

Wind dynamic and energy-efficiency path planning for unmanned aerial vehicles in the lower-level airspace and urban air mobility context

Y.Y. Chan^a, Kam K.H. Ng^{a,*}, C.K.M. Lee^b, Li-Ta Hsu^a, K.L. Keung^b

^a Department of Aeronautical and Aviation Engineering, The Hong Kong Polytechnic University, Hung Hom, Hong Kong Special Administrative Region

^b Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, Hung Hom, Hong Kong Special Administrative Region



ARTICLE INFO

Keywords:

Urban air mobility
Path planning
Energy efficiency
Dynamic wind turbulence and disturbance
Unmanned aerial vehicle
Wind dynamic

ABSTRACT

Unmanned aerial vehicles (UAVs) have been extensively used in urban environments for logistics, parcel delivery and surveillance, and the development of air taxi services. Given the dynamic nature of urban air mobility in terms of decision-making time limit, wind dynamics and other external factors, one should consider their safe and efficient operations in an urban context. Therefore, we propose an energy-efficient path-planning model for UAVs under large and complex urban environments and wind dynamics. The proposed method adopted Voronoi Diagram to decompose the complex urban environment into a simplified network model, given the presence of no-fly zones and restricted areas as obstacles. One could obtain the feasible initial path by solving the network model using the Dijkstra shortest path algorithm concerning the distance matrix. Given the nonlinearity of energy consumption along a path. We further model the UAV energy consumption and propose an efficient particle swarm optimisation (PSO) metaheuristic algorithm to achieve better solution quality. Compared to traditional PSO, the proposed algorithm achieved a 5% to 22% improvement under different wind scenarios by simulating real-life situations. In conclusion, the proposed method can achieve safe UAV operations with sufficient separation and less energy consumption.

Introduction

Urban air mobility (UAM) is a novel transportation system for flying vehicles carrying passengers or goods in point-to-point and short-distance mobility or delivery in urban or suburban contexts [1]. Compared to air traffic management, we need to place special care on UAM safety and ensure an efficient, affordable, safe air transport environment concerning wind dynamics and energy consumption [2–5]. According to the United Nation's final report of World Urbanisation Prospects: The 2018 Revision, over half of the world's population resides in urban areas, and this proportion is expected to enlarge to nearly 70 % by 2050 [6–10]. When developing a new transportation system, many concerns need to be considered, like social acceptance, air traffic management, regulation and safety [11–15]. For example, regulating aircraft maintenance, production, and operation and governing the licensing regime for pilots, aircrew, and aircraft are significant challenges for promoting UAM under government regulation [16–20]. Managing flight risk is essential in safety issues, which includes ensuring human and property safety, and confirming the stability of flight control

systems and ground-based air traffic controllers [15,21]. If all the barriers can be controlled, then the potential market of UAM in the business field is enormous [22,23].

One could expect that, with the increase of UAV adoption in urban air mobility and heightened urbanisation progress, traffic congestion in airspace will subsequently increase and raise several safety concerns for human activities [24–26]. In 2016, Uber published the Elevate Whitepaper for compiling the framework and requirements for providing air taxis service, including planned or non-scheduled on-demand flights [20,27]. Benefiting from the rapid development of modern science and technology, the UAM helps transform mobility from 2D to 3D space movement [28,29]. The vehicle design and requirements, including the criteria of cruise speed, emission, and seat capacity, are essential for UAM implementation. Under UAM, there are different types of aerial vehicles, including Vertical Take-Off and Landing (VTOL), Short Take-Off and Landing (STOL), Small Unmanned Aircraft (SUA), Unmanned Aerial Vehicles (UAV), and Unmanned Aircraft (UA). The passengers of air taxis and the user of logistics transportation services for delivering goods with unmanned control take advantage of UAM to shorten their

* Corresponding author.

E-mail addresses: yin-yuen.chan@connect.polyu.hk (Y.Y. Chan), kam.kh.ng@polyu.edu.hk (K.K.H. Ng), ckm.lee@polyu.edu.hk (C.K.M. Lee), lt.hsu@polyu.edu.hk (L.-T. Hsu), dicky-kin-lok.keung@connect.polyu.hk (K.L. Keung).

traveling time or delivery time and save unexpected costs because of traffic congestion [11].

The key concerns to public safety posed by UAV operations have grown in prominence as their application area has expanded. The increased usage of UAVs in urban areas raises potential accidents and incidents resulting in the loss of life and property if no sufficient safety measures are in place. The UAV operations may overlap with airspace crossed with human activities [30–33]. This move creates several challenges and difficulties for the development of UAV management. Air traffic management may need to be redesigned in the city or region since the order of UAM vehicle's altitude flight needs to be managed to avoid unexpected situations and conflict-free strategies. Therefore, the path design of UAVs must inherently consider the safety factors towards the surrounding obstacles, buildings and terrains [30,31].

Another concern would be energy consumption under wind dynamics. Designing a UAV route from households to commercial regions will always be possible, but the flight time might increase drastically based on changing wind dynamics. More turning points in a path will also increase energy consumption and flight time. Path optimisation with safety concerns can enjoy the benefits of flight time and energy consumption reduction. However, for small aerial vehicles, the impact of wind direction and intensity will exponentially impact the overall flight time and energy consumption. Therefore, considering wind dynamics, one could produce an energy-efficiency UAV path planning with hedging against headwinds and tailwinds.

The key research objectives are enhancing safety awareness among the UAM's stakeholders, improving the effectiveness of flight path distance searching, and considering the energy consumption of UAVs with respect to wind dynamics. The solution method requires special arrangement in network graph design. The most common way to solve path planning problems is by A* and Dijkstra algorithms [34–39]. However, the distance measurement must be linear with a single objective only. Imposing additional operational constraints may require algorithm modification. Furthermore, it is challenging to model the UAV path planning in a free-space map. Based on the nature of path planning in urban air mobility, one could utilise the terrain constraints and formulate a safe network model using by Voronoi diagram [40]. The topological concept of retraction underpins a Voronoi diagram which refers to a skeleton-like structure formed by gradually reducing the open space [41,42]. The Voronoi diagram can be perceived as a foundation as it has widespread usage for finding the best route in path planning and optimisation [41,43]. Dai and Cochran Jr [44] used the Voronoi diagram and parameterised Cornu spirals to design a UAV route, beneficial for path planning and UAV control, which can improve the path in the initial stage by utilising the Cornu spirals' continuous-curvature properties and attempting to match the Voronoi points with the shortest orthogonal distance. Zhang, et al. [45] presented the effect of different parameters on the overall performance of the algorithm proposed and the Voronoi graph by constructing a quantitative model, then discovered that the time required to establish the Voronoi graph and path smoothing is proportional to the number of threats. Pehlivanoglu [46] found that using a multi-frequency vibrational genetic algorithm (mVGA) with a Voronoi diagram in the initial population phase and additional genetic algorithms could be applied to solve the problem of path planning for the autonomous UAVs in a faster and more efficient way. This paper also stated that the simulation using mVGA supported by Voronoi could significantly reduce the computational times in the three-dimensional environment's path planning of autonomous UAVs.

Moreover, the planning algorithms differ in terms of the intended path and environmental information, as simulating the flight environment for route implementation is essential for the path planning and optimisation of multiple aerial vehicles. In a static environment, the consistency theory approach with the Voronoi diagram is used to solve the multiple UAVs path planning problem, which can ensure that the optimal path of multiple UAVs is found and allow multiple UAVs to take off and arrive at predetermined target sites simultaneously. Multiple

targets are hit by the multiple UAVs cooperatively [47,48]. To enhance the realness, the dynamic environment needs to be considered; the obstacles are permitted to move, and the environment can vary substantially over time. Chen, et al. [49] proposed using the Voronoi diagram approach to separate complex airspace challenges, efficiently reduce search space, and shorten planning time. Furthermore, utilising the Dijkstra algorithm with the Voronoi diagram can provide optimum path planning, prompt trajectory alteration, and attack new target points efficiently while the target is shifting, which meets the requirements of path planning and optimisation. However, graph search algorithms are guaranteed to find an optimal solution if it exists, which is the shortest path. Meanwhile, the shortest path is not the most energy-efficient in many cases. As a rule of thumb, the path's maximum curvature must be minimised to avoid sharp turns, such that rapid deceleration and acceleration are avoided. Wind dynamics is also of paramount importance to UAV energy cost. Wind disturbance constitutes most of the drag force experienced by the UAV. In general, the path following the downwind would be more energy efficient. Taking wind drag force into path planning consideration will save the total energy consumption during the whole mission.

Motivated by the abovementioned problems, this paper focuses on a multiple UAV system with varying wind directions under urban air mobility. A novel path planning approach integrating Voronoi diagram generation and Dijkstra shortest path planning is proposed in **Section 2**. **Section 3** presents the construction of the optimisation problem, followed by a modified particle swarm optimisation algorithm with energy cost. **Section 4** conducts multiple instances comparison under different parameter settings for simulating the real-life scenario. **Section 5** highlights the conclusions and future works of this research.

Energy-efficient path planning model for UAVs

This work proposes a novel path planning approach integrating Voronoi diagram generation, Dijkstra shortest path planning, and a modified PSO algorithm with energy cost. To simplify the planning approach, a few assumptions have been made. Firstly, the UAV is assumed to fly at a fixed altitude above building level so the building will impose no disturbance to the wind flow. The configuration space of the UAV is confined to a 2D plane instead of a general 3D area. Secondly, the velocity magnitude is assumed to be constant throughout a mission. It could simplify computation when calculating the path's energy cost. Thirdly, the wind dynamics experienced by the UAV are assumed not to be affecting its stability. Fourthly, as wind dynamics are highly nonlinear, modelling it in real-time is a computationally heavy task, and general mission time is short relative to the change in wind conditions. As a result, the wind dynamics are assumed to be spatially constant throughout the UAV mission. Lastly, other meteorological conditions are assumed as non-factor for the model.

