \ k = 1 \ldots K;\ \ i = 1 \ldots n;\ \ j = 2 \ldots n, \tag{18}$$

$$\sum_{l=1}^{L} \sum_{k=1}^{K} \overline{{}^{l}c_i^k} = z_i;\ \ i = 1 \ldots n, \tag{19}$$

$$\sum_{i=1}^{n} \overline{{}^{l}c_i^k} = \overline{{}^{l}cs^k};\ \ k = 1 \ldots K;\ \ l = 1 \ldots L, \tag{20}$$

$$\left(\overline{{}^{l}x_{i,j}^k} = 1\right) \Rightarrow \left(\overline{{}^{l}fc_j^k} = \overline{{}^{l}cs^k}\right)\ ; j = 1 \ldots n;\ \ k = 1 \ldots K,\ \ l = 1 \ldots L, \tag{21}$$

$$\left(\overline{{}^{l}x_{i,j}^k} = 1\right) \Rightarrow \left(\overline{{}^{l}fc_j^k} = \overline{fc_i^k} - \overline{{}^{l}c_i^k}\right)\ ; i,j = 1 \ldots n;\ \ k = 1 \ldots K,\ \ l = 1 \ldots L, \tag{22}$$

$$\left(\overline{{}^{l}x_{i,j}^k} = 1\right) \Rightarrow \left(\overline{{}^{l}f_{1,j}^k} = \overline{{}^{l}cs^k}\right)\ ; j = 1 \ldots n;\ \ k = 1 \ldots K,\ \ l = 1 \ldots L, \tag{23}$$

$$\left(\overline{{}^{l}x_{i,j}^k} = 1\right) \Rightarrow \left(\overline{{}^{l}f_{i,j}^k} = \overline{{}^{l}fc_j^k}\right)\ ;\ \ i,j = 1 \ldots n;\ \ k = 1 \ldots K,\ \ l = 1 \ldots L, \tag{24}$$

2) **Energy consumption**. To ensure waterproofness of the ${}^{l}S$ sub-mission (i.e., its robustness to weather condition changes $Z(\theta)$), the amount of energy required to complete the task carried out by an UAV must not exceed the capacity of its battery:

$$\Upsilon_{k,l}(\theta) \ge \mathcal{F}(\theta);\qquad \forall\theta \in [0°, 360°), \tag{25}$$

$$\Upsilon_{k,l}(\theta) = \max \Gamma_{k,l}(\theta), \tag{26}$$

$$\Gamma_{k,l}(\theta) = \left\{vw\ \mid\ vw \in R_+^0\ \ \wedge\ \ \forall_{k\in\{1\ldots K\}} \overline{{}^{l}bat^k}(\theta, vw) \le CAP\right\}, \tag{27}$$

$$\overline{{}^{l}bat^k}(\theta, vw) = \sum_{i=1}^{n} \sum_{j=1}^{n} \overline{{}^{l}x_{i,j}^k} \times t_{i,j} \times {}^{l}P_{i,j}^k(\theta, vw), \tag{28}$$

$${}^{l}P_{i,j}^k(\theta, vw) = \frac{1}{2} C_D \times A \times D \times \left({}^{l}va_{i,j}(\theta, vw)\right)^3 + \frac{\left(\left(ep + \overline{{}^{l}f_{i,j}^k}\right) \times \mathrm{g}\right)^2}{D \times b^2 \times {}^{l}va_{i,j}(\theta, vw)}, \tag{29}$$

where, ${}^{l}va_{i,j}(\theta, vw)$ and $t_{i,j}$ depend on the assumed goods delivery strategy.

If the ground speed $vg_{i,j}$ is constant, then an air speed ${}^{l}va_{i,j}$ is calculated from:

$${}^{l}va_{i,j}(\theta, vw) = \sqrt{\left(vg_{i,j} \times cos\vartheta_{i,j} - vw \times cos\theta\ \right)^2 + \left(vg_{i,j} \times sin\vartheta_{i,j} - vw \times sin\theta\ \right)^2} \tag{30}$$

$$t_{i,j} = d_{i,j} / vg_{i,j} \tag{31}$$

3) **The objective function**. A mission $\overline{{}^{l}S}$ plan to maximize customer satisfaction expressed by the following function is sought:

$$f_o\left(\overline{{}^{l}S}\ \right) = \ \ \sum_{i=1}^{n} \left(np_i \times \overline{{}^{l}cp_i}\right) \tag{32}$$

The adoption of such a function means that customers with the lowest level of expectations to be met are served first.

Constraints (1)–(13) describe the relationship between routes (represented by the variables: $\overline{{}^{l}x_{i,j}^k}$) and the delivery schedule (variables $\overline{{}^{l}y_j^k}$ and $\overline{{}^{l}s^k}$). Constraints (5)–(7) provide, that the plan before disturbance should be the same like proactive plan (represented by ${}^{l}s^k$, ${}^{l}x_{i,j}^k$, ${}^{l}y_j^k$). Other constrains provide, that it is not possible for multiple UAVs to take off from the base at the same time (8), a

delivery points cannot by simultaneously served by several UAVs (9), deliveries are made in accordance with the adopted route (10), (11) and on time (12) and it guarantees closed loops of the routes (13).

Constraints (14)–(24), link UAV routes ($\overline{{}^{l}x_{i,j}^{k}}$) to the amounts of delivered goods (variables $\overline{{}^{l}c_{j}^{k}}$). They also ensure that the UAVs are not overloaded (16), (17), and that correct amounts are delivered (19). Constraints (20)–(24) determine the weight ($f_{i,j}^{k}$) of the goods at each section of the taken route.

Constraints (25)–(31) describe the values of the determined weather resistance functions $\Upsilon_{k,l}(\theta)$ for the fleet ${}^{l}\mathcal{U}$ and ensure that these values exceed the value of the function $\mathcal{F}(\theta)$ (forecasted wind speed). The value of the $\Upsilon_{k,l}(\theta)$ function depend on the amount of energy consumed by a UAV in flight which in turn depends non-linearly on the speed value ${}^{l}va_{i,j}(\theta, vw)$. This means that some of the constraints, e.g. (29), (30), in the adopted model, have a non-linear character, thus implying the necessity to use the capabilities of declarative environments (in particular constraints programming).

Since the re-planning of the mission delivery plan S is the result of the disturbance $IS(t^*)$, the new set of sub-missions $\overline{{}^{1}S}, \ldots, \overline{{}^{l}S}, \ldots, \overline{{}^{L}S}$ guaranteeing timely delivery are determined by solving the following COP (33):

$$CS\left({}^{\circ}\mathcal{U}, S, IS(t^*)\right) = \left((\mathcal{V}, \mathcal{D}), \mathcal{C}\left({}^{\circ}\mathcal{U}, S, IS(t^*)\right), f_o\right), \tag{33}$$

where:

$\hat{\mathcal{V}} = \left\{\overline{{}^{l}\Pi}, \overline{{}^{l}Y}, \overline{{}^{l}C} | l = 1 \ldots L\right\}$—the set of decision variables: $\overline{{}^{l}\Pi}$—the set of routes determining the schedule $\overline{{}^{l}Y}$; $\overline{{}^{l}Y}$—schedule of the fleet ${}^{\circ}\mathcal{U}$ guarantees timely service of delivery points in the case of disturbance $IS(t^*)$; and $\overline{{}^{l}C}$—sequence of weights of delivered goods by the fleet ${}^{\circ}\mathcal{U}$;

$\mathcal{D}$—the finite set of decision variable domains: $\overline{{}^{l}x_{i,j}^{k}} \in \{0,1\}$, $\overline{{}^{l}y_{\lambda}^{k}} \in \mathbb{N}$, $\overline{{}^{l}c_{i}^{k}} \in \mathbb{N}$;

$\hat{\mathcal{C}}$—the set of constraints that takes into account the set of routes $\overline{{}^{l}\Pi}$, schedules $\overline{{}^{l}Y}$, and the disturbance $IS(t^*)$, while determining the relationships linking the operations executed by UAVs (1)–(31);

f_o—the objective function defined by (32).

To solve COP (33), the values of the decision variables from the adopted set of domains for which the given constraints are satisfied and guarantee the maximum value of objective function must be determined.

5. Constraint relaxation

The previous section demonstrated that solving the COP (33) problem enables the designation of an ${}^{*}S$ mission resistant to $IS(t^*)$ disturbances caused by changing weather conditions specified by $\mathcal{F}(\theta)$. Solving this problem, however, is very time-consuming, which results from the necessity to verify the inequality (25) of the wind direction change $\Upsilon_{k,l}(\theta) \geq \mathcal{F}(\theta)$ carried out for each increment $\theta \in [0°, 360°)$. To replace condition (25) with an equivalent that is less time consuming computationally, it was assumed that the continuous (smooth) $\Upsilon_{k,l}(\theta)$ function will be approximated by a discrete function represented by a finite set $\mathbb{Y}_{k,l} = \left\{\Upsilon_{k,l}(\theta_i) | i = 1 \ldots lq; \theta_i < \theta_{i+1}\right\}$, where lq is an arbitrarily taken number of samples. This set contains the vertices of a polygon $\mathbb{Y}_{k,l}^{*}(\theta)$ depicted in Figure 7.

