Multi-Objective Cooperated Path Planning of Multiple UAVs Based on Revisit Time

Hassan Haghighi, Davood Asadi, Daniel Delahaye

Multi-Objective Cooperated Path Planning of Multiple UAVs Based on Revisit Time

Hassan Haghighi[1]
ENAC, University of Toulouse, 31400, France

Davood Asadi[2]
Adana Alparslan Turkes Science and Technology University, Adana, 01250, Turkey

and
Daniel Delahaye[3]
ENAC, University of Toulouse, 31400, France

This paper investigates multi-objective optimization of coordinated patrolling flight of multiple unmanned aerial vehicles in the vicinity of terrain, while respecting their performance parameters. A new efficient modified A-star (A*) algorithm with a novel defined criterion known as individual revisit time cell value is introduced and extended to the whole area of the 3D mountainous environment. As a contribution to solving trade-offs in the optimization problem, revisit time is conjugated with other contrary costs effective in flight planning through Pareto analysis. By introducing the revisit time and applying a specific setup to mitigate computational complexity, the proposed algorithm efficiently revisits the desired zones, which are more important to be revisited during the patrolling mission. The results of the introduced modified A* algorithm are compared in various scenarios with two different algorithms including a complete and optimal algorithm known as Dijkstra, and an evolutionary algorithm known as the genetic algorithm. Simulation results demonstrate that the proposed algorithm generates faster and more efficient trajectories in complex multi-agent scenarios due to the introduced cell selection method and dynamic-based simplifications applied in this research.

Keywords: Cooperative path planning, multi-objective, multiple unmanned aerial vehicles, revisit time.

[1] Ph.D., Department of Aerospace Engineering, Toulouse, France, hasan.haghighi-ext@enac.fr.

[2] Assistant Professor, Department of Aerospace Engineering, Adana Alparslan Türkeş Science and Technology University, Adana, Turkey, dasadihendoustani@atu.edu.tr.

[3] Professor, Department of Aerospace Engineering, Head of optimization Lab. ENAC, Toulouse, France, daniel.delahaye@enac.fr.

Nomenclature

A, D, F, f, G, H, Z	= general *matrix or functions*
C_l, C_d, C_y	*= aerodynamic lift, drag, and side force coefficients*
F_y	*= UAV Side force, N*
g	*= gravitational acceleration, m/s*^{2}
h, h_{min}, h_{max}	*= flight altitude, m*
h_o	*= desired/operational flight level, m*
K	*= control constant to identify the types of revisit area*
$L,\ L_{path},\ L_{SP},\ L_{OIL},\ L_{NFIL}$	*= nominal length*
L_{max}	*= UAV maximum lift, N*
n	*= number of cells/areas in the path*
N	*= number of UAVs*
RTC	*= total revisit time function*
$R,\ R_{FOV},\ R_{min}$	*= related turn radius, m*
S_C	*= covered area, m*^{2}
S_{FOV}	*=UAV field of view area, m*^{2}
S_T	*= total area of the environment, m*^{2}
S_w	*= UAV wing area, m*^{2}
t	*= time, sec*
$Th,\ Th_{min},\ Th_{max}$	*= engine thrust, N*
T_{age}	*= aging time, sec*
T_r	*= revisit time matrix*
V	*= UAV velocity, m/s*
x, y, h	*= longitude, latitude, and elevation coordinates, m*
X	*= position vector*
w_i	*= weight of the cost function*
$\delta_e(t), \delta_a(t), \delta_r(t)$	*=elevator, aileron, rudder deflection, rad*
Δd	*= displacement distance, m*
θ	*= pitch angle, rad*
λ	*= inclination angle from the straight path, rad*
ψ_{turn}	*= turning angle, rad*
ϕ_{max}	*= UAV maximum roll angle, rad*
$\emptyset$	*= empty matrix*
ρ	*= air density, kg/m*^{3}

I. Introduction

Unmanned Aerial Vehicles (UAVs) are increasingly considered capable of performing hazardous missions in adversarial environments. Applications of UAVs include wildfire management, geology, ecology, climatology, forestry, agricultural monitoring, border surveillance, reconnaissance, geophysical survey, meteorological investigation, aerial photography, and search-rescue mission as described in [1-5]. One of the most well-known problems in multi-agent path planning is the Vehicle Routing Problem (VRP), which transfers between a fixed station and a set of customers. Typically, the total route cost or the travel time is minimized by various approaches such as heuristic and evolutionary methods [6, 7]. The goal is to minimize the weighted sum of waiting time, travel time, and the average delay between visits. One solving approach is to construct a traveling salesman problem (TSP)-route and to place the agents in that route [8-10]. Another approach is to split the area into clusters, one for each vehicle, and solve a TSP for each cluster [11, 12]. Other methods include negotiation-based algorithms [13], reinforcement learning [14] Markov decision process [15], divide-and-conquer algorithms [16], ant colony algorithm [17], simulated annealing in an adversarial environment [18], and bee colony algorithm [19]. Only a few works consider revisiting on patrolling problems where each point has a deadline for each visit, aims at minimizing the cost of a relevant parameter such as fuel, priority, and important areas. The major drawback of all previously described problems is the cooperation model of vehicles in the maneuvering area and the lack of proper strategy for cooperated decision making.

Path planning is a key element of the UAVs autonomous control framework [20]. In most cases in the literature, a deterministic search algorithm is implemented to find the shortest path, which is accounted as the best-desired path. Depending on the problem, this definition has been developed and the best path is referred to as trajectories optimizing several objectives including traveling distance, coverage, obstacle avoidance, flight altitude, control effort, etc. To cope with problem complexity, researchers have gradually moved away from deterministic algorithms application towards non-deterministic and heuristic algorithms [21].

The problem of UAV path planning, which avoids static obstacles in a dynamic environment is relatively mature. In [22], the corresponding position of the UAV is described by fuzzy information to solve the problem of dynamic obstacle representation. In [23], a two-dimensional real-time dynamic path planning algorithm based on the velocity vector was proposed to establish different spatial characteristics of the velocity field model. Ceccarelli [24] used the artificial potential field and the fuzzy virtual force to simulate the dynamic environment. Genetic Algorithm (GA) [25] and an improved ant colony algorithm [26] were applied in dynamic path planning for the mobile threat source. These studies mostly aim to avoid static threats and are mainly limited to two-dimensional space, without considering moving threats and their maneuvers.

Due to the limited capabilities of a single UAV in large areas and long-range missions, more than one UAV is mostly required to be applied to accomplish the mission. Thus, cooperative algorithms should control the overall framework. Although the above references implement suitable methods in path planning but are mostly restricted to either single UAV applications or Single-Objective Optimization (SOO). A Multi-Objective Optimization (MOO) problem typically includes a set of solutions that are favorable for the rest of the solutions in the total search space. These solutions are known as Pareto-optimal solutions or non-dominated solutions [27-29], where the rest of the solutions are known as dominated solutions. All solutions in the non-dominated set are acceptable and none of them has the privilege respecting the other solutions [30].

The most common way to solve MOO problems is to derive a scalar objective function from the vector of objectives by applying a weight vector [28, 29]. Since mission planning is basically a search problem, Multi-Objective Evolutionary Algorithms (MOEAs), as well as Multi-Objective Genetic Algorithm (MOGA), are prevalent options to cope with these types of problems [31].

In search and coverage missions, to cover all possible activities, the area must be visited within a specific time interval [32, 33]. In the present work, to cope with interval/aging patrolling, a novel timeline cellular searching area method is introduced, in which the value of each cell is determined by a special parameter called revisit time. Revisit time is a function of cell aging time, in which UAVs must visit a series of cells on a priority basis for three reasons: first, to identify the important area, second, to assign the important area for each UAV during the process of task allocation as a criterion for cooperation, and third, to patrol across the valuable path and avoid the local minimum. The patrolling path composes of a queue of cells with the greatest value of revisit time, which incorporate into the weighted form cost function. A solution is a group of path possibilities that impose linear requirements on dynamic constraints of the form $|Df(x)| \leq b$ for a performance value $b \in \mathbb{R}$, given a differential operator $D = \sum_{j=1}^{N} \sum_{i=1}^{n} (\gamma_i \partial^i)_j$, where γ_i is objective coefficient and $\partial^r f(x) = \frac{\partial^r f(x)}{\partial x_1 \partial x_2 \dots \partial x_r}$ is a *r-th* of dominant variable x. In order to satisfy the fast runtime requirement of the optimization algorithm in three-dimensional (3D) space, a modified A-star (MA*) algorithm is applied. The results of the presented approach are compared with the output of the analytical Dijkstra and GA evolutionary algorithm.

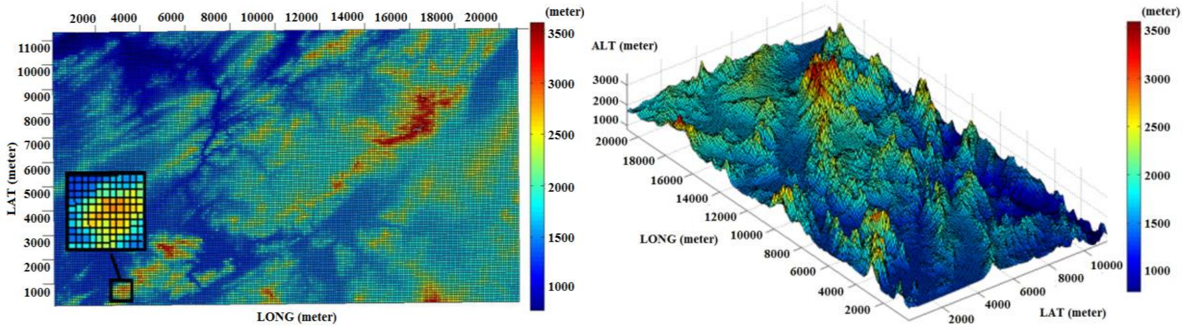
To summarize, the state of the art in cooperated trajectory planning includes several challenging problems. This paper advances the state of the art in the following contributions:

- A novel approach to solve the cooperated multi-agent patrolling with online mapping and dynamic situation of cooperated agents. The formulation captures the features of the dynamic environment, i.e. variable revisit values.

- A multi-objective compatible heuristic algorithm is proposed to estimate the value of the admissible path and provide an online algorithm to solve the cooperated patrolling problem based on the introduced cost functions while considering the UAVs' dynamics in the path planning. Moreover, we propose a multi-agent algorithm that sequentially computes policies for individual UAVs.
- In comparison with different algorithms, simulation results demonstrate the performance benefits of the proposed algorithm, which can effectively reduce the solver complexity by efficiently selecting the sequence of areas with a lower amount of unvisited cell production.

II. Mathematical Representation

The path planner strategy, which is proposed in this paper, has a hierarchical structure according to our previous researches [9, 20]. The first step of path planning is to discretize the environment space into a representation that will be meaningful to the path planning algorithm. Therefore, an approximate 3D cell decomposition of the "Zagros forest protected area" is applied using a 2D grid. Each element of the matrix (cells) represents the elevation of the terrain. This representation allows for applying digital elevation maps repository with no further processing.



a) *b)*

Fig. 1 2D cell decomposition in a 3D environment Zagros mountain, Iran- *a*) 2D cell decomposition, *b*) 3D environment (21*km* (Long.) ×12*km* (Lat.)) with 100*m*×100*m*.