The general framework of the novel path-planning approach. It is divided into four modules: environmental data input, offline roadmap generation, global planner callback, and the UAV itself. The environmental data includes the UAV restricted fly zone and the current wind field data. Typically, the restricted fly zone in the urban area includes the airspace above airports, military facilities such as barracks, hills, prisons, political gatherings, and border control points. Wind field data is an essential data for path optimisation. It is generated through numerical weather forecasting models such as the European Centre for Medium-Range Weather Forecasts (ECMWF) and the Global Forecast System (GFS). In general, it is in the form of a 2D or 3D vector field. The restricted fly zone data usually is in the form of Geographic Information System (GIS) data. In the offline roadmap generation phase, those data would be pre-processed for our path planning algorithms depending on the geographic coordinate system from the GIS data. After that, all data would be stored on the central server [50–57].

The global planner callback is triggered upon the planning request, with the start and destination points as the key parameters. It includes

five sub-processes. Firstly, the pre-processed restricted fly zone data is adapted to generate the Voronoi diagram, the collision-free diagram representing the configuration space of the drone while not using randomised or grid-based approaches. Unreachable nodes and edges must be deleted in the graph whenever edges intersect or when nodes are located inside the obstacles. One additional advantage of the Voronoi diagram approach over the grid-based search and randomised approach is that the Voronoi diagram could be pre-processed, reducing computation time during online planning. In the case of dynamic time spatial obstacles, for example, temporary no-fly zones, Gowda, et al. [58] showed that dynamic addition and deleting of Voronoi nodes and edges is possible on the existing Voronoi diagram.

In the second sub-process, after the diagram generation, given the start point and destination point specified for the UAV mission, the closest Voronoi nodes connecting them have to be determined. The shortest path between the two nodes could be found using graph-based search algorithms. Similar to path planning work by Gowda, et al. [58], our work uses the Dijkstra algorithm. After connecting the starting point and destination with the shortest path, we can acquire the first path candidate.

The first path is far from optimal in terms of distance and energy efficiency. Finding an energy-efficient, short-distance, collision-free path is a multi-objective optimisation problem, and our work proposes utilising a modified PSO to solve it. Depending on the UAV dynamics, a minimum clearance distance C_{min} is used as an input parameter to determine the safety margin between the path and the nearest obstacles. The third sub-process presents an algorithm to generate random paths based on C_{min} and the initial path. The formulation of the fitness function for the path is introduced, which assumes the total energy consumption of the path. The fitness function requires dynamic wind data, as apart from the total distance travelled by the UAV, wind flow constitutes part of the drag force for the UAV to overcome.

The fourth sub-process is path optimisation for determining an energy-efficient path. Canonical PSO is prone to local optimum results, and modifications are made in our work. Time-varying inertia weight is adopted as it has proven successful in various problems. Canonical PSO adopts a global topology where all particle stores the same global best position. Our work adopted the idea of multi-swarm PSO and cluster topology. Particles are divided into subgroups, each with its best global particles. A mutation operator is applied at each subgroup to improve the exploitation ability while avoiding local optimum. All cluster shares information about the global best, and crossover occurs when one cluster suffers from a stagnation problem. Experimental results show that the modified PSO outperforms the canonical PSO and PSO with cluster topology. Regarding the number of clusters, two to six subgroup structures have been tested, and results show that improvements in fitness values are not significant structures with more than three subgroups in our case.

The final subprocess for global planner callback is path smoothing. After the optimisation algorithm, a piece linear path with optimised energy cost would be obtained. However, it would contain sharp or angular turns, which is infeasible for real life UAV operations. A path-smoothing algorithm has to be applied to generate a continuous path for the UAV to follow. In our proposed approach, cubic B-spline smoothing is adopted to obtain a C^2 continuous path that obeys the UAV's acceleration and deceleration constraints. The path would be sent to the UAV in the UAV plant module, which it would report its status back to the central server.

Voronoi diagram and Dijkstra algorithm

The Voronoi diagram is one of the fundamental data structures in computation geometry. In general, it divides a plane into different regions. Each region contains one vertex point such that for every point in the area, the distance between the point and the corresponding region vertex is less than the distance between the point and every other vertex.

Denote S to be a set of n vertices in the configuration space C , namely Voronoi vertices. For a vertex $p \in S$, define the Voronoi region V_p of p as the subset of point in C such that its closest vertex is p . Formally, $V_p = \{x \in C | d(x, p) < d(x, q) \forall q \in S\}$, where $d(x, y)$ is the Euclidean distance between x and y . The Voronoi diagram is the collection of all Voronoi regions.

The computation of the Voronoi diagram construction is based on Barber, et al. [59], which they have further developed and modified, using the Delaunay triangulation as the base. Each Voronoi vertex is the circumcentre of a facet of the Delaunay triangulation. Each Voronoi region corresponds to a vertex (i.e., input site) of the Delaunay triangulation. Then it can be converted back to the Voronoi diagram. Python bindings are available in the Scipy python package. In our research the restricted UAV fly zone issued by the Hong Kong Civil Aviation Department (HKCAD) is adopted for Voronoi Roadmap generation (VD), where the computed VD is shown in Fig. 1(a). The red zone represents the restricted fly zone of UAVs. There are 242 no-fly zones, with each zone represented as a set of points in latitude and longitude. Each point is projected to the World Geodetic System 1984 (WGS84) coordinate system. There are unreachable edges present in the VD. Post-processing of the VD involves two steps. The first step removes nodes within the obstacles. The second step removes edges intersecting or within the obstacles. Fig. 1(b) shows the modified VD. To compute the desired path, determine the nodes closest to the start point and destination. Dijkstra algorithm is applied to find the shortest path between the start node and the end node. Fig. 1(c) is an example of the path generated.

Path expansion with the clearance distance considering wind dynamics

The piecewise linear path is at the maximum distance between obstacles far from optimal. To generate a collision-free path with the consideration of optimised energy efficiency, our work proposes a minimum clearance distance C_{min} and the path would be expanded based on the maximum parameter of C_{min} . C_{min} is the safety factor of the path, and it is a user-defined input that depends on the characteristics of the UAV and the distance between the nearest obstacles. Consider a part of an unsmoothed track $P_{i-1}P_iP_{i+1}$ as shown in Fig. 2. For C_{min} , the free space of the UAV mission would be extended to be the polygon $P'_{i-1}P''_{i-1}P'_{i+1}P''_{i+1}P'_iP'_{i-1}$, and these vertexes are responsible for the polygon's description. Segment $P_{i-1}P_i$, $P'_{i-1}P'_i$ and $P''_{i-1}P''_i$ are parallel. The perpendicular distance is set as C_{min} , which is also the case for interval i to $i + 1$. The area bounded by the red line represents the new free space for path planning based on the initial path, shown as the blue line. The magenta line is an example of the random path generated by the algorithm GENERATE_RANDOM_PATH inside the new free space. The detailed steps are described in Algorithm 1(a) and 1(b) which are based on pure geometry. The generated random path could be visualised as the initial candidate for PSO and MGMPSO.

Algorithm 1(a)..

The random waypoint generation

	GENERATE_TRACK_OFFSET
Input	$C_{min}, P_{i-1}, P_i, P_{i+1}$
Output	$PointP_{r,i}$
1	$\theta_{i-1} := atan2(P_i.y - P_{i-1}.y, P_i.x - P_{i-1}.x)$
2	$\theta_i := atan2(P_{i+1}.y - P_i.y, P_{i+1}.x - P_i.x)$
3	$\phi_i := \frac{\pi - \theta_i - \theta_{i-1}}{2}$
4	$l_i := \frac{C_{min}}{\sin(\phi_i)}$
5	$l_i := l_i * random(0, 1)$
6	$P_{r,i}.x = P_i.x + l_i \cos(\phi_i)$
7	$P_{r,i}.y = P_i.y - l_i \sin(\phi_i)$
8	RETURN $P_{r,i}$