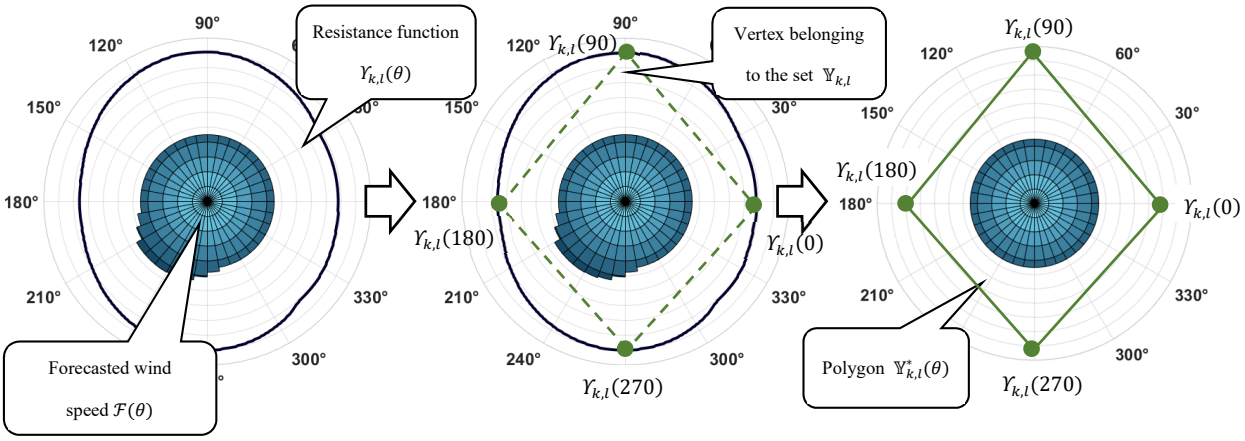


Figure 7. Discretization of the function $\Upsilon_{k,l}(\theta)$.

For such assumptions, the following property holds:

Property 1: For any wind direction $\theta \in [0°, 360°)$ *function of* $\Upsilon_{k,l}(\theta)$ *mission,* ${}^*S$ *takes values no less than its discrete form* $\mathbb{Y}^*_{k,l}(\theta)$: $\forall\theta \in [0°, 360°)$, $\Upsilon_{k,l}(\theta) \geq \mathbb{Y}^*_{k,l}(\theta)$.

This means that the function $\Upsilon_{k,l}(\theta)$ can be replaced by the function $\mathbb{Y}^*_{k,l}(\theta)$ and, in fact, the set of vertices in $\mathbb{Y}_{k,l}$. In practice, it means reducing the quantity of condition (25) checks to the number of vertices of the adopted polygon. Consequently, constraint (25) can be replaced by the following one:

$$\mathbb{Y}^*_{k,l}(\theta) \geq \mathcal{F}(\theta); \quad \theta \in \{\theta_1, \ldots, \theta_{lq}\} \tag{34}$$

6. Evolutionary algorithm

The introduction of constraints relaxation (34) of the considered problem (33) reduces the computational effort but not adequately to solve problems of the scale of those occurring in practice. This means that in online mode, problems of the type (33) involve networks not exceeding 100 nodes. To speed up the calculations, a genetic algorithm dedicated to this problem has been developed.

For the purposes of the developed algorithm, it is assumed that sub-mission $\overline{{}^lS}$ is represented by chromosons $\overline{{}^lCH}$ describing deliveries made by the UAVs of fleet ${}^l\mathcal{U}$:

$$\overline{{}^lCH} = \left(\overline{{}^l\pi_1}, \overline{{}^lY_1}, \overline{{}^lC_1}, \ldots, \overline{{}^l\pi_k}, \overline{{}^lY_k}, \overline{{}^lC_k}, \ldots, \overline{{}^l\pi_K}, \overline{{}^lY_K}, \overline{{}^lC_K}\right) \tag{35}$$

In other words, $\overline{{}^lCH}$ consists of the routes $\overline{{}^l\pi_k}$ $(k = 1, \ldots, K)$; schedules $\overline{{}^lY_k}$ $(k = 1, \ldots, K)$; and volumes of deliveries $\overline{{}^lC_k}$ $(k = 1, \ldots, K)$ describing the sub-mission $\overline{{}^lS}$.

The framework of the proposed genetic algorithm implementing such chromosomes is presented in Fig. 8. According to the algorithm flowchart, its input data are: $S = \left({}^1S, \ldots, {}^lS, \ldots, {}^LS\right)$—the proactive mission plan, $IS(t^*)$—the disturbance occurring at the moment t^*, ${}^lS$—the submission affected by the disturbance, ${}^l\mathcal{U}$—the fleet of UAVs available in reactive mode (while following rules 1–4).

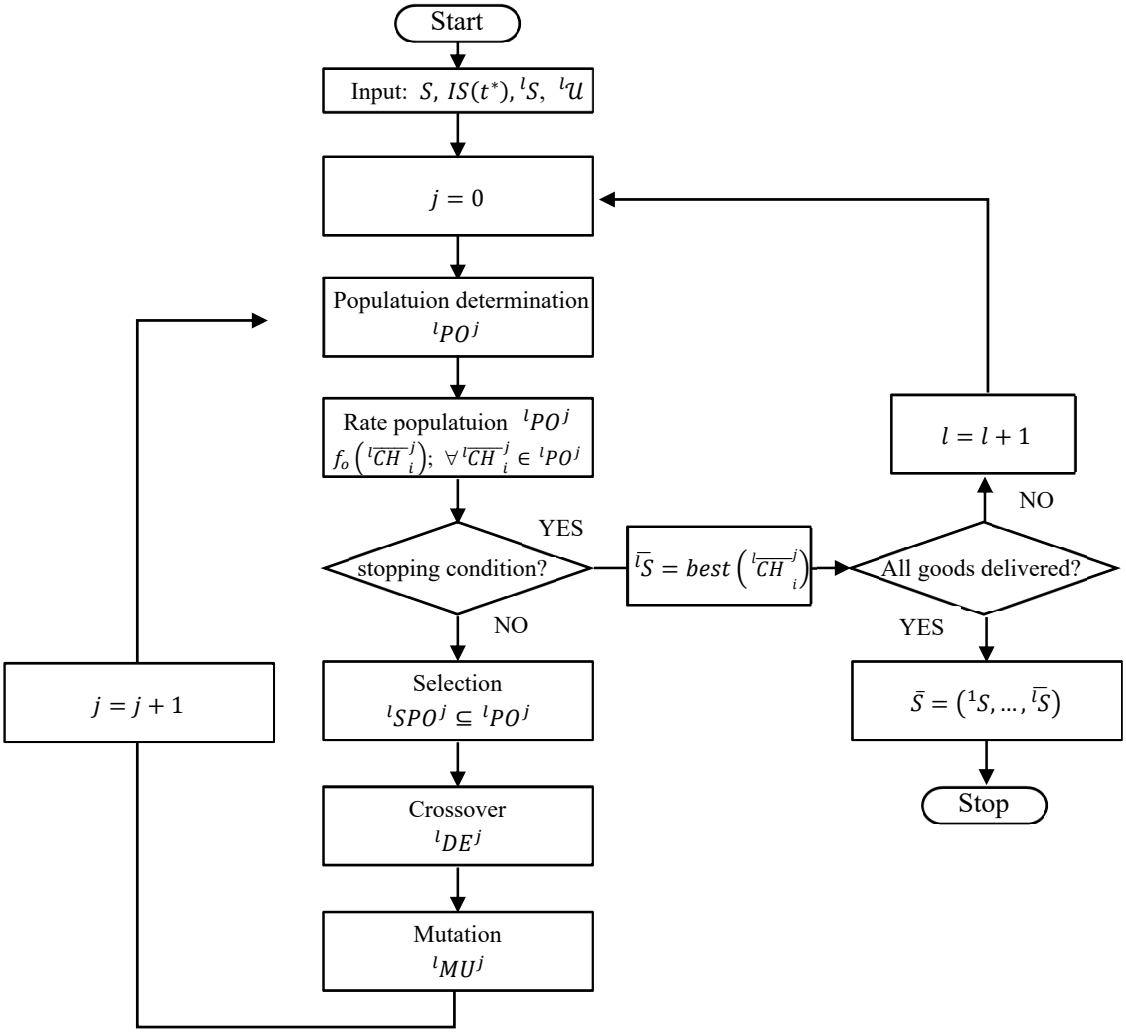


Figure 8. Framework of the proposed genetic algorithm.

The next stages of the algorithm include:

1) Determination of the *j-th* population ${}^{l}PO^{j}$ represented by the set of chromosomes describing different ways to accomplish a sub-mission ${}^{l}S$: ${}^{l}PO^{j} = \left\{ {}^{l}\overline{CH}_i^{\,j} | i = 1 \ldots LP \right\}$, where ${}^{l}\overline{CH}_i^{\,j}$—is the *i-th* chromosome from *j-th* population following submission ${}^{l}S$ (defined according to (35)); LP—population size. Individuals of the initial population ($j = 0$) are selected at random from among the so-called "alive individuals"—solutions that meet the constraints (1)–(31). Individuals of population ${}^{l}PO^{j}$ for $j > 0$ are selected from the set ${}^{l}PO^{j-1} \cup {}^{l}DE^{j-1} \cup {}^{l}MU^{j-1}$ (LP best individuals population).

2) Population growth rate. For each chromosome ${}^{l}\overline{CH}_i^{\,j} \in {}^{l}PO^{j}$, the value of the objective function is determined $f_o\left({}^{l}\overline{CH}_i^{\,j}\right)$.

3) Selection. From among the individuals of the population ${}^{l}PO^{j}$, a subset ${}^{l}SPO^{j} \subseteq {}^{l}PO^{j}$ is drawn (it is assumed that that the parameter ps determines the probability of the event that ${}^{l}\overline{CH}_i^{\,j}$ belongs to the set ${}^{l}SPO^{j}$: $ps = P\left({}^{l}\overline{CH}_i^{\,j} \in {}^{l}SPO^{j}\right)$).

4) Crossover. Subsequent pairs of the set ${}^{l}SPO^{j}$ are crossed with each other. The result of the

operation of crossing two chromosomes ${}^{l}\overline{CH}{}_{u}^{j}$ and ${}^{l}\overline{CH}{}_{v}^{j}$ is a pair of chromosomes ${}^{l}\dot{\overline{CH}}{}_{u}^{J}$ and ${}^{l}\dot{\overline{CH}}{}_{v}^{J}$, in which one (randomly selected) k-th element of the route $\overline{{}^{l}\pi_{u,k}^{J}}$ is replaced by another selected at random k-th element of the route $\overline{{}^{l}\pi_{v,k}^{J}}$. The idea of crossing operation is illustrated by Figure 9. It should be noted that along with the replacement of route elements, the corresponding elements of the schedule $\overline{{}^{l}Y_{u,k}^{J}}$ and delivery ${}^{l}C_{u,k}^{j}$ sequences are replaced. The result of the crossing process is a set of descendants ${}^{l}DE^{j}$ of the population ${}^{l}PO^{j}$ (only alive individuals are included in the set ${}^{l}DE^{j}$).

