Transportation Research Part E 193 (2025) 103859

Contents lists available at ScienceDirect

Transportation Research Part E

journal homepage: www.elsevier.com/locate/tre

Air Corridor Planning for Urban Drone Delivery: Complexity Analysis and Comparison via Multi-Commodity Network Flow and Graph Search

Xinyu He [a], Lishuai Li [a,b,*], Yanfang Mo [d], Zhankun Sun [c], S. Joe Qin [d]

[a] Hong Kong Institute for Data Science, City University of Hong Kong, Hong Kong Special Administrative Region of China
[b] School of Data Science, City University of Hong Kong, Hong Kong Special Administrative Region of China
[c] Department of Management Sciences, City University of Hong Kong, Hong Kong Special Administrative Region of China
[d] School of Data Science, Lingnan University, Hong Kong Special Administrative Region of China

ARTICLE INFO

Keywords:
Unmanned aircraft system traffic management (UTM)
Drone delivery
Multi-path planning
Network flow theory
Graph search

ABSTRACT

Urban drone delivery, a rapidly evolving sector, holds the potential to enhance accessibility, address last-mile delivery issues, and alleviate ground traffic congestion in cities. Effective Unmanned Aircraft System Traffic Management (UTM) is essential to scale drone delivery. A critical aspect of UTM involves planning a city-wide network with spatially-separated air corridors (air routes). Most existing works have focused on routing problems or air traffic management. Compared to these problems, the air corridor planning problem requires much higher spatial and temporal resolutions and presents computational challenges due to the scale, complexity, and density of urban airspace, along with the coupling issues of multi-path planning. Therefore, we conducted this research to understand the complexity and computational resources required to optimally solve the air corridor planning problem. In this paper, we use a minimum-cost Multi-Commodity Network Flow (MCNF) model, a mathematical model, to model the problem and demonstrate the complexity of air corridor planning through the complexity of MCNF. We then apply Gurobi's and GLPK's integer programming (IP) solvers to find optimal solutions. Additionally, we present two existing multi-path graph search algorithms, the Sequential Route Network Planning (SRP) algorithm and the Distributed Route Network Planning (DRP) algorithm, to address this corridor planning problem. Numerical experiments conducted at various scales and settings using IP solvers and graph search algorithms indicate that finding an optimal solution requires significant computational resources and yields only a slight improvement in optimality compared to graph search algorithms. Thus, air corridor planning is complex both theoretically and numerically, and graph search algorithms can provide a feasible solution with good enough optimality for corridor planning in real-world scenarios. Moreover, the multi-path graph search algorithms can easily incorporate side constraints that are known to be impossible to solve with polynomial algorithms, making it more practical for real-world applications. Finally, we demonstrate the application of SRP and DRP in real-world 3D urban scenarios.

* Corresponding author at: School of Data Science, City University of Hong Kong, Hong Kong Special Administrative Region of China.
E-mail addresses: x.y.he@my.cityu.edu.hk (X. He), lishuai.li@cityu.edu.hk (L. Li), yanfangmo@LN.edu.hk (Y. Mo), zhanksun@cityu.edu.hk (Z. Sun), joeqin@ln.edu.hk (S.J. Qin).

https://doi.org/10.1016/j.tre.2024.103859
Received 11 June 2024; Received in revised form 6 November 2024; Accepted 7 November 2024
Available online 27 November 2024

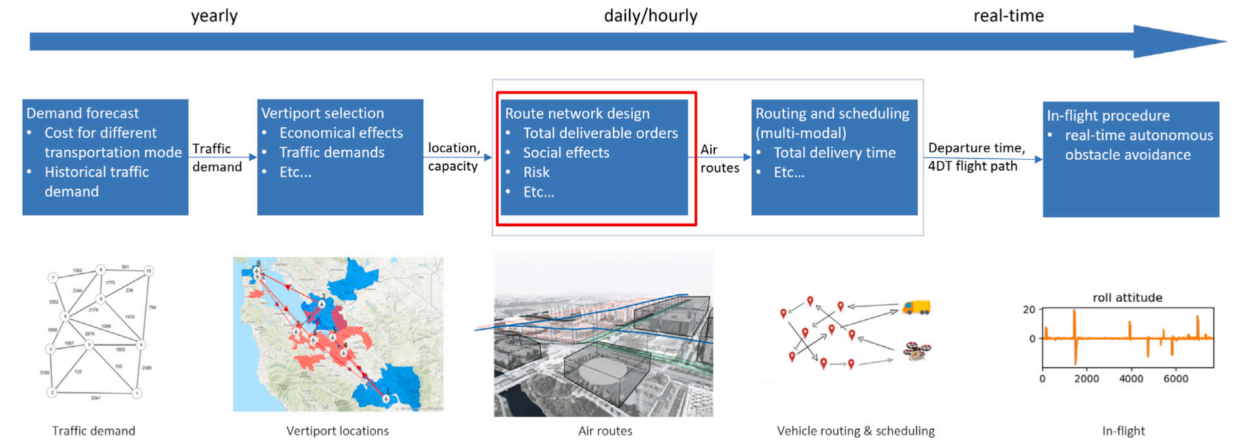


Fig. 1. Planning and operations in drone delivery service.

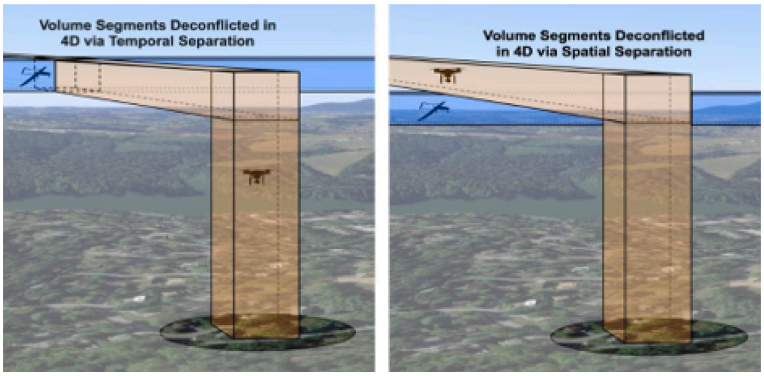


Fig. 2. Temporal separation vs. spatial separation in different UTM ConOps (NASA, 2022).

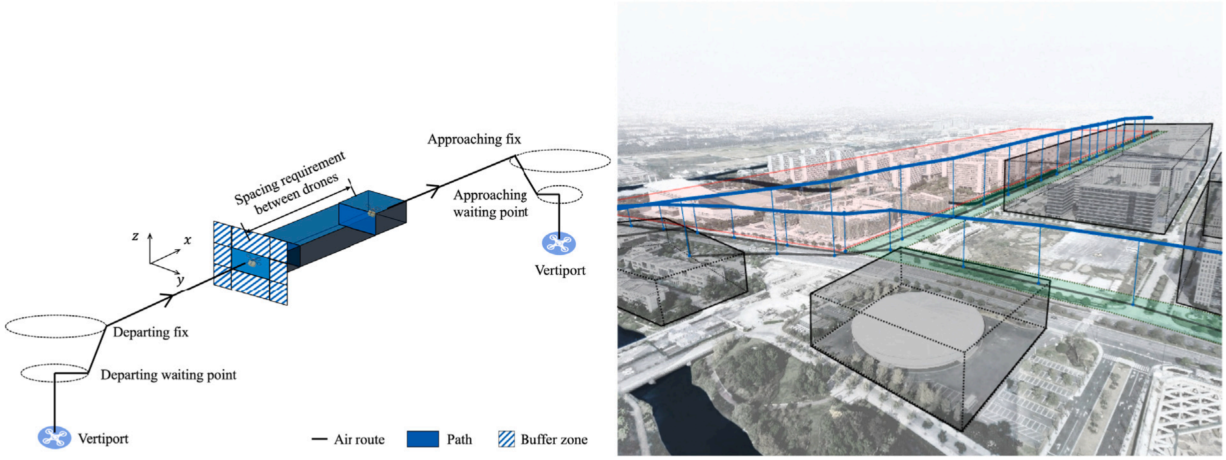
1. Introduction

Urban drone delivery is an emerging aspect of Urban Air Mobility (UAM), set to revolutionize urban logistics by mitigating vehicular traffic congestion and diminishing human labor in distribution networks (Lemardelé et al., 2021). The planning and operational aspects of drone delivery services are illustrated in Fig. 1. Crucial to the management of a large number of drone operations in complex urban environments is the route network design, as highlighted in the figure. This involves the planning of designated air corridors (air routes). Currently, most research focuses on the routing (Murray and Raj, 2020) and air traffic management problems (Chin et al., 2021); in contrast, this research focuses on the air corridor planning problem. This problem requires much higher spatial and temporal resolution, and this presents computational challenges due to the scale, complexity, and density of urban airspace, along with the coupling issues of multi-path planning. Therefore, the primary focus of this research is on the computational complexity and solutions for designing such air corridor networks.

Several concepts of operations (ConOps) for Unmanned Aircraft System Traffic Management (UTM) have been explored (NASA, 2022; SESAR, 2021; Mohamed Salleh et al., 2018; Ushijima, 2017). These ConOps specify the airspace management approaches, including when and how to utilize temporal and/or spatial separation techniques to regulate drone traffic, thereby facilitating the integration of drones into the urban transportation matrix, as depicted in Fig. 2.

UAV operations employing a structured network of air corridors or routes have been implemented for routine drone delivery operations in some Chinese cities (Yang, 2023). These drones are designated specifically for transporting goods to strategically placed collection hubs located near clusters of residential or commercial buildings, rather than offering direct-to-consumer doorstep delivery. This strategy facilitates the use of predetermined flight paths that connect launching sites to collection hubs, thereby streamlining navigation challenges in complex urban landscapes. Fig. 3 depicts a schematic illustration of this organized air corridor.

In this route network, multiple vertiports serve as nodal points for UAV transit. The air routes are conceptualized as unidirectional air corridors that facilitate connectivity between the vertiports, more precisely, linking their respective approach and departure waypoints. These air corridors are spatially separated to avoid potential conflicts. UAVs adhere to a sequential flight pattern along these air corridors, remaining within the designated corridor and maintaining a regulated inter-spatial distance from adjacent UAVs to ensure safety and efficiency.



(a) An air corridor (air route) connecting an OD pair

(b) A section of a network with two air corridors

Fig. 3. Illustration for the air corridors (structured route network) (He et al., 2024).

The term "the path", as used in the present discourse, refers to the rectangular corridor encompassing the air route, with its dimensions meticulously calibrated to ensure the isolation of UAVs on adjacent paths. To enhance safety protocols, an additional "buffer zone" envelops the path, acting as a supplementary safety interstice. For an exhaustive elucidation of the air route network's architecture and its constituent elements, one is directed to consult the comprehensive analysis presented in He et al. (2022).

The detailed design requirements of route networks vary under different ConOps, yet a core and universal problem is multi-path planning for air corridor planning. Most existing methodologies are inadequate for addressing the complexities involved in planning air route networks. Current Multi-Agent Path Finding (MAPF) algorithms are predominantly tailored for executing specific tasks within a drone fleet. However, these algorithms were not developed with consideration for the design of systems or infrastructures that accommodate various types of drones managed by different operators. In other words, they do not facilitate the construction of "roads" in the sky. Some graph-search-based algorithms have been applied to solve the air corridor planning problem by considering its unique constraints in real-world scenarios (He et al., 2022, 2024). However, their optimality and computational performances have not been systemically analyzed.

Network flow theory is a mathematical model that focuses on optimizing certain types of network systems, such as logistical networks, routing networks, etc. In such a network, the target is to send an amount of flow, such as packages and data, from the source to the target while satisfying a set of constraints. The Network flow model can potentially provide a benchmark for air corridor planning problems, as an optimal solution can be sought. Yet, to the best of our knowledge, no one has conceptualized and formulated the air corridor planning problem into network flow models, and the computational complexity of such an application has not been thoroughly investigated.

Therefore, this research develops a minimum-cost Multi-Commodity Network Flow (MCNF) mathematical model for air corridor planning in an urban environment and analyzes its computational complexity. Then we use Gurobi's (Gurobi, 2021) and the GLPK's (GLPK, 2021) integer programming (IP) solvers to find an optimal solution for the MCNF model. We also present two existing multi-path graph search algorithms, "Sequential Route Network Planning (SRP)" (He et al., 2022) and "Distributed Route Network Planning (DRP)" (He et al., 2024), to solve the optimization problem, and compared their performance with IP solvers.

2. Literature review

2.1. UTM and air route network planning

There is a vast literature in air traffic management focusing on air route design, collision risk models, and collision avoidance (Blom et al., 2003; Delahaye et al., 2003; Hoekstra et al., 2002; Kochenderfer et al., 2010; Perrin et al., 2007). However, existing air route design is considering traditional aircraft operated by human pilots in airspace organized by sectors, which is significantly different from UTM. Recent UTM studies focus on ConOps development and system-level design, e.g. the safe and efficient integration of UAS into ATM (Aweiss et al., 2018; Cone et al., 2018; EUROCONTROL, 2018; Kopardekar and Bradford, 2017; Vascik et al., 2018; Vascik and Hansman, 2017). Operational constraints for urban air mobility from the perspective of air traffic control have been identified and studied in a number of papers (Vascik et al., 2018; Vascik and Hansman, 2017; Vascik and Jung, 2016; Wang et al., 2022a,b). The capacity envelop is analytically estimated via an Integer Programming (IP) approach (Vascik and Hansman, 2019). Some initial analysis has been performed to explore suitable traffic management concepts and their associated risks (Aweiss et al., 2018; Jung et al., 2018; Sunil et al., 2015). Chin et al. (2021) considers the fairness of delay assignment in the context of UAS Traffic Flow Management (UTFM). However, very limited efforts have been made to develop specific air corridors for UAS operations in cities (Tang et al., 2021; Wang et al., 2021; Wu and Zhang, 2021).