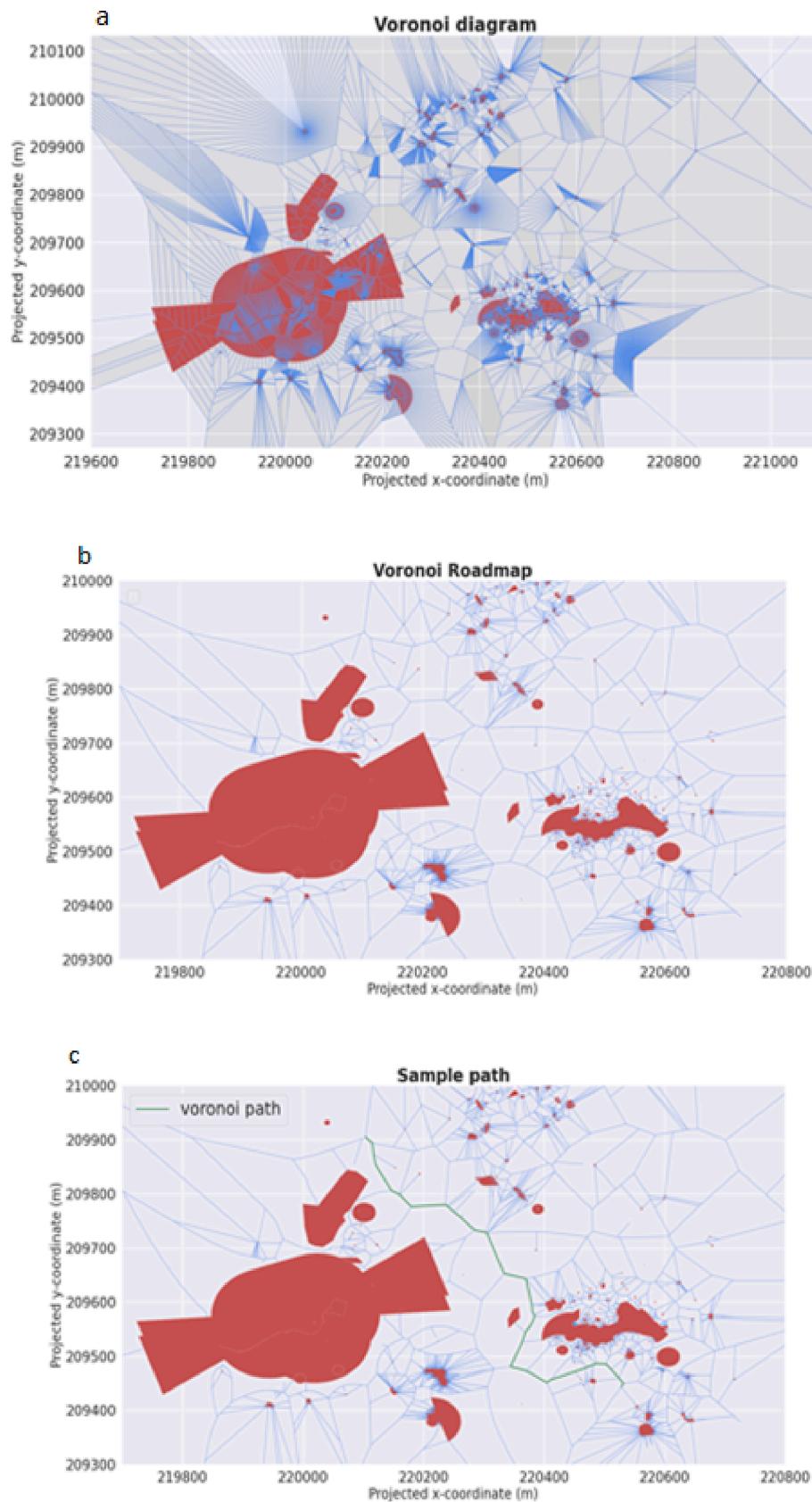


Fig. 1. (a) The Voronoi diagram generated by the obstacle information, including 24,775 vertices and 42,065 edges. (b) The Voronoi diagram after unreachable nodes and edges are eliminated, including 11,835 nodes and 30,052 edges. (c) Dijkstra shortest path algorithm in the Voronoi diagram (An example of the start point and end point in (latitudes and longitudes) is (114.19, 22.26) and (114.19, 21.24) respectively).

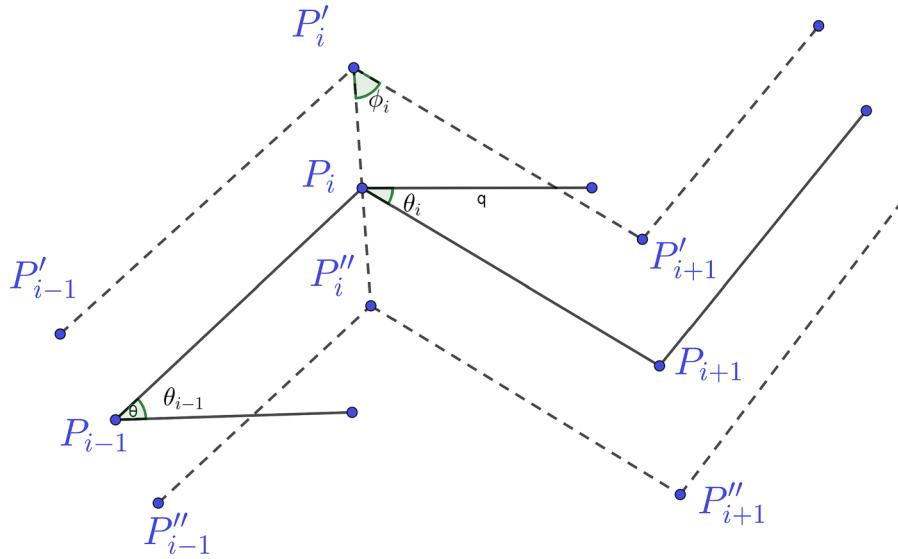


Fig. 2. The logic of generating a random waypoint.

Algorithm 1(b).

The random path generation

GENERATE_RANDOM_PATH

Input	C_{min} , setofpoints $P_i, i = 1, 2, \dots, N$
Output	setofpoints $\{P_i^{random_path}\}, i = 1, 2, \dots, N$

```

1    $P_{random\_path}^i = [] //$  set of waypoints of the random path
2    $P_{random\_path}^i.append(P_0)$  // start point is set as the same point
3   FOR  $i = 2$  to  $n = 1$  DO
4      $P_{r,i} = GENERATE_TRACK_OFFSET(C_{min}, P_{i-1}, P_i, P_{i+1})$ 
5      $P_{random\_path}^i.append(P_{r,i})$ 
6      $P_{random\_path}^i.append(P_n)$  // endpoint is set as the same point
7   RETURN  $P_{upper\_path}, P_{lower\_path}$ 
```

Fitness function

The fitness function is to evaluate the relative energy consumption of the path. Based on the results from Krell, et al. [60], accurate measurements of UAV energy consumption require computation on different operation modes, such as in idle mode, taking-off, hovering, and flying horizontally. Therefore, under these circumstances, our fitness function aims to estimate the path's energy level while minimising the fitness calculation's computation burden. The proposed fitness function has the form from **Equation (1)**.

$$E(Path) = \sum_{i=1}^{N-1} w(P_i, P_{i+1}) \quad (1)$$

w is the energy function for the path segment $\overline{P_i P_{i+1}}$. As stated in the assumption, UAV velocity \vec{v}_{UAV} remains constant throughout the mission. Along $\overline{P_i P_{i+1}}$, with wind speed \vec{v}_{wind} , the relative UAV velocity \vec{v}_r is given by **Equation (2)**.

$$\vec{v}_r = \vec{v}_{UAV} - \vec{v}_{wind} \quad (2)$$

Without consideration of nonlinearity in aerodynamics, the drag force experienced by UAV is under **Equation (3)**.

$$D = \frac{1}{2} \rho \vec{v}_r^2 C_d A \quad (3)$$

ρ is the air density, C_d is the drag coefficient and A is the reference area of the UAV. Therefore $w(P_i, P_{i+1})$ is given by the work done to overcome

the drag force on the segment $\overline{P_i P_{i+1}}$.

$$w(P_i, P_{i+1}) = |D|d(P_i, P_{i+1}), \quad (4)$$

where $d(P_i, P_{i+1})$ is the euclidean distance of P_i and P_{i+1} .

Modified particle swarm optimisation (MGMPSO) for UAV energy-efficiency path planning model

Given the nonlinearity of the model and consideration of wind dynamics, the path model cannot be solved directly by a mixed-integer linear programming solver. Particle Swarm Optimisation (PSO) is a popular *meta-heuristics* optimisation method that has proven its ability to optimise various nonlinear constrained optimisation problems [60–63]. The origin of the algorithm is to mimic the flocking of birds or schooling of fish to look for food. The original PSO algorithm would be introduced first, followed by the modifications for our proposed optimisation problem, multi-group mutation PSO (MGMPSO). A swarm of particles P is conceptualised as a set of moving points in d-dimensional problem space, each point $x_i = (x_{i1}, x_{i2}, \dots, x_{iD})^T$ with its associated vectors $v_i = (v_{i1}, v_{i2}, \dots, v_{iD})^T$. A fitness function for the particles has to be defined for evaluating the particle's cost value. Denote each particle's local best position as $p_{best,i}$, where $i = 1, 2, \dots, N$ for total of N particles. The best position among all particle is set as $g_{best} = (x_{g_{best,1}}, x_{g_{best,2}}, \dots, x_{g_{best,D}})^T$. At each iteration, particles are updating its position according to the associated velocity vector, shown in **Equation (5)**.

$$x_{k+1,i} = x_{k,i} + v_{k+1,i} \quad (5)$$

where k is the current step number. At each iteration, the velocity of every particle will be calculated as follows:

$$v_{k+1,i} = \omega v_{k,i} + c_1 r_1 (p_{best,i} - x_{k,i}) + c_2 r_2 (g_{best} - x_{k,i}) \quad (6)$$

where ω is the current velocity weight, c_1 and c_2 are the local and global weights, r_1 and r_2 are two random numbers in the range $[0, 1]$, $x_{k,i}$ is the current position of the particle.

A modified particle swarm optimiser

The problem of the canonical PSO is its vulnerability to local optimum. There are many variations developed and proven to improve PSO optimisation. We have adopted the idea of diminishing inertia weight,

mutation mechanism in velocity update, and PSO in multi-group structure into our algorithm.