Let $N \in \mathbb{N}$ and $n_j \in \mathbb{Z}^+$ respectively denote the number of UAVs as a value within the set of natural numbers $\mathbb{N} = \{1, 2, 3, \dots\}$, and the number of cells in the individual path of the jth UAV within the set of semi-positive integers $\mathbb{Z}^+ = \{0, 1, 2, \dots\}$, with the following constraints:

$$\begin{cases} min(N) = 1 \\ \max\left(n = \sum_{j=1}^{N} n_j\right) = [S_T/S_{FOV}] \end{cases} \tag{1}$$

We discretize the environment and limit the size of the discrete cells to be no smaller than the sensor footprint, thus the maximum number of cells is $[S_T/S_{FOV}]$ in which S_T and S_{FOV} are the total area of the environment and the area of the UAV field of view, respectively.

The p-norm of any arbitrary vector is defined as $\|\vec{v}\|_p = (\sum_i (v_i)^p)^{1/p}, (1 \leq p < \infty)$, which is used to evaluate the effect of multi-agents cooperated costs in terms of path variables. Let X be a set of position vectors including a nominal sequence of position values of longitude x, latitude y, and altitude h. Considering any function of X, which can be defined as $f(X): X^N \to \mathbb{R}$ to demonstrate the multi-objective cost functions. The novel Revisit Time Cost (RTC) is defined as below:

$$RTC(X) = \begin{cases} \dfrac{e^{K\tau(X)} - 1}{e^{K\tau(X)}} & T_r \geq T_{age} \\ 1 & T_r \leq T_{age} \end{cases} \tag{2}$$

where K as a control parameter determines the rate of RTC growth and τ is a time-based function of the path X, which depends on revisit time (T_r). The parameter τ indicates when a cell should be revisited when it reaches a certain amount of aging time, the elapsed time since the last visit. It is defined as below for a set of cells $(1,2,\ldots,n)$, to evaluate the RTC function.

$$\tau(X) = \sum_{i=1}^{n} \tau(X_i) = \sum_{i=1}^{n} \frac{T_{age}(X_i)}{\left|T_{r_i} - T_{age}(X_i)\right|} \tag{3}$$

where T_{age} is the aging time, which is the elapsed time since the last visit and is determined for each important zone. The mathematical formulation of the multi-objective path planning formulation uses the RTC and the following parameters, which can be summarized in three main groups: variables, constraints, and objectives as presented in Table 1.

Table 1 Mathematical components

Group	Description	
Variables	Fight Altitude (h) Turning Rates ($\dot{\psi}$) Field Of View (R_{FOV}) Revisit Time Matrix ($[T_r]$) Cooperation Factor (CF)	
Constraints	Velocity:	$V = V_{trim}$
	Operational altitude:	$h_{min} \leq h_o \leq h_{max}$
	Rate of climb:	$\dot{h}_{min} \leq ROC \leq \dot{h}_{max}$
	Turning rate:	$\dot{\psi}_{min} \leq \dot{\psi} \leq \dot{\psi}_{max}$
	Minimum turning radius:	R_{min}
	Minimum inclination from straight:	λ_{min}
	Thrust:	$Th_{min} \leq Th \leq Th_{max}$
	Revisit time matrix:	$[0] \leq T_r \leq [Tr_{max}]$
Objectives	Maximum Revisited Value (max(RTC)) Minimum Cooperation Factor Minimum number of producing critical cells Maximum Coverage	

III. Multi-Objective Path Planning

In the case of multi-objective path planning, the optimal solution is more complex and includes different dynamic characteristics. A path fulfilling all the attributes to a high degree would result in low-cost functions. Our planning cost functions are defined based on the revisit time, UAVs dynamics, and searching environment. There are delicate tradeoffs between the objectives in the path planning; one imposes a shorter path in order to have a minimum cost, and the other makes a longer path, therefore, the cost functions are adjusted by weighted functions. This means that the multi-objective optimization problem is transformed into a single objective optimization problem by using the weighted sum of the cost functions. The outcome of path optimization is a set of Pareto-optimal solutions that reflect the trade-off between the objectives.

A. Cost Functions

To design a 3D path planner $\min_X\left\{\left(X(tf) - X_f\right) + \sum f(X)\right\}$ in a complex environment, multiple objective functions are combined using the weighted sum method into a total cost function of *Z*, as follows:

$$Z_j = w_1 f_1(\vec{X}_j) + w_2 f_2(\vec{X}_j) + w_3 f_3(\vec{X}_j) + \cdots \tag{4}$$

$$Z = \sum_{j=1}^{N} \sum_{i=1}^{k} w_i f_i(\vec{X}_j) \tag{5}$$

where $\vec{X}_j = [xyh_1, xyh_2, \dots, xyh_n]_j$ represents the *n*-dimensional vector of feasible cells' location $[x, y, h]'$ (longitude, latitude, and altitude), k is the number of individual costs, w_i are the weights ($0 < w_i < 1$), which all sum up to unity ($\sum_{i=1}^{k} w_i = 1$), and the $f_i(X): \mathbb{R}^{3\times n} \longrightarrow \mathbb{R}$ is a group of multi-objective cost functions, which will be introduced. The proposed multi-objective formulation in this paper uses the RTC and the following parameters, which are divided into six groups as defined in Table 2.

Accordingly, Eq. (6) denotes the revisited cost, Eqs. (7, 8) are the costs related to the UAV's performance and dynamics. Accordingly, Eq. 8 is related to the turn maneuver and specifies that straight motions are more favorable respecting the curvilinear motions. The Eqs. (9-11) determine the environmental constraints and limitations. Mathematically, a solution obtained by equal weights for all objectives may offer the least objective conflict, but the real-world requirements demand a satisfying solution, while priorities must be included in the formulation. The advantage of this strategy is to control the impact of one objective over the other one, and the obtained solution is usually a Pareto-optimum solution.

Table 2 Path cost function components

Cost Function	Parameters	Formula	
Revisit	Revisit time cost (*RTC*)	$f_1(X) = \frac{1}{RTC(X)}$	(6)
Altitude	Flight altitude (h) Desired altitude (h_d) Max. admissible altitude (h_{max}) Min. admissible altitude (h_{min})	$f_2(X) = \frac{\lvert h(X) - h_d \rvert}{h_{max} - h_{min}}$	(7)
Turn	Turning angle (ψ_{turn}) Path length (L_{path}) Inclination angle from straight ($\lambda = \pi - \psi_{turn}$) Minimum turning radius (R_{min}) Field of view radius (R_{FOV})	$f_3(X) = \frac{\frac{\pi - \lvert \psi_{turn}(X) \rvert}{\pi}}{L_{path}}$	(8)
Min. distance	Straight path length (L_{SP}) Path length (L_{path})	$f_4\ (X) = 1 - \frac{L_{SP}}{L_{path}(X)}$	(9)
Collision avoidance	Obstacle inside length (L_{OIL}) Path length (L_{path})	$f_5(X) = \frac{L_{OIL}}{L_{path}(X)}$	(10)
No-fly zone	No-fly zone inside length (L_{NFIL}) Path length (L_{path})	$f_6(X) = \frac{L_{NFIL}}{L_{path}(X)}$	(11)

To achieve the best flight performance [34], the *j-th* UAV path variable $X_j = \left[[x_1, y_1, h_1]', \dots, [x_{nj}, y_{nj}, h_{nj}]'\right]$, $n_j \in \mathbb{Z}^+$ includes a sequence of points/cells with a specific revisit time value $T_{r_i} \geq 0$, for each cell $i \in \{1,2, \dots, n_j\}$. Thus optimal $X_j{}^*$, is the path with the minimum accumulated revisit cost function $\min_{X_j} 1/\sum_{j=1}^{n_j} RTC(X_j)$, where $RTC(X_j)$ is the revisited value on each distinct trajectory.

B. UAV Dynamics

The desired UAV path implements minimum turning maneuvers to have less control effort in path planning. In addition, according to Fig 2*a*, UAVs might lose the center of the search cells in sharp turns. Therefore, the cost of admissible turns with the goal of the smaller number of turns is defined in Eq. (12). According to Eqs. (12, 13), let λ be the inclination angle from straight in turn maneuvers as $\lambda = \pi - \psi_{turn}$ shown in Fig. 2*b*, then the dynamics of the best turn with R_{min} is calculated as a function of UAV Field of View (FoV), lift (L), and bank angle (ϕ) as follows.

$$\sin\left(\frac{\pi - \psi_{turn_{max}}}{2}\right) = \sin\left(\frac{\lambda_{min}}{2}\right) = \frac{R_{min}}{R_{min} + R_{FOV}} \tag{12}$$

consequently

$$\lambda_{min} = 2\sin^{-1}\left(\frac{R_{min}}{R_{min} + R_{FOV}}\right) = 2\sin^{-1}\left(\frac{1}{1 + \frac{R_{FOV}}{R_{min}}}\right) \tag{13}$$

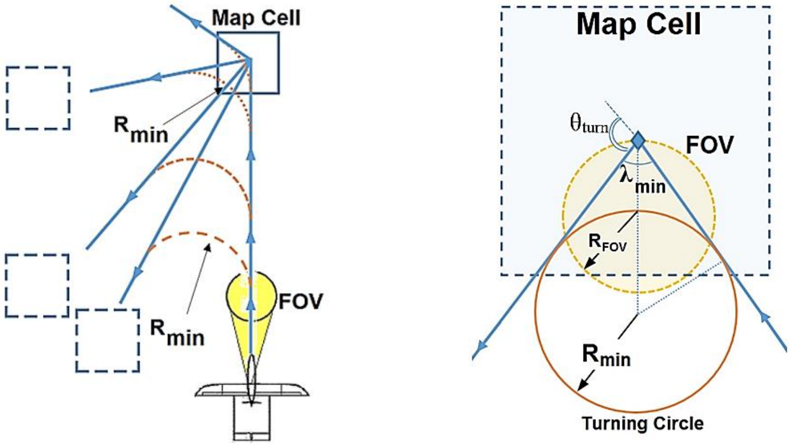
where

$$R_{min} = \frac{mV^2}{L_{max} \ \ sin\phi_{max}} \tag{14}$$

$$R_{min} = \frac{2m}{\rho S_w C_{l_{max}} \ \ sin\phi_{max}} \tag{15}$$

where ρ is the air density and m is the mass of UAV, S_w is the wing area, ϕ_{max} is the maximum roll angle, V is the velocity, $C_{l_{max}}$ is the maximum lift coefficient, and L_{max} is the maximum lift of the desired UAV. In level flight, the lift to weight ratio is considered to be equal to one $(L/mg \approx 1)$, where g is the gravitational acceleration. In steady-state flight conditions, the UAV minimum radius of turn can be estimated by Eq. (16) as below:

$$R_{min} = \frac{V^2}{gsin\phi_{max}} \tag{16}$$



a) *b)*

Fig. 2 Turning limitation. *a*) divers approaches to candidate cells with a minimum turn radius, *b*) maximum performance in turn due to cell coverage.