5) Mutation. From among the elements of the set ${}^{l}PO^{j} \cup {}^{l}DE^{j}$, a set of individuals undergoing mutation is selected at random (with a probability of pm). Operation of the chromosome ${}^{l}\overline{CH}{}_{u}^{j}$ mutation consists of a random change of one (randomly selected) element of the k route $\overline{{}^{l}\pi_{u,k}^{J}}$. As a result of this operation, a set of mutated individuals ${}^{l}MU^{j}$ of population ${}^{l}PO^{j}$ is formed.

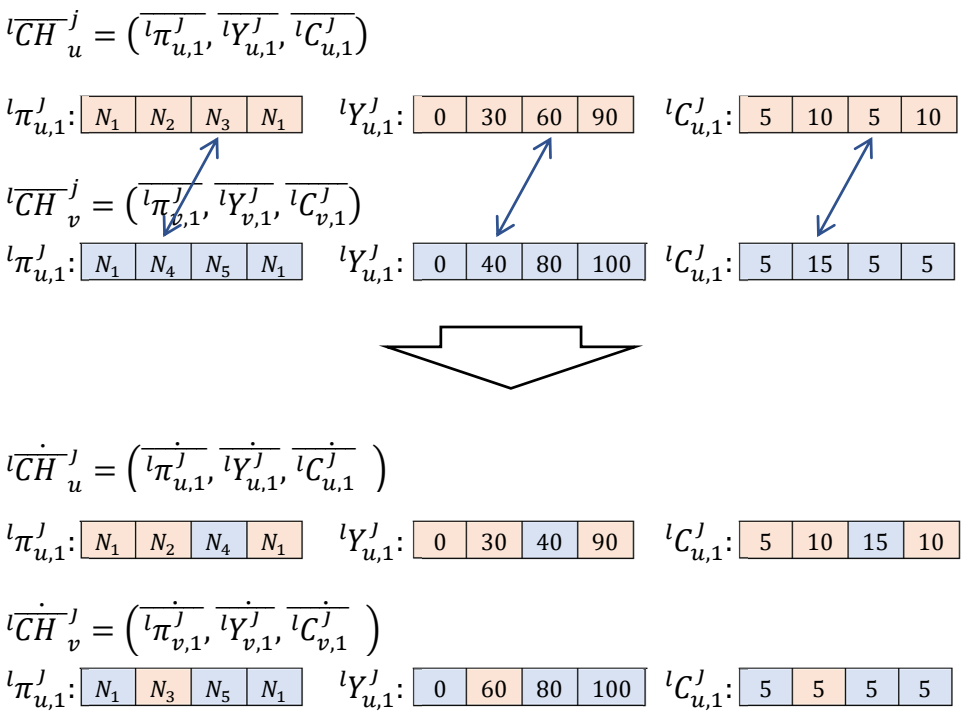


Figure 9. The idea of the chromosomes ${}^{l}\overline{CH}{}_{u}^{j}$ and ${}^{l}\overline{CH}{}_{v}^{j}$ crossing operation.

According to the presented algorithm, operations used for Selection, Crossover, and Mutation of individuals of the population ${}^{l}PO^{j}$ are repeated until the stopping condition (that can be seen as a stopping rule or simply stopping criteria) is satisfied. In the considered version of the algorithm, the following types of the stopping condition are distinguished:

- Condition 1 (C1): value of the objective function of the best individual of the population $\max_{i=1\ldots LP}\left\{f_o\left({}^{l}\overline{CH}{}_{i}^{j}\right)\right\}$ has reached the expected value RV: $\max_{i=1\ldots LP}\left\{f_o\left({}^{l}\overline{CH}{}_{i}^{j}\right)\right\} = RV$;
- Condition 2 (C2): value of the objective function of the best individual of the population $\max_{i=1\ldots LP}\left\{f_o\left({}^{l}\overline{CH}{}_{i}^{j}\right)\right\}$ has not changed in q generations: $\max_{i=1\ldots LP}\left\{f_o\left({}^{l}\overline{CH}{}_{i}^{j}\right)\right\} =$

$\max_{i=1\ldots LP}\left\{f_o\left({}^{l}\overline{CH}_i^{j-1}\right)\right\} = \cdots = \max_{i=1\ldots LP}\left\{f_o\left({}^{l}\overline{CH}_i^{j-q}\right)\right\}$;

- Condition 3 (C3): the number of generations has reached the limit value le: $j = le$.

Satisfaction of the stopping condition allows you to determine the next submissions ${}^{l}\overline{S}$ (where ${}^{l}\overline{S} = best\left({}^{l}\overline{CH}_i^{j}\right)$) constituting the reactive UAV fleet mission plan $\bar{S} = \left({}^{1}S, \ldots, {}^{l-1}S, {}^{l}\overline{S}, \ldots, {}^{L}\overline{S}\right)$.

7. Computational experiments

We consider the network from Figure 1, in which the four UAVs $\mathcal{U} = \{U_1, U_2, U_3, U_4\}$ service delivery points $N_2 - N_{40}$. The structure of the implementation of subsequent sub-missions ${}^{1}S, {}^{2}S, \ldots, {}^{6}S$ of the adopted mission plan S is presented in Figure 2.

Let us consider a situation related to the appearance of a disturbance IS(3000), specified in Section 2, where at the time $t^* = 3000$ [s], during the execution of the sub-mission ${}^{2}S$, the wind speed increased to $vw = 11\ m/s$ with the same wind intensity and direction $\theta = 210° - 230°$. With such a change in weather conditions, the implementation of the adopted plan turns out to be impossible. It becomes necessary to re-plan the implemented mission, including the introduction of the new sub-missions ${}^{2}\overline{S}, {}^{3}\overline{S}, {}^{4}\overline{S}, {}^{5}\overline{S}, {}^{6}\overline{S}$ to correct its course. For this purpose, the considered problem has been modeled in COP (33) formalism. The Intel Core i7-M4800MQ 2.7 GHz, 32 GB RAM has been used to carry out the necessary calculations, employing:

1) The constraint programming environment IBM ILOG,
2) The genetic algorithm (shown on the Figure 8) implemented in Matlab environment.

Ad. 1. Implementation of considered problem in the constraint programming environment IBM ILOG has shown that the solution time for problems of size considered does not exceed 25 s. Figure 10 demonstrates the mission ${}^{*}S$ schedule being adapted to the weather conditions caused by the disturbance IS(3000). Rule 2, i.e., *If there are UAVs that (within the set $\mathcal{UR}$) cannot continue to fly due to disturbance $IS(t^*)$, then they should be returned to the base to allow, if possible, airborne UAVs (the set $\mathcal{U}\backslash\mathcal{UR}$) to take over their tasks*, was used in the selection of mission ${}^{*}S$.

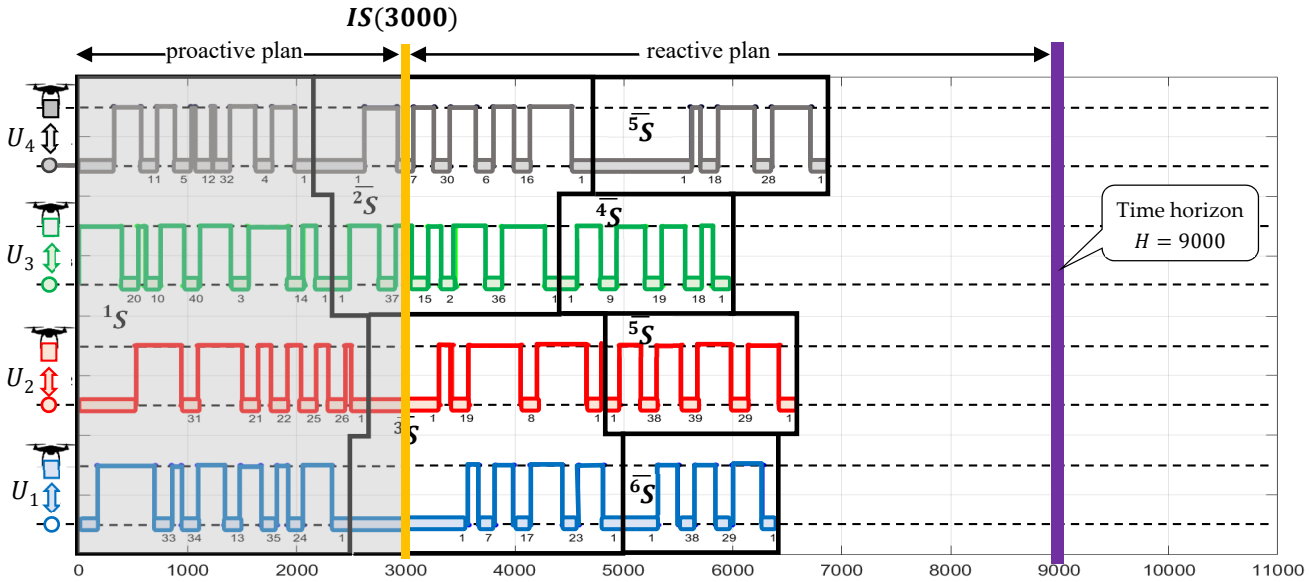


Figure 10. Change in the mission plan as per rule 2 (solution obtained from IBM ILOG).

Consequently, due to disturbance IS(3000) increasing the risk of premature battery depletion, a decision to turn U_3 back to the base was made (see sub-mission $\overline{{}^{2}S}$). At the same time U_4 continued its mission unchanged. The undertaken decision forced the necessity to reschedule subsequent sub-missions $\overline{{}^{3}S}, \overline{{}^{4}S}, \overline{{}^{5}S}, \overline{{}^{6}S}$, creating a new alternative mission plan ${}^{*}S$. The mission implementing such a modified plan will end at 6900 s.