Diminishing particle swarm optimiser

In general, velocity inertia weight ω is an essential factor affecting the exploration-exploitation trade-off in PSO. Large ω encourages global exploration, whereas small ω is favorable for local exploitation. At the beginning of the search, more search space should be explored to avoid falling into local optimum. A delicate search is required for a better near-optimal solution in the end phase. Our algorithm adopted the results from Tsujimoto, et al. [64], decreasing ω linearly from 0.95 to 0.4, by [Equation \(7\)](#), where K is the number of maximum iteration and k and the current iteration.

$$\omega = \omega_{0.95} - \frac{(\omega_{0.95} - \omega_{0.4})}{K} k \quad (7)$$

Multi-group PSO and crossover

The PSO swarm P is divided into M sub-swarms G_1, G_2, \dots, G_M , associated with their own $g_{best,j}, j = 1, 2, \dots, M$. $g_{best'}$ is defined as the current best particle among all sub-swarms. Our objective is to detect the g_{best} by falling stagnation and potentially local optimum. After each iteration, g_{global_best} would be updated. For each, the stagnation status of the sub-swarms would be found by Worasucheep [65]. When the sub-swarm shows signs of stagnation, the mutation would be triggered in that its g_{best} would be replaced by $g_{best'}$ with a small Gaussian disturbance, keeping its sub-swarm different from the original sub-swarm with $g_{best'}$.

Path smoothing by cubic spline interpolation

After the optimisation algorithm, a linear path with optimised energy cost is obtained. However, it does not obey the acceleration and deceleration characteristics of the UAV. A path-smoothing algorithm has to be applied to generate a continuous path for the UAV to follow. One of the popular approaches would be smoothing using Dubins curves. In the piecewise linear path case, this algorithm would concatenate line segments by arcs of circles, and the circle's radius is determined by considering the maximum rate of change of turn of the robot. The Dubins method has been extended to other more-complex vehicle models but is still limited to line segments and arcs of circles [66–69]. One drawback is that the Dubins curve is C^1 continuous only. Its 1st order derivative is not continuing at the junction of the line and arc. The final path would be smoothed by cubic spline interpolation to ensure that the path is C^2 continuous, where no sharp turn is present in the path. We followed the approach described in Worasucheep [65] for the path synthesis using B-spline curves, which provides evident advantages when compared to other geometry-based smoothing approaches [70–73].

Results and discussion

All numerical simulations ran on a desktop computer with AMD® Ryzen 7 5800x 8-core CPU, with 32 GB ram in operation system Ubuntu 20.04 LTS. All codes are written in Python 3.8. There are, in total, 242 no-fly zones on the Hong Kong flight map. The initial Voronoi diagram construction generated 24,775 nodes and 42,089 edges. Subsequent refinements on the Voronoi diagram removed 12,920 nodes and 12,013 edges. Voronoi diagram-related computations and adjacency matrix construction for running the Dijkstra algorithm are realised by the python NumPy library, with a total computation time of 46.74296 s. The initial path computed by the Dijkstra algorithm served as the benchmark for comparing with the path optimised by canonical and modified PSO. Three sample sets of start and destination points are selected for path

planning. The effect of PSO and MGMPSO optimisation under different wind fields, different UAV speeds, and different C_{min} is investigated. To evaluate the overall performance of Voronoi path, PSO, and MGMPSO from a statistical view, significant differences between them were determined by using the non-parametric Wilcoxon signed-rank test. Three indicators, including p-value, R^+ and R^- can be obtained. P-value presents the p-value of the paired, two-sided test for the null hypothesis. If the p-value is less than 0.05, we can reject the null hypothesis at the 5 % significance level. R^+ indicates the sum of all the ranks of the sign “+”, while R^- indicates the sum of the ranks of a sign. It has been adopted to compare the performance of different PSO-based algorithms in different applications [70–73].

Wind speed is of paramount importance for the calculation of the energy fitness function. Wind field data from 07 to 09-2022 to 09–09-2022, recorded at noon (GMT + 8) are used of estimation of wind speed. The primary source of wind dynamics is wind forecasting services based on Numerical Weather Predictions (NWP), Windy [74]. The European Centre for Medium-Range Weather Forecasts (ECMWF) data provided by Windy is adopted. It is a global forecast system that generates meteorological data, including wind dynamics.

The start point and end point are defined in latitude and longitude. Three sets of random points (113.97, 22.24), (114.29, 22.44), (113.82, 22.47), (114.35, 22.44) and (114.19, 22.26), (114.20, 21.24) are selected. Wind field data are sampled on 07–09-2022, 08–09-2022, and 09–09-2022, defined as wind fields 1, 2, and 3, respectively, and are provided by the Hong Kong Observatory. It is assumed that the wind field is spatially invariant in the mission. Following the initial recommendation from Kohler, et al. [73] and Zhang, et al. [72], the population size of PSO is set to 50 [66,67], whereas, for MGMPSO, 51 is divided into three subgroups. Both maximum iterations of PSO and MGMPSO are limited to 50. The summary of the Voronoi path, PSO path, and MGMPSO path data are formalised in [Table 1\(a\)](#).

Same wind field under different routes

The three sets of missions under the three sets of wind fields and the paths' results are shown in [Fig. 3a, b, and c](#), respectively. The blue line represents the Voronoi path, the orange line represents the path optimised by canonical PSO, and the green line is the path optimised by MGMPSO. The light blue arrows represent the wind field. Both PSO and MGMPSO-optimised graphs can generate a more energy-efficient path. Extreme edges are eliminated in both approaches. We defined the following set of parameters [start-end-point pair = (113.97, 22.24), (114.29, 22.44), wind field set = set 1, UAV speed = 10.0 m/s, $C_{min}=10.0$ m] as the baseline. Changing only one parameter while keeping the others at baseline values during each mission, we conducted missions with 2 other start-end-point pairs under the same wind field for general comparison; 2 with alternative wind field sets for testing headwind, tailwind and crosswind; 3 mission sets with changing UAV velocity, and 2 other sets of C_{min} . To summarise, there were 10 sets of scenarios completed. Set $a = \{1, 2, 3, \dots, 10\}$ is a set of instances considering different parameters for testing purposes, including the different start and end points, wind field scenario, and speed of UAV. Pair (113.97, 22.24), (114.29, 22.44) is adopted to evaluate the performance in the different wind fields, different C_{min} and different UAV speeds. The parameters are generated with the rationale of simulating real-life scenarios. [Table 1\(a\)](#) shows the results of different PSOs and the improvement from the improved PSO under set a circumstances. Set {2} has the best optimisation result, with a 14.09 % improvement compared to the Voronoi path. It is attributed to the wind flow from west to east, which formed a tailwind condition favorable for set {1} and {2}, and the magnitude of the wind field in the free space of the set {2} is more significant than in set {1}.

Table 1a

Results of the tested instances.

Set	start point (lat, lon)	end point (lat, lon)	Wind field set	UAV speed (m/s)	C_{min} (m)	Voronoi path cost	Canonical PSO path cost	Voronoi-PSO Improvement (%)	MGMPSO path cost	Voronoi-MGMPSO improvement (%)
1	(113.97,22.24)	(114.29, 22.44)	1	10.0	10.0	3372.04	3159.16	6.31	3101.98	8.01
2	(113.82,22.47)	(114.35, 22.44)	1	10.0	10.0	3938.54	3444.09	12.55	3383.74	14.09
3	(114.19, 22.26)	(114.20, 21.24)	1	10.0	10.0	3208.41	3088.32	3.74	3048.08	5.00
4	(113.97,22.24)	(114.29, 22.44)	2	10.0	10.0	3223.53	3031.38	5.96	2974.38	7.73
5	(113.97,22.24)	(114.29, 22.44)	3	10.0	10.0	3552.62	3,375.30	4.99	3,357.22	5.50
6	(113.97,22.24)	(114.29, 22.44)	1	5.0	10.0	3318.91	3022.88	8.92	3051.39	8.06
7	(113.97,22.24)	(114.29, 22.44)	1	15.0	10.0	3800.60	3515.94	7.49	3463.38	8.87
8	(113.97,22.24)	(114.29, 22.44)	1	20.0	10.0	4604.40	4156.82	9.72	4127.85	10.35
9	(113.97,22.24)	(114.29, 22.44)	1	10.0	20.0	3372.04	3088.98	8.73	3060.97	9.56
10	(113.97,22.24)	(114.29, 22.44)	1	10.0	30.0	3372.04	3038.84	10.08	2620.93	22.44

Different wind fields under fixed routes

Set {1}, {4} and {5} compare the effect of different fields, which is shown in Table 1(a). All three sets share the same start point (113.97, 22.24) and endpoint (114.29, 22.44). Set {1} and {4} have a similar magnitude of improvement of 8.01 % and 7.73 %, respectively, where set {5} improvement magnitude is lower at 5.5 %. It is attributed to the fact that the direction of wind field 3 is from west to east, but the flight path is from east to west. The UAV would be facing a headwind, so extra energy is required. Also, the magnitude of the wind is more significant than wind field 1. The two factors led to less capability for optimisation. In terms of the energy cost, set {4} has a lower energy cost with a value of 2974.38. It is because the direction of the second wind field flow is from northeast to southwest. The path faced a partially favourable crosswind condition. However, the magnitude of the second wind field is much larger than the first wind field. It follows the energy cost from set {1} with a value of 3101.98. Set {5} has an enormous path cost of 3357.22, as a higher amount of extra energy is needed to overcome the headwind.

Same wind field, same route under different energy costs of UAV

Set {1}, {6}, {7} and {8} were used to compare the effect of different UAV speeds, and the result is shown in Table 1. Regarding energy cost, it could be deduced that when UAV speed is higher than 10 m/s, the energy consumption would also increase corresponding to UAV speed. For set {6} with the UAV speed set to 5 m/s, the energy cost of the MGMPSO path was 3051.39, whereas the cost was 4127.85 for the UAV speed set to 20 m/s. It can be attributed to the drag force experienced by UAVs as it is proportional to the square of UAV velocity, and the total energy consumption increases despite the reduction in flight time. In terms of optimal performance in the proposed algorithm, different UAV speeds did not lead to a difference in performance. The percentage of optimisation of MGMPSO of the four sets is similar, at 8.01 %, 8.09 %, 8.06, and 10.35 %.