2.2. Network flow models for network planning problems

A network flow model is a mathematical tool for modeling and addressing how items are transported between locations, using a network with limited capacities on arcs. A flow network is a directed graph $G = (\mathcal{N}, \mathcal{A})$ where $\mathcal{N}$ denotes the set of nodes and $\mathcal{A}$ denotes the set of arcs connecting the nodes (Ahuja et al., 1993). Each arc imposes a cost on the flow passing through it and has an associated capacity that limits the amount of flow it can carry. The incoming and outgoing flows at each node must be equal unless the node is designated as a source (origin) or sink (destination), which is the flow conservation constraint for the flow models. Two relevant network flow problems are the minimum cost flow problem and its generalization — minimum cost multi-commodity network flow problem.

Minimum cost flow problems seek to route a certain amount of flow through a network from source nodes $\mathcal{S}$ to sink nodes $\mathcal{T}$ so as to minimize the total cost. Let $(i,j) \in \mathcal{A}$ represent an arc from node i to node j, with associated cost per unit flow c_{ij} and lower and upper capacity bounds l_{ij} and u_{ij} respectively. Letting x_{ij} represent the amount of flow on arc (i,j), to send an amount of flow equal to d_j, the minimum cost flow problem can be formulated as follows:

$$\min z(x) = \sum_{(i,j)\in\mathcal{A}} c_{ij} x_{ij} \tag{1a}$$

$$\text{s.t.} \quad \sum_{\{j:(i,j)\in\mathcal{A}\}} x_{ij} - \sum_{\{j:(j,i)\in\mathcal{A}\}} x_{ji} = \begin{cases} -d_j & \forall j \in \mathcal{S} \\ 0 & \forall j \in \mathcal{N}/(\mathcal{S}\cup\mathcal{T}) \\ d_j & \forall j \in \mathcal{T} \end{cases} \tag{1b}$$

$$l_{ij} \le x_{ij} \le u_{ij} \quad \forall (i,j) \in \mathcal{A} \tag{1c}$$

$$x_{ij} \ge 0 \quad \forall (i,j) \in \mathcal{A} \tag{1d}$$

Eq. (1a) minimizes the total cost of flow passing through the edges. Eq. (1b) represents the flow conservation constraint, which ensures that the difference between inflow and outflow at each node is equal to $-d_j/0/d_j$ for source/transshipment/sink nodes. Here d_j refers to the demand for the flow to node j. Eq. (1c) imposes capacity constraints on each arc, while Eq. (1d) ensures that the flow on each arc is non-negative. In minimum cost network flow problems, when multiple sources and sinks exist, they can be clustered into a virtual source node and a virtual sink node, reducing the network to a single-source single-sink $s-t$ flow network. The demand for the virtual source node and the virtual sink node will be $\sum_{j\in\mathcal{S}} -d_j$ and $\sum_{j\in\mathcal{T}} d_j$. Many polynomial algorithms have been developed to derive the solution, like cycle canceling methods (Klein, 1967), cost scaling methods (Hassin, 1983), network simplex methods that are based on linear programming simplex (Orlin, 1997), etc. Minimum-cost flow is extensively applied in many applications, like resources allocation (Cappanera and Scaparra, 2011; Wang et al., 2008), assortment (Frank, 1965; Pentico, 2008), interdependent infrastructure systems modeling (Ahangar et al., 2020; Goldbeck et al., 2019; Lee II et al., 2007).

The shortest path problem is a minimum-cost flow problem with infinite arc capacities that sends one unit of flow. It finds the shortest directed path from a source node to a sink node, which is similar to the single path finding problem in the network planning problem. Many label-setting and label-correcting algorithms are developed for shortest path problem (Bellman, 1958; Floyd, 1962; Fredman and Tarjan, 1987; Hart et al., 1968), among which Dijkstra's algorithm is a classic one (Dijkstra, 1959). Shortest path problems are extensively applied in robotic path (Hu et al., 1993), digital mapping services (Saab and VanPutte, 1999; Quddus and Washington, 2015), IP routing problem (Amaldi et al., 2013), etc. Maximum flow problems are complementary and can be transformed to minimum-cost flow problems. They maximize the total amount of flow while honoring the arc capacities. Many polynomial algorithms have been developed like Dinic's algorithm (Dinitz, 1970), Ford-Fulkerson algorithm (Ford and Fulkerson, 1956), and push-relabel algorithm (Goldberg and Tarjan, 1988). Multi-agent path planning in Yu and LaValle (2013) was formulated as a maximum flow problem, but the considered problem only brought each agent to any goal position in the set of goal positions instead of a specific goal position.

The minimum cost flow problem models the flow of a single commodity over a network. In contrast, multi-commodity network flow (MCNF) problems arise when multiple commodities share the same underlying network. It is a generalization of minimum cost flow problems and can be applied to describe multi-path finding problems. The minimum cost MCNF problem aims to achieve a given amount of flow for each commodity while minimizing the total cost. Let x_{ij}^n represent the amount of flow for commodity n on arc (i,j) and c_{ij}^n represent the cost per unit flow for commodity n on arc (i,j). Given N commodities with respective flows $d_{n,j}$ from source nodes $\mathcal{S}_n$ to sink nodes $\mathcal{T}_n$, the minimum cost MCNF problem can be formulated as follows:

$$\min \sum_{n=1}^{N} \sum_{(i,j)\in\mathcal{A}} c_{ij}^n x_{ij}^n \tag{2a}$$

$$\text{s.t.} \quad \sum_{\{j:(i,j)\in\mathcal{A}\}} x_{ij}^n - \sum_{\{j:(j,i)\in\mathcal{A}\}} x_{ji}^n = \begin{cases} -d_{n,j} & \forall j \in \mathcal{S}_n \\ 0 & \forall j \in \mathcal{N}/(\mathcal{S}_n\cup\mathcal{T}_n) \\ d_{n,j} & \forall j \in \mathcal{T}_n \end{cases} \quad \forall n \in [N] \tag{2b}$$

$$l_{ij} \le \sum_{n=1}^{N} x_{ij}^n \le u_{ij} \quad \forall (i,j) \in \mathcal{A} \tag{2c}$$

$$x_{ij}^n \ge 0 \quad \forall (i,j) \in \mathcal{A}, \forall n \in [N] \tag{2d}$$

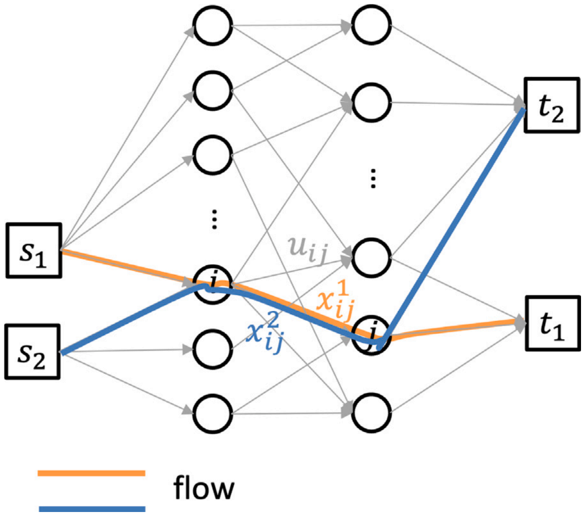


Fig. 4. An illustration for an MCNF problem.

Eq. (2a) minimizes the total cost of flow for all commodities. Eq. (2b) represents the flow conservation constraint for each commodity. Eq. (2c) imposes capacity constraints on each arc, where the flow on an arc is the sum of flows for all commodities. Eq. (2d) ensures that the flow for each commodity on each arc is non-negative. The primary difference between problems Eqs. (1) and (2) lies in the number of commodities. The coupling of commodities in capacity constraints (Eq. (2c)) makes the MCNF problem significantly more challenging to solve than its single-commodity counterpart. An illustration of an MCNF problem is shown in Fig. 4. Though multi-source multi-sink minimum cost network flow problems can be reduced to single-source single-sink minimum cost network flow problems, such reduction is impossible for multi-commodity network flow problems among different commodities because different commodities cannot be merged on any edge. While the minimum cost flow problem belongs to class P, the MCNF problem is NP-hard. MCNF models are extensively utilized in the transportation industry and communication networks (Salimifard and Bigharaz, 2022). Specifically, the traffic flow management rerouting problem in air traffic control is formulated as an MCNF problem to distribute flights over a national air traffic network to minimize delays (Bertsimas and Patterson, 2000). MCNF was also applied in driver rostering problems to schedule schemes (Mesquita et al., 2015). Product distribution is an area where MCNF was applied, like vehicle routing problems (Vaziri et al., 2019; Zhang et al., 2019). Network planning and design for multimodal transportation network utilizes given infrastructures and chooses transportation modes to optimally transfer multicommodity (Elbert et al., 2020; Katayama, 2019; Qu et al., 2016; Yaghini et al., 2012). Though multi-path planning was formulated as an MCNF problem in Yu and LaValle (2016), it only considers temporal conflicts and generated routes may still have spatial conflicts. However, MCNF still has the potential to be extended to solve the network planning problem due to the similarity of problem features.

The considered network planning problem has similarities to these minimum-cost network flow problems and minimum-cost multi-commodity network flow problems. In the network planning problem, the planning for a single route is related to single-source single-sink minimum-cost network flow problems. The planning for the whole network is similar to the minimum-cost multi-commodity network flow problems in that route (flow) for each OD (commodity) should be found while satisfying a set of constraints.

2.3. Graph-search-based methods for network planning problems

Existing graph-search-based multi-path planning methods traverse the graph to determine feasible routes. They aim at minimizing the sum of individual route costs and avoiding conflicts among routes (Felner et al., 2017). The graph-search-based methods include heuristic and non-heuristic methods that use a single entity for computation, as well as communication-based approaches where each route is treated as a separate computational entity.

Heuristic methods resolve conflicts using strategies such as priority-based searches, where routes are planned sequentially with existing routes considered as moving obstacles (Bnaya and Felner, 2014; Silver, 2005). An alternative approach involves using rule-based methods to plan routes for each OD pair. These methods follow specific rules to avoid collisions, such as changing velocity profiles or delaying departures (Alotaibi and Al-Rawi, 2018; De Wilde et al., 2014; Luna and Bekris, 2011; Saha and Isto, 2006).

Non-heuristic methods undertake a comprehensive search of the state space for solutions, omitting mechanisms designed to expedite conflict resolution. Such methods employ state space representations, whereby the positions of all paths to plan at any given moment are depicted as state vectors (Wagner and Choset, 2015; Standley, 2010). Conflict-based state spaces utilize a tree structure, with nodes encapsulating a set of conflicts as constraints and a set of potential conflict resolutions and the associated cost. The tree structure evolves from a root node devoid of constraints to leaf nodes that fulfill all constraints (Cohen et al., 2019, 2016; Sharon et al., 2015; Barer et al., 2014).

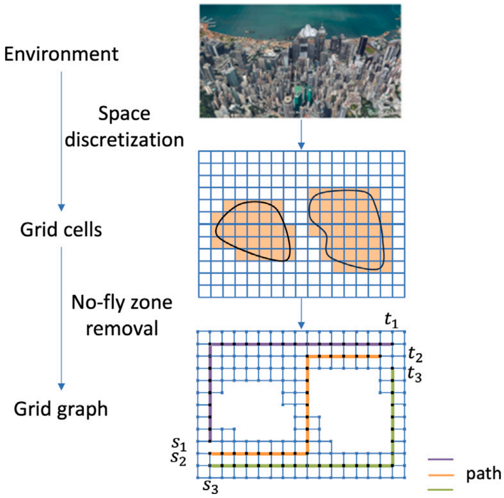


Fig. 5. MCNF by extracting grid graph from environment.

Communication-based methods coordinate routes collaboratively to achieve consensus (Wang and Rubenstein, 2020; Ferrera et al., 2017; Desaraju and How, 2012; Purwin et al., 2008; Olfati-Saber et al., 2007). For example, multiple robots navigate by exchanging messages, deciding on movements, broadcasting intentions, receiving approvals, and executing actions as proposed in Gilboa et al. (2006).

In summary, heuristic methods can quickly generate feasible solutions in scenarios with a large number of state spaces. Non-heuristic methods are capable of finding optimal (or near-optimal) solutions but may require significant computational resources, particularly in complex 3D urban settings. Communication-based methods, although they enable efficient coordination, impose substantial demands on the communication and computation capabilities of each drone. These requirements may not be practical given the regulatory and business complexities of the market.

In previous research, we have developed two graph-search-based methods for network planning. The Sequential Route Planning (SRP) approach leverages a hierarchical prioritization to partition the network planning problem into solvable single-path tasks, executed in sequence (He et al., 2022). To enhance fairness and efficiency among routes, the Distributed Route Planning (DRP) method applies a distributed planning approach, allowing each OD pair to optimize their routes within a system-level coordinated framework (He et al., 2024). This study aims to develop a standardized network flow model for the air corridor planning problem, assess the computational complexity of corridor planning through the complexity of the network flow model, and understand the computation resources required by comparing the performance of finding optimal and sub-optimal solutions in solving this optimization problem.

3. The air corridor planning problem formulation

3.1. The conceptual problem statement

In this section, we introduce the conceptual statement of the route network planning problem to support urban drone delivery service. While various ConOps can address diverse route network planning challenges, our focus here is on the most fundamental problem that necessitates resolution.

Air routes are unidirectional corridors between vertiports, as shown in Fig. 3(a). Different corridors should not intersect with each other. Drones follow these corridors with minimum spacing. Each corridors is 20 m wide and 10 m high with a 10 m buffer zone. No obstacles are allowed in the buffer zone. Buffer zones of different corridors can be either overlapped or not allowed to be overlapped, depending on different ConOps. In this fundamental problem, we do not overlap buffer zones.