Ad. 2. Solving the problem under consideration using the genetic algorithm from Figure 8 took 2 s. Figure 11 shows the schedule of obtained mission ${}^{*}S$. To determine a proactive plan, as before, rule 2 was used. According to this rule U_3 is to turn back to the base. The further course of the mission is carried out in 8 sub-missions $\overline{{}^{2}S}, \overline{{}^{3}S}, \overline{{}^{4}S}, \overline{{}^{5}S}, \overline{{}^{6}S}, \overline{{}^{7}S}, \overline{{}^{8}S}, \overline{{}^{9}S}$. It is apparent that the obtained solution is associated with a longer mission time (8500s).

This example demonstrates that the constraint programming environment allows us to obtain solutions of better quality (mission completion time is reduced by 19%) but at the expense of increased calculation time (more than 10 times longer solution determination time).

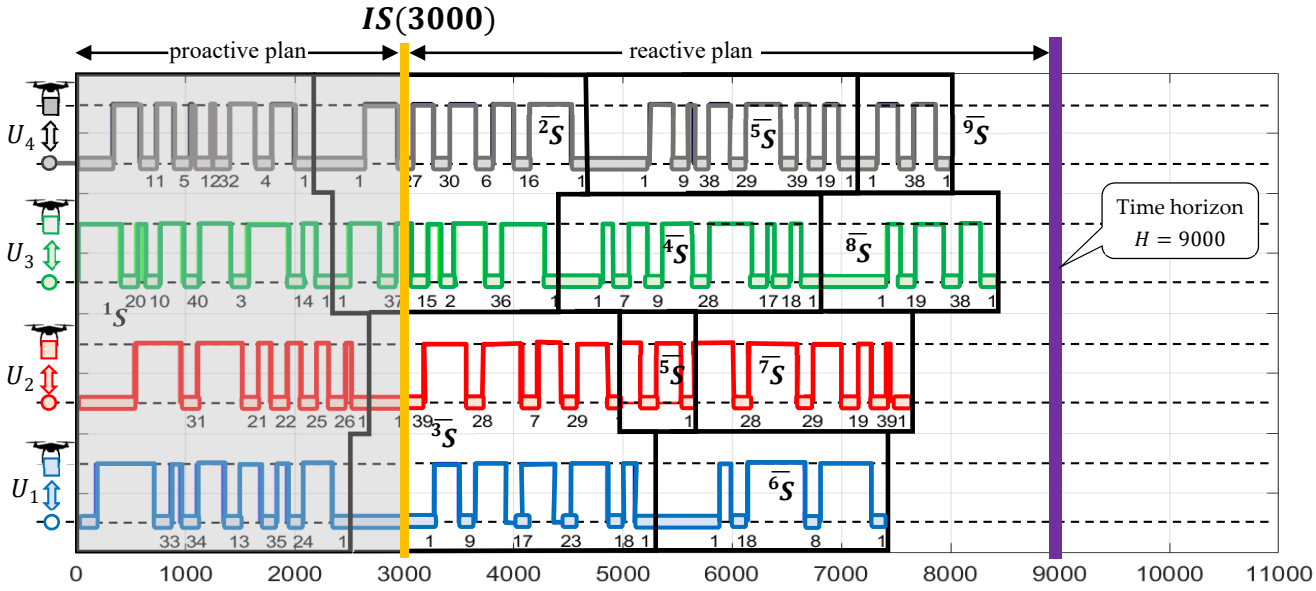


Figure 11. Change in the mission plan as per rule 2.

To assess the scalability of the proposed approach in terms of the possibility of its use in an online mode (i.e., to solve the problem in < 600 s) in decision support systems, a series of quantitative experiments have been carried out. Tables A1–A3 (Appendix A) contain the results of the experiments that were conducted for the three functions of forecasted weather $\mathcal{F}\ (\theta) = 9, 10, 11\ m/s$. The experiments were carried out for the network of $n = 40,60, \ldots, 220$ randomly designated delivery points (on an area of 10 km × 10 km) and a fleet consisting of $K = 2,3,4$ UAVs within the technical parameters, as shown in the table from Figure 1c). For each of the considered variants of the network, a proactive mission plan has been set out to guarantee the deliveries in the time horizon $H = 10{,}000$ s. It was assumed that at the moment $t^* =$ 2000 s, there is a change in the weather forecast (disturbance IS (2000)) that lasts until the end of the considered time horizon. The change in weather involves increasing the expected wind speed by 2m/s and equals accordingly $\mathcal{F}^*(\theta)$=11, 12, 13 m/s. The UAVs route planning is aimed at the reactive performance of delivery missions in a given time

horizon, under expected weather conditions $\mathcal{F}^*(\theta)$.

In the conducted experiments, two approaches were compared:

1) The constraint programming environment IBM ILOG CPLEX implemented the following stopping condition:

C0. The optimal value of the objective function has been reached or the time $TC = 600s$ allotted for calculations has elapsed.

2) The Genetic Algorithm (GA) shown in Figure 8 specified by: $LP = 1000$ (population size); $ps = 0.25$ (probability of selection); $pm = 0.01$ (probability of mutation), and implemented in the Matlab environment using three distinct stopping conditions:

C1. value of the objective function of the best individual of the population has reached the value of solution obtained from the IBM ILOG CPLEX environment (see condition C0);

C2. value of the objective function of the best individual of the population has not changed through $q = 10$ generations;

C3. the number of generations has reached the limit value le: $j = 200$.

The results (i.e., the times with relaxation TC determining the reactive mission plan design) are presented in Figures 12–15 (and Tables A1–A3). Figure 12 compares the time expenditure incurred in determining the reactive plan in the ILOG environment and the developed GA (implementing the stopping condition C1). In the GA variant under consideration, it is assumed that the calculation is stopped when the value of the objective function specified by the solution obtained from the ILOG environment is reached. Calculations were carried out for weather conditions $\mathcal{F}(\theta) = 9\frac{m}{s}, \forall\Theta \in [0°, 360°)$. Disruption (for which a reactive plan is sought) consists of changing the wind speed to $\mathcal{F}^*(\theta) = 11\ \frac{m}{s}$.

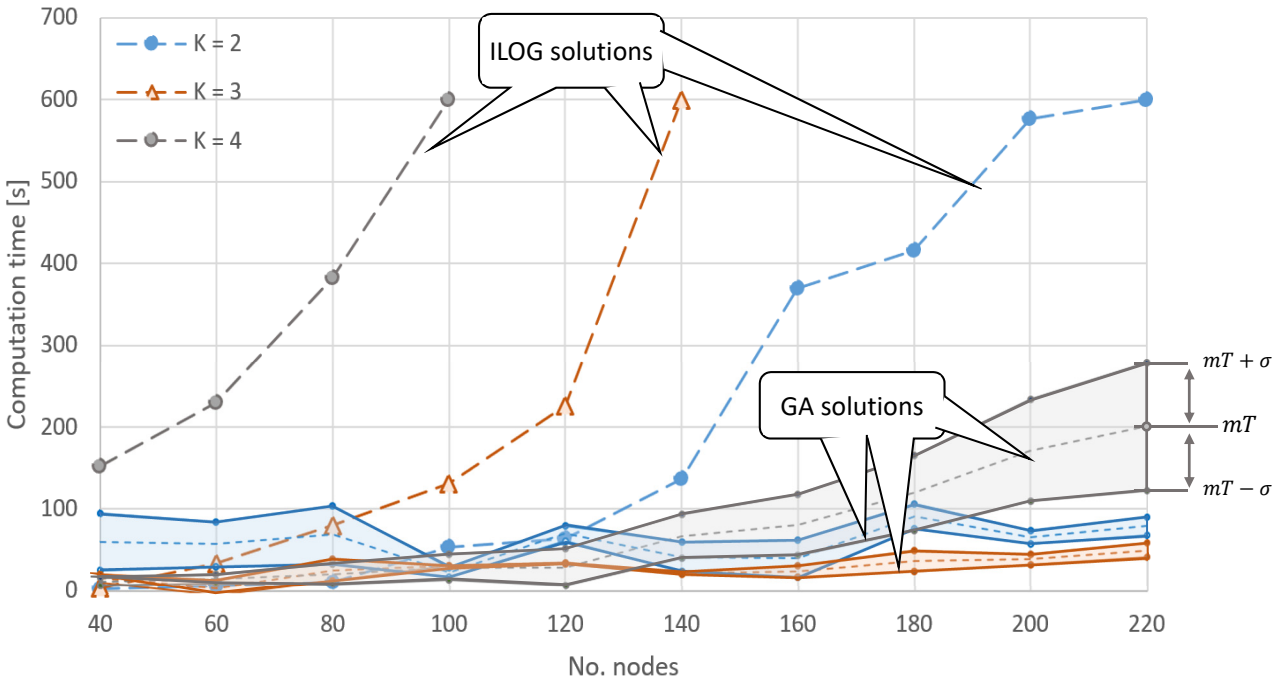


Figure 12. Comparison of computation times for the ILOG environment and GA (for stopping condition C1) and weather conditions $\mathcal{F}(\theta) = 9\frac{m}{s}$, $\forall\Theta \in [0°, 360°)$ (see Table A1).

It is evident that determining the optimal solution (maximizing objective function (32)) in the ILOG environment (dashed lines in Figure 12) requires significant computational effort. The scale of problems (number of network nodes n) for which it is possible to determine a solution on-line (< 600 s) decreases with the size of the UAV fleet and amounts to: $n = 220$ for $k = 2$; $n = 140$ for $k = 3$; and $n = 100$ for $k = 4$. Determining reactive plans for the set values of the objective function (obtained from the ILOG environment) using the implemented GA requires, in turn, a much lower computation time. As such, the considered scale of the network for all solutions were received in a time below 300 s—see ribbon-like lines presented means (mT) and standard deviation ($mT \pm \sigma$) of time computations. Such advantages of the GA results from the assumption that the solution (mission plan) was sought at the set (optimal) value of the objective function. Subsequent experiments were carried out on the assumption that the value of the objective function is unknown (stopping conditions C2 and C3 are used).

