Transportation Research Part C 169 (2024) 104882

Contents lists available at ScienceDirect

Transportation Research Part C

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

Airspace network design for urban UAV traffic management with congestion[☆]

Leanne Stuive, Fatma Gzara [*]

Department of Management Science and Engineering, University of Waterloo, Canada

ARTICLE INFO

Keywords:
Airspace network design
UAV traffic management
Airspace congestion

ABSTRACT

To support the safe and widespread use of unmanned aerial vehicles (UAVs) in urban environments, industry stakeholders and regulatory authorities are partnering to develop urban airspace traffic management systems (UTMs). UTM system providers face strategic decisions in how to design and manage airspace available to UAV flights. We consider a provider that plans to open an urban airspace in which UAV flights are routed above existing roads in 3D corridors corresponding to segmented altitude levels. The provider aims to select a subset of the road network to form an air-network with the goal of providing safe and cost effective service for UAV traffic. The air-network selected must provide routes that respect UAV technology restrictions, and must have adequate capacity to support the expected flight volume. We develop a 3D airspace network design model that selects a subset of roads whose 3D projection into the sky will be used for routing flights. The constrained system optimum (CSO) traffic assignment model is used to evaluate the quality of the network; the CSO user constraints represent battery restrictions while minimizing the total travel time ensures realistic routing in the face of congestion. To incorporate the 3D nature of flights, we use simulation to calibrate a Bureau of Public Roads capacity parameter that reflects the multiple vertical layers of airspace made available when a road is selected for the network. We introduce a methodology to derive candidate maps for urban areas and use it on open-source data to build a case study for Chicago city center. We assess the impact of budget, congestion, minimum-path deviation, and demand patterns on network designs.

1. Introduction

As unmanned aerial vehicle (UAV), or drone, technology evolves, UAVs have become an increasingly viable mode for transporting goods. UAVs are not limited by road traffic and provide a greener alternative to trucks for last mile delivery of consumer packages in some settings (Kirschstein, 2020). In 2022, Amazon Prime Air (Staff, 2022) announced trials in Lockeford California, Alphabet's Project Wing trialed UAV delivery in Ireland (O'Brien, 2022), and Walmart made UAV delivery available to four million customers (Walmart, 2022). Smaller companies are also active in the UAV delivery space, with Flytrex receiving Federal Aviation Administration (FAA) permission to increase the flight radius of their delivery zones (Edwards, 2022) and SkyDrop partnering with Dominos Pizza in New Zealand (Skydrop, 2022). Continued advances in UAV battery life and control capability – especially in inclement weather – will further the case for wide-spread UAV usage.

[☆] This research work is carried out by the authors in collaboration with industry partner AirMatrix under the Sponsored Research Agreement SRA# 089089, jointly funded by NSERC Alliance grant, Canada ALLRP576624 – 22. The authors would like to acknowledge AirMatrix team's support in providing expert knowledge about UAV traffic management.

[*] Corresponding author.
E-mail address: fgzara@uwaterloo.ca (F. Gzara).

https://doi.org/10.1016/j.trc.2024.104882
Received 11 October 2023; Received in revised form 15 July 2024; Accepted 5 October 2024
Available online 25 October 2024

However, a major hindrance to UAVs at scale is the lagging development of government regulations to support the safe and fair operation of many simultaneous flights. Government regulations surrounding UAV usage differ by country and continue to evolve; commercial users must be aware of and adapt to local requirements (Stöcker et al., 2017). Various frameworks for regulating UAV flights have been proposed (Jiang et al., 2016); many of these frameworks depend on a UAV traffic management system (UTM) to track ongoing UAV flights and facilitate the approval of new flight requests. The UTM concept of operations provides necessary roles (ex. UAV pilot, traffic manager), processes (ex. flight request, monitor traffic), and technologies (ex. supporting software and datasets) to manage UAV traffic in an airspace (Kopardekar, 2019). The concept is similar to existing air traffic management systems and the civil aviation authority (ex. FAA in USA, NavCanada in Canada) is leading the integration of UAVs in many countries. We focus here on a pressing technical design issue for UTMs; for an economics and policy perspective see Decker and Chiambaretto (2022), and for a management perspective see Merkert and Bushell (2020).