In order to formulate and compute the network planning problem, we need to discretize the urban environment into a grid graph, as shown in Fig. 5. An urban environment includes essential elements such as obstacles, geofencing, vertiports, and zoning information. The obstacles and geofencing constitute no-fly zones that drones are prohibited from traversing. The space is first divided into grid cells, and cells are excluded from feasible airspace if they contain no-fly zones. The remaining cells constitute the grid graph $\mathcal{G}$, characterized by nodes $\mathcal{N}$ and arcs $\mathcal{A}$ which represent the vertices and edges, respectively.

Given a set of OD pairs $\{(o_n, d_n), n \in [N]\}$ in an urban environment per the demand of drone delivery service, we aim to look for a set of air routes that connect these OD pairs while spatially separated from each other. A route $\boldsymbol{r}_n$ for an OD pair (o_n, d_n) is specified by a sequence of points along which drones can travel in a sequence from the origin o_n to the destination d_n, namely $\boldsymbol{r}_n = \left(v_n^0, v_n^1, \dots, v_n^{l_n}\right)$

where the route consists of l_n segments $\left\{\left(v_n^{i-1}, v_n^i\right), i \in [l_n]\right\}$ and $o_n = v_n^0, d_n = v_n^{l_n}$. We can derive the objectives and constraints of the route network planning problem as below,

$$\min_{\boldsymbol{r}_{[N]}} \sum_{n\in[N]} C_o(\boldsymbol{r}_n) + \alpha_I \sum_{n\in[N]} C_i(\boldsymbol{r}_n) \tag{3a}$$

$$\mathcal{M}(\boldsymbol{r}_n) \cap \mathcal{B} = \emptyset, \tag{3b}$$

$$\mathcal{M}(\boldsymbol{r}_i) \cap \mathcal{M}(\boldsymbol{r}_j) = \emptyset, \forall i, j \in [N] \ \&\ i \neq j. \tag{3c}$$

where C_o is the operation cost and C_i is the impact cost, and α_I is a weighting factor. In this equation, the function $\mathcal{M}$ maps a route to the set of cells that the route goes through. $\mathcal{B}$ is the set of cells blocked by obstacles. In summary, Eq. (3a) is the objective function, Eq. (3b) is the obstacle avoidance constraint, and Eq. (3c) is the route separation constraint. Next, we will illustrate them in detail.

The aim of the route network planning problem is twofold. Firstly, there is a need to minimize the operation cost of each individual route, which is primarily influenced by energy consumption. Specifically, the operation cost $C_o(\boldsymbol{r}_n)$ of a route $\boldsymbol{r}_n$, is broken down into four distinct parts,

$$C_o(\boldsymbol{r}_n) = \sum_{i\in[l_n]} \alpha_g D_g(v_n^{i-1}, v_n^i) + \alpha_c D_c(v_n^{i-1}, v_n^i) + \alpha_d D_d(v_n^i, v_n^{i+1}) + \alpha_y \left| Y\left(v_n^{i-1}, v_n^i, v_n^{i+1}\right)\right|, \tag{4}$$

where $D_g(v_n^{i-1}, v_n^i)$ represents the distance between consecutive waypoints (v_n^{i-1}, v_n^i), D_c and D_d denote the climb and descent heights, respectively. The last item Y captures the turning angles between route segments. The coefficients $\alpha_g, \alpha_c, \alpha_d, \alpha_y$ are predetermined weighting factors that account for the relative importance of each factor. In the context of a simplified network planning problem, the operational cost formula is reduced by focusing only on the path distance, altitude differences of climbing and descending. The turning angle is not considered here because it needs complicated transformations on the network and requires additional vertices and edges, which can be incorporated in the graph-search algorithms but not the network flow model with commercial solvers. To make the comparison fair between the graph search algorithms and the network flow model with commercial solvers, the operation cost is simplified as

$$C_o(\boldsymbol{r}_n) = \sum_{i\in[l_n]} D_g(v_n^{i-1}, v_n^i) + D_c(v_n^{i-1}, v_n^i) + D_d(v_n^i, v_n^{i+1}). \tag{5}$$

Secondly, there is a requirement to minimize system-level costs associated with drone operations. In the context of this simplified problem, system-level costs include the cumulative effects on safety risks, noises, privacy concerns, and other issues resulting from drone delivery operations, collectively referred to as impact costs. The impact cost, denoted as $C_i(\boldsymbol{r}_n)$, is largely influenced by the geographic extent of air corridors. Each cell a within the grid graph $\mathcal{G}$ is assigned an impact cost $I(a)$, and the total impact cost $C_i(\boldsymbol{r}_n)$ for a given route $\boldsymbol{r}_n$ is computed as follows:

$$C_i(\boldsymbol{r}_n) = \sum_{a\in\mathcal{M}(\boldsymbol{r}_n)} I(a). \tag{6}$$

For the impacts that involve multiple operations at each location, our impact cost function can be easily modified to include different impacts for different operations.

Let $\mathcal{B}$ be the set of cells obstructed by physical buildings and terrains. Clearly, the route of the OD pair (o_n, d_n) must entirely detour these blocked regions. In other words, we require

$$\mathcal{M}(\boldsymbol{r}_n) \cap \mathcal{B} = \emptyset, \tag{7}$$

where the function $\mathcal{M}$ maps a route to the set of cells that the route goes through. This constraint ensures that the path for each OD pair does not intersect with any obstructed cell. However, the network planning problem is intended to design multiple conflict-free routes. To ensure spatial separation among all OD pairs and thereby prevent any shared use of cells by different routes, the following constraint is required:

$$\mathcal{M}(\boldsymbol{r}_i) \cap \mathcal{M}(\boldsymbol{r}_j) = \emptyset, \forall i, j \in [N] \ \&\ i \neq j. \tag{8}$$

It is important to note that Eq. (3) addresses a simplified network problem, focusing on the most fundamental problem that necessitates a resolution. There could be additional constraints in real-world route network planning problems. For example, we may require the planned routes to accommodate the flight dynamics of specific types of UAVs, climbing/descending rate limits, turning angle limits, speed limits, etc. or we may allow buffer zones to overlap. These additional constraints significantly increase the complexity of the network problem compared to the simplified version.

3.2. The Multi-Commodity Network Flow (MCNF) formulation

This section introduces how to transform the conceptual problem statement described into an MCNF model. Given a discretized grid graph $\mathcal{G}$, N OD pairs $(o_n, d_n), n \in [N]$ are represented as N commodities with source nodes $\mathcal{S} = \{s_n\}_{n=1}^N$ and sink nodes $\mathcal{T} = \{t_n\}_{n=1}^N$, where $s_n = o_n$ and $t_n = d_n$. Let x_{ij}^n denote the existence of flow for commodity n on arc (i, j). The cost of flow for commodity n on arc (i, j) is represented by c_{ij}^n.

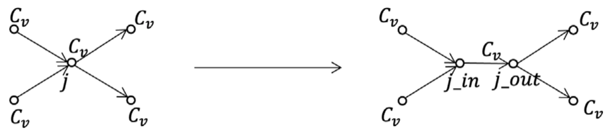


Fig. 6. Illustration for node splitting.

Eq. (3a) is the objective function that minimizes operation and impact costs for all routes. To transform the objective function to the MCNF format, we first define a route in an MCNF model. A route for OD pair n can be represented as a set of arcs where x_{ij}^n is true, i.e., $\boldsymbol{r}_n = \{(i,j)|x_{ij}^n = 1\}$. The calculation of operation and impact cost for each route can be decomposed to the sum of operation and impact cost for the arcs that the route traversed. Thus we can define the cost of flow for commodity n on arc (i,j) as $c_{ij}^n = 1 + \alpha_I I(a)$, where arc (i,j) is inside the cell a, and $I(a)$ represents the impact of cell a. Consequently, the objective function can be transformed as $\sum_{n=1}^N \sum_{(i,j)\in\mathcal{A}} c_{ij}^n x_{ij}^n$.

Eq. (3b) is the obstacle avoidance constraint. The transformation of this constraint is conducted before conceptual problem statement. That is, nodes and arcs inside obstacles are excluded from the graph when constructing graph $(\mathcal{N},\mathcal{A})$ from grid cells $\mathcal{G}$. Therefore, in constructing a graph, we first transform the discretized grid cells of the environment, which is an undirected network, into a directed network, and then we delete the nodes inside the obstacles and all edges that connect these nodes.

Eq. (3c) is the route separation constraint. This constraint requires that tubes cannot intersect with each other. Correspondingly, in the MCNF model, this constraint indicates that no node or arc of the graph can be occupied by two or more commodities. It can be transformed into capacity constraints on arcs and nodes such that each arc or node allows only one unit of flow. The arc capacity constraint is $0 \le \sum_{n=1}^N x_{ij}^n \le 1$, and the node capacity constraint is $0 \le \sum_{n=1}^N \sum_{j:(i,j)\in\mathcal{A}} x_{ij}^n \le 1$.

Therefore, we can derive an MCNF model based on these objectives and constraints together with flow conservation constraints. The formulation is as follows:

$$\min \sum_{n=1}^N \sum_{(i,j)\in\mathcal{A}} c_{ij}^n x_{ij}^n \tag{9a}$$

$$\text{s.t.} \quad \sum_{\{j:(i,j)\in\mathcal{A}\}} x_{ij}^n - \sum_{\{j:(j,i)\in\mathcal{A}\}} x_{ji}^n = \begin{cases} -1 & j = s_n \\ 0 & \forall j \neq s_n, t_n \\ 1 & j = t_n \end{cases} \quad \forall n \in [N] \tag{9b}$$

$$0 \le \sum_{n=1}^N x_{ij}^n \le 1 \quad \forall (i,j) \in \mathcal{A} \tag{9c}$$

$$0 \le \sum_{n=1}^N \sum_{j:(i,j)\in\mathcal{A}} x_{ij}^n \le 1 \quad \forall j \in \mathcal{N}/\{\mathcal{S}\cup\mathcal{T}\} \tag{9d}$$

$$x_{ij}^n \in \{0,1\} \quad \forall (i,j)\in\mathcal{A}, \forall n \in [N] \tag{9e}$$

In Eq. (9), Eq. (9a) minimizes the operation and impact cost, Eq. (9b) represents flow conservation constraints, Eq. (9c) and (9d) represent arc and node capacity constraints arising from route separation constraints.

3.3. MCNF formulation standardization

We then transform the MCNF model of Eq. (9) into a standardized minimum cost MCNF problem, which can then be solved by IP solvers. The primary challenge in transformation lies in handling node capacity constraints, which are not considered in standard minimum-cost MCNF problems. To address this challenge, we apply node splitting to transform node capacity constraints into arc capacity constraints, as illustrated in Fig. 6. The details are described below.

Let the node capacity threshold be 1. By splitting node j into nodes j_in and j_out, the capacity constraint on node j can be transferred to a capacity constraint on arc (j_in, j_out). Mathematically, this means that the inflow to node j must equal the inflow to node j_in, i.e.,

$$\sum_{\{j:(i,j)\in\mathcal{A}\}} x_{ij}^n = \sum_{\{j\_in:(i,j\_in)\in\mathcal{A}\}} x_{ij\_in}^n.$$

Since flow conservation constraints are still satisfied after transformation (i.e., $\sum_{\{j\_in:(i,j\_in)\in\mathcal{A}\}} x_{ij\_in}^n - x_{j\_in\ j\_out}^n = 0$), we have $\sum_{\{j:(i,j)\in\mathcal{A}\}} x_{ij}^n = x_{j\_in\ j\_out}^n$. Thus, the node capacity constraint is transformed from

$$0 \le \sum_{n=1}^N \sum_{j:(i,j)\in\mathcal{A}} x_{ij}^n \le 1$$

to

$$0 \le \sum_{n=1}^N x_{j\_in\ j\_out}^n \le 1.$$

Table 1
Summary of the size of graph and MCNF formulation.

Scenario size	$m \times n \times l$ graph, N commodities	
Graph after transformation	Number of nodes $\approx 6(m \cdot n \cdot l)$	Number of arcs $\approx 7(m \cdot n \cdot l)$
MCNF formulation	Number of variables $\approx 7(m \cdot n \cdot l \cdot N)$	Number of constraints $\approx 7(m \cdot n \cdot l) + (m \cdot n \cdot l \cdot N)$

Therefore, we can derive a standard minimum cost MCNF problem from Eq. (9) as below:

$$\min \sum_{n=1}^{N} \sum_{(i,j)\in\mathcal{A}} c_{ij}^n x_{ij}^n \tag{10a}$$

$$\text{s.t.} \quad \sum_{\{j:(i,j)\in\mathcal{A}\}} x_{ij}^n - \sum_{\{j:(j,i)\in\mathcal{A}\}} x_{ji}^n = \begin{cases} -1 & j = s_n \\ 0 & \forall j \neq s_n, t_n \\ 1 & j = t_n \end{cases} \quad \forall n \in N \tag{10b}$$

$$0 \leq \sum_{n=1}^{N} x_{ij}^n \leq 1 \quad \forall (i,j) \in \mathcal{A} \tag{10c}$$

$$0 \leq \sum_{n=1}^{N} x_{j\_in\ j\_out}^n \leq 1 \quad \forall j \in \mathcal{N}/\{\mathcal{S} \cup \mathcal{T}\} \tag{10d}$$

$$x_{ij}^n \in \mathbb{N} \quad \forall (i,j) \in \mathcal{A}, \forall n \in [N] \tag{10e}$$

The standard minimum cost MCNF problem in Eq. (10) has the key property of being integer flow and a unit capacity network, meaning that each arc has a capacity of one flow unit. However, even with only two commodities and unit capacities, this problem remains NP-hard (Even et al., 1975).