Figure 13 shows the results of experiments in which the stopping conditions C2 (Figure 13a) and C3 (Figure 13b) were implemented in the GA. In the first case, as before, the solution designation times for the GA do not exceed 400 s for the network size $n \leq 220$.

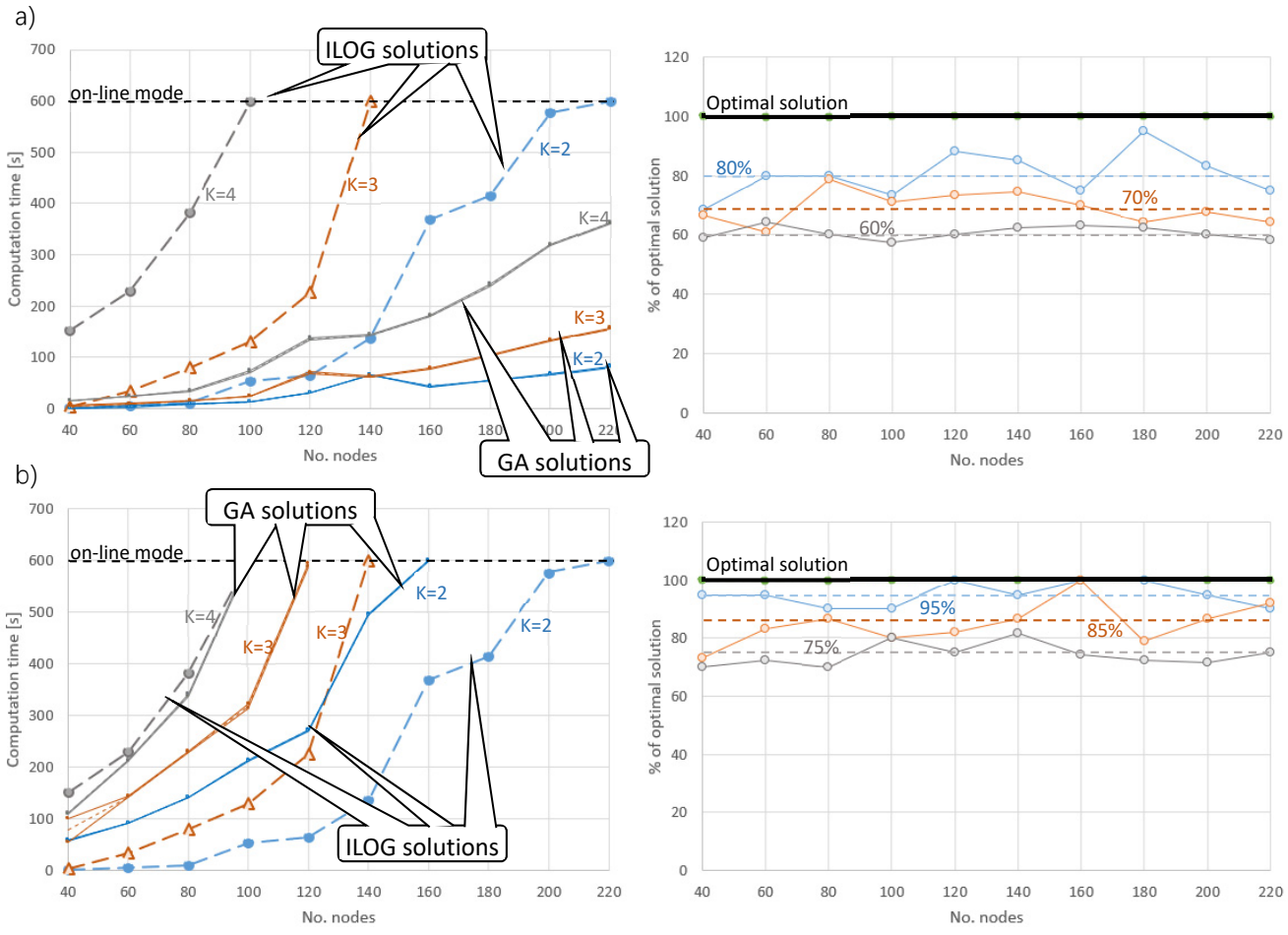


Figure 13. Comparison of the results of experiments carried out for the ILOG and GA (for stopping conditions C2 a) and C3 b)) under weather conditions $\mathcal{F}(\theta) = 9\frac{m}{s}, \forall \Theta \in [0°, 360°)$ (see Appendix A-Table A1).

Shorter computation times, however, came at the cost of lower "quality" solutions. This is

illustrated in Figure 13, which shows the changes in the value of the objective function for the solutions obtained. These values are expressed in % of optimal solutions obtained from the ILOG environment (e.g., a value of 70% means that the solution obtained from the GA has a value of the objective function equal to 0.7 of the result obtained from the ILOG environment). Mission plans determined using the GA are characterized by a lower value of the objective function than the solutions obtained in the ILOG environment, representing respectively: $\approx 80\%$ (for $k = 2$); $\approx 70\%$ (for $k = 3$); and $\approx 60\%$ (for $k = 4$) values obtained when using the ILOG environment.

In the second case (condition C3—see Figure 13b), the mission plans determined using the GA are characterized by a higher value of the objective function: $\approx 95\%$ (for $k = 2$); $\approx 85\%$ (for $k = 3$); and $\approx 75\%$ (for $k = 4$). However, obtaining such values requires significant computation times, which in the considered case are greater than in the ILOG environment (see Figure 13). It is also worth noting that the calculation times in both analyzed cases (C2 and C3) are characterized by a very low standard deviation value ($\sigma < 2$ for C2 and $\sigma < 25$ for C3; see Table A1), which means that despite the randomly generated population, the solution is usually obtained at the same time.

The presented experiments demonstrate that the planning of reactive missions using accurate methods (e.g., in the ILOG environment) is limited to small-scale networks (up to 100–220 nodes). This range can be increased using the proposed GA (implementing stopping condition C2); the solution is obtained on-line (<600 s). However, this result coincides with a lower value of the target function (e.g., at the level of 60%–80% of optimal solutions).

In further experiments, the influence of weather conditions on the time and quality of the solutions obtained was examined. Figures 14 and 15 show the results of experiments carried out for weather conditions $\mathcal{F}(\theta) = 10\frac{m}{s}$ and $\mathcal{F}(\theta) = 11\frac{m}{s}$, in which a disturbance in the form of changes in wind speed $\mathcal{F}^*(\theta) = 11\ \frac{m}{s}$ oraz $\mathcal{F}^*(\theta) = 12\ \frac{m}{s}$, respectively, was considered. The results of the experiments are collected in Tables A2 and A3 (see Appendix A).

Figures 14 and 15 compare the solutions obtained in the ILOG and GA environments with the stopping condition C2. It is evident that the change in weather conditions affects the time it takes for the ILOG environment to obtain a solution:

- for $\mathcal{F}(\theta) = 10\frac{m}{s}$, the scale of problems for which it is possible to determine a solution on-line (< 600 s) amounts to: $n = 200$ for $k = 2$; $n = 120$ for $k = 3$ and $n = 80$ for $k = 4$;
- for $\mathcal{F}(\theta) = 11\frac{m}{s}$, the scale of problems for which it is possible to determine a solution on-line (< 600 s) amounts to: $n = 200$ for $k = 2$; $n = 100$ for $k = 3$, $n = 80$ for $k = 4$.

It is also apparent that the change in weather conditions has increased the GA calculation time to 500 s. However, the change in weather conditions does not affect the quality of the solutions obtained. The value of the target function for the missions set by the GA remains at a similar level as before:

- for $\mathcal{F}(\theta) = 10\frac{m}{s}$, the value of the objective function is equal to: $\approx 78\%$ (for $k = 2$); $\approx 68\%$ (for $k = 3$); $\approx 60\%$ (for $k = 4$);
- for $\mathcal{F}(\theta) = 11\frac{m}{s}$, the value of the objective function is equal to: $\approx 75\%$ (for $k = 2$); $\approx 65\%$ (for $k = 3$); $\approx 60\%$ (for $k = 4$).

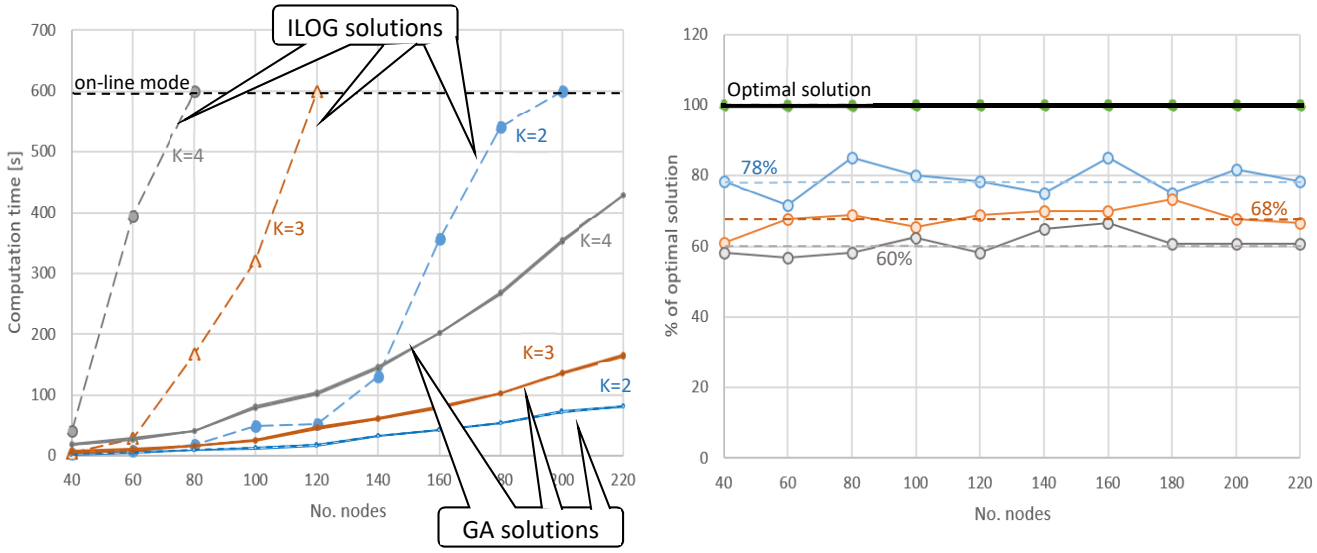


Figure 14. Comparison of the results of experiments carried out for the ILOG and GA (for stopping condition C2) under weather conditions $\mathcal{F}(\theta) = 10\frac{m}{s}$, $\forall\Theta \in [0°, 360°)$ (see Appendix A-Table 2).