Within the UTM concept of operations, we use the term *UTM provider* for an entity that provides the tools required to develop and administer a UAV airspace. In other words, a UTM provider builds the tools, software, and system required to implement an urban airspace within the UTM framework. One key process in a UTM provider solution is the submission and approval of flight requests via a Web API, or similarly automated system. In some cases the corresponding aviation authority ultimately grants the flight request approval. For example, in the United States, UAV flights under 400ft in controlled airspace must receive approval from the FAA. This approval can be obtained through a UTM provider that has integrated with the automated Low Altitude Authorization and Notification Capability system. In other cases the approval of flights may be required only at the UTM level; flight notification and approval allow the traffic manager to ensure that there are no conflicting flights. Although detect-and-avoid capability for UAVs is improving, reservation of airspace coupled with flight request approval is the next step towards unlocking UAV flights beyond visual line of sight, and at scale.

Current iterations of UTMs are focused on flight request approvals in low risk areas. As flight requests in densely populated cities increase, the design of airspace and management of simultaneous UAV flights in urban environments is a critical area of UTM design. The "best", or *greedy*, path for a single UAV to travel between two locations is point-to-point (i.e. via the Euclidean shortest path), avoiding obstacles, and at its ideal altitude (which depends on flight length, weather, etc.). If many flights are requested in a short period of time, their greedy paths may conflict and UAV pilots would have to either wait long times to start with a cleared path, or drain battery while hovering en route. A well-designed urban airspace could segment and separate traffic to prevent the likelihood of such path conflicts. Bauranov and Rakas (2021) summarize recent academic and industry literature on urban airspace design and various UTM design proposals. They define the *physical structure of urban airspace* to refer to the position and size of airspace elements such as flying trajectories, tubes, corridors, and layers, as well as the associated rules of operation.

The design of airspace for traditional aircraft provides guidance for UTM airspace, but key differences apply. As UAVs fly at low altitudes, tall buildings and ground features are physical obstacles that must be avoided and can cause communication interference when flying nearby. The technical capabilities of UAVs are more disparate than for traditional aircraft because a UAV may be sized, built, and equipped for a very specific task. One defining technical difference may be the propulsion mechanics of the vehicle; some UAVs generate lift using wings ("fixed-wing"), but the more maneuverable UAVs are rotorcraft that rely on rotating blades to generate lift. Many UAVs rely on electric power, meaning flight length – in terms of both time and distance – is limited by battery capacity, rather than a conventional fuel supply. Most consumer UAVs need to recharge after 30-45 min of flight time. The distance between vehicles required for safe operation, or *separation standards*, are much smaller for UAVs than for traditional airplanes. Exact separation standards are still in development and expected to evolve over time to reflect the maturity of the underlying systems and technology.

Although UAVs are capable of direct point-to-point flight, an airspace design that restricts UAVs to fly along existing road networks provides many advantages in urban settings. A similar design is discussed by Jang et al. (2017) who propose using "sky lanes" in urban centers to mimic street organization. Doole et al. (2021) focus on flying above roads, observing it provides separation, reduces privacy concerns, and enables ground vehicle coordination. To test the airspace design, they develop one-way and two-way airspace configurations with altitude levels segmented by turning/through traffic and heading direction. Then, they simulate air traffic over all the streets of Manhattan, assuming conflict avoidance procedures instead of flight approval processes. As UAVs in the city scale, a version of this design which integrates flight approvals is a more practicable next step. A UAV airspace above roads is also a practical choice because it enables the rapid collection of the telecommunication, location, and weather data required to provide precise safe paths for UAV flight. Some of this data must be obtained with physical measurements made by traversing the underlying terrain with a vehicle. This procedure is easiest to perform when the ground below is an already navigable street. The term *mapping* will refer to the procedure of collecting data on the streets above which UAVs will fly.

Mapping and maintaining the data required for safe UAV flight over every city street would be cost-prohibitive. UTM providers may prefer to map a portion of streets so that they can safely scale capacity, be responsive to a change in regulation, and build infrastructure that is supported by demand. In this work, we take the point of view of our industry partner, whose needs align with a UTM provider opening up UAV airspace by implementing a UTM system. The tactical decision they face is determining which roads of a city to map for UTM service first. We develop an optimization model to select a set of roads in a city to support UAV flights. Once selected and mapped, the roads, sky above, and supporting data provide the initial 3D pathways of a scalable UAV traffic network, or *UTM network*. UAV pilots will request to fly between points on the network and, with approval, execute flight plans restricted to a layer of the airspace above the mapped network.