4. Complexity of the air corridor planning problem

The last section has formulated the air corridor network planning problem into a standard MCNF model. This section analyzes the complexity of corridor network planning by exploring the MCNF model's complexity. We consider both the problem size and its NP-hardness in our analysis.

4.1. Problem size of MCNF formulation

The number of nodes and arcs in the network directly impacts the computational resources required; more nodes and arcs typically mean a larger problem space to explore, and more commodities and capacity constraints can also complicate the solution in terms of scalability, solution feasibility, and optimization difficulty.

The process of obtaining a minimum-cost MCNF model and the associated changes in network size can be summarized as follows. First, the urban environment under network planning is first discretized into grid cells, from which a directed network is generated. Nodes and edges within obstacles are subsequently removed, followed by node splitting across the entire graph. For a grid graph of size $m \times n \times l$ in the absence of obstacles, it contains approximately $(m+1)(n+1)(l+1) \approx m \cdot n \cdot l$ nodes and $m(n+1)(l+1) + (m+1)n(l+1) + (m+1)(n+1)l \approx 3(m \cdot n \cdot l)$ arcs. Applying node splitting to transform node capacity constraints into arc capacities doubles the number of nodes from $3(m \cdot n \cdot l)$ to $6(m \cdot n \cdot l)$, while the arcs of split nodes increase the number of arcs from $6(m \cdot n \cdot l)$ to $7(m \cdot n \cdot l)$. Thus for a grid graph size of $m \times n \times l$ in the absence of obstacles, the network after the transformation has $6(m \cdot n \cdot l)$ nodes and $7(m \cdot n \cdot l)$ arcs.

If there are N commodities in such a network, the number of 0–1 decision variables is the product of the number of arcs and commodities: $7(m \cdot n \cdot l \cdot N)$. The number of capacity constraints equals the number of arcs: $7(m \cdot n \cdot l)$, while the flow balance constraints equal the product of the number of nodes and commodities: $m \cdot n \cdot l \cdot N$. Thus, in a route network planning problem with a grid graph size of $m \times n \times l$ in the absence of obstacles and N commodities, the multi-commodity network flow formulation has $7(m \cdot n \cdot l \cdot N)$ variables and $7(m \cdot n \cdot l) + (m \cdot n \cdot l \cdot N)$ constraints.

It should be noted that the current MCNF model extracts networks from grid cells; however, diagonal nodes and others that can "see" each other are not connected in the current networks. To make generated routes more realistic, such nodes should be connected, which will significantly increase the arcs in the network. Additionally, other side constraints that are omitted in the current MCNF model can also increase the variables and constraints in the model for real applications. Therefore, the problem size for the real-world air corridor planning problem can be much larger than that of the current MCNF model, although the problem size for the current MCNF model is already very large (see Table 1).

4.2. NP-hardness of MCNF formulation

Proposition 1. *A flow problem is simple if the capacities of all edges are equal to one. Simple two-commodity integral flow in directed graphs(D2CIF) is NP-hard (Even et al., 1975).*

Theorem 1. *The MCNF model for route network planning is NP-hard.*

Proof. For simple D2CIF, besides the flow conservation constraints, the constraints can be represented as

$$0 \le x_{ij}^1 + x_{ij}^2 \le 1,$$

$$x_{ij}^1, x_{ij}^2 \in \mathbb{N}.$$

For MCNF model with N commodities, it has constraints

$$0 \le \sum_{n=1}^{N} x_{ij}^n \le 1,$$

$$x_{ij}^n, \in \mathbb{N}, \quad \forall n \in [N],$$

let $y_{ij} = \sum_{n=1}^{N-1} x_{ij}^n$, then the constraints will be transformed to

$$0 \le y_{ij} + x_{ij}^N \le 1,$$

$$y_{ij}, x_{ij}^N \in \mathbb{N}.$$

This is identical to the constraints for the simple D2CIF, thus allowing the simple D2CIF to be considered a special case of the MCNF model for route network planning. Given that the simple D2CIF is NP-hard, the MCNF model for route network planning is also NP-hard.

It should be noted that while the MCNF model addresses the simplified network planning problem, for real-world network planning problems, the MCNF model can become more complicated due to additional constraints. Since the MCNF model for the simplified problem is NP-hard, real-world network planning will be at least as challenging, being NP-hard or more so.

5. Solutions for the air corridor planning problem

5.1. Integer programming (IP) solvers for MCNF model

This section solves the MCNF model to get an optimal solution. We use IP solvers to obtain such solutions for benchmarking. Different optimization software uses different algorithms and different programming in their IP solvers. Here we consider two software, one is commercial software – Gurobi (Gurobi, 2021), and one is open-source package – GNU Linear Programming Kit (GLPK) (GLPK, 2021).

The Gurobi solver leverages an advanced branch-and-cut (B&C) algorithm, which enhances the traditional branch and bound (B&B) methodology by incorporating cutting planes and heuristic strategies to efficiently resolve IP problems. The core principle of B&B is to systematically partition the search space of the problem into smaller, manageable subproblems using a tree structure, and to solve each subproblem individually to find an optimal solution (Clausen, 1999; Balas and Toth, 1983). At the base of the tree is the IP problem that awaits resolution. Each node within the tree processes the problem's linear programming (LP) relaxation, which temporarily relaxes integer constraints to allow non-integer solutions. If the LP relaxation is infeasible, the node and its descendants are pruned. If the LP relaxation provides a non-integer solution, the solver branches on a non-integer variable and creates two subproblems corresponding to the variable being fixed to either its floor value $\lfloor x \rfloor$ or its ceiling value $\lceil x \rceil$. If the LP relaxation provides an integer solution, it is compared to the current best integer solution. If it is better, the current best solution is updated. If the root node's LP relaxation results in an integer solution, this solution is established as optimal. Techniques such as the primal simplex and primal–dual interior-point methods are employed to solve LP relaxations and find a warm start for the B&B algorithm (Dantzig et al., 1955; Potra and Wright, 2000).

The B&B algorithm is a powerful technique for solving IP problems, but it can be computationally expensive for large problems. Cutting planes is a technique used by Gurobi to improve the efficiency of the IP solver. Cutting planes are linear inequalities that are added to the LP relaxation of the problem to tighten the LP relaxation and improve the lower bound (Padberg and Rinaldi, 1991; Mitchell, 2002). The solver generates cutting planes by analyzing the structure of the problem and identifying valid inequalities that are satisfied by all integer solutions. Heuristics are also used by Gurobi to quickly find good integer solutions for the problem. These heuristics are based on problem-specific knowledge and customized by the user. These techniques make B&C more effective than B&B for some types of IP problems, especially when the LP relaxation is loose and cutting planes can significantly tighten it.

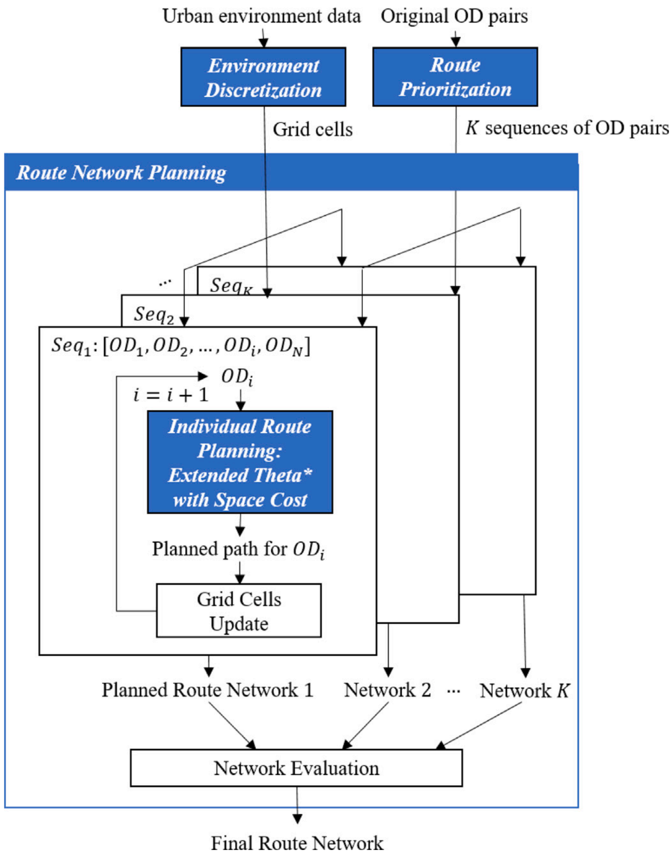


Fig. 7. The Sequential Route Network Planning (SRP) method (He et al., 2022).

GLPK is an open-source software package for solving IP problems. It relies primarily on a B&B algorithm to solve IP problems, but it still integrates the cutting plane function. Both Gurobi and GLPK are powerful tools for solving integer programming (IP) problems. However, differences in algorithms, implementation in terms of performance optimization and parallelization, solver parameters, and problem characteristics can affect their efficiency for different problems. Therefore, it is appropriate to apply both solvers to the network flow problem to see which one performs better.

5.2. A sequential graph-search-based method: Sequential Route Network Planning Method (SRP)

This section introduces a graph-search-based approach for UTM planning, known as the Sequential Route Network Planning method (SRP), developed in our prior work (He et al., 2022). This approach employs a hierarchical prioritization scheme that separates the planning of a route network into a sequence of single-path planning tasks executed in sequence. Prioritization rules are applied to order OD pairs, ensuring that important routes have higher priority for airspace usage to enhance overall system performance. In the SRP framework (as shown in Fig. 7), the routes to be planned are ordered into sequences according to predetermined criteria, which guide the subsequent *Network Planning* module in designing a feasible route network. Each route sequence runs the *Individual Route Planning* repeatedly to form a route network. After generating all potential networks, the Network Evaluation module assesses each network, selecting the one with the minimal costs. This selected network is then evaluated for risk across all air corridors; only networks meeting established safety thresholds are feasible. If no network meets the criteria, the planning fails.

5.3. A distributed graph-search-based method: Distributed Route Planning with congestion pricing (DRP)

This section introduces a graph-search-based planning method — the distributed route network planning method (DRP) developed in our prior work (He et al., 2024) for UTM planning. The DRP method uses a distributed planning framework where each origin–destination (OD) pair optimizes its route independently to minimize travel distance, following overall coordination strategies. A distinctive aspect of the DRP approach is the incorporation of congestion costing, which functions as a flexible constraint to

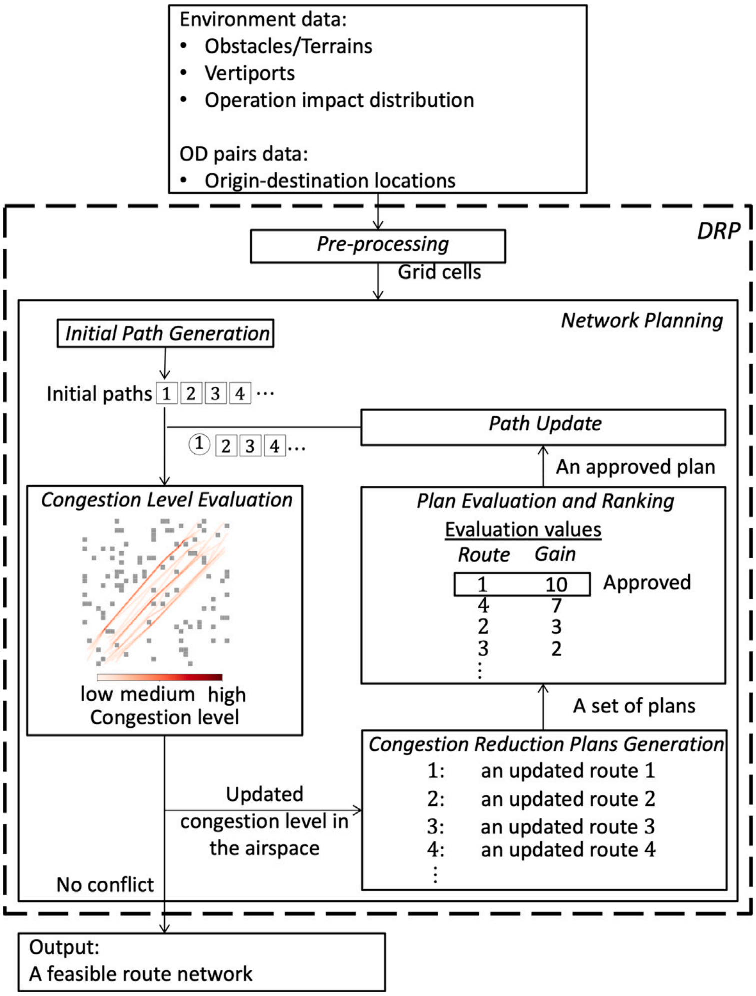


Fig. 8. The distributed route network planning (DRP) method (He et al., 2024).

facilitate airspace allocation. Instead of relying on rigid constraints to manage conflicts, as traditional methods do, this approach uses adjustable congestion costs. These costs are dynamically modified to reroute traffic from busy airspaces to less crowded ones, improving overall system efficiency and minimizing potential conflicts.

The DRP method is illustrated in Fig. 8 and further detailed in He et al. (2024). In the DRP method, initially, the Extended Theta* algorithm in the Initial Path Generation module calculates optimal routes for each OD pair without accounting for congestion. Subsequently, congestion levels are analyzed by calculating the frequency of grid cell usage across routes. In the Congestion Reduction Plans Generation module, alternate paths are proposed independently for each route to mitigate congestion, facilitated by dynamic congestion costs. These proposals are then evaluated using a function that balances congestion reduction and minimal travel distance increases. The process concludes with the Path Update module, which implements the optimized routes based on these evaluations.