Let $\vec{X}$ include a sequence of independent position values $\vec{X} = [[x_1, y_1, h_1]', [x_2, y_2, h_2]', \ldots]$ to connect the environmental model to the multi-objective function. The UAV displacement (Δd) from the current position $[x_k, y_k, h_k]'$ to the next desired position $[x_{k+1}, y_{k+1}, h_{k+1}]'$, during the time interval of Δt, which depends on drag force F_D, inclination angle λ, total velocity V, and pitch angle θ, can be calculated based on the UAV discretized dynamics equation as follows:

$$\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ z_{k+1} \end{bmatrix} = \begin{bmatrix} x_k \\ y_k \\ z_k \end{bmatrix} + \begin{bmatrix} -cos\lambda & -sin\lambda & 0 \\ sin\lambda & -cos\lambda & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} F_D sin\lambda \\ F_D(1 + cos\lambda) \\ V\Delta t \ \ sin\theta \end{bmatrix} \tag{17}$$

where

$$F_D = \frac{V^2}{g\sqrt{F_y - 1}} \tag{18}$$

$$F_y = \frac{1}{2}\rho V^2 S_w C_{y_{max}} \tag{19}$$

where F_y is the lateral force, and C_y is the lateral force coefficient. The pitch angle (θ) and the inclination angle (λ), which are the control input parameters, should be determined by the optimization of turning flight. Supposing a constant speed UAV, the displacement is equal to $\Delta d = V\Delta t$. Due to the limitations of λ, θ, and Δt, the search space of the next time step is restricted to certain surface spaces. Therefore, the problem here has two control inputs λ, θ to create a sequence of position vector $X = [[x_1, y_1, h_1]', [x_2, y_2, h_2]', \dots]$. Thus, the optimization problem changes to the minimization of the total cost function Z (X) as follows:

$$Z^*(X) = \begin{matrix} min \\ \lambda, \theta \end{matrix} \left(\sum_{i=1}^{k} w_i f_i(X) \right) \tag{20}$$

Subjected to the dynamics constraints described in Table 2.

IV. Resolution Algorithms

Given the size of the search area and the limitation of flying time, the desired areas must be searched by several UAVs by applying optimal algorithms. Accordingly, different paths are allocated to each UAV by their priority to have the best overall revisit cost [35]. Therefore, a Cooperation Factor (CF) is defined as a criterion to guarantee the maximum coverage by producing separate paths. A function must be defined to minimize the cell intersections among all designed paths. The number of intersections (n_{is}) of the *jth* UAV, relative to the path length is a measure of the cooperation and efficiency of the coverage between the two separate points during multi-agent patrolling.

$$CF_j = \left. \frac{n_{is}}{L_{path}} \right)_{UAV_j} \tag{21}$$

The resolution for a cooperation flight is based on the assigned revisited cells. Cell distribution is based on revisit critical value and UAV priority during hierarchical processing. Accordingly, X_j *is* a sequence of the position vector, which express the path of *jth* UAV, and D_j indicates the position set of critical revisited cells. The distributed analysis calculates cells' subsets (C_1, C_2, ...) on sequential operations as follows:

$$\begin{cases} C_1 = \{[x, y, h]'| \quad [x, y, h]' \in X_1 + D_1\} \\ C_2 = \{[x, y, h]'| \quad [x, y, h]' \in X_2 - C_1 + D_2\} \\ C_3 = \{[x, y, h]'| \quad [x, y, h]' \in X_3 - C_1 - C_2 + D_3\} \\ \vdots \\ C_N = \left\{ [x, y, h]'| \quad [x, y, h]' \in X_N - \sum_{j=1}^{N-1} C_j + D_N \right\} \end{cases} \tag{22}$$

where C_j is the set of cells that indicate the path of the j-th UAV in a cooperated patrolling.

The ideal no-intersection situation is for all i and j, $i \neq j$; $C_i \cap C_j = \emptyset$ were based on intersection distribution, $C_1 \cap C_2 \ \cap \dots \cap C_N = \emptyset$. According to the practical environment, this condition is usually violated, therefore the optimal solutions try to have a minimum intersection. Efficient methods try to minimize the number of critical cells *D* as a function of the UAV path length L_{path}:

$$\begin{cases} D = \bigcup_{j=1}^{N} D_j \\ count(D) = \sum_{j=1}^{N} n_{cr\,j} \end{cases} \tag{23}$$

The minimum count of critical cells n_{cr}, which implies that the cell needs to be revisited as soon as possible due to revisit time, can be expressed as a function of covered cells (L_{path}) at each time step for *N* number of UAVs, according to the following equation.

$$D_{min} = min\left(\sum_{j=1}^{N} \frac{1}{\left(n_{cr} \cap L_{path}\right)_j}\right) \tag{24}$$

Coverage performance is measured by a criterion named Percentage of Coverage (POC). Let S refers to the area, S_T is the total area and S_C is a covered area, which depends on the field of view radius R_{FOV} and the length of the path X_i as follows.

$$POC = \frac{S_C}{S_T} \tag{25}$$

$$S_C = \sum_{i=1}^{n_j} (S_{FOV})_i \approx \sum_{j=1}^{N} 2R_{FOV}|X_j| \tag{26}$$

Where $|X_j|$ denotes the traveling distance for agent *j*. Therefore, it can be followed for all UAVs by Eq. (27).

$$POC = \frac{\sum_{j=1}^{N}\sum_{i=1}^{n_j}(S_{FOV})_i}{S_T} = \frac{\sum_{j=1}^{N} 2R_{FOV}|X_j|}{S_T} \tag{27}$$

The following algorithms include the cooperated planning (*Algorithm 1*), minimum critical cell approach (*Algorithm 2*), and achieving the maximum percent of coverage (*Algorithm 3*) in terms of reference cells, the cellular transformation of the searching environment, the allowable cells, and the free area without any obstacles or non-critical passed cells. In the following algorithms, the OpenList includes all the initial candidate cells and the CloseList represents the selected cells of the path derived from the optimization process.

Algorithm 1: Cooperated function

```
function multiUavCooperation
    OpenList←referenceCells() as environment
    CloseList← []
    $T_r$← cellValues() as revisit value
    for i in numberOfUavs
        startPoint ← OpenList(Max($T_r$))
        OpenList←{OpenList}\ {startPoint}
        endPoint ← OpenList(Max($T_r$))
        OpenList←{OpenList}\ {startPoint}
        Ci← findBestPath(startPoint,endpoint)
        Add Ci to CloseList
        OpenList←{OpenList}\ {CloseList}
        $T_r$(CloseList) ← 0
    endfor
return CloseList
```

Algorithm 2: Minimum critical cells

```
function minimumCriticalCells
    OpenList←referenceCell()
    O←OpenCell
    CloseList← []
    $T_r$← cellValues()
    for i in numberOfCells
        if  $T_r i$ >= criticalTr
            startPoint ←  {$T_r i$}
            $T_r$  (startPoint) ← 0
            O← {O}\{ startPoint}
            for j in (i:numberOfCells)
                if  $T_r j$ >= critical$T_r$
                    endPoint ←{$T_r j$  }
                    $T_r$(endPoint) ←0
                    O ← {O}\{endPoint}
                Break
            endfor
        Break
        startPoint ← OpenList(Max($T_r$))
        OpenList ← {OpenList} \{startPoint}
        endPoint ← OpenList(Max($T_r$))
        $T_r$(startPoint) ← 0
        $T_r$(endPoint) ← 0
        Ci←findBestPath(startPoint,endPoint)
        O ← {O}\{Ci}
        $T_r$({Ci}) ← 0
        Add Ci to CloseList
    endfor
return  $T_r$, CloseList
```

Algorithm 3: Cooperated percent of coverage

```
function maxPecentOfCoveage
    OpenList← referenceCell()
    O←OpenCell
    CloseList← []
    $T_r$← cellValues
    Percent OfCoverage ← 0
    startPoint ← OpenList(Max($T_r$))
    OpenList ← {OpenList} \{startPoint}
    endPoint ← OpenList(Max($T_r$))
    for i in numberOfUavs
        $C_i$ ← findBestPath(startPoint,endpoint)
        O ← {O}\{$C_i$}
        $T_r$({$C_i$}) ← 0
        percentOfCoverage ← percentOfCoverage + area({$C_i$})/totalArea
        Add Ci to CloseList
    endfor
return    percentOfCoverage
```

A. Cooperated Modified *A-star*

According to the problem strategy, the optimization method must be consistent with the cost function. In real-time applications, the convergence rate is very important. Based on the identification of key parameters, such as area visibility, cell cost value, and elevation map, many algorithms can be applied to solve the problem. Given the sensitivity of the vehicles and aerial operations, the fastest one, which can be compatible with multi-objective solutions, is more favorable. For this purpose, the capabilities of the Modified *A-star* (MA*) algorithm are used, and the results are compared with other approaches such as GA and Dijkstra algorithms.

Inspiring from typical $A*$ during searching among trajectories, the algorithm first considers the ones with the fastest solution. It is formulated in terms of weighted graphs: starting from a specific node of the graph, constructs a tree of paths from the starting node, expanding paths one step at a time until one path ends at the predetermined End Point (EP). At each iteration of its main loop, the algorithm needs to determine which one of the partial trajectories to expand into one or more trajectories. The cell with the lowest cost value is chosen as a candidate for the next one in the sequence. According to the basic formula of $A*$ [36], the structure of the multi-objective cost function of Eq. (5) is rewritten as Eq. (28), which determines the $MA*$ structure.

$$Z_{{A^*}_j} = Z_g(X_j^{n-1}) + Z_h(X_j^n) \tag{28}$$

The path variable consists of a sequence $X = \left[X_j^1, X_j^2, \dots, X_j^{n-1}, X_j^n\right]$ for j-th UAV where $Z_g(X_j^{n-1})$ is defined as the multi-objective cost of the path from the known start node to any node $n-1$, and $Z_h(X_j^n)$ is defined as a heuristic term that estimates the cost of the trajectories from $n-1$ to the destination node within the entire set of all the target nodes. Each adjacent cell is evaluated by the value Z_{A^*}. The node with the lowest Z_{A^*} value is removed from the queue, the Z_{A^*} and Z_g values of its neighbors are updated accordingly, and the neighbors are added to the queue. The algorithm continues until the endpoint node has a lower Z_{A^*} value respecting other nodes in the queue or until the queue is empty.