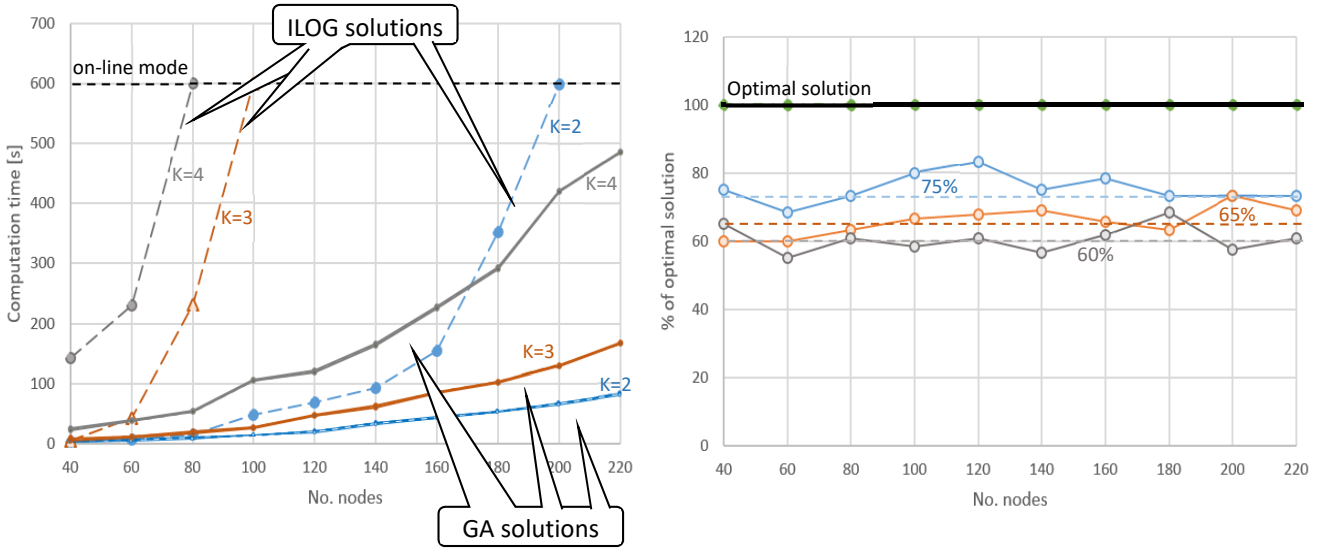


Figure 15. Comparison of the results of experiments carried out for the ILOG and GA (for stopping condition C2) under weather conditions $\mathcal{F}(\theta) = 11\frac{m}{s}$, $\forall\Theta \in [0°, 360°)$ (see Appendix A-Table A3).

The conducted experiments indicate that the greatest impact on the time of determining the solution (reactive mission) is primarily the number of nodes n of the network and the size K of the UAV fleet.

8. Conclusions

We propose a reactive routing method to solve the problem of UAV fleet mission contingency planning in a dynamically changing environment. We have considered plans for UAV route missions in the event that the weather changes beyond the previously predicted situation and/or that the previously agreed order fulfillment terms change. The need to react in such situations enforces the establishment of condition-action rules that allow for the designation of appropriate possible end-to-end routes and enabling safe completion of the mission or its continuation in a modified version during an emergency scenario.

The main advantage of the proposed model is its open structure, which allows for taking into account several variables and constraints, particularly the conditions, enabling a significant acceleration of calculations related to the variants of UAV routes caused by changes in weather conditions.

The developed model was implemented in the IBM ILOG declarative programming environment and the author's Genetic Algorithm. Computational results show that the proposed approach is suitable for online applications. The best results (i.e., the shortest time to determine the solution) were obtained for the GA in which the C2 stopping condition was applied (the value of the objective function has not changed through q generations). Experiments have shown that the scale of problems for which on-line, reactive UAV missions can be determined includes networks containing $n < 220$ nodes and fleets $K \leq 4$ UAVs.

In the general case, however, it should be noted that the aforementioned advantage of heuristic methods is paid for by the lack of guarantee that the trajectories of the UAVs motion determined with their help are collision-free. This disadvantage does not have an exact method that implements CP techniques, so it allows planning UAVs routes carrying out their missions at the same flight ceiling.

In our future research, we want to take into account the uncertain nature of the real-world variables which are not deterministic. Thus, a fuzzy approach could be applied to the UAV mission planning problem while allowing for a more accurate estimation of the timeliness of deliveries.

Conflict of interest

The authors declare there is no conflict of interest.

References

1. K. Dorling, J. Heinrichs, G. G. Messier, S. Magierowski, Vehicle routing problems for drone delivery, *IEEE Trans. Syst. Man Cybern. Syst.*, **47** (2017), 70–85. https://doi.org/10.1109/TSMC.2016.2582745
2. J. J. Enright, E. Frazzoli, M. Pavone, S. Ketan, UAV routing and coordination in stochastic, dynamic environments, in *Handbook of Unmanned Aerial Vehicles* (eds. K. P. Valavanis and G. J. Vachtsevanos), Springer, Dordrecht, (2015), 2079–2109. https://doi.org/10.1007/978-90-481-9707-1_28
3. Y. Khosiawan, A. Khalfay, I. Nielsen, Scheduling unmanned aerial vehicle and automated guided vehicle operations in an indoor manufacturing environment using differential evolution-fused particle swarm optimization, *Int. J. Adv. Rob. Syst.*, **15** (2018). https://doi.org/10.1177/1729881417754145

4. S. M. Patella, G. Grazieschi, V. Gatta, E. Marcucci, S. Carrese, The adoption of green vehicles in last mile logistics: a systematic review, *Sustainability*, **13** (2021), 6. https://doi.org/10.3390/su13010006
5. I. Sung, P. Nielsen, Zoning a service area of unmanned aerial vehicles for package delivery services, *J. Intell. Rob. Syst.*, **97** (2020), 719–731. https://doi.org/10.1007/s10846-019-01045-7
6. A. Thibbotuwawa, G. Bocewicz, G. Radzki, P. Nielsen, Z. Banaszak, UAV mission planning resistant to weather uncertainty, *Sensors*, **20** (2020), 515. https://doi.org/10.3390/s20020515
7. A. Troudi, S. A. Addouche, S. Dellagi, A. E. Mhamedi, Sizing of the drone delivery fleet considering energy autonomy, *Sustainability*, **10** (2018), 3344. https://doi.org/10.3390/su10093344
8. A. Thibbotuwawa, P. Nielsen, B. Zbigniew, G. Bocewicz, Energy consumption in unmanned aerial vehicles: A review of energy consumption models and their relation to the UAV routing, in *International Conference on Information Systems Architecture and Technology*, (2018), 173–184. https://doi.org/10.1007/978-3-319-99996-8_16
9. J. Hall, D. Anderson, Reactive route selection from pre-calculated trajectories—application to micro-UAV path planning, *Aeronaut. J.*, **115** (2011), 635–640. https://doi.org/10.1017/S0001924000006321
10. R. Shirani, M. St-Hilaire, T. Kunz, Y. Zhou, J. Li, L. Lamont, On the delay of reactive-greedy-reactive routing in unmanned aeronautical ad-hoc network, *Procedia Comput. Sci.*, **10** (2012), 535–542. https://doi.org/10.1016/j.procs.2012.06.068
11. W. Bożejko, A. Gnatowski, J. Pempera, M. Wodecki, Parallel tabu search for the cyclic job shop scheduling problem, *Comput. Ind. Eng.*, **113** (2017), 512–524. https://doi.org/10.1016/j.cie.2017.09.042
12. B. N. Coelho, V. N. Coelho, I. M. Coelho, L. S. Ochi, R. H. Koochaksaraei, D. Zuidema, et al., A multi-objective green UAV routing problem, *Comput. Oper. Res.*, **88** (2017), 306–315. https://doi.org/10.1016/j.cor.2017.04.011
13. M. A. R. Estrada, A. Ndoma, The uses of unmanned aerial vehicles—UAV's-(or drones) in social logistic: natural disasters response and humanitarian relief aid, *Procedia Comput. Sci.*, **149** (2019), 375–383. https://doi.org/10.1016/j.procs.2019.01.151
14. P. Golinska, M. Hajdul, Multi-agent coordination mechanism of virtual supply chain, in *KES International Symposium on Agent and Multi-Agent Systems: Technologies and Applications*, Springer, Berlin, (2011), 620–629. https://doi.org/10.1007/978-3-642-22000-5_64
15. P. Golinska, M. Hajdul, Virtual logistics clusters—IT support for integration, in *Asian Conference on Intelligent Information and Database System*, Springer, Berlin, (2012), 449–458. https://doi.org/10.1007/978-3-642-28487-8_47
16. M. Lohatepanont, C. Barnhart, Airline schedule planning: integrated models and algorithms for schedule design and fleet assignment, *Transp. Sci.*, **38** (2004), 19–32. https://doi.org/10.1287/trsc.1030.0026
17. P. Sitek, J. Wikarek, A multi-level approach to ubiquitous modeling and solving constraints in combinatorial optimization problems in production and distribution, *Appl. Intell.*, **48** (2018), 1344–1367. https://doi.org/10.1007/s10489-017-1107-9
18. A. Thibbotuwawa, G. Bocewicz, B. Zbigniew, P. Nielsen, A solution approach for UAV fleet mission planning in changing weather conditions, *Appl. Sci.*, **9** (2019), 3972. https://doi.org/10.3390/app9193972