The SRP and DRP methods applied in this work are simplified versions of the methods developed separately in He et al. (2022) and He et al. (2024). The baseline algorithm for searching a single route is a degradation from Extended Theta*, where routes are not restricted to the edges of the grids, to A*, which generates routes that strictly follow the edges of the grid network, akin to the network flow method. Additionally, the cost objective function and constraints in SRP and DRP are aligned with Eq. (3) in this work.

6. Computational experiments

6.1. Testing on a standard 2D scenario

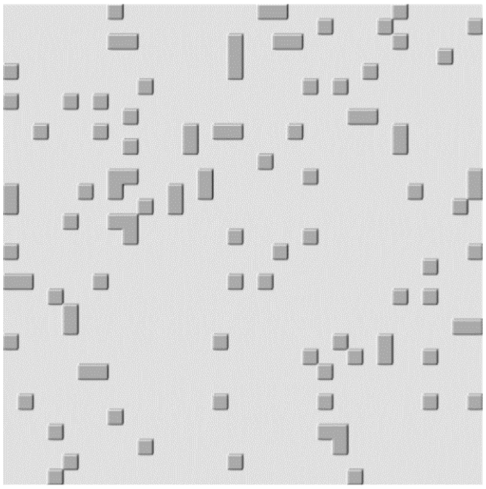


Fig. 9. 2D scenario for testing (Sturtevant, 2012).

Table 2
Summary for network sizes for testing.

Numbers of grid cells	256 × 256	128 × 128	64 × 64
Number of nodes	112 590	31 942	7366
Number of arcs	277 391	78 636	17 492

6.1.1. Testing scenario

We conduct a set of tests to compare the graph-search-based methods, namely SRP and DRP methods, against applying commercial solvers (Gurobi and GLPK) to the MCNF formulation, for solving route network planning problems in a standard 2D scenario (Sturtevant, 2012) with a coverage of 2.5 km by 2.5 km, as depicted in Fig. 9. In this scenario, the impacts of flight operations are uniformly distributed, thus, the impact is directly proportional to the distance of flight operations.

In the baseline experiments, we discretize the map into a grid graph with a resolution of 10 m (256 × 256). It includes 256 by 256 grid cells. A total of 10 routes need to be generated in the baseline experiments. Then sensitivity analysis were performed to understand the effect of the numbers of grid cells and the numbers of routes on computational resources and optimality. Specifically, we use resolutions of 10 m (256 × 256), 20 m (128 × 128), and 40 m (64 × 64), and a number of routes ranging from 1 to 20. Table 2 shows the number of nodes and arcs for different grid sizes in this 2D scenario.

6.1.2. Computational implementation

The MCNF model with Gurobi/GLPK solver is implemented following the steps as illustrated in Fig. 10. The first step, *environment formulation*, generates nodes, arcs, and commodities from the environment data. The process begins with environment discretization to generate grid cells. Each grid cell is then assigned a unique node ID. OD pairs are recorded as sets of {In_ID} and {Out_ID}. Arcs are defined based on node IDs, with an in-flow node From_ID and an out-flow node To_ID. Each arc has a cost for traversing and a maximum capacity of 1 for the network planning problem. Commodities are determined by their origin nodes (Source_ID) and destination nodes (Sink_ID), with a quantity of 1 for the network planning problem. The second step, *modeling*, uses the Pyomo optimization interface (Bynum et al., 2021) to formulate objective functions and constraints and imports them into optimization software such as Gurobi. The third step, *optimize()*, employs optimization software to derive optimal objective and decision variable values, yielding flows for each commodity or routes for each OD pair. The optimization software utilizes Integer Programming (IP) solvers for optimization.

When implementing the SRP and the DRP algorithm, the input environment data and the OD pairs data are consistent with the ones used in the MCNF model. Only the network planing modules used different methods as illustrated in Section 5.2 for SRP and Section 5.3 for DRP.

All computations were executed on an Ubuntu server equipped with 32 Intel Xeon Gold 5218 processors and 13 GB of RAM.

6.1.3. Evaluation metrics

We use the *total distance of routes [km]* as a proxy of the total cost, measuring optimality, and the *computational time [s]* needed to generate a feasible route network, measuring the computational resources needed. The total distance is a suitable proxy for solution optimality since the impact cost is directly proportional to the total distance in this scenario. The primary objective of the following tests is to understand the differences in optimality and computational time of the MCNF model and the graph-search-based methods, especially for large maps with many OD pairs.

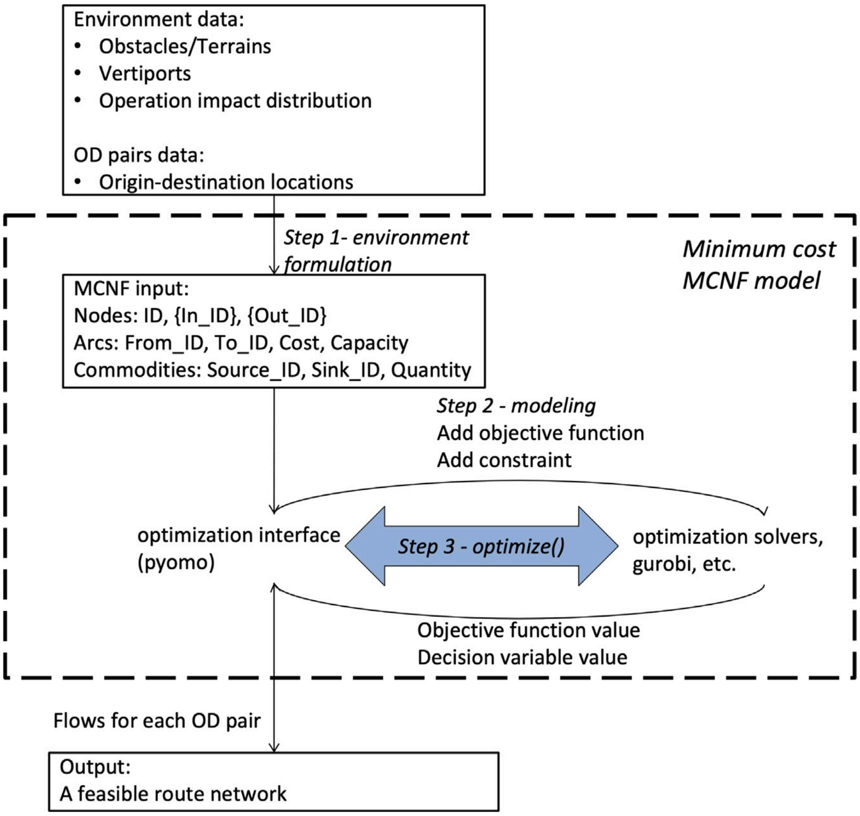


Fig. 10. Implementation of the MCNF model.

Table 3
Baseline results of 10 routes for the methods on 256 × 256 grid cells.

MCNF model: GLPK		MCNF model: Gurobi		Graph-search: SRP		Graph-search: DRP	
Computation time [s]	Total route distance [km]	Computation time [s]	Total route distance [km]	Computation time [s]	Total route distance [km]	Computation time [s]	Total route distance [km]
–	–	14 534	3.42	#	#	3.1	3.57

–: A feasible solution may exist, but the algorithm failed to converge in 10 h
#: A feasible solution may exist, but the algorithm returned no feasible solution.

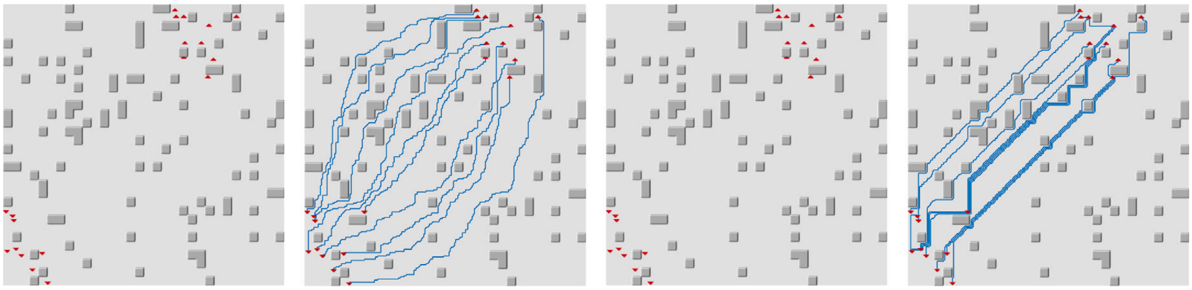
6.1.4. Baseline results

In the baseline experiments, all methods were used to plan a network with 10 routes connecting the pre-specified origins and destinations on the 256 × 256 grid cells, as shown in Fig. 11 and Table 3. For the MCNF model, Gurobi found a feasible solution, but GLPK failed to find a feasible solution within 10 h. For the graph-search-based methods, SRP failed to find a feasible solution, while DRP found a feasible solution. Comparing DRP and Gurobi, DRP took much less computational time with a small gap in total distance. The results indicates that DRP can find near-optimal solutions within a short computational time for the baseline experiments.

6.1.5. Sensitivity analysis on number of routes

We conducted a sensitivity analysis on the number of routes, ranging from 1 to 20. The results are presented in Table 4 and Fig. 12. With the increase of the number of routes, the computation times for the MCNF model using either Gurobi or GLPK grow rapidly, exceeding 10 h for just 12 routes. In contrast, the graph-search-based methods, SRP and DRP, exhibit significantly shorter computation times, and their rate of increase is much slower than that of the MCNF models. However, these methods yield slightly longer route distances compared to the MCNF models.

Specifically, within the MCNF models, Gurobi's IP solver demonstrates shorter computation times than GLPK's as the number of routes increases. This can be attributed to Gurobi's superior performance optimization and parallelization capabilities as commercial software. In this experiment, GLPK is limited to using a single CPU, whereas Gurobi efficiently utilizes multiple CPUs in parallel at its peak. This experiment was conducted using default solver parameters without any optimization. Among the graph-search-based methods, DRP not only provides better route distances but also identifies more feasible routes.



(a) GLPK (failed)   (b) Gurobi   (c) SRP (failed)   (d) DRP

Fig. 11. 10 routes planned by MCNF model using Gurobi and GLPK solvers, and graph-search methods SRP and DRP on 256 × 256 grid cells.

Table 4
Results on 256 × 256 grid cells for different number of routes.

Number of routes	MCNF: GLPK		MCNF: Gurobi		Graph-search: SRP		Graph-search: DRP	
	Computation time [s]	Route distance [km]	Computation time [s]	Route distance [km]	Computation time [s]	Route distance [km]	Computation time [s]	Route distance [km]
1	1224	0.39	17	0.39	0.01	0.39	0.03	0.39
2	12 241	0.73	99	0.73	0.02	0.73	0.08	0.73
3	–	–	196	1.08	0.12	1.08	0.1	1.08
4	–	–	1314	1.42	0.73	1.52	0.6	1.43
5	–	–	3015	1.67	1.28	1.68	0.7	1.68
6	–	–	4517	2.00	3.22	2.05	1.3	2.00
7	–	–	4884	2.33	4.35	2.36	1.6	2.34
8	–	–	7464	2.66	7.43	2.85	2.3	2.70
9	–	–	10 145	2.99	9.43	3.14	2.6	3.04
10	–	–	14 534	3.42	#	#	3.1	3.57
11	–	–	16 438	3.78	#	#	6.9	3.84
12	–	–	31 306	4.14	#	#	7.6	4.19
13	–	–	–	–	#	#	8.0	4.61
14	–	–	–	–	#	#	8.9	4.81
15	–	–	–	–	#	#	11.5	5.17
16	–	–	–	–	#	#	21.3	5.64
17	–	–	–	–	#	#	28.8	6.32
18	–	–	–	–	#	#	30.7	6.57
19	–	–	–	–	#	#	31.5	6.97
20	–	–	–	–	#	#	39.2	7.18

–: A feasible solution may exist, but the algorithm failed to converge in 10 h
#: A feasible solution may exist, but the algorithm returned no feasible solution.