Same wind field, same route under different C_{min}

Set {1}, {9} and {10} were used to compare the effect of different minimum clearance C_{min} . Increase in C_{min} leads to an increase in computational efficiency. When $C_{min} = 10m$, the MGMPSO path cost was 3101.98 with 8.01 % optimisation relative to the Voronoi path,

while for $C_{min} = 30m$, the MGMPSO path cost was 2620.93 with a 22.44 % optimisation. It contributes to the fact that increasing C_{min} increases the free space extended by the Voronoi path. Larger free space means there exists a more energy-efficient path. Meanwhile, enlarging C_{min} infinitely would cause the path to collide with obstacles. C_{min} should be chosen with consideration of safety regulations and obstacle zones on the map.

Validation and discussion

To compare the performance of canonical PSO and MGMPSO, the performance of PSO and MGMPSO in all ten sets is plotted in Fig. 4. It was observed that MGMPSO outperforms PSO in all 10 cases. In addition, pairwise comparison results using a non-parametric Wilcoxon signed-rank test were conducted. The improvement of PSO and MGMPSO towards the initial path by the Voronoi diagram is summarised in Table 1(b). The result of the Wilcoxon test. R_+ denotes the sum of all positive ranks, while R_- denotes the sum of all opposing ranks. The level of significance considered for the statistical test is 0.05. The p-value calculated is 0.012, indicating that MGMPSO outperforms the canonical PSO. Fig. 4 shows the comparison of PSO and MGMPSO in all ten sets of experimental conditions. The proposed MGMPSO performs better than PSO in all ten cases. In terms of convergence rate, in all ten cases, MGMPSO converges faster than PSO in the first ten iterations. When a multi-group of particles exists in the search space, there would be more potential global optimum found by the swarm of particles at the start of the search process. Hence, the exploration efficiency would be higher than PSO, with the initial convergence rate being faster. Fig. 4(e) and (g) show the multi-group crossover mechanism's effect. The convergence rate is slow in both cases after only a few iterations. The optimisation process showed signs of stagnation, implying the particles were trapped in the local optimum. The algorithm detects the presence of stagnation by checking the convergence rate in each iteration. When the convergence rate is slow, it triggers the crossover mechanism. It leads to alternating global optimum among sub-groups and eventually reactivating the particles to explore the free space. Fig. 4(e) and (g) echo with the mechanism, and the optimisation result improves after the mutation. In terms of the convergence rate of the MGMPSO algorithm, it was observed that MGMPSO reaches optimum in fewer iterations than PSO on average. Exceptional cases occurred in set {10}, $c_{min} = 3$ and set {6}, $speed = 5$. In set {10} where the UAV speed is reduced to 5 m/s, PSO reached optimum after 16 iterations while MGMPSO was still

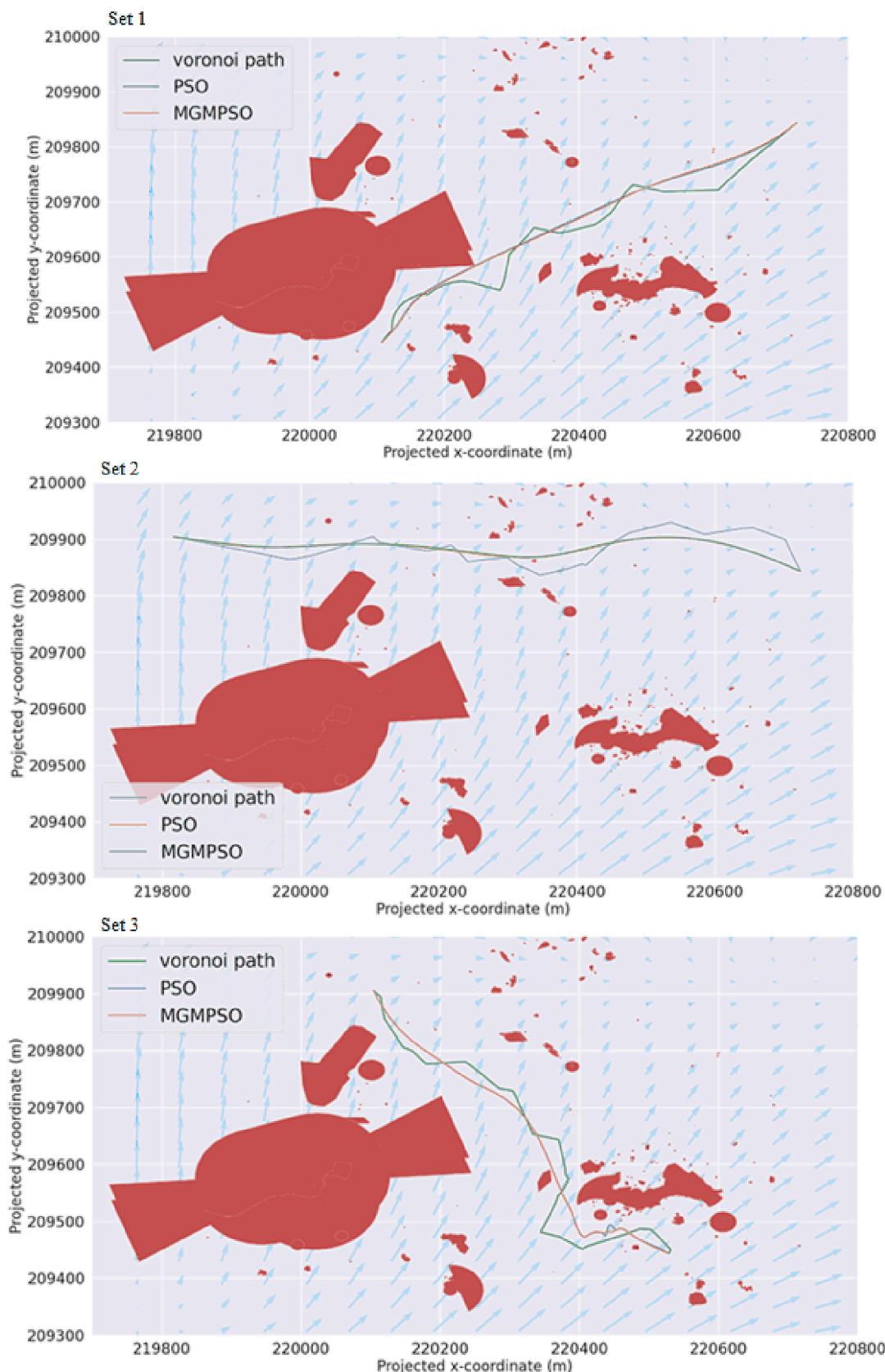


Fig. 3. Voronoi path, PSO optimised path and MGMPSO optimised path in different wind yield set: Set 1 from (113.97,22.24) to (114.29, 22.44); Set 2 from (113.82,22.47) to (114.35, 22.44) and Set 3 from (114.19, 22.26) to (114.20, 21.24).

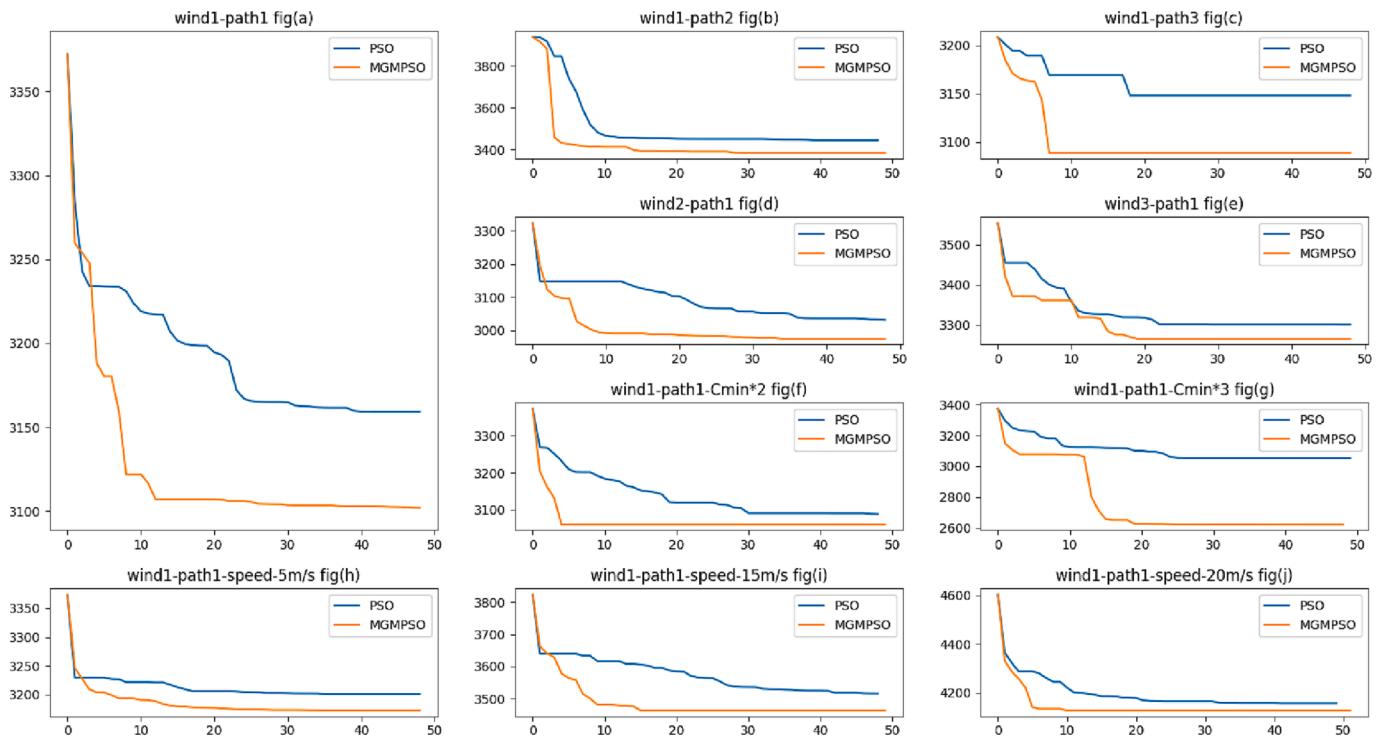


Fig. 4. A comparison figure between canonical PSO and MGMPSO under set α .