The quality of a candidate UTM network is measured by its ability to service expected flight requests. UAV pilots provide flight requests by specifying their origin, destination, and battery limit. A network has adequate connectivity to service a flight request if there is an origin to destination 3D route whose trajectory is limited to airspace above mapped roads and whose trajectory can be traversed by the UAV within the time limit imposed by its battery life. If certain roads are oversubscribed, however, a flight request may not be accommodated in real time because the network has inadequate capacity and therefore there is congestion along the route. Either the UAV will need to wait too long for traffic to clear, or experience untenable delays en route due to hovering and excessive altitude switching. To incorporate both route length limits and congestion effects in our model, we use constrained system optimum (CSO) traffic assignment to evaluate routings allowed by a candidate network. CSO traffic assignment minimizes total travel time, where travel time along a road section is a function of the amount of traffic using the road. For example, The Bureau of Public Roads (BPR) function (Bureau of Public Roads, 1964) is often used to model travel time as a function of traffic flow on ground highways.

The main contribution of this paper is the first UAV network design model for UTM providers. The model differs from existing UAV research in that it considers the 3D nature of flights through a congestion function, assumes UAVs fly over existing streets rather than via point-to-point flight, and explicitly evaluates routings which are designed in respect of regulatory constraints and airspace design. We use simulation to develop BPR congestion function parameters consistent with routing UAVs in 3D space in respect of separation standards and along multiple corridors according to altitude. The single level model we develop is the first to extend the constrained system optimum road traffic model to include a network design component. As demand for UAV flights increases, UTM providers need to be agile to expand their network rapidly. To this end, the models and methodology developed can also be used in a network expansion context. We focus on initial network planning in this paper because UTMs are in early stages of development.

The remainder of the paper is structured hence. Section 2 summarizes related literature in UAV Operations Research, traffic modeling, and network design. Section 3 provides an exact mathematical formulation of the model we develop, explains how it captures various real-world constraints, and introduces the approximation we solve which uses piecewise linear approximation. Section 4 discusses designing congestion functions that account for the projection of 2D ground roadways into 3D pathways in the sky. Section 5 contains the results of numerical experiments. Its primary focus is a case study of UTM network design for Chicago city center, covering end-to-end modeling methodology from instance generation to sensitivity analysis to runtimes. Section 6 concludes with a summary of the results and future research directions.

2. Literature review

To the best of our knowledge, no academic literature addresses selecting ground roadways in the interest of managing UAV traffic by projecting the chosen roads to 3D pathways in the sky. Of the abundant UAV-specific Operations Research work, the most closely related models focus on designing a UAV distribution network for a company that is integrating a UAV fleet into their logistics operations. These models assume point-to-point flight with recharging, and disregard regulatory constraints and path conflicts. In fact, the UTM network design problem we formulate shares many similarities to transportation planning for ground vehicles. In our literature review, we focus first on existing UAV-specific Operations Research models and then discuss the most relevant concepts in ground traffic routing and network design.

2.1. Operations research models for UAV applications

As the use of UAVs in civilian and commercial enterprises has become more economically viable, many Operations Research practitioners and researchers have studied the related strategic, tactical, and operational challenges. The authors of Moshref-Javadi and Winkenbach (2021), Otto et al. (2018), and Poikonen and Campbell (2021) survey the breadth of logistics models for civilian UAV applications. Each survey points to integrating regulatory constraints into models within its discussion of future research directions. In particular, Moshref-Javadi and Winkenbach (Moshref-Javadi and Winkenbach, 2021) observe that risk, chaos, and safety issues are of “growing concern” as UAV usage increases. They identify dedicated aerial highways, standard routing protocols, and stochastic risk models as possible ways to mitigate these concerns. Otto et al. (2018) observe that it is “important to work out technological and regulatory solutions that will address public concerns of privacy and safety without impeding value-creating drone deployment”. Lastly, Poikonen and Campbell (Poikonen and Campbell, 2021) observe that “thus far, there has been little research on drone routing related to UTM and constraints UTMs may impose”.