6.1.6. Sensitivity analysis on number of grid cells

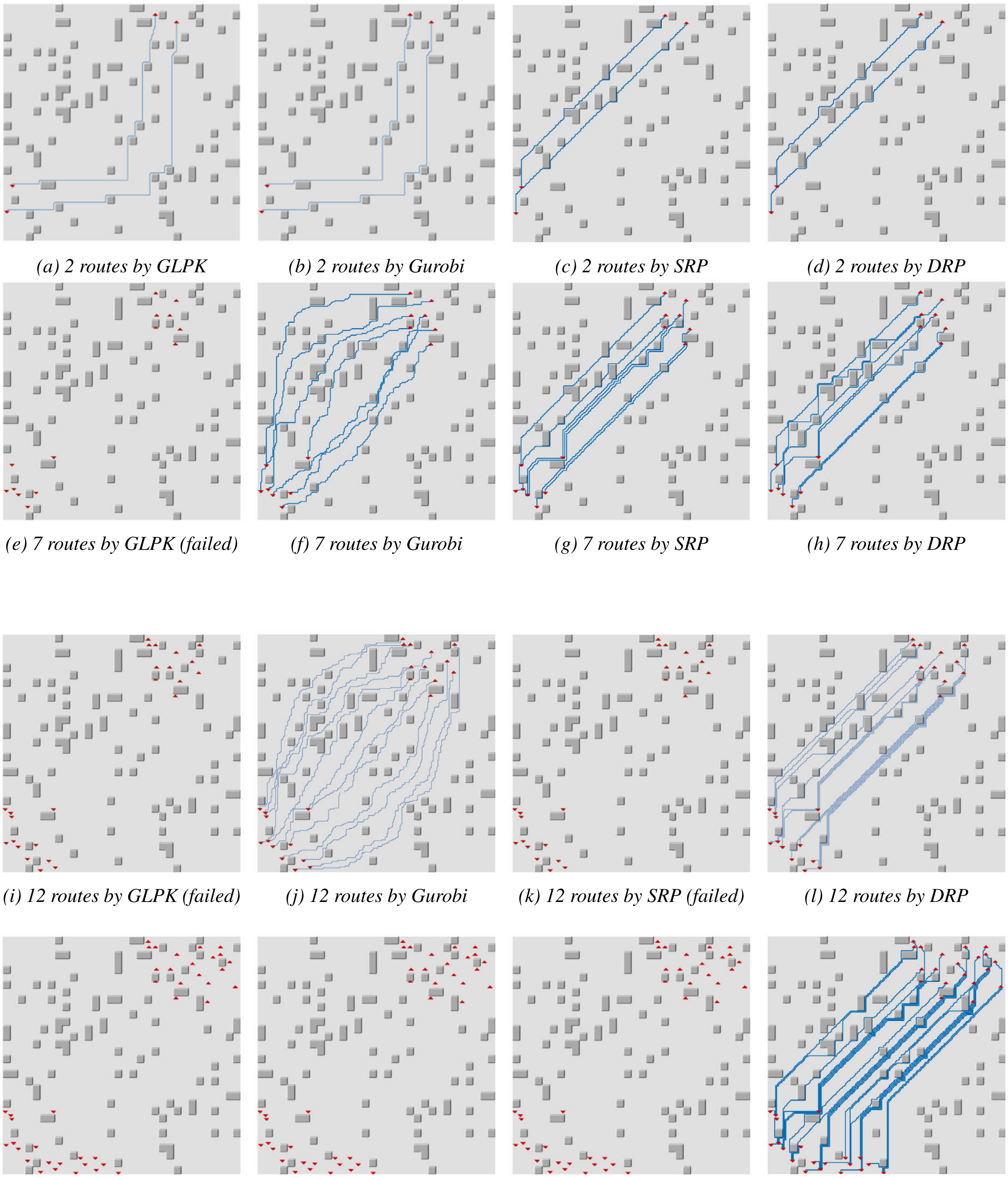
We also conducted a sensitivity analysis on the number of grid cells, which ranged from 64 × 64 to 256 × 256. The results are summarized in Tables 5 through 8.

For a small region with a large number of routes to plan, it is possible that no feasible solution exists. Therefore, graph-search-based methods, such as SRP and DRP, may fail to identify a conflict-free route network even for a small number of routes if the region is over-crowded.

As the number of grid cells increases, the MCNF model demands significantly more computational time and easily to exceed 10 h, especially when using the GLPK solver. In contrast, the graph-search-based methods are capable of finding a large number of routes while still maintaining relatively small computational times.

6.1.7. Test on relaxed MCNF problem with fractional flows

The MCNF model is NP-hard even for only two commodities and unit capacities if flows are integer. If fractional flows are allowed, the problem is only a P problem and can be solved by linear programming (LP) in polynomial time. Therefore, we also show the computational time for solving the relaxed problems as an informative benchmark. As shown in Table 9, MCNF with fractional flows can be solved much faster than integer MCNF. However, the relaxed problems permit fractional flows, which makes the resulting solution not immediately applicable for tackling the path planning problem in UTM. Further investigation is warranted in this area.



(a) 2 routes by GLPK (b) 2 routes by Gurobi (c) 2 routes by SRP (d) 2 routes by DRP

(e) 7 routes by GLPK (failed) (f) 7 routes by Gurobi (g) 7 routes by SRP (h) 7 routes by DRP

(i) 12 routes by GLPK (failed) (j) 12 routes by Gurobi (k) 12 routes by SRP (failed) (l) 12 routes by DRP

(m) 20 routes by GLPK (failed) (n) 20 routes by Gurobi (failed) (o) 20 routes by SRP (failed) (p) 20 routes by DRP

Fig. 12. Route networks planned by MCNF model using Gurobi solver and GLPK solver, and graph-search methods SRP and DRP on 256 × 256 grid cells.

6.2. Demonstration in real-world scenarios

This section shows the applications of graph-search-based methods in real-world scenarios. Real-world constraints are considered in these scenarios, e.g. 3D space and altitude constraints, aerial vehicle dynamics, different risk levels of urban zones, etc. The two

Table 5
Results of MCNF model using GLPK solver on different size of grid cells.

Number of routes	Numbers of grid cells: 64 × 64		Numbers of grid cells: 128 × 128		Numbers of grid cells: 256 × 256	
	Computation time [s]	Route distance [km]	Computation time [s]	Route distance [km]	Computation time [s]	Route distance [km]
1	2.6	0.38	104.1	0.38	1224	0.39
2	13.4	0.71	525.5	0.71	12 241	0.73
3	101.4	1.06	2735.0	1.06	–	–
4	3349.6	1.412	–	–	–	–
5	–	–	–	–	–	–
6	–	–	–	–	–	–
7	–	–	–	–	–	–
8	–	–	–	–	–	–
9	–	–	–	–	–	–
10	–	–	–	–	–	–
11	–	–	–	–	–	–
12	–	–	–	–	–	–
13	–	–	–	–	–	–
14	–	–	–	–	–	–
15	–	–	–	–	–	–
16	–	–	–	–	–	–
17	–	–	–	–	–	–
18	–	–	–	–	–	–
19	–	–	–	–	–	–
20	–	–	–	–	–	–

–: A feasible solution may exist, but the algorithm failed to converge in 10 h.

Table 6
Results of MCNF model using Gurobi solver on different size of grid cells.

Number of routes	Numbers of grid cells: 64 × 64		Numbers of grid cells: 128 × 128		Numbers of grid cells: 256 × 256	
	Computation time [s]	Route distance [km]	Computation time [s]	Route distance [km]	Computation time [s]	Route distance [km]
1	1.9	0.38	3.7	0.38	17	0.39
2	4.9	0.71	16.8	0.71	99	0.73
3	6.4	1.06	56.1	1.06	196	1.08
4	8.5	1.41	74.9	1.40	1314	1.42
5	11.6	1.66	494.7	1.66	3015	1.67
6	14.8	1.98	807.5	1.98	4517	2.00
7	68.9	2.31	1572.6	2.31	4884	2.33
8	92.5	2.64	2877.9	2.64	7464	2.66
9	94.2	2.98	3191.4	2.97	10 145	2.99
10	177.5	3.40	5100.2	3.39	14 534	3.42
11	383.0	3.76	11 887.3	3.77	16 438	3.78
12	403.5	4.12	17 391.2	4.13	31 306	4.14
13	[a]	[a]	–	–	–	–
14	[a]	[a]	–	–	–	–
15	[a]	[a]	–	–	–	–
16	[a]	[a]	–	–	–	–
17	[a]	[a]	–	–	–	–
18	[a]	[a]	–	–	–	–
19	[a]	[a]	–	–	–	–
20	[a]	[a]	–	–	–	–

–: A feasible solution may exist, but the algorithm failed to converge in 10 h.
[a] The model was proven to be infeasible, and no feasible solution exists.

areas shown below came from real-world applications of the SRP algorithm and the DRP algorithm, where our company collaborators have operations or plan to have operations. The scale and specification of these two areas are different, therefore direct numerical comparison is not provided in this section.

Fig. 13 presents a network of 12 air corridors developed using the SRP algorithm in a district in Hangzhou, China, where drone delivery service has been deployed by Antwork Technology since 2020. The specific area is an urban zone of approximately 15.75 km^2 (5.35 km by 2.95 km).

Fig. 14 shows a network of 16 air corridors using the DRP algorithm in Mong Kok, a district within Hong Kong SAR. Drone delivery services in Hong Kong are still in the developmental and testing stages, primarily due to the city's complex airspace, dense population, and strict regulatory framework. Hong Kong high-rise urban environment presents unique challenges for drone

Table 7
Results of graph-search-based method SRP on different size of grid cells.

Number of routes	Numbers of grid cells: 64 × 64		Numbers of grid cells: 128 × 128		Numbers of grid cells: 256 × 256	
	Computation time [s]	Route distance [km]	Computation time [s]	Route distance [km]	Computation time [s]	Route distance [km]
1	0.01	0.38	0.01	0.38	0.01	0.39
2	0.01	0.71	0.01	0.71	0.02	0.73
3	0.02	1.09	0.05	1.07	0.12	1.08
4	#	#	0.20	1.42	0.73	1.52
5	#	#	0.43	1.74	1.28	1.68
6	#	#	0.80	2.10	3.22	2.05
7	#	#	1.45	2.32	4.35	2.36
8	#	#	3.92	2.67	7.43	2.85
9	#	#	#	#	9.43	3.14
10	#	#	#	#	#	#
11	#	#	#	#	#	#
12	#	#	#	#	#	#
13	#	#	#	#	#	#
14	#	#	#	#	#	#
15	#	#	#	#	#	#
16	#	#	#	#	#	#
17	#	#	#	#	#	#
18	#	#	#	#	#	#
19	#	#	#	#	#	#
20	#	#	#	#	#	#

#: A feasible solution may exist, but the algorithm returned no feasible solution.

Table 8
Results of graph-search-based method DRP on different size of grid cells.

Number of routes	Numbers of grid cells: 64 × 64		Numbers of grid cells: 128 × 128		Numbers of grid cells: 256 × 256	
	Computation time [s]	Route distance [km]	Computation time [s]	Route distance [km]	Computation time [s]	Route distance [km]
1	0.004	0.38	0.01	0.38	0.03	0.39
2	0.01	0.71	0.03	0.71	0.08	0.73
3	0.02	1.06	0.04	1.06	0.1	1.08
4	0.07	1.41	0.4	1.41	0.6	1.43
5	0.1	1.66	0.4	1.66	0.7	1.68
6	0.1	1.99	0.8	2.02	1.3	2.00
7	0.4	2.33	1.2	2.32	1.6	2.34
8	0.6	2.82	2.3	2.67	2.3	2.70
9	#	#	2.6	3.01	2.6	3.04
10	#	#	2.93	3.43	3.1	3.57
11	#	#	3.23	4.03	6.9	3.84
12	#	#	3.85	4.39	7.6	4.19
13	#	#	4.69	4.50	8.0	4.61
14	#	#	4.92	4.94	8.9	4.81
15	#	#	5.87	5.53	11.5	5.17
16	#	#	#	#	21.3	5.64
17	#	#	#	#	28.8	6.32
18	#	#	#	#	30.7	6.57
19	#	#	#	#	31.5	6.97
20	#	#	#	#	39.2	7.18

#: A feasible solution may exist, but the algorithm returned no feasible solution.

navigation and safety. This route network planning demonstration is conducted as part of an exploration and evaluation effort, to navigate these challenges and prove the viability of drone delivery technologies in Hong Kong's unique environment.

6.3. Managerial insights

Solving the air corridor planning problem is a complex task if we want to get an optimal solution. While human operators might design similar individual air corridors, the complexity of managing dozens or hundreds of air routes simultaneously in urban environments exceeds human capability. An automated design process is essential for handling this complexity at scale. Our testing results show that the network-flow-based method with generic IP solvers takes a long time to solve the multi-path planning problem, making it less preferable in practice, but it can provide an optimality reference to evaluate other methods. Additionally,

Table 9
Computational time [s] for solving integral and fractional MCNF using Gurobi IP and LP solver.

Number of ODs	Numbers of grid cells: 64 × 64		Numbers of grid cells: 128 × 128		Numbers of grid cells: 256 × 256	
	Integral (IP)	Fractional (LP)	Integral (IP)	Fractional (LP)	Integral (IP)	Fractional (LP)
1	1.9	0.5	3.7	2.6	17	10.6
2	4.9	0.8	16.8	3.2	99	15.4
3	6.4	1.4	56.1	5.5	196	30.7
4	8.5	1.8	74.9	8.7	1314	43.3
5	11.6	2.1	494.7	11.3	3015	50.9
6	14.8	2.6	807.5	13.8	4517	62.8
7	68.9	3.3	1572.6	15.9	4884	76.5
8	92.5	3.9	2877.9	18.6	7464	87.3
9	94.2	4.7	3191.4	21.1	10 145	109.4
10	177.5	5.3	5100.2	24.5	14 534	120.0
11	383.0	6.1	11 887.3	29.5	16 438	134.8
12	403.5	6.6	17 391.2	31.2	31 306	162.5
13	[a]	7.8	–	35.0	–	168.44
14	[a]	9.4	–	41.4	–	199.21
15	[a]	10.2	–	45.5	–	222.6
16	[a]	11.4	–	50.2	–	247.3
17	[a]	12.7	–	54.6	–	287.4
18	[a]	13.3	–	58.9	–	304.8
19	[a]	14.7	–	63.6	–	333.8
20	[a]	[a]	–	71.3	–	359.1

–: A feasible solution may exist, but the algorithm failed to find one in 10 h.
[a] The model was proven to be infeasible, and no feasible solution exists.