19. P. Traverso, E. Giunchiglia, L. Spalazzi, F. Giunchiglia, Formal theories for reactive planning systems: some considerations raised from an experimental application, in *AAAI Technical Report WS-96-07,* 1996.
20. O. Oubbati, A. Lakas, M. Güneş, F. Zhou, M. B. Yagoubi, UAV-assisted reactive routing for urban VANETs, in *Proceedings of the Symposium on Applied Computing*, (2017), 651–653. https://doi.org/10.1145/3019612.3019904
21. O. S. Oubbati, N. Chaib, A. Lakas, S. Bitam, P. Lorenz, U2RV: UAV-assisted reactive routing protocol for VANETs, *Int. J. Commun. Syst.*, **33** (2020), e4104. https://doi.org/10.1002/dac.4104
22. G. Radzki, P. Nielsen, G. Bocewicz, Z. Banaszak, A proactive approach to resistant UAV mission planning, in *Conference on Automation*, Springer, Cham, (2020), 112–124. https://doi.org/10.1007/978-3-030-40971-5_11
23. M. Relich, G. Bocewicz, K. B. Rostek, Z. Banaszak, A declarative approach to new product development project prototyping, *IEEE Intell. Syst.*, **36** (2020), 88–95. https://doi.org/10.1109/MIS.2020.3030481
24. A. Thibbotuwawa, G. Bocewicz, P. Nielsen, Z. Banaszak, Unmanned aerial vehicle routing problems: a literature review, *Appl. Sci.*, **10** (2020), 4504. https://doi.org/10.3390/app10134504
25. T. Elmokadem, A. V. Savkin, A hybrid approach for autonomous collision-free UAV navigation in 3D partially unknown dynamic environments, *Drones*, **5** (2021), 57. https://doi.org/10.3390/drones5030057
26. M. Halat, Ö. Özkan, The optimization of UAV routing problem with a genetic algorithm to observe the damages of possible Istanbul earthquake, *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi*, **27** (2021), 187–198. https://doi.org/10.5505/pajes.2020.75725
27. K. E. C. Booth, *Constraint Programming Approaches to Electric Vehicle and Robot Routing Problems*, Ph. D thesis, University of Toronto, 2021.
28. M. A. Russell, G. B. Lamont, A genetic algorithm for unmanned aerial vehicle routing, in *Proceedings of the 7th Annual Conference on Genetic and Evolutionary Computation*, (2005), 1523–1530. https://doi.org/10.1145/1068009.1068249
29. J. E. Baculi, C. A. Ippolito, Onboard decision-making for nominal and contingency sUAS flight, in *AIAA Scitech 2019 Forum*, 2019. https://doi.org//10.2514/6.2019-1457
30. S. Bhargava, A note on evolutionary algorithms and its applications, *Adults Learning Math.*, **8** (2013), 31–45.
31. C. Guettier, F. Lucas, A constraint-based approach for planning unmanned aerial vehicle activities, *Knowl. Eng. Rev.*, **31** (2016), 486–497. https://doi.org/10.1017/S0269888916000291
32. I. K. Nikolos, E. S. Zografos, A. N. Brintaki, UAV path planning using evolutionary algorithms, in *Innovations in Intelligent Machines-1* (eds. J. S. Chahl, L. C. Jain, A. Mizutani, and M. Sato-Ilic), Springer, Berlin, (2007), 77–111. https://doi.org/10.1007/978-3-540-72696-8_4
33. G. Radzki, M. Relich, G. Bocewicz, Z. Banaszak, Declarative approach to UAVs mission contingency planning in dynamic environments, in *International Symposium on Distributed Computing and Artificial Intelligence*, Springer, 2021. https://doi.org/10.1007/978-3-030-86887-1_1
34. Z. Fu, J. Yu, G. Xie, Y. Chen, Y. Mao, A heuristic evolutionary algorithm of UAV path planning, *Wireless Commun. Mobile Comput.*, **2018** (2018). https://doi.org/10.1155/2018/2851964
35. R. Nagasawa, E. Mas, L. Moya, Model-based analysis of multi-UAV path planning for surveying postdisaster building damage, *Sci. Rep.*, **11** (2021), 1–14. https://doi.org/10.1038/s41598-021-97804-4

36. B. B. K. Ayawli, R. Chellali, A. Y. Appiah, F. Kyeremeh, An overview of nature-inspired, conventional, and hybrid methods of autonomous vehicle path planning, *J. Adv. Transp.*, **2018** (2018). https://doi.org/10.1155/2018/8269698
37. J. Hu, H. Wu, R. Zhan, M. Rafik, X. Zhou, Self-organized search-attack mission planning for UAV swarm based on wolf pack hunting behavior, *J. Syst. Eng. Electron.*, **32** (2021), 1463–1476. https://doi.org/10.23919/JSEE.2021.000124
38. N. Lin, J. Tang, X. Li, L. Zhao, A novel improved bat algorithm in UAV path planning, *J. Comput. Mater. Contin.*, **61** (2019), 323–344. https://doi.org/10.32604/cmc.2019.05674
39. V. Rodríguez-Fernández, H. D. Menéndez, D. Camacho, Design and development of a lightweight multi-UAV simulator, in *2015 IEEE 2nd International Conference on Cybernetics (CYBCONF)*, (2015), 255–260. https://doi.org/10.1109/CYBConf.2015.7175942
40. J. C. Rubio, J. Vagners, R. Rysdyk, Adaptive path planning for autonomous UAV oceanic search missions, in *AIAA 1st Intelligent Systems Technical Conference*, 2004. https://doi.org/10.2514/6.2004-6228
41. P. Calégari, G. Coray, A. Hertz, D. Kobler, P. Kuonen, A taxonomy of evolutionary algorithms in combinatorial optimization, *J. Heuristics*, **5** (1999), 145–158. https://doi.org/10.1023/A:1009625526657
42. A. Slowik, H. Kwasnicka, Evolutionary algorithms and their applications to engineering problems, *Neural Comput. Appl.*, **32** (2020), 12363–12379. https://doi.org/10.1007/s00521-020-04832-8
43. J. Stork, A. E. Eiben, T. Bartz-Beielstein, A new taxonomy of global optimization algorithms, *Nat. Comput.*, **2020** (2020), 1–24. https://doi.org/10.1007/s11047-020-09820-4

Appendix A

Table A1. Results of experiments conducted for $\mathcal{F}(\theta) = 9\frac{m}{s}, \forall\Theta \in [0°, 360°)$.

		$\mathcal{F}(\theta) = 9\frac{m}{s}, \forall\Theta \in [0°, 360°)$													
		ILOG		Genetic Algorithm (GA)											
				C1				C2				C3			
n	K	TC	f_0^{ILOG}	mT	σ	f_0^{GA}	%	mT	σ	f_0^{GA}	%	mT	σ	f_0^{GA}	%
40	2	2,17	300	60	34,214	300	100	0,06	0,015	205	68,33	58,51	0,45	285	95,00
	3	3,73	450	15,45	4,726	450	100	5,68	1,883	300	66,67	78,37	23,258	330	73,33
	4	151,69	600	11,27	4,472	600	100	15,69	0,236	355	59,17	111,36	1,133	420	70,00
60	2	6,62	300	57,33	27,25	300	100	5,28	1,903	240	80,00	92,11	0,577	285	95,00
	3	34,25	450	3,52	8,94	450	100	9,78	0,264	275	61,11	142,55	0,618	375	83,33
	4	230	600	14,86	5,613	600	100	24,17	0,592	385	64,17	212,7	1,009	435	72,50
80	2	11,1	300	68,7	35,29	300	100	8,52	0,059	240	80,00	141,76	0,585	270	90,00
	3	80	450	25,7	12,96	450	100	16,05	0,307	355	78,89	230,05	0,934	390	86,67
	4	382,29	600	20,97	12,468	600	100	34,66	0,89	360	60,00	339,92	1,729	420	70,00
100	2	53,41	300	23,03	6,041	300	100	12,39	0,064	220	73,33	212,62	1,625	270	90,00
	3	131	450	31	0,407	450	100	23,68	0,435	320	71,11	318,17	3,49	360	80,00
	4	t>600	×	29,24	15,966	600	100	71,84	1,832	345	57,50	600	2,834	480	80,00
120	2	64,56	300	70	9,36	300	100	30,57	0,118	265	88,33	272,3	1,079	300	100,00

		$\mathcal{F}(\theta) = 9\frac{m}{s}, \forall\Theta \in [0°, 360°)$													
		ILOG		Genetic Algorithm (GA)											
				C1				C2				C3			
n	K	TC	f_0^{ILOG}	mT	σ	f_0^{GA}	%	mT	σ	f_0^{GA}	%	mT	σ	f_0^{GA}	%
	3	226,34	450	34	0,997	450	100	70,12	1,282	330	73,33	591,49	2,769	370	82,22
	4	t>600	×	29	22,613	600	100	136,26	1,768	360	60,00	t>600	×	×	×
140	2	136,67	300	42	17,73	300	100	65,05	0,059	255	85,00	495,11	0,75	285	95,00
	3	599,32	450	21,68	1,25	450	100	62,51	1,03	335	74,44	t>600	×	×	×
	4	t>600	×	66,53	26,45	600	100	143,45	0,74	375	62,50	t>600	×	×	×
160	2	368,99	300	40,21	22,58	300	100	43,19	0,43	225	75,00	599,98	0,58	300	100,00
	3	t>600	×	24,11	7,06	450	100	78,12	1,01	315	70,00	t>600	×	×	×
	4	t>600	×	81	37,04	600	100	180,77	1,43	380	63,33	t>600	×	×	×
180	2	415,55	300	91	15,15	300	100	54,27	0,47	285	95,00	t>600	×	×	×
	3	t>600	×	37	12,55	450	100	103,34	0,98	290	64,44	t>600	×	×	×
	4	t>600	×	119,66	45,72	600	100	241,23	1,6	375	62,50	t>600	×	×	×
200	2	576,72	300	65,66	8,2	300	100	66,61	0,74	250	83,33	t>600	×	×	×
	3	t>600	×	38,7	6,54	450	100	133,09	0,67	305	67,78	t>600	×	×	×
	4	t>600	×	171,35	61,81	600	100	318,89	1,7	360	60,00	t>600	×	×	×
220	2	t>600	×	79,22	11,21	300	100	82,72	1,59	225	75,00	t>600	×	×	×
	3	t>600	×	49,89	8,73	450	100	156,67	1,01	290	64,44	t>600	×	×	×
	4	t>600	×	200,73	78,4	600	100	363,03	1,65	350	58,33	t>600	×	×	×

Note: n—number of nodes (delivery points); K—size of the UAV fleet; TC—time of computation for IBM ILOG; mT—average calculation time for GA; σ—standard deviation of calculation time; f_0—value of the objective function; %—percentage assessment of the difference between solutions obtained from ILOG and GA: $\frac{f_0^{GA}}{f_0^{ILOG}}$ [%].