The UAV logistical challenges receiving the most attention have been operational problems and, more specifically, problems related to routing (i.e. choosing flight paths and schedules). Models and solutions have been proposed for: routing fleets of delivery UAVs, coordinating UAV and truck deliveries, routing UAVs for inspecting physical infrastructure, and routing UAVs for surveillance purposes. For surveys on some of these problems, see Chung et al. (2020) and Macrina et al. (2020). Most routing-based work assumes UAVs fly directly point-to-point and few consider flight approval requirements, flight delays caused by waiting for traffic, and regulatory restrictions on where a UAV may fly. One exception is Chin et al. (2021) which considers optimization of traffic managed within a UTM as an extension of classical air traffic flow management. The authors consider approaches to fairly integrate flights known in advance with on-demand requests, showing fairness can be improved while maintaining efficiency. They perform a case study for package delivery in Toulouse France.

The managerial models most closely related to UTM network design pertain to designing distribution networks that integrate a UAV fleet. The strategic challenge faced is determining the commercial viability of using a UAV fleet for certain deliveries. To perform the analysis, optimized decisions regarding infrastructure placement are considered. For example, Shavarani et al. (2019a) and Shavarani et al. (2019b) use genetic algorithms to solve models minimizing cost, travel distance, and lost demand for UAV delivery systems. These models capture: the strategic costs of building charging stations, building warehouses, and procuring UAVs;

and the operational costs incurred when operating the UAVs to serve demand. Chauhan et al. (2019) address a similar problem using heuristics, but do not incorporate as many operational features such as congestion at facilities. Other work addresses only the problem of charging station placement; for example Cokyasar et al. (2021), Hong et al. (2018), and Pinto and Lagorio (2022). Baloch and Gzara (Baloch and Gzara, 2020) also model strategic network design for UAV parcel delivery and study sensitivity to technological limitations and government regulation. They deviate from other strategic models in that the objective is not to minimize delivery cost but rather to maximize return from a set of delivery services offered by an e-retailer where customer demand is utility driven. All of these models incorporate battery restrictions by limiting the capabilities of a UAV within the model constraints or instance considered.

Hou et al. (2021) provide the first UAV distribution network design model that incorporates lane capacity in addition to charging needs. The model is also differentiated in its use of distributionally robust modeling to deal with demand uncertainty. The authors observe that "total infrastructure costs can be saved by pooling drone flows into a small number of high-capacity channels/transfer airports". However, the lanes they consider are Euclidean shortest paths between candidate facility locations, rather than 3D corridors above existing ground roadways. To model airspace congestion, the capacity of these lanes is taken to be a hard limit determined by an investment decision.

All the distribution network design models cited so far assume, often implicitly, that: (1) the fleet operator does not need to compete with other airspace traffic, and (2) UAVs may fly freely in the local airspace. Our work provides an alternative point of view — that of the UTM provider which will eventually operate the high demand airspace in urban centers and impose system constraints on UAV flight. This point of view investigates a research gap identified by the UAV logistic model surveys, and may inform future operational models which will need to integrate UTM constraints.

2.2. Related concepts in traffic modeling and network design

Determining how UAV traffic would be routed on a fixed network is, in and of itself, a nontrivial problem. The class of mathematical models that determine routings are referred to as *traffic assignment* problems. What is common to all these models is that they predict traffic flow patterns while accounting for congestion effects. Traffic assignment problems date back to the seminal work of Wardrop, who introduced the user equilibrium (selfish routing on shortest path) and system optimal (smallest possible total travel time) routing principles (Wardrop, 1952). From a practitioner's point of view, traffic assignment is one aspect of a transportation forecasting model, alongside trip generation, trip distribution, and mode choice; for more information see Patriksson (2015), Chiu et al. (2011) and Perederieieva et al. (2015). Here we restrict our discussion to CSO traffic assignment because it closely models the routing requirements we choose for our UTM network design problem; see Section 3.1 for details on this design decision. The CSO traffic assignment problem is a static model which performs system optimal routing subject to user constraints. As a static model, the CSO problem assesses steady-state behavior via a congestion function. The goal is to route vehicles in a system optimal way; that is, the objective is to minimize the total travel time on the network. However, the paths chosen are subject to a user constraint which requires the individual travel time from origin to destination for each vehicle to respect some corresponding limit.