In multi-agent problems with several endpoints $(n_1, n_2, \dots, n_j, \dots, n_N)$, the multi-objective cost function specifies a multivariable function (*Z*), which is evaluated by the real-time expansion of cells as follow:

$$Z\left(X_1^{n1}, X_2^{n2}, \dots, X_j^{nj}\right) = \left[\sum_{i=1}^{n1-1} Z_g\left(X_1^i\right) + Z_h\left(X_1^{n1}\right) + \sum_{i=1}^{n2-1} Z_g\left(X_2^i\right) + Z_h\left(X_2^{n2}\right) + \cdots + \sum_{i=1}^{nj-1} Z_g\left(X_j^i\right) + Z_h\left(X_j^{nj}\right)\right] \tag{29}$$

Consequently, multi-agent Z_{ij} for global position variable $X \ in$ the search space is as follows:

$$\begin{cases} Z_{ij}(X) = min\left[\sum_{j=1}^{N}\left(\sum_{i=1}^{n_j-1} Z_g(X_j) + Z_h(X_j)\right)\right] \\ \qquad finding\colon X_1, X_2, \dots, X_N \end{cases} \tag{30}$$

Subject to

$$
\begin{cases}
Constraints: V, h, ROC, R_{min}, R_{FOV}, \dot{\psi}, T, Tr \\
\min CF = \sum_{j=1}^{N} \frac{(n_{is})_j}{(L_{path})_j} \\
\min Critical \ \ Cell = min\left(\sum_{j=1}^{N} \frac{1}{(n_{Tr} \cap L_{path})_j}\right) \\
\max POC: \ max\left(\frac{\sum_{j=1}^{N} S_{C_j}}{S_T}\right)
\end{cases}
\tag{31}
$$

The pseudo-code of the cooperative MA* expansion and the final component of the cooperated multi-objective approach are presented below in Algorithm 4 and Fig. 3:

*Algorithm 4: Best revisit path with MA**

```
function findBestPath(startPoint,endPoint)
    O← OpenList
    CloseList← []
    C← CloseList
    $T_r$← cellValues
    $Z_{best}$ ← Z(StartPoint)
    for j in O
        if $\left[\sum_{k=startPoint_i}^{j} \boldsymbol{Z_g(k)} + \sum_{k=j}^{endPoint_i} \boldsymbol{Z_h(k)}\right] < (\boldsymbol{Z_{best}})$
            Cj← Cj +cell(j)
            $\boldsymbol{Z_{best}} \leftarrow \left[\sum_{k=startPoint_i}^{j} Z_g(k) + \sum_{k=j}^{endPoint_i} Z_{\hbar}(k)\right]$
        endif
    endfor
        O ← O \Cj
        $T_r$(Cj) ← 0
         Add Cj to CloseList
     endfor
return CloseList
```

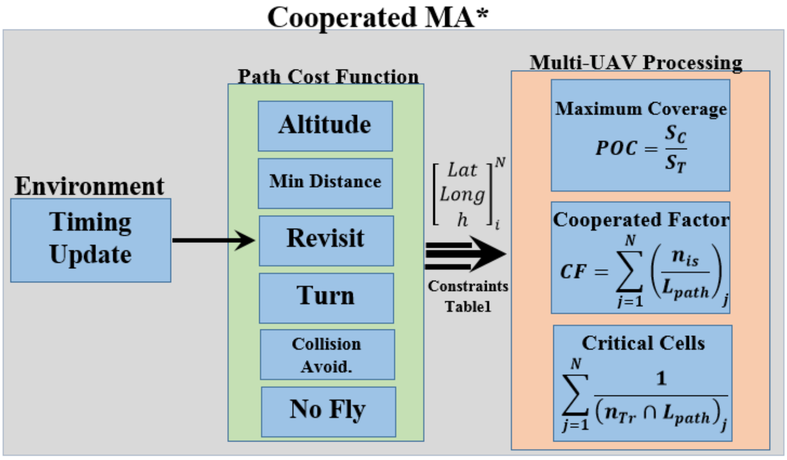


Fig. 3 Multi-objective cooperated modified A* (MA*) during optimal path planning for multiple UAVs

V. Simulation Results and Discussions

According to Eq. (5), the first step of trajectory planning is the cost function verification in terms of weights. In the simulations, some kinematic limitations were considered as the following equation.

$$
\begin{gathered}
V_{trim} = 30 \ (\frac{m}{s}) \\
1000 \ (m) \leq h \leq 3500 \ (m) \\
-5 \ \left(\frac{m}{s}\right) \leq ROC \leq 5(\frac{m}{s}) \\
-0.08 \ \left(\frac{rad}{s}\right) \leq \dot{\psi} \leq 0.08 \ (\frac{rad}{s}) \\
R_{min} = 375 \ (m) \\
R_{FOV} = 375 \ (m) \\
\lambda_{min} = 60 \ (deg) \\
100(N) < Th < 200(N) \\
[0] < T_r < [T_r]
\end{gathered}
\tag{32}
$$

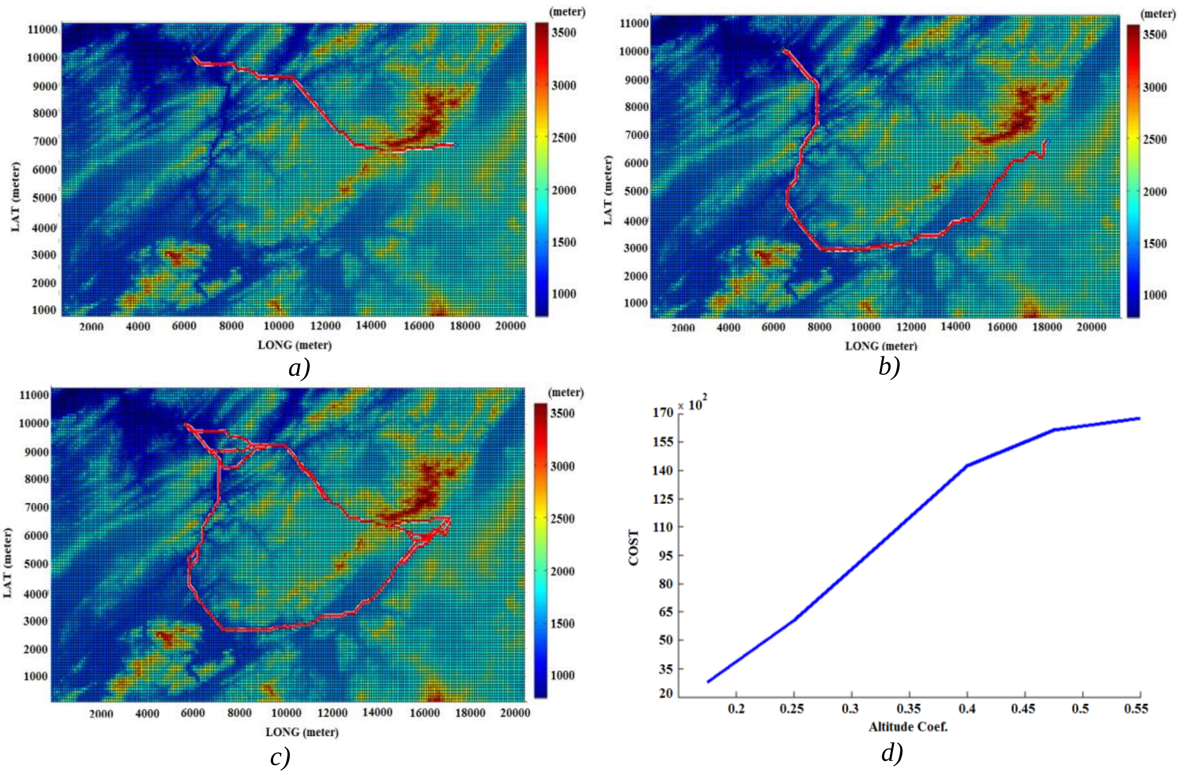
where the maximum flight altitude of UAVs is considered to be 3500 meters in terms of V_{trim} and rate of climb ROC. The trajectories are created in a discrete way using Bresenham's line drawing algorithm [37].

A. Multi-Objective Weight Analysis

The goal here is just to analyze the effect of the multi-objective cost function (Eq. (5)) in a 3D environment. The choice of one solution over others requires problem prevailing conditions. Therefore, in multi-objective optimization problems, it is useful to apply Pareto-optimal analysis [38]. Consider a planning system with N agents and k cost. An allocation $\{X_1, X_2, \cdots, X_N\}$, where $X_j \in \mathbb{R}^k$ for all $j = 1, \cdots, N$, is Pareto optimal if there is no other feasible allocation $\{X_1', X_2', \cdots, X_N'\}$ for the objective function Z_i for each objective $i = 1, \cdots, k$ as $Z_i(X_j') \leq Z_i(X_j)$. Pareto sets make an efficient frontier of solutions in the performance space [39] to demonstrate that the weighted cost solution is an optimal response.

Fig. 4a represents the simulation result of path planning for a specific value of the altitude coefficient, while Fig.4b illustrates the results of weight increase by 300%. In Fig. 4c, different trajectories derived for the different possible values of the altitude weight coefficient are illustrated. Fig. 4d depicts the altitude cost threshold for variations of the altitude weight coefficient which demonstrates that the cost function saturates for the altitude weights larger than 0.5. Table 3 represents the important parameters of three different scenarios with different altitude weight coefficients. Accordingly, the trajectory length and number of control efforts increase by increasing the altitude weight coefficient.

The minimum distance, unlike altitude, enforces shorter trajectories. Fig. 5a represents the trajectory with a specific value of min distance weight, while Fig. 5b illustrates the trajectory with a 300% increase of the related weight. Accordingly, more straight trajectories are derived by increasing the min distance weight. The overall trajectory pattern for a variety of weights is illustrated in Fig. 5c, while Fig. 5d represents the cost values with different weight coefficients. The trajectory performance parameters are shown in Table 4 respecting three different scenarios with different values of minimum distance weight coefficient.

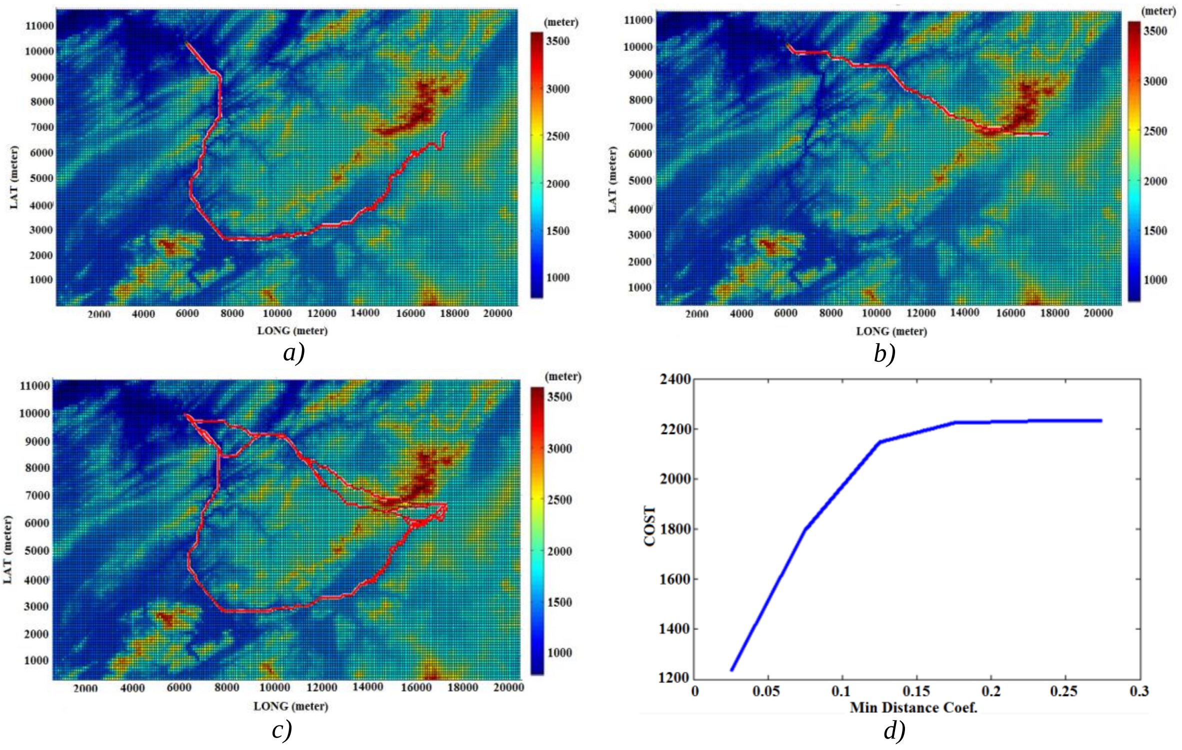


a) b) c) d)

Fig. 4 Altitude weight effect: *a*) initial value, *b*) 300% increased, *c*) the range of possible trajectories, *d)* overall trajectory pattern concerning altitude weight variations.