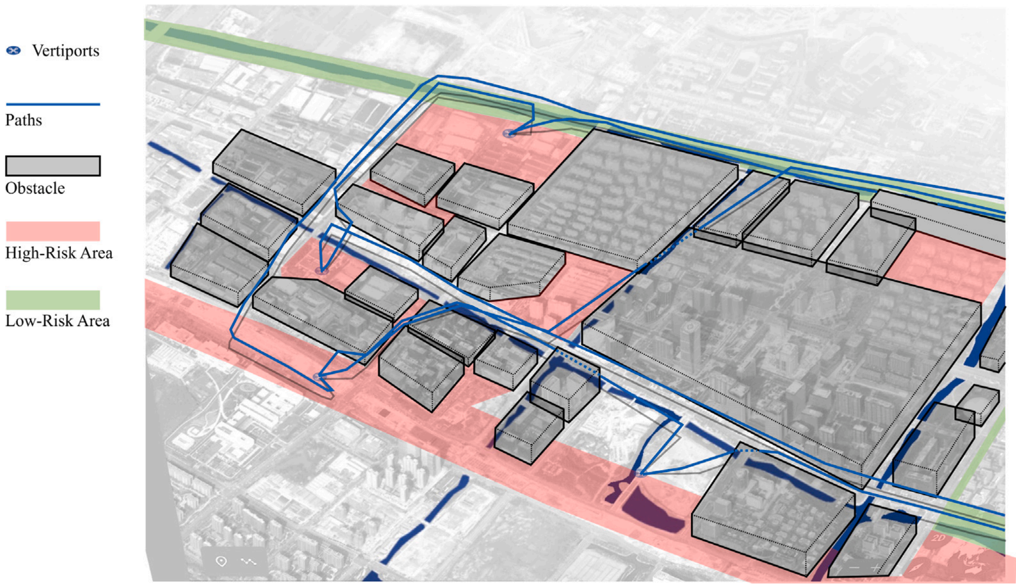


Fig. 13. A set of paths planned by the SRP method for an area in Hangzhou, China.

Gurobi's IP solver is more efficient in solving the MCNF model than GLPK's IP solver. In comparison, in real-world scenarios, the graph-search-based methods may be more viable considering computational time and side constraints.

A limitation here is that the current MCNF model extracts networks from grid cells. Diagonal nodes and other nodes that can "see" each other are not connected in current networks. As a result, generated routes can only follow three coordinate axis directions, which is not desired in real applications. To make generated routes more realistic, nodes that can "see" each other should be connected, but this would make the network extremely large and difficult to solve. Moreover, as urban drone delivery scales up, the airspace could become even more congested, and the graph-search-based methods may fail to solve all potential conflicts and handle such a high level of traffic density. Other emerging computational techniques, e.g. reinforcement learning, evolutionary optimization, etc., shall be explored in the future.

As we grapple with the swift advancement in fields such as drone delivery, air mobility, and other low-altitude aerial activities, the establishment of dynamic and resilient Unmanned Aircraft System Traffic Management (UTM) infrastructures and policies becomes vital. This research focuses on evaluating the complexity of the air corridor planning problem and existing computational

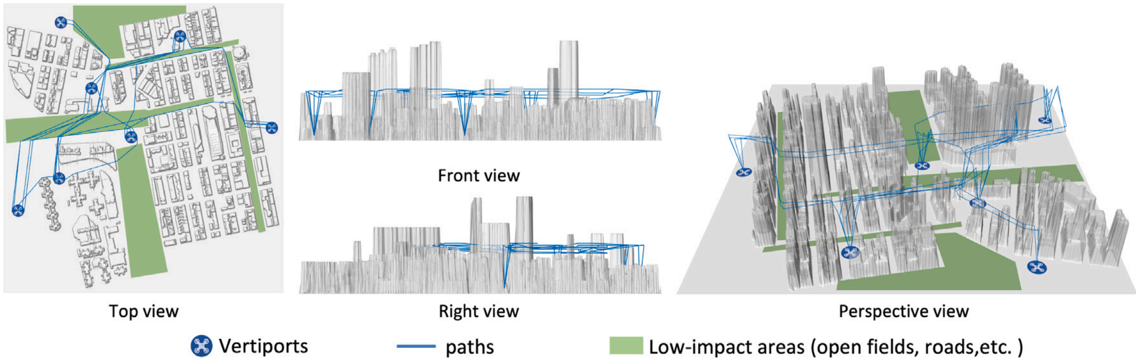


Fig. 14. A set of paths planned by the DRP method for Mong Kok, Hong Kong SAR.

techniques to aid in the planning and operations of UTM. It represents an initial effort to connect two distinct modeling and computational methodologies: the Network Flow model with Integer Programming (IP) and the graph-search-based algorithm.

Our findings offer valuable insights that could guide the formulation of UTM policies on two levels. On the first level, the results of this work demonstrate the significance of computational complexity involved in solving drone route network planning problems. The findings indicate that structured routes are already highly complex; the transition towards free flight in ConOps will introduce even greater computational challenges, thereby increasing complexity. On the second level, the findings in this work show that, for UTM in low-altitude airspace, a centralized multi-path planning approach alone will not be enough. Pre-planned structures and distributed planning would be needed to guarantee the safety, robustness, and efficiency of drone operations in low-altitude airspace. This also implies that the digital sharing of each user's planned and actual flight details would be essential for a reliable, safe, and efficient UTM. This will require new communication, navigation, and surveillance infrastructure and systems to be developed and implemented. Policymakers might consider devising incentives or frameworks to champion such initiatives and investments.

7. Conclusion

Establishing the necessary infrastructure and formulating Concepts of Operations (ConOps) for Unmanned Aircraft System Traffic Management (UTM) is absolutely critical for facilitating large-scale drone operations. Different from most existing research focuses on routing and traffic management problems, this study focuses on the complexity of the air corridor planning problem and computational methodologies used for the planning and operations of UTM. It is one of the few attempts to compare two unique modeling and computation methods: the Network Flow model with Integer Programming (IP) and the graph-search-based algorithm, to address the corridor network planning problem. We conduct a set of experiments to compare an MCNF model, DRP, and SRP algorithm. In summary, the air corridor planning problem is very complex; it is NP-hard and has a large problem size for real-world scenarios. The MCNF using generic IP solvers generates a route network that is optimal with respect to the objective function but requires a significant amount of time. In contrast, the DRP method finds near-optimal solutions quickly and efficiently; the SRP method also finds near-optimal solutions quickly but may quickly fail to find all routes in denser airspace.

Future work is planned in two directions: (1) to advance the current methods to achieve super performance in both computational efficiency and optimality, leveraging the theoretical frameworks of optimization modeling and the computational power of artificial intelligence (AI) techniques; (2) to explore UTM ConOps towards free flight with temporal separations, therefore developing computation methods to support 4-Dimensional Trajectory (4DT) planning for a route network of urban drone delivery service.

To enable a large scale of drone operations in low-altitude airspace, UTM policies need to be adaptive as methods and systems evolve. The evolving nature of technologies underscores the necessity for policies to adjust. A consistent policy review mechanism, which ensures UTM policies remain updated, efficacious, and in tandem with technological advancements, should be an essential consideration for stakeholders.

CRediT authorship contribution statement

Xinyu He: Writing – review & editing, Writing – original draft, Software, Methodology, Investigation, Formal analysis. **Lishuai Li:** Writing – review & editing, Writing – original draft, Validation, Supervision, Project administration, Methodology, Investigation, Funding acquisition, Conceptualization. **Yanfang Mo:** Validation, Formal analysis. **Zhankun Sun:** Validation, Formal analysis. **S. Joe Qin:** Writing – review & editing, Supervision, Funding acquisition.

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.

Acknowledgments

The work was supported by the Hong Kong Research Grants Council General Research Fund (Project No. 11215119, 11200823), City University of Hong Kong Strategic Research Grant (Project No. 7005569) and Strategic Interdisciplinary Research Grant (Project No. 7020098), and Hong Kong Innovation and Technology Commission Innovation and Technology Fund (Project No. GHP/145/20).

Data availability

Data will be made available on request.

References

Ahangar, N.E., Sullivan, K.M., Nurre, S.G., 2020. Modeling interdependencies in infrastructure systems using multi-layered network flows. Comput. Oper. Res. 117, 104883.

Ahuja, R.K., Magnanti, T.L., Orlin, J.B., 1993. Network Flows: Theory, Algorithms, and Applications. Prentice Hall, URL https://dspace.mit.edu/bitstream/handle/1721.1/49424/networkflows00ahuj.pdf, (Accessed 17 June 2024).

Alotaibi, E.T.S., Al-Rawi, H., 2018. A complete multi-robot path-planning algorithm. Auton. Agents Multi-Agent Syst. 32 (5), 693–740. http://dx.doi.org/10.1007/s10458-018-9391-2.

Amaldi, E., Capone, A., Gianoli, L.G., 2013. Energy-aware IP traffic engineering with shortest path routing. Comput. Netw. 57 (6), 1503–1517.

Aweiss, A.S., Owens, B.D., Rios, J., Homola, J.R., Mohlenbrink, C.P., 2018. Unmanned aircraft systems (UAS) traffic management (UTM) national campaign II. In: 2018 AIAA Information Systems-AIAA Infotech@ Aerospace, January 08-12, 2018. p. 1727. http://dx.doi.org/10.2514/6.2018-1727.

Balas, E., Toth, P., 1983. Branch and Bound Methods for the Traveling Salesman Problem. Tech. rep., Carnegie-Mellon University Pittsburgh Pa Management Sciences Research Group.

Barer, M., Sharon, G., Stern, R., Felner, A., 2014. Suboptimal variants of the conflict-based search algorithm for the multi-agent pathfinding problem. In: Seventh Annual Symposium on Combinatorial Search, August 15-17, 2014. Prague, Czech Republic.

Bellman, R., 1958. On a routing problem. Quart. Appl. Math. 16 (1), 87–90.

Bertsimas, D., Patterson, S.S., 2000. The traffic flow management rerouting problem in air traffic control: A dynamic network flow approach. Transp. Sci. 34 (3), 239–255.

Blom, H., Bakker, B., Everdij, M., Park, M.V.D., 2003. Collision risk modeling of air traffic. In: Proceedings of the 2003 European Control Conference (ECC), September 1-4, 2003. IEEE, University of Cambridge, UK, pp. 2236–2241.

Bnaya, Z., Felner, A., 2014. Conflict-oriented windowed hierarchical cooperative a*. In: 2014 IEEE International Conference on Robotics and Automation (ICRA), May 31-June 05, 2014. IEEE, Hong Kong, China, pp. 3743–3748.

Bynum, M.L., Hackebeil, G.A., Hart, W.E., Laird, C.D., Nicholson, B.L., Siirola, J.D., Watson, J.-P., Woodruff, D.L., 2021. Pyomo–Optimization Modeling in Python. Vol. 67, Springer Science & Business Media.

Cappanera, P., Scaparra, M.P., 2011. Optimal allocation of protective resources in shortest-path networks. Transp. Sci. 45 (1), 64–80.

Chin, C., Gopalakrishnan, K., Egorov, M., Evans, A., Balakrishnan, H., 2021. Efficiency and fairness in unmanned air traffic flow management. IEEE Trans. Intell. Transp. Syst. 22 (9), 5939–5951.

Clausen, J., 1999. Branch and Bound Algorithms-Principles and Examples. Tech. rep., Department of Computer Science, University of Copenhagen.

Cohen, L., Uras, T., Kumar, T.S., Koenig, S., 2019. Optimal and bounded-suboptimal multi-agent motion planning. In: Twelfth Annual Symposium on Combinatorial Search, July 16-17, 2019. Napa, California, USA.

Cohen, L., Uras, T., Kumar, T.K.S., Xu, H., Ayanian, N., Koenig, S., 2016. Improved solvers for bounded-suboptimal multi-agent path finding. In: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, July 9-15, 2016. IJCAI '16, AAAI Press, pp. 3067–3074.

Cone, A., Thipphavong, D., Sridhar, B., 2018. TCAS alerts from simulated urban air mobility flights along faa helicopter routes in dallas-fort worth. In: Proceedings of the Aviation Forum 2018.

Dantzig, G.B., Orden, A., Wolfe, P., et al., 1955. The generalized simplex method for minimizing a linear form under linear inequality restraints. Pacific J. Math. 5 (2), 183–195.

De Wilde, B., Ter Mors, A.W., Witteveen, C., 2014. Push and rotate: a complete multi-agent pathfinding algorithm. J. Artificial Intelligence Res. 51, 443–492. http://dx.doi.org/10.1613/jair.4447.

Delahaye, D., Puechmorel, S., Hansman, J., Histon, J., 2003. Air traffic complexity based on non linear dynamical systems. In: Proceedings of the 5th USA/Europe Air Traffic Management Research and Development Seminar, June 23 - 27, 2003. Vol. 12, Budapest, Hungary.

Desaraju, V.R., How, J.P., 2012. Decentralized path planning for multi-agent teams with complex constraints. Auton. Robots 32 (4), 385–403.

Dijkstra, E.W., 1959. A note on two problems in connexion with graphs. Numer. Math. 1 (1), 269–271. http://dx.doi.org/10.1007/BF01386390.

Dinitz, Y.A., 1970. An algorithm for the solution of the problem of maximal flow in a network with power estimation. Dokl. Akad. Nauk 194 (4), 754–757.

Elbert, R., Müller, J.P., Rentschler, J., 2020. Tactical network planning and design in multimodal transportation–a systematic literature review. Res. Transp. Bus. Manage. 35, 100462.

EUROCONTROL, 2018. Unmanned aircraft systems (UAS) ATM integration. URL https://www.eurocontrol.int/publication/unmanned-aircraft-systems-uas-atm-integration.

Even, S., Itai, A., Shamir, A., 1975. On the complexity of time table and multi-commodity flow problems. In: 16th Annual Symposium on Foundations of Computer Science (Sfcs 1975), October 13-15, 1975. pp. 184–193. http://dx.doi.org/10.1109/SFCS.1975.21.

Felner, A., Stern, R., Shimony, S.E., Boyarski, E., Goldenberg, M., Sharon, G., Sturtevant, N., Wagner, G., Surynek, P., 2017. Search-based optimal solvers for the multi-agent pathfinding problem: Summary and challenges. In: Proceedings of the 10th Annual Symposium on Combinatorial Search, June 16-17, 2017. 2017-Janua, Pittsburgh, Pennsylvania, USA, pp. 29–37.

Ferrera, E., Capitan, J., Castano, A.R., Marron, P.J., 2017. Decentralized safe conflict resolution for multiple robots in dense scenarios. Robot. Auton. Syst. 91, 179–193. http://dx.doi.org/10.1016/j.robot.2017.01.008.