The CSO problem first appeared in Jahn et al. (2000, 2005), where the authors studied traffic assignment models that minimize total travel time subject to route lengths being within a given multiplicative factor (usually between 105% and 150%) of the length of the shortest path. The length of a path is determined by the sum of the lengths of the arcs en route. The model is restricted in that the arc lengths must be *normal*, meaning that the length of an arc cannot depend on the amount of traffic assigned, and further implying that the set of allowable paths is fixed prior to the optimization. The authors use a Frank Wolfe type algorithm alongside column generation to solve the model and they analyze the results for a subset of the Transportation Networks for Research (Transportation Networks for Research Core Team, 2022) instances. Angelelli et al. (2020, 2021) also study practical instances of the CSO problem. They solve the problem using piecewise linear approximation and heuristic path generation techniques. These techniques allow for the arc lengths to depend on the amount of traffic assigned, and computational results show that the solutions found are of good quality.

While traffic assignment pertains to routing vehicles/flow on a fixed network, *network design problems* are primarily concerned with selecting arcs to form a network. The corresponding models typically include arc-based binary variables that are used to model the inclusion or exclusion of arcs in the network. Network design models are prevalent in transportation planning; for more information see Crainic et al. (2021). Here, we restrict our discussion to network design models which incorporate congestion and path length limits. Bayram et al. (2015) consider a disaster response application where individuals are instructed to evacuate their homes and drive along roadways to a specific shelter location. The objective is to minimize total congestion subject to CSO routing of evacuees and a budget limit on the number of shelters that may be opened. The network design decision is which nodes to open; a node represents the candidate location of a shelter. The authors solve the model with second order cone programming (SOCP) techniques. SOCP techniques are also used by Gürel (2011) to minimize congestion plus arc capacity installation cost subject to routing all flow on a network. However, the network routing is not subject to path length limits. Most transportation modeling work that considers congestion but not path length limits is done respecting user equilibrium routing (Marcotte, 1986). For example, Mathew and Sharma (2009) consider capacity expansion decisions for a road network with the goal of minimizing total congestion.

In capacity expansion applications, an arc's capacity may be increased incrementally at a per unit cost. Our UTM network design problem differs in that UTM network capacity decisions are binary. Either the arc is open, and the corresponding layers of airspace are available, or, the arc is closed and no airspace is available. The capability of open airspace is not a model decision but a parameter determined by separation requirements coupled with flight ceilings, and is reflected in the congestion function. Contrary to work on expanding existing road traffic networks, our UTM network design model applies to transportation planning for an emerging mode of transport with no existing infrastructure. We compare CSO traffic assignment for any networks within budget, as opposed to weighed against capacity installation cost.

3. UTM network design model

The core decision to model is simple: which roads should be selected for use by UAV flights? There is a cost of selecting a section of road which reflects both the initial mapping cost and any ongoing maintenance costs. Throughout, we will refer to the roads selected as forming a *UTM network* because this represents the primary network design decision. However, it is important to recognize that mapping a road will make multiple vertical layers of airspace available to UAV traffic because UAVs operate in three dimensional space.

3.1. Design choices

To assess a candidate UTM network, we require as input an indication of where UAV flights are expected to originate and terminate. This information will be represented as an OD-list where each element is a tuple that indicates: the starting location (*origin*), ending location (*destination*), and a demand number specifying the expected number of flights per hour from origin to destination. This is similar to static road traffic assignment problems where an OD-matrix is used to specify road traffic flows between every pair of intersections in a road network. A list of *OD-pairs* is a more appropriate data structure than a matrix in our application because the demand is sparse; it will usually include locations of known UTM clients (often early adopters) and other pairs that target a coverage-based objective. This is in contrast to road traffic modeling which is based on existing, observed traffic patterns of much larger volumes (i.e. where the number of vehicles in the system per hour is magnitudes larger). We return to the discussion of demand patterns for UAV networks in Section 5.1.2. Note that a sparser demand pattern is also required for the models developed to be solvable.