Table 3 Altitude weight performance according to the cost function component

Altitude weight	Trajectory cost	Number of control effort	Trajectory length (m)	Cost of revisit term	Cost of altitude
Initial value	2631	126	10379	241	352
Increased by 100%	4528	341	12971	518	3411
Increased by 300%	10432	831	24312	795	8745

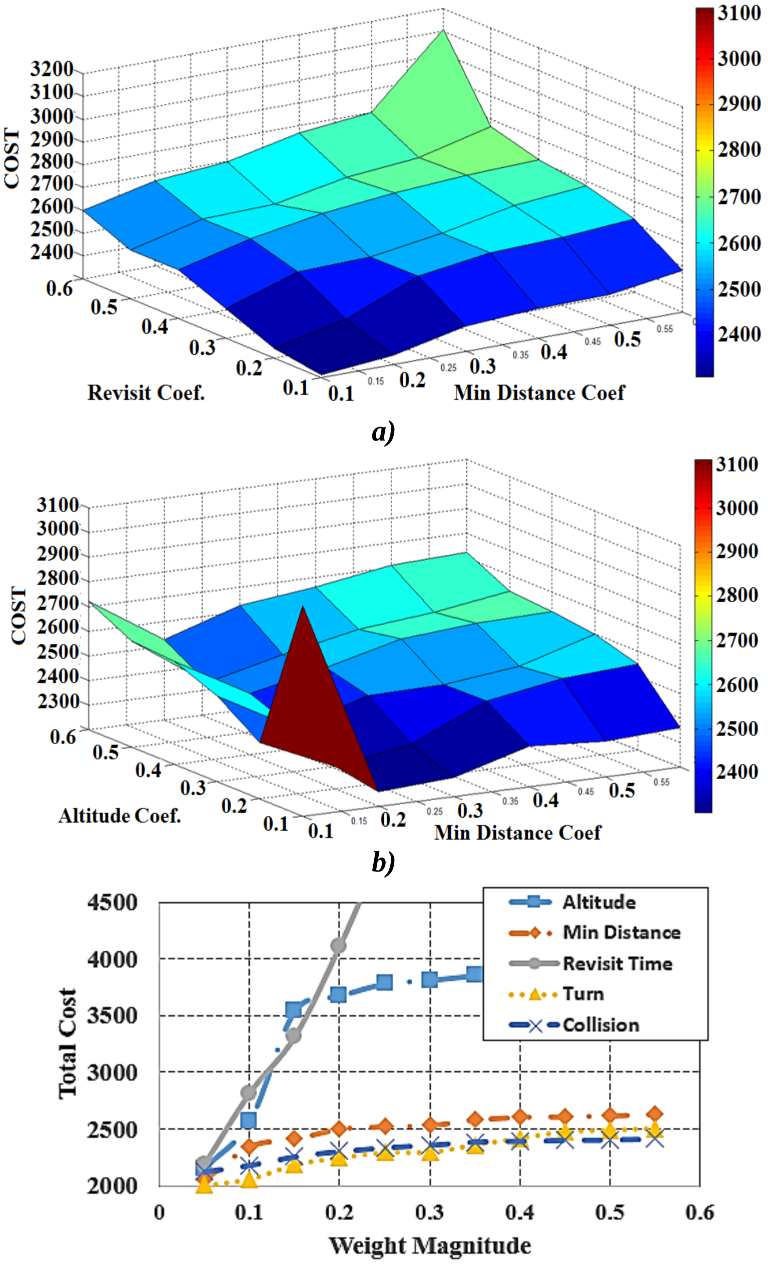


a) b) c) d)

Fig. 5 Min distance weight effect: *a*) initial value, *b)* 300% increased, *c*) the range of possible trajectories, *d*) overall trajectory pattern concerning altitude weight changes.

Table 4 Min distance weight performance according to the cost function components

Altitude weight	Trajectory cost	Number of control effort	Trajectory length (m)	Cost of revisit term	Cost of distance
Initial value	1310	811	24879	113	341
Increased by 100%	1540	298	14121	211	961
Increased by 300%	2320	112	10965	231	1869



a)

b)

c)

Fig. 6 Dual comparison of weights in multi-objective cost: *a*) between revisit time and altitude, *b*) between min distance and altitude, *c*) total cost.

According to Fig. 6*a* and Fig. 6*b*, a dual comparison of cost functions is applied by changing the cost functions respecting the weights. Fig. 6*c* illustrates the effect of each cost function weight on the total value of the cost function according to Eq. (5).

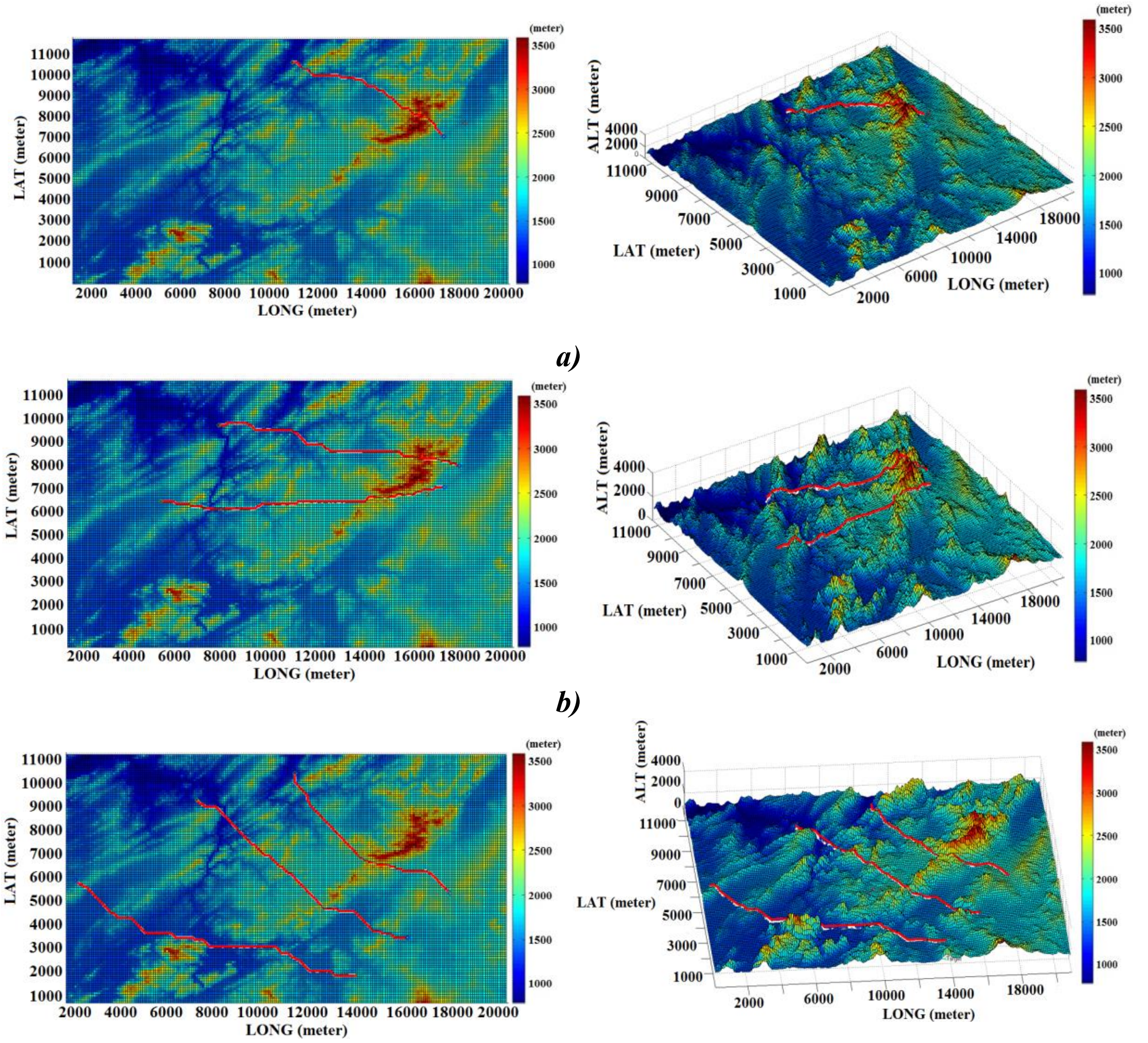
In Pareto analysis, based on the normalized weight coefficients and dual comparison of weights, the following weight factors are selected according to Table 6. The criterion for choosing the coefficients is based on the individual cost threshold and optimality in a common patrol mission. The simulation results hereafter are based on the weights derived by the Pareto Analysis.

Table 6 Min distance weight performance according to the cost function component

Altitude coefficient	Min-distance coefficient	Revisit time coefficient	Turn coefficient	Collision avoidance coefficient
0.175	0.125	0.275	0.2	0.225

B. Multiple UAV Analysis

Based on previous analysis, the simulations are illustrated for a cooperated flight using the MA* optimal solution for a single, double, and triple-UAV regarding revisit values.



a)

b)

c)

Fig. 7 Multi-agent simulation of cooperated patrolling in the start of mission between different high revisit time values; *a)* a single-UAV, *b*) double-UAV *c)* triple-UAV.

Table 7 Multi-agent mean performance values according to the cost function component

Number of UAVs	Trajectory cost	Number of control effort	Trajectory length (m)	Cost of revisit term	Cost of altitude	POC
1	1893	75	8312	241	311	5.32%
2	3185	182	21351	518	534	12.01%
3	6257	289	30132	795	721	18.54%

Fig. 7 illustrates the trajectories based on the MA* optimal solution for a single, double, and triple-UAV cooperated flight respecting revisit values. Table 7 represents the mean performance values for different searching scenarios of Fig. 7. As presented in Table 7, simulations demonstrate that for the same flight endurance with similar UAVs, multi-agent flights provide a higher percentage of coverage by 5.32%, 12.01%, and 18.54% for one, two, and three UAVs accordingly in only one iteration from initial to the goal destination.