Table A2. Results of experiments conducted for $\mathcal{F}(\theta) = 10\frac{m}{s}, \forall\Theta \in [0°, 360°)$.

		$\mathcal{F}(\theta) = 10\frac{m}{s}, \forall\Theta \in [0°, 360°)$													
		ILOG		Genetic Algorithm (GA)											
				C1				C2				C3			
n	K	TC	f_0^{ILOG}	mT	σ	f_0^{GA}	%	mT	σ	f_0^{GA}	%	mT	σ	f_0^{GA}	%
40	2	3,845	300	48,2	14,928	300	100	3,26	1,592	235	78,33	47,85	18,871	255	85,00
	3	5,131	450	2,45	3,773	450	100	6,2	1,838	275	61,11	71,74	18,385	390	86,67
	4	40,021	555	13,46	34,729	600	100	18,27	0,51	350	58,33	107,18	8,773	405	67,50
60	2	7,859	300	65,32	21,081	300	100	5,67	1,771	215	71,67	83,56	16,7	270	90,00
	3	28,583	450	4,14	6,781	450	100	10,51	1,666	305	67,78	132,41	10,643	370	82,22
	4	394,009	600	17,67	12,881	600	100	28,24	1,576	340	56,67	200,59	6,408	410	68,33
80	2	18,647	300	25	3,83	300	100	9,13	0,614	255	85,00	135,9	16,463	285	95,00
	3	167,592	450	5,7	4,599	450	100	16,26	0,625	310	68,89	216,76	7,469	400	88,89
	4	t>600	×	23,07	6,502	600	100	40,63	0,607	350	58,33	318,51	1,111	370	61,67

		$\mathcal{F}(\theta) = 10\frac{m}{s}, \forall\Theta \in [0°, 360°)$													
		ILOG		Genetic Algorithm (GA)											
				C1				C2				C3			
n	K	TC	f_0^{ILOG}	mT	σ	f_0^{GA}	%	mT	σ	f_0^{GA}	%	mT	σ	f_0^{GA}	%
100	2	49,724	300	3,04	26,858	300	100	12,5	1,027	240	80,00	197,63	13,839	285	95,00
	3	49,194	450	145	10,953	450	100	25,42	1,213	295	65,56	317,09	1,416	355	78,89
	4	t>600	×	152	12,186	600	100	79,71	1,639	375	62,50	t>600	×	×	×
120	2	52,216	300	196	0,912	300	100	17,39	1,71	235	78,33	271,71	2,794	285	95,00
	3	t>600	×	212	8,147	450	100	45,84	0,991	310	68,89	t>600	×	×	×
	4	t>600	×	61,28	7,747	600	100	103,37	1,571	350	58,33	t>600	×	×	×
140	2	131,188	300	9,71	10,639	300	100	32,44	1,324	225	75,00	499,47	7,476	280	93,33
	3	t>600	60	21,54	3,511	450	100	61,14	0,935	315	70,00	t>600	×	×	×
	4	t>600	×	83,21	3,003	600	100	144,88	1,046	390	65,00	t>600	×	×	×
160	2	356,773	300	10,24	5,985	300	100	42,83	0,453	255	85,00	t>600	×	×	×
	3	t>600	×	28,05	4,633	450	100	80,28	1,808	315	70,00	t>600	×	×	×
	4	t>600	×	65	6,726	600	100	202,13	0,641	400	66,67	t>600	×	×	×
180	2	541,505	300	98	15,498	300	100	53,36	1,072	225	75,00	t>600	×	×	×
	3	t>600	×	35,02	8,192	450	100	102,86	0,62	330	73,33	t>600	×	×	×
	4	t>600	×	150,85	8,599	600	100	268	1,556	365	60,83	t>600	×	×	×
200	2	599,257	300	19,27	0,625	300	100	72,3	1,418	245	81,67	t>600	×	×	×
	3	t>600	×	44,91	1,009	450	100	135,6	1,036	305	67,78	t>600	×	×	×
	4	t>600	×	202,68	1,436	600	100	354,15	1,509	365	60,83	t>600	×	×	×
220	2	t>600	×	19,37	0,54	300	100	81,89	0,754	235	78,33	t>600	×	×	×
	3	t>600	×	57,78	5,329	450	100	165,16	1,751	300	66,67	t>600	×	×	×
	4	t>600	×	239,33	0,801	600	100	428,7	0,626	365	60,83	t>600	×	×	×

Note: n—number of nodes (delivery points); K—size of the UAV fleet; TC – time of computation for IBM ILOG; mT—average calculation time for GA; σ - standard deviation of calculation time; f_0—value of the objective function; % - percentage assessment of the difference between solutions obtained from ILOG and GA: $\frac{f_0^{GA}}{f_0^{ILOG}}$ [%].

Table A3. Results of experiments conducted for $\mathcal{F}(\theta) = 11\frac{m}{s}, \forall\Theta \in [0°, 360°)$.

		$\mathcal{F}(\theta) = 11\frac{m}{s}, \forall\Theta \in [0°, 360°)$													
		ILOG		Genetic Algorithm (GA)											
				C1				C2				C3			
n	K	TC	f_0^{ILOG}	mT	σ	f_0^{GA}	%	mT	σ	f_0^{GA}	%	mT	σ	f_0^{GA}	%
40	2	3,957	300	96	30,905	300	100	3,53	1,058	225	75,00	47,771	21,925	270	90,00
	3	5,599	450	9,32	34,568	450	100	6,95	1,379	270	60,00	71,685	20,719	330	73,33
	4	342,597	555	19,72	31,304	600	100	24,3	1,781	390	65,00	112,225	16,231	390	65,00
60	2	6,845	300	32	7,138	300	100	6,11	1,083	205	68,33	83,249	21,772	255	85,00
	3	12,292	450	45,61	9,525	450	100	11,43	0,841	270	60,00	131,21	14,801	325	72,22

		$\mathcal{F}(\theta) = 11\frac{m}{s}, \forall\Theta \in [0°, 360°)$													
		ILOG		Genetic Algorithm (GA)											
				C1				C2				C3			
n	K	TC	f_0^{ILOG}	mT	σ	f_0^{GA}	%	mT	σ	f_0^{GA}	%	mT	σ	f_0^{GA}	%
	4	29,543	600	25,54	10,312	600	100	38,16	1,102	330	55,00	204,368	14,424	385	64,17
80	2	14,705	300	12	11,621	300	100	9,85	1,296	220	73,33	134,987	16,747	285	95,00
	3	31,342	450	7,27	6,468	450	100	18,72	1,435	285	63,33	213,583	15,717	345	76,67
	4	464,70	×	32,29	5,84	600	100	53,94	1,32	365	60,83	332,77	21,899	440	73,33
100	2	47,528	300	3,73	5,277	300	100	14,25	0,485	240	80,00	196,489	5,106	285	95,00
	3	t>600	×	24	5,703	450	100	26,79	1,146	300	66,67	320,935	17,292	405	90,00
	4	t>600	×	36	5,605	600	100	105,38	1,101	350	58,33	t>600	×	×	×
120	2	69,069	300	58	12,056	300	100	20,05	1,357	250	83,33	272,852	22,023	285	95,00
	3	271,93	450	18,2	5,425	450	100	47,08	1,23	305	67,78	595,066	20,186	360	80,00
	4	486,909	600	35	1,515	600	100	119,51	1,746	365	60,83	t>600	×	×	×
140	2	93,477	300	98	8,019	300	100	33,55	1,701	225	75,00	493,384	5,415	270	90,00
	3	476,9	450	24,52	5,884	450	100	62,62	1,753	310	68,89	t>600	×	×	×
	4	578,965	600	118,43	3,601	600	100	164,53	1,592	340	56,67	t>600	×	×	×
160	2	155,018	300	12,71	6,243	300	100	43,23	1,808	235	78,33	t>600	×	×	×
	3	t>600	×	32,04	1,503	450	100	85,6	0,579	295	65,56	t>600	×	×	×
	4	t>600	×	154	5,646	600	100	226,91	1,781	370	61,67	t>600	×	×	×
180	2	351,507	300	222	30,871	300	100	53,73	0,867	220	73,33	t>600	×	×	×
	3	t>600	×	40,46	3,673	450	100	102,23	0,861	285	63,33	t>600	×	×	×
	4	t>600	×	193,15	18,532	600	100	292,14	1,027	410	68,33	t>600	×	×	×
200	2	584,659	300	50	0,621	300	100	66,79	1,518	220	73,33	t>600	×	×	×
	3	t>600	×	48	1,259	450	100	129,77	1,351	330	73,33	t>600	×	×	×
	4	t>600	×	281,58	1,426	600	100	420,34	0,77	345	57,50	t>600	×	×	×
220	2	t>600	×	23,56	0,7	300	100	82,01	1,361	220	73,33	t>600	×	×	×
	3	t>600	×	65,46	1,086	450	100	167,36	0,671	310	68,89	t>600	×	×	×
	4	t>600	×	331,64	1,112	600	100	485,18	0,854	365	60,83	t>600	×	×	×

Note: n—number of nodes (delivery points); K – size of the UAV fleet; TC—ime of computation for IBM ILOG; mT—average calculation time for GA; σ—standard deviation of calculation time; f_0—value of the objective function; %—percentage assessment of the difference between solutions obtained from ILOG and GA: $\frac{f_0^{GA}}{f_0^{ILOG}}$ [%].