A UTM provider will have many competing criteria for evaluating the quality of a UTM network. In this paper, we focus on *connectivity* and *capacity*; in Section 6 we outline other criteria which provide future research directions. In evaluating connectivity we aim to answer: for how many OD-pairs is there an acceptable route on the UTM network? In evaluating capacity, we aim to answer: how many UAV flights can the UTM network support? The answers to these questions, in large part, depend on how the UTM provider would route UAV flight requests on the network as they arrive in real-time. However, we do not embed an exact routing protocol into our optimization model because traffic rules and routing strategies for UAVs are in early stages of development. Instead, we use industry-informed assumptions and approximations to capture an indicative assessment of the network's capabilities. This approach is consistent with road traffic methodology where static macroscopic traffic assignment models are used for high-level assessment and coupled with further investigation using simulation and queuing models. The time-dimension, or *dynamic*, nature of traffic is not considered in such models; a steady-state behavior is assumed. Similarly, our model does not capture the propagation of traffic over time — for example, a bottleneck slowing traffic on incoming roads. The model does consider the high-level flight routing strategy in that it concurrently assigns to each OD-pair an acceptable path through the UTM network it selects.

A fundamental choice in routing strategy is whether a UAV flight request provides both the endpoints and a specific flight path between them, or only the endpoints because the UTM provider dictates the flight path. Based on our industry partner's insights into the future regulatory environment and safety requirements, our operating assumption is that the UTM provider dictates the flight path. As the UTM provider maps and maintains the data required to safely route UAVs on exact paths, they are well-positioned for this task. Looking beyond regulatory environments that will require adherence to provided flight paths, UAV operators may continue to rely on the UTM provider's expertise to recommend safe flight paths and to receive quick approvals. The primary requirement on the flight paths the UTM provider offers is to respect battery life restrictions which limit the total distance a UAV can safely travel. To summarize the design decision and primary justifications, our network design model uses system-optimal routing in accordance with future regulations and incorporates user constraints to restrict flight lengths because of battery limits and other technology limitations. That is to say, CSO traffic assignment will be used for determining routings in the network design model.

There are other secondary considerations that make CSO traffic assignment appropriate. The UTM provider may also want to offer short paths for economic reasons; if the flight paths differ too greatly from the greedy path or shortest route on the road, the company funding the UAV flight may otherwise find substitute ground transport. As UTM providers are incentivized to service as many flights as safe capacity allows, they must offer relatively short paths, and may also try to offer the same path for flights with the same endpoints to build customer trust. CSO traffic assignment also allows the UTM provider to obtain user buy-in by aiming for routings where UAVs travel on short paths with respect to travel times in the congested network. This can be achieved by setting appropriate path length limits in the user constraints. For a survey of work on bridging system optimal and user-equilibrium routing by techniques such as CSO traffic assignment, see Morandi (2021).

A second fundamental choice in routing strategy is how congestion, i.e., traffic buildup at a network component, is ameliorated. When a UAV flight request is made, the flight path chosen by the UTM provider must not conflict with airspace reservations made by past flights. The UAV will need to either wait to begin its journey, or, hover en route and deplete its battery while it waits for preceding traffic to clear. This dynamic problem manifests in a steady-state model whenever there is a road segment that appears on the assigned paths of many OD-pairs. One technique for limiting congestion is to use a capacitated model which enforces a strict limit on the number of OD-pairs assigned paths through a given arc. Such hard limits are not ideal for capturing congestion trade-offs in traffic models, however, because capacity is not so rigid. It is more realistic to consider the observed effect of congestion on the model, rather than strictly forbid it.

In road traffic networks, congestion effects are captured using *congestion*/latency/volume-delay functions which measure the effect of the volume of traffic flow on travel time. The amount of time to traverse a road increases gradually as the number of vehicles using it increases, up until the point at which the number of vehicles on the road exceeds its *saturation capacity* (hypothetical maximum capacity). In a dynamic setting, the number of vehicles that can traverse the road would then decrease. However, in traditional steady-state traffic modeling, the objective is to minimize the total travel time on the network, where travel time is obtained by evaluating the congestion function for the amount of demand assigned to the road (which may not represent a steady-state flow). Above the saturation capacity, the congestion function is designed to increase sharply, to indicate the disutility incurred by traffic flows outside of steady state behavior. However, such flows are not strictly forbidden. Congestion functions are also appropriate in our UAV context and we choose a model objective of minimizing total travel time, where travel time accounts for congestion effects. We choose a BPR congestion function; see the discussion in Section 4.

Following the model design decisions just discussed, Section 3.2 present a mathematical formulation of the model as a mixed integer convex optimization problem. Section 3.3 describes how to use piecewise linear approximation to obtain a mixed integer linear problem which approximates the nonlinear model.