C. Revisit time analysis

Revisit time is assessed as the main cost function in our trajectory planning problem. From a cellular point of view, when a cell/area is revisited by the UAV field of view, the cell lifespan should be reduced to zero and the incremental process should be resumed. Fig.8 illustrates the planning process to follow four revisit constraints (RTC1, RTC2, …, RTC4) in the general plane. The objective is to cross these desired four points on the desired revisit time $\left(\boldsymbol{T_{r_1}}, \boldsymbol{T_{r_2}}, \boldsymbol{T_{r_3}}, \boldsymbol{T_{r_4}}\right)$ shown as constraints time circles. Most paths (more than 67%) cover the constraints and pass the area in the correct aging time. It can be improved by increasing the number of UAVs. For a set of areas, the closer the value of the average revisit time to the desired value, the greater the efficiency of the optimization algorithm in implementing the revisit time. Fig. 9 shows the experience of ten areas under revisit time lifespan for one, two, three, and four UAVs in ten lifespans. The average revisits life span of these nominated ten cells shows 2.25 hours, 1.51 hours, 1.17 hours, and 0.85-hour revisit time for one, two, three, and four UAVs, respectively. It is also inferred that the ten life-span will finish at 22 hours, 16 hours, 12 hours, and 9 hours for selected scenarios.

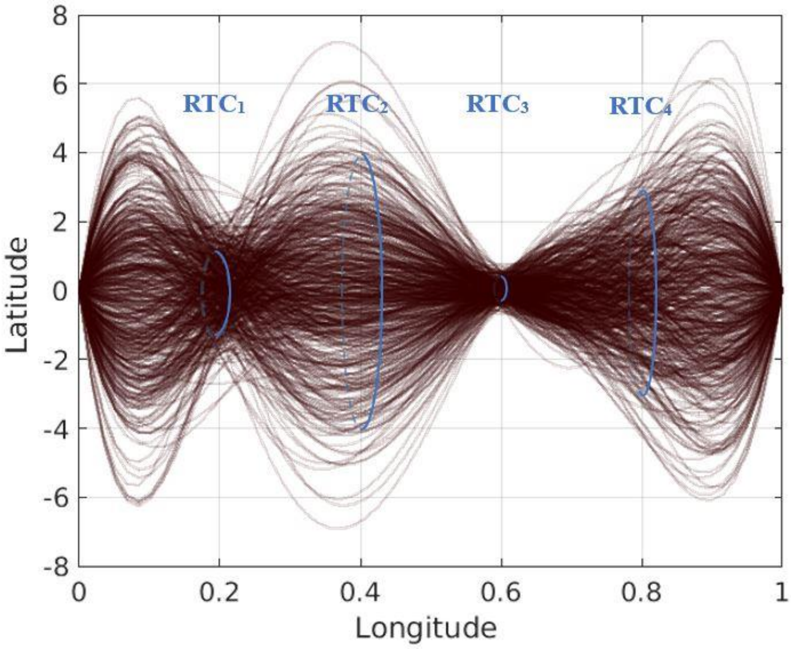


Fig. 8 Path planning analysis to follow 4 revisit time cost (RTC1~4) constraints ($T_{r_3} < T_{r_1} < T_{r_4} < T_{r_2}$)

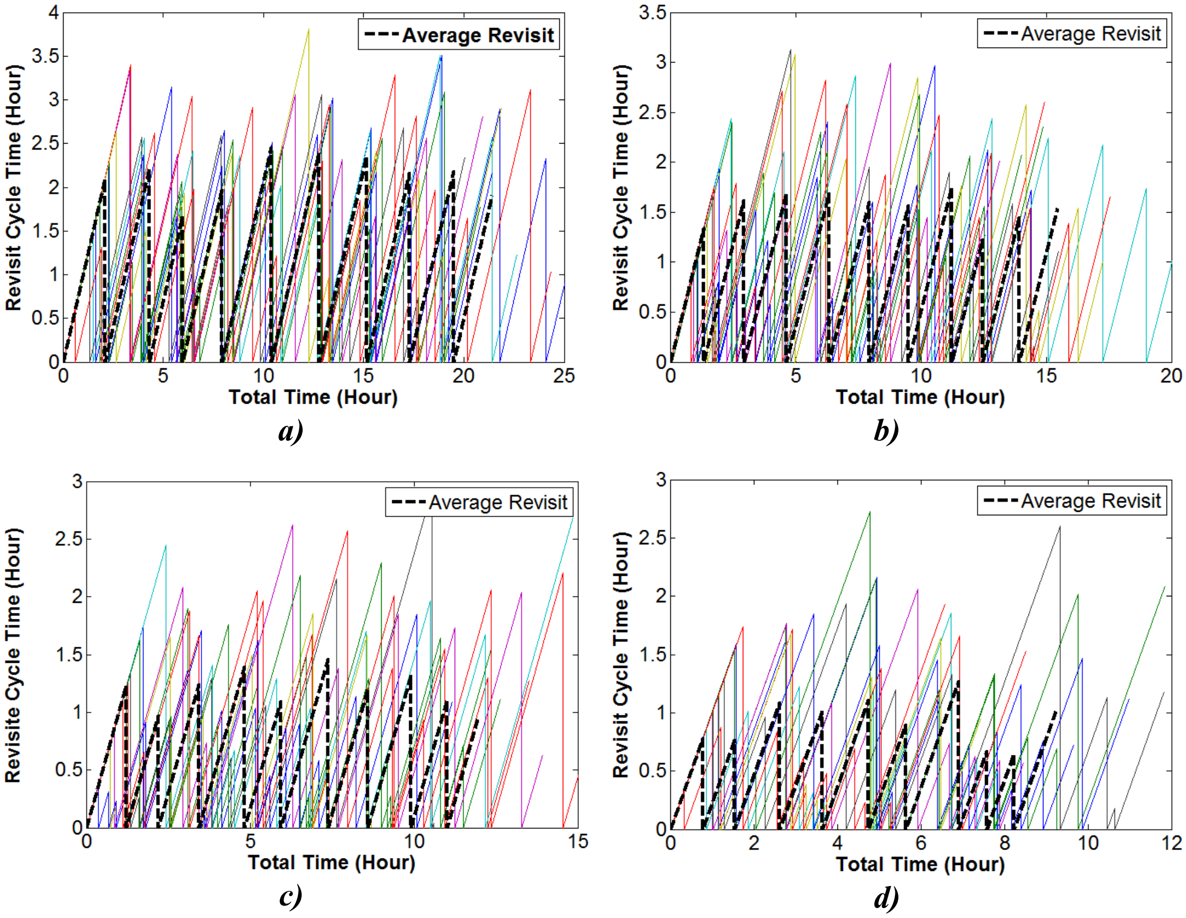
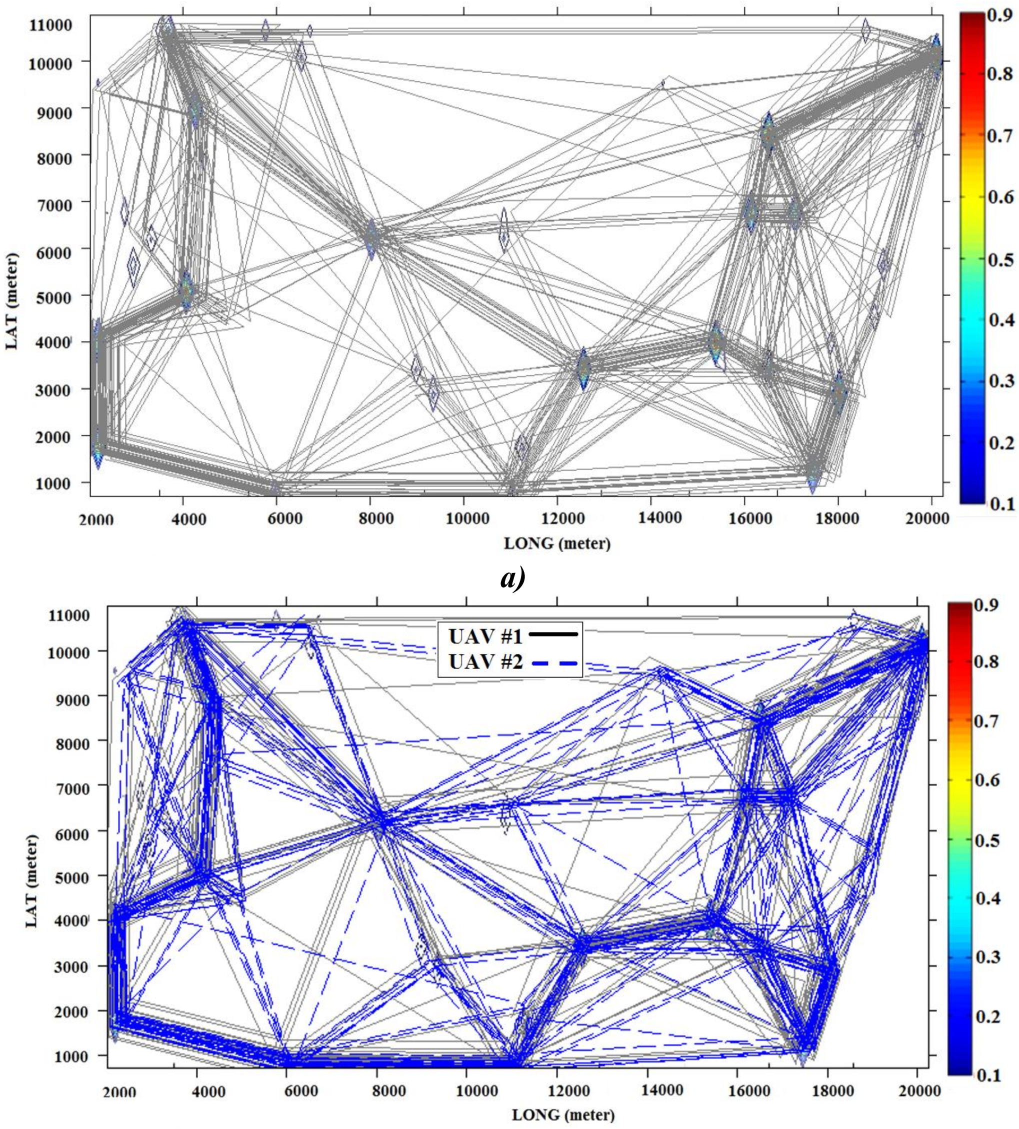


Fig. 9 Revisit time life-span for 10 nominated areas for 10 life-span; *a*) single UAV patrolling, *b*) 2-UAV cooperated patrolling, *c*) 3-UAV cooperative patrolling, *d*) 4-UAV cooperated patrolling.