Floyd, R.W., 1962. Algorithm 97: shortest path. Commun. ACM 5 (6), 345.

Ford, L.R., Fulkerson, D.R., 1956. Maximal flow through a network. Canad. J. Math. 8, 399–404.

Frank, Jr., C.R., 1965. A note on the assortment problem. Manage. Sci. 11 (7), 724–726.

Fredman, M.L., Tarjan, R.E., 1987. Fibonacci heaps and their uses in improved network optimization algorithms. J. ACM 34 (3), 596–615.

Gilboa, A., Meisels, A., Felner, A., 2006. Distributed navigation in an unknown physical environment. In: Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems, May 8 - 12, 2006. ACM Press, New York, New York, USA, pp. 553–560. http://dx.doi.org/10.1145/1160633.1160735.

GLPK, 2021. GLPK (GNU linear programming kit). URL https://www.gnu.org/software/glpk/.
Goldbeck, N., Angeloudis, P., Ochieng, W.Y., 2019. Resilience assessment for interdependent urban infrastructure systems using dynamic network flow models. Reliab. Eng. Syst. Saf. 188, 62–79.
Goldberg, A.V., Tarjan, R.E., 1988. A new approach to the maximum-flow problem. J. ACM 35 (4), 921–940.
Gurobi, 2021. Gurobi optimizer reference manual. URL https://www.gurobi.com/documentation/.
Hart, P.E., Nilsson, N.J., Raphael, B., 1968. A formal basis for the heuristic determination of minimum cost paths. IEEE Trans. Syst. Sci. Cybern. 4 (2), 100–107.
Hassin, R., 1983. The minimum cost flow problem: a unifying approach to dual algorithms and a new tree-search algorithm. Math. Program. 25, 228–239.
He, X., He, F., Li, L., Zhang, L., Xiao, G., 2022. A route network planning method for urban air delivery. Transp. Res. E 166, 102872. http://dx.doi.org/10.1016/J.TRE.2022.102872, URL www.sciencedirect.com/science/article/pii/S1366554522002526.
He, X., Li, L., Mo, Y., Huang, J., Qin, S.J., 2024. A distributed route network planning method with congestion pricing for drone delivery services in cities. Transp. Res. C 160, 104536. http://dx.doi.org/10.1016/j.trc.2024.104536, URL https://www.sciencedirect.com/science/article/pii/S0968090X24000573.
Hoekstra, J.M., Van Gent, R.N., Ruigrok, R.C., 2002. Designing for safety: The 'free flight' air traffic management concept. Reliab. Eng. Syst. Saf. 75 (2), 215–232. http://dx.doi.org/10.1016/S0951-8320(01)00096-5.
Hu, T., Kahng, A.B., Robins, G., 1993. Optimal robust path planning in general environments. IEEE Trans. Robot. Autom. 9 (6), 775–784.
Jung, J., Ippolito, C., Rogers, C., Kerczewski, R., Downey, A., Matheou, K., 2018. Small unmanned aircraft electromagnetic interference (EMI) initial assessment. In: Proceedings of the 2018 Integrated Communications, Navigation, Surveillance Conference (ICNS), April 10-12, 2018. IEEE, pp. 2A3–1.
Katayama, N., 2019. A combined fast greedy heuristic for the capacitated multicommodity network design problem. J. Oper. Res. Soc. 70 (11), 1983–1996.
Klein, M., 1967. A primal method for minimal cost flows with applications to the assignment and transportation problems. Manage. Sci. 14 (3), 205–220.
Kochenderfer, M.J., Edwards, M.W.M., Espindle, L.P., Kuchar, J.K., Griffith, J.D., 2010. Airspace encounter models for estimating collision risk. J. Guid. Control Dyn. 33, 487–499.
Kopardekar, P., Bradford, S., 2017. UAS Traffic Management (UTM): Research Transition Team (RTT) Plan. Tech. rep., Federal Aviation Administration, URL https://www.faa.gov/sites/faa.gov/files/uas/research_development/traffic_management/utm_pilot_program/FAA_NASA_UAS_Traffic_Management_Research_Plan.pdf, (Accessed 17 June 2024).
Lee II, E.E., Mitchell, J.E., Wallace, W.A., 2007. Restoration of services in interdependent infrastructure systems: A network flows approach. IEEE Trans. Syst. Man Cybern. C (Appl. Rev.) 37 (6), 1303–1317.
Lemardelé, C., Estrada, M., Pagès, L., Bachofner, M., 2021. Potentialities of drones and ground autonomous delivery devices for last-mile logistics. Transp. Res. E 149, 102325. http://dx.doi.org/10.1016/j.tre.2021.102325.
Luna, R., Bekris, K.E., 2011. Efficient and complete centralized multi-robot path planning. In: 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, September 25-30, 2011. San Francisco, CA, USA, pp. 3268–3275. http://dx.doi.org/10.1109/IROS.2011.6095085.
Mesquita, M., Moz, M., Paias, A., Pato, M., 2015. A decompose-and-fix heuristic based on multi-commodity flow models for driver rostering with days-off pattern. European J. Oper. Res. 245 (2), 423–437.
Mitchell, J.E., 2002. Branch-and-cut algorithms for combinatorial optimization problems. Handb. Appl. Optim. 1 (1), 65–77.
Mohamed Salleh, M.F.B., Wanchao, C., Wang, Z., Huang, S., Tan, D.Y., Huang, T., Low, K.H., 2018. Preliminary concept of adaptive urban airspace management for unmanned aircraft operations. In: AIAA Information Systems-AIAA Infotech@ Aerospace, January 8-12, 2018. Kissimmee, Florida, USA, p. 2260.
Murray, C.C., Raj, R., 2020. The multiple flying sidekicks traveling salesman problem: Parcel delivery with multiple drones. Transp. Res. C 110, 368–398. http://dx.doi.org/10.1016/j.trc.2019.11.003.
NASA, 2022. UTM concept of operations v2.0. URL https://www.faa.gov/sites/faa.gov/files/2022-08/UTM_ConOps_v2.pdf.
Olfati-Saber, R., Fax, J.A., Murray, R.M., 2007. Consensus and cooperation in networked multi-agent systems. Proc. IEEE 95 (1), 215–233.
Orlin, J.B., 1997. A polynomial time primal network simplex algorithm for minimum cost flows. Math. Program. 78, 109–129.
Padberg, M., Rinaldi, G., 1991. A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems. SIAM Rev. 33 (1), 60–100.
Pentico, D.W., 2008. The assortment problem: A survey. European J. Oper. Res. 190 (2), 295–309.
Perrin, E., Kirwan, B., Stroup, R., 2007. A systemic model of ATM safety: the integrated risk picture. In: Proceedings of the 7th USA/Europe ATM R&D Seminar, Barcelona, July 02-05,2007. Citeseer, Barcelona. Spain.
Potra, F.A., Wright, S.J., 2000. Interior-point methods. J. Comput. Appl. Math. 124 (1–2), 281–302.
Purwin, O., D'Andrea, R., Lee, J.-W., 2008. Theory and implementation of path planning by negotiation for decentralized agents. Robot. Auton. Syst. 56 (5), 422–436.
Qu, Y., Bektaş, T., Bennell, J., 2016. Sustainability SI: multimode multicommodity network design model for intermodal freight transportation with transfer and emission costs. Netw. Spat. Econ. 16, 303–329.
Quddus, M., Washington, S., 2015. Shortest path and vehicle trajectory aided map-matching for low frequency GPS data. Transp. Res. C 55, 328–339.
Saab, Y., VanPutte, M., 1999. Shortest path planning on topographical maps. IEEE Trans. Syst. Man Cybern. A 29 (1), 139–150.
Saha, M., Isto, P., 2006. Multi-robot motion planning by incremental coordination. In: 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, October 09-15, 2006. IEEE, Beijing, China, pp. 5960–5963. http://dx.doi.org/10.1109/IROS.2006.282536.
Salimifard, K., Bigharaz, S., 2022. The multicommodity network flow problem: state of the art classification, applications, and solution methods. Oper. Res. 22 (1), 1–47.
SESAR, 2021. Sustainability, airspace optimization and urban air mobility - focus of latest very large-scale demonstrations. URL https://www.sesarju.eu/news/sustainability-capacity-and-urban-air-mobility-focus-newly-launched-sesar-demonstration-call.
Sharon, G., Stern, R., Felner, A., Sturtevant, N.R., 2015. Conflict-based search for optimal multi-agent pathfinding. Artificial Intelligence 219, 40–66. http://dx.doi.org/10.1016/j.artint.2014.11.006.
Silver, D., 2005. Cooperative pathfinding. In: Proceedings of the First Artificial Intelligence and Interactive Digital Entertainment, June 01-05, 2005. Vol. 1, Marina Del Rey, Marina del Rey, California, USA, pp. 117–122.
Standley, T., 2010. Finding optimal solutions to cooperative pathfinding problems. In: Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, July 11-15, 2010. Vol. 24, Atlanta, Georgia, USA, pp. 173–178.
Sturtevant, N.R., 2012. Benchmarks for grid-based pathfinding. IEEE Trans. Comput. Intell. AI Games 4 (2), 144–148.
Sunil, E., Hoekstra, J., Ellerbroek, J., Bussink, F., Nieuwenhuisen, D., Vidosavljevic, A., Kern, S., 2015. Metropolis: Relating airspace structure and capacity for extreme traffic densities. In: Proceedings of the 11th USA/Europe Air Traffic Management Research and Development Seminar, June 23-26, 2015. Lisbon, Portugal.
Tang, H., Zhang, Y., Mohmoodian, V., Charkhgard, H., 2021. Automated flight planning of high-density urban air mobility. Transp. Res. C 131, 103324. http://dx.doi.org/10.1016/J.TRC.2021.103324.
Ushijima, H., 2017. UTM project in Japan. In: Proceedings of the Global UTM Conference, Montreal, QC, Canada. Vol. 26.
Vascik, P.D., Hansman, R.J., 2017. Evaluation of key operational constraints affecting on-demand mobility for aviation in the los angeles basin: ground infrastructure, air traffic control and noise. In: Proceedings of the 17th AIAA Aviation Technology, Integration, and Operations Conference, June 5-9, 2017. Denver, Colorado, USA, p. 3084, URL https://arc.aiaa.org/doi/abs/10.2514/6.2017-3084.
Vascik, P.D., Hansman, R.J., 2019. Development of vertiport capacity envelopes and analysis of their sensitivity to topological and operational factors. In: Proceedings of the AIAA Scitech 2019 Forum, January 07-11, 2019. p. 0526.

Vascik, P.D., Hansman, R.J., Dunn, N.S., 2018. Analysis of urban air mobility operational constraints. J. Air Transp. 26, 133–146.

Vascik, P.D., Jung, J., 2016. Assessing the impact of operational constraints on the near-term unmanned aircraft system traffic management supported market. In: Proceedings of the 16th AIAA Aviation Technology, Integration, and Operations Conference, June 13-17, 2016. p. 4373, URL https://arc.aiaa.org/doi/abs/10.2514/6.2016-4371.

Vaziri, S., Etebari, F., Vahdani, B., 2019. Development and optimization of a horizontal carrier collaboration vehicle routing model with multi-commodity request allocation. J. Clean. Prod. 224, 492–505.

Wagner, G., Choset, H., 2015. Subdimensional expansion for multirobot path planning. Artificial Intelligence 219, 1–24.

Wang, Z., Delahaye, D., Farges, J.-L., Alam, S., 2021. Air traffic assignment for intensive urban air mobility operations. J. Aerosp. Inf. Syst. 18, 860–875.

Wang, Z., Delahaye, D., Farges, J.L., Alam, S., 2022a. Complexity optimal air traffic assignment in multi-layer transport network for urban air mobility operations. Transp. Res. C 142, 103776. http://dx.doi.org/10.1016/J.TRC.2022.103776.

Wang, Z., Delahaye, D., Farges, J.-L., Alam, S., 2022b. Route network design in low altitude airspace for future urban air mobility operations: A case study of urban airspace of Singapore. In: International Conference on Research in Air Transportation, June 19-23, 2022. Tampa, United States.

Wang, L., Fang, L., Hipel, K.W., 2008. Basin-wide cooperative water resources allocation. European J. Oper. Res. 190 (3), 798–817.

Wang, H., Rubenstein, M., 2020. Walk, stop, count, and swap: decentralized multi-agent path finding with theoretical guarantees. IEEE Robot. Autom. Lett. 5 (2), 1119–1126.

Wu, Z., Zhang, Y., 2021. Integrated network design and demand forecast for on-demand urban air mobility. Engineering 7, 473–487.

Yaghini, M., Momeni, M., Sarmadi, M., 2012. A simplex-based simulated annealing algorithm for node-arc capacitated multicommodity network design. Appl. Soft Comput. 12 (9), 2997–3003.

Yang, Z., 2023. Food delivery by drone is just part of daily life in shenzhen. URL https://www.technologyreview.com/2023/05/23/1073500/drone-food-delivery-shenzhen-meituan, (Accessed 17 June 2024).

Yu, J., LaValle, S., 2013. Structure and intractability of optimal multi-robot path planning on graphs. In: Proceedings of the AAAI Conference on Artificial Intelligence, July 14-18, 2013. Vol. 27, pp. 1443–1449.

Yu, J., LaValle, S.M., 2016. Optimal multirobot path planning on graphs: Complete algorithms and effective heuristics. IEEE Trans. Robot. 32 (5), 1163–1177.

Zhang, Z., Cheang, B., Li, C., Lim, A., 2019. Multi-commodity demand fulfillment via simultaneous pickup and delivery for a fast fashion retailer. Comput. Oper. Res. 103, 81–96.