Table 1b

Wilcoxon signed-rank test and improvement of fitness score by PSO and MGMPSO.

Set	Improvement by PSO	Improvement by MGMPSO
1	212.88	270.06
2	494.45	554.80
3	120.09	160.33
4	192.15	249.15
5	177.32	195.40
6	296.03	267.52
7	284.66	337.22
8	447.58	476.55
9	283.06	311.07
10	333.20	751.11

optimising. In set {10} where c_{\min} was increased to 20 m/s, PSO started converging after 10 iterations, while MGMPSO was observed to escape the local optimum at the 13th iteration. However, in both cases, MGMPSO outperforms PSO despite showing slower convergence rate. It is worth mentioning that the absolute running time is not a revealing measure for our case because the code is still not optimised. The performance would increase when it is rewritten in a compiled language, especially C and C++.

Finally, the proposed approach is implemented and tested using the rotorS simulator [75]. It is a UAV simulator using Gazebo as the simulation engine with inbuilt APIs using the ROS framework, which includes a custom wind field API and an API for trajectory input. The custom environment is built with the help of the OpenStreetMap (OSM) database. It could be used for further analysis, such as tests for dynamic route re-planning, multi-UAV navigation, and real-time collision avoidance. In terms of energy consumption, our experimental results shows that MGMPSO outperforms PSO by 1.25 % for set {1}, 11.2 % for set {2}, and 6.67 % for set {3}. The cost reduction percentage does not echo the fitness cost in Table 1(a). It is attributed to the simplification of the fitness function compared to the drone dynamics in the rotorS simulation. A simplified fitness function boosts the search process. It is expected that the discrepancy will increase regarding the real-world

test. Therefore, one of the possible directions for future research could be the design of the fitness function to increase the correlation between the fitness value and the real-world flight performance of the UAV.

Future works and conclusion

The new era of sustainable urban air mobility attracts academics' and industrial practitioners' attention to develop an energy-efficiency approach for achieving carbon neutrality and energy-efficient goals [76–79]. As a factor affecting path allocation, wind dynamic is always omitted by the current literature on path planning for UAVs. In a lower-level airspace context, a Voronoi Diagram is considered to decompose the complex environment into a simplified diagram, given the presence of no-fly zones as obstacles. A cost function for the path considering total energy consumption was formulated. To solve the path optimisation problem, a modified PSO was adapted to provide a near-optimal solution with reduced computational time. The current method was investigated based on numerical simulations. It needs to be brought into real-life applications for the justification and realisation of our work. One extension of our work is a large-scale implementation of a multiple UAV system or the urban air mobility system, which utilises urban airspace to a full extent to address the traffic congestion problem. The proposed work could predetermine the route for the airmobile, which consumes much less energy. In short, it could be a reference for stakeholders to support the development of UAM or other multiple UAV-based systems.

In the future, the focus will be threefold. First, as the proposed algorithm is 2D-based, it could be extended to a 3D to prepare for scenarios where more layers of airspace are open to UAVs. In the Voronoi roadmap generation phase, 3D Generalised Voronoi Diagram (GVD) could be adopted to construct the initial Voronoi network, and shortest path algorithms could still be applied as the graph structure is retained. However, there are two hurdles in the 3D path optimisation phase. Firstly, the solution search space for optimisation would be augmented from $N \times 2$ dimensions to $N \times 3$ dimensions, where N is the number of control waypoints. In general, poor quality results are generated when metaheuristic-based algorithms are used to deal with large and complex search spaces. When the search space is large, its convergence speed

becomes very slow near the global optimum [69–73]. Especially, PSO-based algorithms often fail to search for a globally optimal solution when the objective function has many dimensions. The reason for this phenomenon is not only owing to the local optima but the velocities of the particles sometimes trapped in degeneracy. The successive range was eventually restricted in the sub-plane of the whole search hyperplane [80–88]. Even though our results showed that MGMPSO possess the ability to escape from local optima, its performance in 3D path optimisation is not yet tested. Secondly, the availability of vertical shear wind data in the urban environment is limited. The wind field data we gathered is limited to mesoscale 2D wind fields. When it comes to complex urban physics modelling, extensive use of the meteorological model coupled with the Computational Fluid Dynamics (CFD) model is required to simulate the wind field data required for optimising the UAV path [69–73]. It is also the rationale behind the assumption that the flight altitude of the UAV plane is assumed to be over the buildings, such that buildings have no impact to the wind field. The two hurdles would be our future research directions: investigating MGMPSO in a higher dimension and incorporating a meteorological-CFD coupled model. Another focus would be extending the algorithm from one UAV to covering multiple UAVs with multiple goals in the environment. Multiple paths would lead to path conflicts, and local collision detection and avoidance strategies with consideration of minimum energy consumption have to be formulated. The other direction that should be further extended is the cross-effect of different parameters tested in Section 4. Different combinations with different parameters could be further tested and connected under different scenarios. A database could be generated for future machine learning training proposes for path recommendations.

CRediT authorship contribution statement

Y.Y. Chan: Conceptualization, Methodology, Data curation, Formal analysis, Validation, Investigation, Writing – original draft, Writing – review & editing. **Kam K.H. Ng:** Conceptualization, Validation, Resources, Writing – review & editing, Supervision, Project administration, Funding acquisition, Supervision. **C.K.M. Lee:** Writing – review & editing, Validation. **Li-Ta Hsu:** Conceptualization, Investigation, Writing – review & editing. **K.L. Keung:** Writing – review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgment

The work described in this paper was supported by the *Department of Aeronautical and Aviation Engineering, The Hong Kong Polytechnic University, Hong Kong SAR (RJ1D)* and by the *Research Institute for Sustainable Urban Development (BBWK), The Hong Kong Polytechnic University, Hong Kong SAR*.