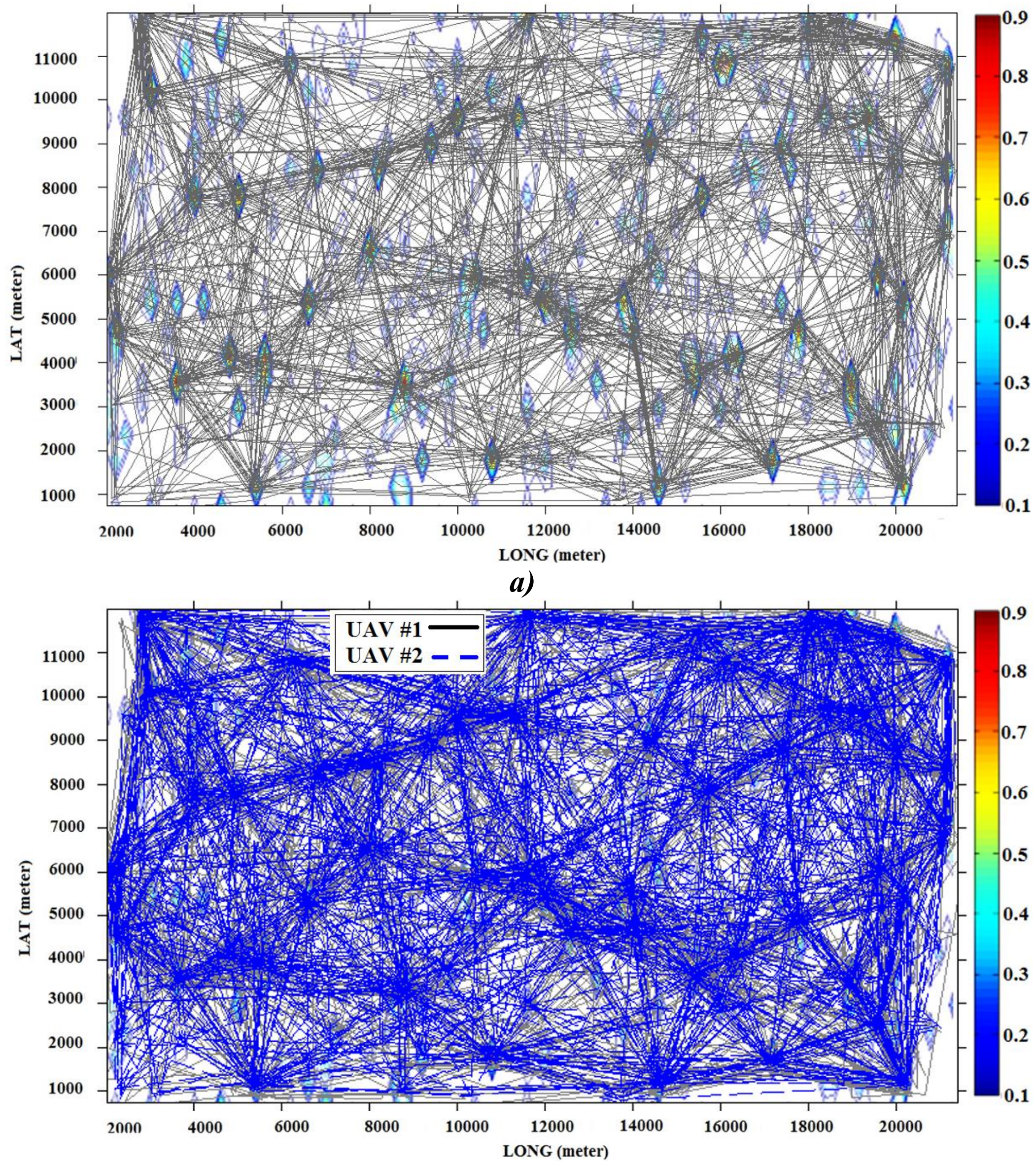
As stated in the mathematical relations, the revisit time as the life-span experience of an important area is a dynamic parameter that changes over time [40]. The best way to evaluate the algorithm's performance respecting the revisit time is to stop the lifespan in the desired development and to check the level of environmental cell revisit density. According to Fig. 2, important areas respecting revisit cost were previously introduced. Figs. 10 and 11 illustrate patrolling over the first and second revisit development for a single and cooperated double-UAV. Fig. 10 shows 200 hours of patrolling by a single (10.*a*) and double-UAV (10.*b*) on the first revisit development. Fig. 11 illustrates the patrolling of 500 hours by a single (11.*a*) and double-UAV (11.*b*) on the second revisit development. According to Figs 12 and 13, trajectory density is much higher around the point of interest, which is more important to be revisited during the patrolling mission.



a)

b)

Fig. 10 Patrolling routes over the initial revisit development for 200 hours of patrolling (Between 0 to 1 based on revisiting cost function); *a*) single-UAV patrolling, *b*) Double-UAV patrolling.



a)

b)

Fig. 11 Patrolling routes over revisit development for 370 hours of patrolling (Between 0 to 1 based on revisiting cost function); *a*) single-UAV patrolling, *b*) Double-UAV patrolling.

For multi-agent traveling over the mentioned environment, the results are compared with a classical and an evolutionary algorithm, namely Dijkstra and GA algorithms. Two comparison criteria for measuring the efficiency of algorithms are the revisit rate of the critical cell production (number of cells per unit of time) and the run-time. Critical cells are the cells that have passed the time of revisiting and are still waiting to be visited. An improper increase in this criterion increases the complexity of the problem or may even lead to the collapse of the problem. Fig. 12 illustrates the revisit rate of the critical cells produced by different approaches. The results demonstrate the optimality of MA* respecting the other two algorithms, especially when the number of UAVs is increased. Accordingly, when 9 UAVs are applied for coordinating patrolling, the revisit rate of the critical cells is around zero, which means full coverage achievement of the desired area by a minimum of 9 agents.

Fig.13 represents the comparison of important parameters including the run-time, accuracy, and number of extended cells for three algorithms of MA*, GA, and Dijkstra. Comparisons of the performance criteria verify the efficiency of the MA* algorithm respecting other algorithms in multi-agent patrolling missions.

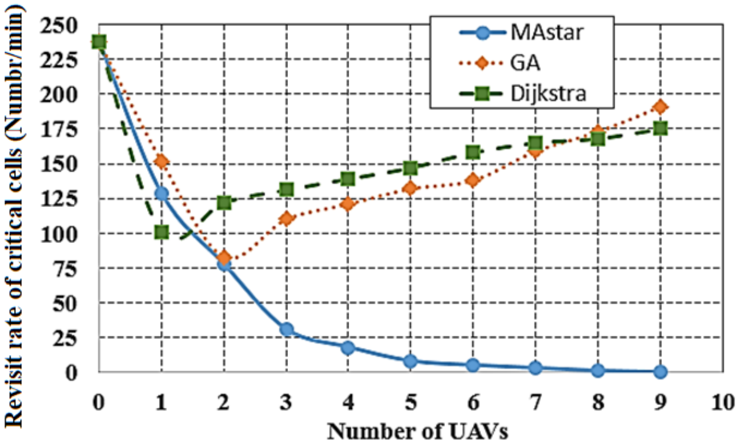


Fig. 12 Different algorithms' performances to prevent critical revisit cells in terms of the number of UAVs during the real-time solution.

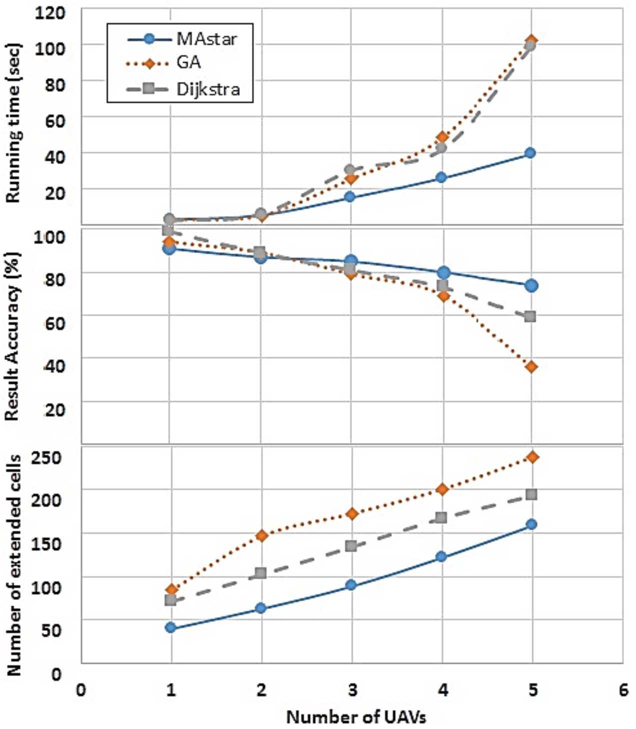


Fig. 13. Algorithms comparisons in terms of extended cell number, result accuracy, and run-time.

An important criterion for evaluating the efficiency of the algorithm, especially for real-time applications, is the number of extended cells to generate the optimal trajectory. According to the results, the MA* is more efficient in comparison with both Dijkstra and GA. Fig 13 indicates some performance criteria in terms of run-time, result accuracy, and the extended number of cells. In multi-agent scenarios, the MA* algorithm shows much better performance, while presenting less computational complexity. The complexity order of GA usually is exponential ($O(e^n)$) and depends on the genetic operators, their implementation, the representation of the individuals, the population, and the fitness function, while the time complexity of MA* is polynomial ($O(n)$), and depends on the heuristic function (Z_h). The time complexity of Dijkstra is polynomial with the base of the neighboring numbers per node but the solutions do not have enough sense of the target point and therefore create unfavorable trajectories.

Although desirable responses are achieved from the proposed approach, some limitations have been neglected. The communication range between UAVs, camera resolution, the field of view limitation, and camera delay time for object recognition have to be applied in the modeling and simulation. In our future research, we plan to focus on coordinated multi-agent path planning, while considering communication and camera limitations.

VI. Conclusions

This paper presents an efficient approach for multi-objective cooperative path planning of multiple UAVs in terms of values of cell's revisit time and several other effective cost functions including the altitude, minimum distance, collision avoidance, and turn costs. The main objective is to develop an implementable optimization algorithm that can be applied in cooperated multiple UAVs patrolling in search and coverage missions. The paper contributes to coordinated patrolling by MA* optimization algorithm to cover the most valuable cells during the flight in a 3D environment.

The design is based on a weighted multi-objective cost function, which is uniformly Pareto regulated and is applied according to the importance of its distinguished costs. In order to formulate the mathematical model of optimized cooperated patrolling, a multi-objective weighted form of the cost function in terms of dynamics constraints is proposed. The proposed approach develops an individual task weighting and cell investigation for persistent surveillance as well as border patrol challenges. The revisit time is considered as a governing parameter in path planning optimization. The variation of cost function weights changes the trajectory and its performance parameters. For revisiting scenarios that were subject to the lifespan revisit cycle, the density of trajectories over the important cells is evident, which demonstrates the effectiveness of the introduced revisit time in path planning. The revisit time changes over time as a dynamic parameter according to the importance of the area and therefore it can generate different trajectories. Simulations show that the proposed method not only effectively avoids low-efficiency trajectories but also has the desired ability to perform a cooperated design while satisfying real-time demands by generating fast and accurate solutions.

Finally, to demonstrate the effectiveness of the proposed approach for minimizing the multi-objective cost function, the performance parameters are compared in terms of critical cell generations that need to be revisited. However, the performance of the algorithm is lower in single-agent patrolling, it shows higher performance in a complex environment in multi-agent patrolling missions. In fact, in multi-agent applications, the results of the proposed algorithm are much more efficient in comparison with Dijkstra and GA algorithms regarding the lower time computational complexity of MA*, the number of extended cells, and a higher percentage of coverage, while considering revisit of important zones.

References

[1] Nex, F., Remondino, F., “UAV for 3*D* mapping applications: a review,” *Applied geomatics*, Vol. 6, No. 1, 2014, pp. 1-15. doi: 10.1007/s12518-013-0120.

[2] Adao, T., Huruska, J., Padua, L., Bassa, J., Pres, E., Morais, R., Sousa, J., “Hyperspectral imaging: A review on UAV-based sensors,” data processing and applications for agriculture and forestry, *Remote Sensing*, Vol. 9, No. 11, 2017, 1110. doi: 10.3390/rs9111110.

[3] Kim Y, Bang H., “Vision-based navigation for unmanned aircraft using ground feature points and terrain elevation data,” *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, Vol. 232, No. 7, 2018, pp.1334-1346.
doi: 10.1177/0954410017690548.

[4] Campion, M., Ranganathan, P., Faruque, S., “UAV Swarm Communication and Control Architectures: A Review,” *Journal of Unmanned Vehicle Systems*, Vol. 7, 2019, pp. 93-106.
doi:10.1139/juvs-2018-0009.

[5] Gai, W., Zhang, N., Zhang., J., Li, Y., “A constant guidance law-based collision avoidance for unmanned aerial vehicles,” *Proceedings of the Institution of Mechanical Engineers, Part G*: *Journal of Aerospace Engineering*, 2018, pp.1-13, doi:0954410017751325.

[6] Mersheeva, V., and Gerhard, F., “*Multi-UAV monitoring with priorities and limited energy resources*,” Twenty-Fifth International Conference on Automated Planning and Scheduling, Jerusalem, Israel, from June 7–11, ICAPS, 2015.

[7] Haghighi, H., Sadati, S. H., Karimi, J., Dehghan, S. M., “Heuristic optimization of multi-agent persistent surveillance revisit time by minimum distance weight functions”, *Journal of aerospace knowledge and technology,* Fall 2017-winter 2018, vol.6, No2, pp. 129-141, 2017

[8] Xu, Y., Chang C., “*A brief review of the intelligent algorithm for traveling salesman problem in UAV route planning*,” IEEE 9th ICEIEC, July 12-14,2018 in Beijing, China, 2019.

[9] Haghighi, H., Heidari, H., Sadati, H., Karimi, J., “*A hierarchical and priority-based strategy for trajectory tracking in UAV formation flight*,”, 8th International Conference on Mechanical and Aerospace Engineering (ICMAE), *IEE*E, Prague, Czech Republic, 22-25 July 2017.

[10] Haghighi, H., Sadati, S. H., Dehghan, S. M., & Karimi, J., “Hybrid Form of Particle Swarm Optimization and Genetic Algorithm For Optimal Path Planning in Coverage Mission by Cooperated Unmanned Aerial Vehicles”, *Journal of Aerospace Technology and Management,* 12, 2020.

[11] Ahmadi, M., Stone, P., “*A multi-robot system for continuous area sweeping tasks*,” IEEE International Conference on Robotics and Automation, Orlando, FL, USA, 26 Jun. 2006.

[12] Smith, S., Rus, D., “*Multi-robot monitoring in dynamic environments with guaranteed currency of observations*,” ICDC-IEEE, Atlanta, GA, USA, 22 Feb., 2010.

[13] Poulet, C., Corruble, V., Seghrouchni, A., working as a Team: Using social criteria in the timed patrolling problem, ICTWAI- IEEE, Athens, Greece, 7-9 Nov., 2012.

[14] Zhang, B., *Mao, Z., Liu, W.,* “Geometric reinforcement learning for path planning of UAVs,” *Journal of Intelligent & Robotic Systems*, Vol. 77, No.2, 2015, pp. 391-409.
doi:10.1007/s10846-013-9901-z.

[15] Capitan, J., Luis M., and Aníbal O., “Cooperative decision-making under uncertainties for multi-target surveillance with multiples UAVs,” *Journal of Intelligent & Robotic Systems*, Vol. 84, 2016, pp. 371-386.
doi: 10.1007/s10846-015-0269-0.

[16] Adamey, E., A. Ersan O., Ümit Ö.; “Collaborative multi-msa multi-target tracking and surveillance: a divide & conquer method using region allocation trees,” *Journal of Intelligent & Robotic Systems*, Vol. 87, 2017, pp. 471-485.
doi: 10.1007/s10846-017-0499-4.

[17] Dentler, J., Rosale, M., Danoy, G., Bouvry, P., Kannan, S., Olivres., M., Voss., H., “Collision avoidance effects on the mobility of a UAV swarm using chaotic ant colony with model predictive control,” *Journal of Intelligent & Robotic Systems*, Vol. 93, 2019, pp. 227-243.
doi:10.1007/s10846-018-0822-8.

[18] Park, C.-H.; Kim, Y.-D.; and Jeong, B. J., “Heuristics for determining a patrol path of an unmanned combat vehicle,” *Computers & Industrial Engineering*, Vol. 63, No.1, 2012, pp. 150 – 160.
doi:10.1016/j.cie.2012.02.007.

[19] Murrieta-Mendoza, A., Botez, R. M., Bunel, A., “Four-Dimensional Aircraft En-Route Optimization Algorithm Using the Artificial Bee Colony,” *AIAA Journal of aerospace information systems*, Vol. 15, No. 6, 2018,

doi: 10.2514/1.I010523.

[20] Haghighi, H., Sadati, H., Karimi, J., Dehghan, M., “A Hierarchical and Prioritized Framework in Coordinated Maneuver of Multiple UAVs Based on Guidance Regulator,” *Journal of Aerospace Technology and Management*, Vol. 11, 2019, pp. 11-19. doi: 10.5028/jatm.v11.999.

[21] Masehian, E., Sedighzadeh, D., “Classic and heuristic approaches in robot motion planning-a chronological review,” *World Academy of Science, Engineering and Technology*, Vol. 23, No 5. 2007, pp. 101-106.

[22] Sabo, C., and Kelly C., “Fuzzy logic unmanned air vehicle motion planning,” *Advances in Fuzzy Systems,* Vol. 13, 2012. doi: 10.1155/2012/989051.

[23] Chuntao, L., Yi X., and Hu M., “Real-time path planning of UAV based on velocity vector,” *Journal of Nanjing University of Aeronautics & Astronautics*, Vol. 44, No.3, 2012, pp. 340-346.

[24] Ceccarelli, N., Enright, J., Frazzoli, E., Rasmussen, S., Schumacher, C., “*Micro UAV path planning for reconnaissance in wind*,” 2007 American Control Conference. *IEEE*, New York, NY, USA, 30 Jul., 2007.

[25] Darrah, M., Wilhem, J., Duling, K., Yokum, S., Sorton, E., Rojas, J., Wathen, M., “A flexible genetic algorithm system for multi-UAV surveillance: algorithm and flight testing.” *Unmanned systems*, Vol. 3, No.1, 2015, pp. 49-62.
doi: https://doi.org/10.1142/S2301385015500041.

[26] Raja, P., and Pugazhenthi, S., “Optimal path planning of mobile robots: A review,” International journal of physical sciences,” Vol 7, No.9, 2012, pp. 1314-1320.
doi: 10.5897/IJPS11.1745

[27] Chankong, V., and Yacov Y. H., “Multi-objective decision making: theory and methodology,” Courier Dover Publications, 2008.

[28] Asadi, D., Atkins E., M., “Multi-Objective Weight Optimization for Trajectory Planning of an Airplane with Structural Damage,” *Journal of Intelligent & Robotic Systems*, Vol. 91, 2017.
doi:10.1007/s10846-017-0753-9.

[29] Asadi, D., Sabzehparvar, M., Atkins, E.M., Talebi, H.A., Damaged airplane trajectory planning based on flight envelope and stability of motion primitives. *J. Aircraft.* Vol 51, No. 6, 2014, pp. 1740–1757.
doi:10.2514/1.C032422.

[30] Taboada, H. A., and David W. C., “Multi-objective scheduling problems: Determination of pruned Pareto sets,” *IEEE* Transactions, Vol. 40, No.5, 2008, pp. 552-564.
doi: 10.1080/07408170701781951.

[31] Ramirez-Atencia, C., Orgaz, G., Moreno., M., Camacho, D., “Solving complex multi-UAV mission planning problems using multi-objective genetic algorithms,” *Soft Computing*, Vol 21, No. 17, 2017, pp. 4883-4900.
doi: 10.1007/s00500-016-2376-7.

[32] Nigam, N., Bieniawski, S., Kroo, I., & Vian, J., Control of multiple UAVs for persistent surveillance: Algorithm and flight test results. *IEEE Transactions on Control Systems Technology*, 20(5), 1236-1251., 2011.
doi: : 10.1109/TCST.2011.2167331.

[33] Peterson, C. K., Casbeer, D. W., Manyam, S. G., & Rasmussen, S. (2020). Persistent Intelligence, Surveillance, and Reconnaissance Using Multiple Autonomous Vehicles with Asynchronous Route Updates, *IEEE Robotics and Automation Letters,* 5(4), 5550-5557., 2020.

[34] Pargett, D., M., and Mark D., A., “Flight path optimization at constant altitude,” *Journal of guidance, control, and* dynamics, Vol 30, No. 4, 2007, pp. 1197-1201.
doi: 10.2514/1.28954.

[35] Standley, Trevor Scott. *"Finding Optimal Solutions to Cooperative Pathfinding Problems",* AAAI. Vol. 1. 2010.

[36] Hart, P., E., Nils J. N., and Bertram R., “A formal basis for the heuristic determination of minimum cost paths,” *IEEE transactions on Systems Science and Cybernetics*, Vol4, No. 2, pp. 100-107,1968.
doi: 10.1109/TSSC.1968.300136.

[37] Bresenham, J., E., “Algorithm for computer control of a digital plotter,” *IBM Systems journal*, Vol 4, No.1, 1965, pp. 25-30.
doi: 10.1147/sj.41.0025.

[38] Kasprzak, E., M., and Kemper L., “Pareto analysis in multi-objective optimization using the collinearity theorem and scaling method,” *Structural and Multidisciplinary Optimization*, Vol. 22, No. 3, 2001, pp. 208-218.
doi: 10.1007/s001580100138.

[39] Geoffrion, A., M., “Proper efficiency and the theory of vector maximization”, *Journal of mathematical analysis and applications*,” Vol. 22, No. 3, 1968, pp. 618-630.
doi: 10.1016/0022-247X(68)90201-1.

[40] Olsen, C. C., Kalyanam, K., Baker, W. P., & Kunz, D. L., Maximal Distance Discounted and Weighted Revisit Period: A Utility Approach to Persistent Unmanned Surveillance. *Unmanned Systems,* 2019, 7(04), 215-232.
doi: 10.1142/S2301385019500079.