References

- [1] Mur  a MCR. Identification and prediction of urban airspace availability for emerging air mobility operations. *Transportation Research Part C: Emerging Technologies* 2021;131:103274. <https://doi.org/10.1016/j.trc.2021.103274>.
- [2] Roga S, Bardhan S, Kumar Y, Dubey SK. Recent technology and challenges of wind energy generation: A review. *Sustainable Energy Technol Assess* 2022;52:102239. <https://doi.org/10.1016/j.seta.2022.102239>.
- [3] D. K. R and S. Chavhan. Shift to 6G: Exploration on trends, vision, requirements, technologies, research, and standardization efforts. *Sustainable Energy Technol Assess* 2022;54:102666. <https://doi.org/10.1016/j.seta.2022.102666>.
- [4] V. patki, et al. Improving the geo-drone-based route for effective communication and connection stability improvement in the emergency area ad-hoc network. *Sustainable Energy Technol Assess* 2022;53:102558. <https://doi.org/10.1016/j.seta.2022.102558>.
- [5] Manikandan M, Vaidya E, Pant RS. Design and analysis of hybrid electric multi-lobed airship for cargo transportation. *Sustainable Energy Technol Assess* 2022;51: 101892. <https://doi.org/10.1016/j.seta.2021.101892>.
- [6] H. Ritchie and M. Roser, "Urbanization," *Our world in data*, 2018.
- [7] P. D. Vascik and R. J. Hansman, "Development of vertiport capacity envelopes and analysis of their sensitivity to topological and operational factors," in *AIAA Scitech 2019 Forum*, 2019, p. 0526.
- [8] K. K. H. Ng, K. L. Keung, C. K. M. Lee, and Y. T. Chow, "A Large Neighbourhood Search Approach to Airline Schedule Disruption Recovery Problem," in *2020 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, 14-17 Dec. 2020 2020, pp. 600-604, doi: 10.1109/IEEM5057.2020.9309768.
- [9] J. Huo, K. L. Keung, C. K. M. Lee, K. K. H. Ng, and K. C. Li, "The Prediction of Flight Delay: Big Data-driven Machine Learning Approach," in *2020 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, 14-17 Dec. 2020 2020, pp. 190-194, doi: 10.1109/IEEM5057.2020.9309919.
- [10] K. L. Keung, C. K. M. Lee, K. K. H. Ng, and C. K. Yeung, "Smart City Application and Analysis: Real-time Urban Drainage Monitoring by IoT Sensors: A Case Study of Hong Kong," in *2018 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, 16-19 Dec. 2018 2018, pp. 521-525, doi: 10.1109/IEEM.2018.8607303.
- [11] Cohen AP, Shaheen SA, Farrar EM. Urban air mobility: History, ecosystem, market potential, and challenges. *IEEE Trans Intell Transp Syst* 2021;22(9):6074–87.
- [12] Wang Z, Xu Z, Wang X, Xie M. A temporal-spatial cleaning optimization method for photovoltaic power plants. *Sustainable Energy Technol Assess* 2022;49:101691. <https://doi.org/10.1016/j.seta.2021.101691>.
- [13] Yiu CY, et al. Towards safe and collaborative aerodrome operations: Assessing shared situational awareness for adverse weather detection with EEG-enabled Bayesian neural networks. *Adv Eng Inf* 2022;53:101698. <https://doi.org/10.1016/j.aei.2022.101698>.
- [14] Li Q, et al. Effects of the multisensory rehabilitation product for home-based hand training after stroke on cortical activation by using NIRS methods. *Neurosci Lett* 2020;717:134682. <https://doi.org/10.1016/j.neulet.2019.134682>.
- [15] Li Q, Ng KK-H, Fan Z, Yuan X, Liu H, Bu L. A human-centred approach based on functional near-infrared spectroscopy for adaptive decision-making in the air traffic control environment: A case study. *Adv Eng Inf* 2021;49:101325. <https://doi.org/10.1016/j.aei.2021.101325>.
- [16] Wang S, Zhan X, Zhai Y, Zheng L, Liu B. Enhancing navigation integrity for Urban Air Mobility with redundant inertial sensors. *Aerospace Sci Technol* 2022;107631.
- [17] Wu P, Yang X, Wei P, Chen J. Safety Assured Online Guidance With Airborne Separation for Urban Air Mobility Operations in Uncertain Environments. *IEEE Trans Intell Transp Syst* 2022.
- [18] Reiche C, Cohen AP, Fernando C. An initial assessment of the potential weather barriers of urban air mobility. *IEEE Trans Intell Transp Syst* 2021;22(9):6018–27.
- [19] Cokirlo O. Urban air mobility: safety challenges. *Transp Res Procedia* 2020;45: 21–9.
- [20] D. P. Thipphavong et al., "Urban air mobility airspace integration concepts and considerations," in *2018 Aviation Technology, Integration, and Operations Conference*, 2018, p. 3676.
- [21] Q. Li, C. Y. Yiu, S. C. M. Yu, and K. K. H. Ng, "Situational Awareness and Flight Approach Phase Event Recognition Based on Psychophysiological Measurements," in *2021 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, 13-16 Dec. 2021 2021, pp. 1308-1312, doi: 10.1109/IEEM50564.2021.9673081.
- [22] Xiao H, Jiang H, Shi F-R, Luo Y, Deng L-P. Energy efficient resource allocation in delay-aware UAV-based cognitive radio networks with energy harvesting. *Sustainable Energy Technol Assess* 2021;45:101204. <https://doi.org/10.1016/j.seta.2021.101204>.
- [23] Wang N, Li B, Duan Y, Jia S. A multi-energy scheduling strategy for orderly charging and discharging of electric vehicles based on multi-objective particle swarm optimization. *Sustainable Energy Technol Assess* 2021;44:101037. <https://doi.org/10.1016/j.seta.2021.101037>.
- [24] Wang Z, Delahaye D, Farges J-L, Alam S. Complexity optimal air traffic assignment in multi-layer transport network for Urban Air Mobility operations. *Transportation Research Part C: Emerging Technologies* 2022;142:103776. <https://doi.org/10.1016/j.trc.2022.103776>.
- [25] Bennaceur M, Delmas R, Hamadi Y. Passenger-centric Urban Air Mobility: Fairness trade-offs and operational efficiency. *Transportation Research Part C: Emerging Technologies* 2022;136:103519. <https://doi.org/10.1016/j.trc.2021.103519>.
- [26] Garrow LA, German BJ, Leonard CE. Urban air mobility: A comprehensive review and comparative analysis with autonomous and electric ground transportation for informing future research. *Transportation Research Part C: Emerging Technologies* 2021;132:103377. <https://doi.org/10.1016/j.trc.2021.103377>.
- [27] Fu M, Rothfeld R, Antoniou C. Exploring preferences for transportation modes in an urban air mobility environment: Munich case study. *Transp Res Rec* 2019;2673 (10):427–42.
- [28] Tahir B, Tariq M. Vulnerability assessment and federated intrusion detection of Air Taxi enabled smart cities. *Sustainable Energy Technol Assess* 2022;53:102686. <https://doi.org/10.1016/j.seta.2022.102686>.

- [29] Li Z, Pu O, Pan Y, Huang B, Zhao Z, Wu H. A study on measuring wind turbine wake based on UAV anemometry system. *Sustainable Energy Technol Assess* 2022; 53:102537. <https://doi.org/10.1016/j.seta.2022.102537>.
- [30] Pang B, Low KH, Lv C. Adaptive conflict resolution for multi-UAV 4D routes optimization using stochastic fractal search algorithm. *Transportation Research Part C: Emerging Technologies* 2022;139:103666. <https://doi.org/10.1016/j.trc.2022.103666>.
- [31] Li A, Hansen M, Zou B. Traffic management and resource allocation for UAV-based parcel delivery in low-altitude urban space. *Transportation Research Part C: Emerging Technologies* 2022;143:103808. <https://doi.org/10.1016/j.trc.2022.103808>.
- [32] Du W, Guo T, Chen J, Li B, Zhu G, Cao X. Cooperative pursuit of unauthorized UAVs in urban airspace via Multi-agent reinforcement learning. *Transportation Research Part C: Emerging Technologies* 2021;128:103122. <https://doi.org/10.1016/j.trc.2021.103122>.
- [33] Chen AY, Chiu Y-L, Hsieh M-H, Lin P-W, Angah O. Conflict analytics through the vehicle safety space in mixed traffic flows using UAV image sequences. *Transportation Research Part C: Emerging Technologies* 2020;119:102744. <https://doi.org/10.1016/j.trc.2020.102744>.
- [34] Keung KL, Lee CKM, Ji P. Industrial internet of things-driven storage location assignment and order picking in a resource synchronization and sharing-based robotic mobile fulfillment system. *Adv Eng Inf* 2022;52:101540. <https://doi.org/10.1016/j.aei.2022.101540>.
- [35] Keung KL, et al. Edge intelligence and agnostic robotic paradigm in resource synchronisation and sharing in flexible robotic and facility control system. *Adv Eng Inf* 2022;52:101530. <https://doi.org/10.1016/j.aei.2022.101530>.
- [36] Keung KL, Lee CKM, Ji P, Ng KKH. Cloud-based Cyber-Physical Robotic Mobile Fulfillment Systems: A Case Study of Collision Avoidance. *IEEE Access* 2020;1. <https://doi.org/10.1109/ACCESS.2020.2992475>.
- [37] Keung KL, Lee CKM, Ji P. Data-driven order correlation pattern and storage location assignment in robotic mobile fulfillment and process automation system. *Adv Eng Inf* 2021;50:101369. <https://doi.org/10.1016/j.aei.2021.101369>.
- [38] Ng KKH, Lee CKM, Zhang SZ, Keung KL. The impact of heterogeneous arrival and departure rates of flights on runway configuration optimization. *Transportation Letters* 2020;1–12. <https://doi.org/10.1080/19427867.2020.1852496>.
- [39] K. L. Keung, C. K. M. Lee, P. Ji, and J. Huo, "Cloud-based Cyber-Physical Robotic Mobile Fulfillment Systems Considering Order Correlation Pattern," in *2020 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, 14-17 Dec. 2020 2020, pp. 113-117, doi: 10.1109/IEEM45057.2020.9309904.
- [40] B. Boots, K. Sugihara, S. N. Chiu, and A. Okabe, "Spatial tessellations: concepts and applications of Voronoi diagrams," 2009.
- [41] Aurenhammer F, Klein R. *Voronoi Diagrams. Handbook of computational geometry* 2000;5(10):201–90.
- [42] Aurenhammer F. Voronoi diagrams—a survey of a fundamental geometric data structure. *ACM Computing Surveys (CSUR)* 1991;23(3):345–405.
- [43] S. Fortune, "A sweepline algorithm for Voronoi diagrams," in *Proceedings of the second annual symposium on Computational geometry*, 1986, pp. 313-322.
- [44] Dai R, Cochran Jr J. Path planning and state estimation for unmanned aerial vehicles in hostile environments. *J Guid Control Dynam* 2010;33(2):595–601.
- [45] C. Zhang, H. Liu, and Y. Tang, "Quantitative evaluation of Voronoi graph search algorithm in UAV path planning," in *2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS)*, 2018: IEEE, pp. 563-567.
- [46] Pehlivanoglu YV. A new vibrational genetic algorithm enhanced with a Voronoi diagram for path planning of autonomous UAV. *Aerosp Sci Technol* 2012;16(1): 47–55.
- [47] X. Chen, G.-y. Li, and X.-m. Chen, "Path planning and cooperative control for multiple UAVs based on consistency theory and Voronoi diagram," in *2017 29th Chinese control and decision conference (CCDC)*, 2017: IEEE, pp. 881-886.
- [48] X. Chen and X. Chen, "The UAV dynamic path planning algorithm research based on Voronoi diagram," in *The 26th chinese control and decision conference (2014 ccc)*, 2014: IEEE, pp. 1069-1071.
- [49] Chen X, Chen X, Xu G. The path planning algorithm studying about UAV attacks multiple moving targets based on Voronoi diagram. *International Journal of Control and Automation* 2016;9(1):281–92.
- [50] Zhu F, Feng J, Xie M, Li L, Lei J, Lee J. Profile Abstract: An Optimization-based Subset Selection and Summarization Method for Profile Data Mining. *IEEE Trans Ind Inf* 2022;1–12. <https://doi.org/10.1109/TII.2022.3227642>.
- [51] Lui CF, Liu Y, Xie M. A Supervised Bidirectional Long Short-Term Memory Network for Data-Driven Dynamic Soft Sensor Modeling. *IEEE Trans Instrum Meas* 2022;71: 1–13. <https://doi.org/10.1109/TIM.2022.3152856>.
- [52] Jing T, Zheng P, Xia L, Liu T. Transformer-based hierarchical latent space VAE for interpretable remaining useful life prediction. *Adv Eng Inf* 2022;54:101781. <https://doi.org/10.1016/j.aei.2022.101781>.
- [53] Xia L, Zheng P, Li X, Gao RX, Wang L. Toward cognitive predictive maintenance: A survey of graph-based approaches. *J Manuf Syst* 2022;64:107–20. <https://doi.org/10.1016/j.jmsy.2022.06.002>.
- [54] Zheng P, Xia L, Li C, Li X, Liu B. Towards Self-X cognitive manufacturing network: An industrial knowledge graph-based multi-agent reinforcement learning approach. *J Manuf Syst* 2021;61:16–26. <https://doi.org/10.1016/j.jmsy.2021.08.002>.
- [55] Xia L, Zheng P, Huang X, Liu C. A novel hypergraph convolution network-based approach for predicting the material removal rate in chemical mechanical planarization. *J Intell Manuf* 2021/05/24 2021,. <https://doi.org/10.1007/s10845-021-01784-1>.
- [56] Liu C, Zheng P, Xu X. Digitalisation and servitisation of machine tools in the era of Industry 4.0: a review. *Int J Prod Res* 2021;1:33. <https://doi.org/10.1080/00207543.2021.1969462>.
- [57] Liu C, Su Z, Xu X, Lu Y. Service-oriented industrial internet of things gateway for cloud manufacturing. *Rob Comput Integr Manuf* 2022;73:102217. <https://doi.org/10.1016/j.rcim.2021.102217>.
- [58] Gowda I, Kirkpatrick D, Lee D, Naamad A. Dynamic voronoi diagrams. *IEEE Trans Inf Theory* 1983;29(5):724–31.
- [59] Barber CB, Dobkin DP, Huhdanpaa H. The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software (TOMS)* 1996;22(4):469–83.
- [60] Krell E, King SA, Carrillo LRG. Autonomous Surface Vehicle energy-efficient and reward-based path planning using Particle Swarm Optimization and Visibility Graphs. *Appl Ocean Res* 2022;122:103125.
- [61] Clerc M. Particle swarm optimization. John Wiley & Sons; 2010.
- [62] Poli R, Kennedy J, Blackwell T. Particle swarm optimization. *Swarm Intell* 2007;1(1):33–57.
- [63] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95-international conference on neural networks*, 1995, vol. 4: IEEE, pp. 1942-1948.
- [64] T. Tsujimoto, T. Shindo, T. Kimura, and K. Jin'no, "A relationship between network topology and search performance of PSO," in *2012 IEEE Congress on Evolutionary Computation*, 2012: IEEE, pp. 1-6.
- [65] C. Worasueep, "A particle swarm optimization with stagnation detection and dispersion," in *2008 IEEE congress on evolutionary computation (IEEE world congress on computational intelligence)*, 2008: IEEE, pp. 424-429.
- [66] Liu Z, Liu H, Lu Z, Zeng Q. A Dynamic Fusion Pathfinding Algorithm Using Delaunay Triangulation and Improved A-Star for Mobile Robots. *IEEE Access* 2021; 9:20602–21. <https://doi.org/10.1109/ACCESS.2021.3055231>.
- [67] T. Gawron and M. M. Michałek, "A G3-Continuous Extend Procedure for Path Planning of Mobile Robots with Limited Motion Curvature and State Constraints," *Applied Sciences*, vol. 8, no. 11, doi: 10.3390/app8112127.
- [68] Elbanhawi M, Simic M, Jazar RN. Continuous Path Smoothing for Car-Like Robots Using B-Spline Curves. *J Intell Rob Syst* 2015/12/01 2015;,80(1):23–56. <https://doi.org/10.1007/s10846-014-0172-0>.
- [69] Y. Guang and V. Kapila, "Optimal path planning for unmanned air vehicles with kinematic and tactical constraints," in *Proceedings of the 41st IEEE Conference on Decision and Control*, 2002, 10-13 Dec. 2002 2002, vol. 2, pp. 1301-1306 vol.2, doi: 10.1109/CDC.2002.1184695.
- [70] Yamada T, Koike K. Downscaling mesoscale meteorological models for computational wind engineering applications. *J Wind Eng Ind Aerodyn* 2011;99(4):199–216. <https://doi.org/10.1016/j.jweia.2011.01.024>.
- [71] Esmin AAA, Coelho RA, Matwin S. A review on particle swarm optimization algorithm and its variants to clustering high-dimensional data. *Artif Intell Rev* 2015/06/01 2015;,44(1):23–45. <https://doi.org/10.1007/s10462-013-9400-4>.
- [72] Zhang D, Ma G, Deng Z, Wang Q, Zhang G, Zhou W. A self-adaptive gradient-based particle swarm optimization algorithm with dynamic population topology. *Appl Soft Comput* 2022;130:109660. <https://doi.org/10.1016/j.asoc.2022.109660>.
- [73] Kohler M, Villasuso MMBR, Tanscheit R. PSO+: A new particle swarm optimization algorithm for constrained problems. *Appl Soft Comput* 2019;85:105865. <https://doi.org/10.1016/j.asoc.2019.105865>.
- [74] "Windy as forecasted," Windy.com/. [Online]. Available: <https://www.windy.com/?22.336%2C114.185%2C5>. [Accessed: 31-Mar-2023].
- [75] F. Furter, M. Burri, M. Achtelik, and R. Siegwart, "RotorS—A Modular Gazebo MAV Simulator Framework," in *Robot Operating System (ROS)*, (Studies in Computational Intelligence, 2016, ch. Chapter 23, pp. 595-625.
- [76] Liu Y, Lin R, Ren J. Developing a life cycle composite footprint index for sustainability prioritization of sludge-to-energy alternatives. *J Clean Prod* 2021; 281:124885. <https://doi.org/10.1016/j.jclepro.2020.124885>.
- [77] Lin R, Lu S, Yang A, Shen W, Ren J. Multi-criteria sustainability assessment and decision-making framework for hydrogen pathways prioritization: An extended ELECTRE method under hybrid information. *Int J Hydrogen Energy* 2021;46(24): 13430–45. <https://doi.org/10.1016/j.ijhydene.2021.01.018>.
- [78] Lin R, Liu Y, Man Y, Ren J. Towards a sustainable distributed energy system in China: decision-making for strategies and policy implications. *Energy, Sustainability and Society* 2019/12/23 2019;,9(1):51. <https://doi.org/10.1186/s13705-019-0237-9>.
- [79] Liu Y, Lin R, Man Y, Ren J. Recent developments of hydrogen production from sewage sludge by biological and thermochemical process. *Int J Hydrogen Energy* 2019;44(36):19676–97. <https://doi.org/10.1016/j.ijhydene.2019.06.044>.
- [80] Fan J, Zheng P, Li S, Wang L. An Integrated Hand-Object Dense Pose Estimation Approach With Explicit Occlusion Awareness for Human-Robot Collaborative Disassembly. *IEEE Trans Autom Sci Eng* 2022;1–10. <https://doi.org/10.1109/TASE.2022.3215584>.
- [81] Xia L, Liang Y, Zheng P, Huang X. Residual-Hypergraph Convolution Network: A Model-Based and Data-Driven Integrated Approach for Fault Diagnosis in Complex Equipment. *IEEE Trans Instrum Meas* 2022;1. <https://doi.org/10.1109/TIM.2022.3227609>.
- [82] Fan J, Zheng P, Li S. Vision-based holistic scene understanding towards proactive human–robot collaboration. *Rob Comput Integr Manuf* 2022;75:102304. <https://doi.org/10.1016/j.rcim.2021.102304>.
- [83] Zhang X, Zheng P, Peng T, He Q, Lee CKM, Tang R. Promoting employee health in smart office: A survey. *Adv Eng Inf* 2022;51:101518. <https://doi.org/10.1016/j.aei.2021.101518>.
- [84] Li S, Fan J, Zheng P, Wang L. Transfer Learning-enabled Action Recognition for Human-robot Collaborative Assembly. *Procedia CIRP* 2021;104:1795–800. <https://doi.org/10.1016/j.procir.2021.11.303>.

- [85] Li S, Wang R, Zheng P, Wang L. Towards proactive human–robot collaboration: A foreseeable cognitive manufacturing paradigm. *J Manuf Syst* 2021;60:547–52. <https://doi.org/10.1016/j.jmsy.2021.07.017>.
- [86] Li S, Zheng P, Fan J, Wang L. Towards Proactive Human Robot Collaborative Assembly: A Multimodal Transfer Learning-Enabled Action Prediction Approach. *IEEE Trans Ind Electron* 2021;1. <https://doi.org/10.1109/TIE.2021.3105977>.
- [87] Li S, Zheng P, Zheng L. An AR-Assisted Deep Learning-Based Approach for Automatic Inspection of Aviation Connectors. *IEEE Trans Ind Inf* 2021;17(3): 1721–31. <https://doi.org/10.1109/TII.2020.3000870>.
- [88] Liu B, Zhang Y, Zhang G, Zheng P. Edge-cloud orchestration driven industrial smart product-service systems solution design based on CPS and IIoT. *Adv Eng Inf* 2019; 42:100984. <https://doi.org/10.1016/j.aei.2019.100984>.