3.2. Formulation

Underlying Network Data: The underlying city street network is represented by a graph denoted by $G = (N, A)$ where N denotes the set of nodes (corresponding to intersections) and A denotes the set of undirected edges (corresponding to sections of road). Let $c_{ij} \geq 0$ denote the cost of including edge $(i, j) \in A$ in the UTM network, indicating the cost of mapping the road segment. Let $B \geq 0$ denote the mapping budget which represents a hard limit on the total cost of the edges mapped. Once mapped, an edge may be traversed in either direction. For the purposes of representing routing decisions, we introduce notation for the graph $G' = (N, A')$, which is the symmetric directed graph obtained from G by replacing each edge in A with an arc in either direction.

Note that although the model supports generic costs, the expected method of data collection gives a good indication of what these costs might be. The length of an edge – measured by driving distance along the center of the road segment, and not Euclidean distance between its endpoints – should be proportional to the cost incurred to map the road. This is because mapping is performed by a vehicle driving along the road capturing precise navigational measurements.

UAV Flight Demand Data: Let K denote the set of OD-pairs, representing expected UAV flight requests. For each OD-pair $k \in K$, there is an affiliated origin $O_k \in N$, destination $D_k \in N$, and demand $d^k \in \mathbb{Z}_+$ which we interpret as the expected number of flight requests per hour from origin to destination. In this paper we assume that all UAV flights originate and terminate at an intersection, but the model can be extended to allow UAV flights to originate anywhere along a road segment.

Distance Data: Let $\ell_{ij} \geq 0$ denote the length of arc $(i, j) \in A'$. Let path length limit L_k for OD-pair $k \in K$ denote the maximum distance a UAV may fly to travel from O_k to D_k. The *path length limit* bound represents the restriction imposed by the battery life of an “average UAV” flying between the two locations. Note that this model can be extended to consider vehicle-specific arc distances and path length limits in a straightforward way by introducing arc lengths ℓ^k_{ij} for $(i, j) \in A$ and $k \in K$.

Travel Time Data: Let $t_{ij} : \mathbb{R}_+ \rightarrow \mathbb{R}_+$ be the congestion function which provides the time a UAV takes to traverse arc $(i, j) \in A'$ as a function of the number of flights using the arc. It is used to evaluate the total travel time on the network. We assume congestion is one-way; that is to say, counter-flow traffic does not affect the travel time on an arc. Note that as arcs become congested, UAVs may become “stuck” behind slower traffic and thus modifying the travel times to be vehicle-specific is more complicated than vehicle-specific path length limits.

Decision Variables: Let design variable $z_{ij} \in \{0, 1\}$ indicate if edge $(i, j) \in A$ is selected for the network. This model restricts all flight requests for an OD-pair to be assigned to the same path in the network. Let routing variable $x^k_{ij} \in \{0, 1\}$ indicate if arc $(i, j) \in A'$ is on the path from origin O_k to destination D_k for OD-pair $k \in K$. Let helper flow variable f_{ij} indicate the amount of traffic assigned to arc $(i, j) \in A'$.

Objective Function and Constraints: For node $i \in N$, let $N^+_i := \{j \in N : (i, j) \in A'\}$ denote the out-nodes from i and let $N^-_i = \{j \in N : (j, i) \in A'\}$ denote the in-nodes to i. Note that $(i, j) \in A' \iff (j, i) \in A'$, and therefore $N^+_i = N^-_i$. We use the distinct notation for consistency with standard directed models.

The mathematical formulation of our model, which will be denoted (UTM-TT), is as follows.

$$\underset{z,x,f}{\text{minimize}} \quad \sum_{(i,j)\in A'} [t_{ij}\left(f_{ij}\right)] \cdot f_{ij} \tag{1a}$$

$$\text{subject to} \quad \sum_{j\in N^+_i} x^k_{ij} - \sum_{j\in N^-_i} x^k_{ji} = \begin{cases} 1 & \text{if } i = O(k) \\ -1 & \text{if } i = D(k) \\ 0 & \text{otherwise} \end{cases} \qquad \forall k \in K,\ \forall i \in N \tag{1b}$$

$$f_{ij} = \sum_{k\in K} d^k x^k_{ij} \qquad \forall (i, j) \in A' \tag{1c}$$

$$x^k_{ij},\ x^k_{ji} \leq z_{ij} \qquad \forall (i, j) \in A,\ \forall k \in K \tag{1d}$$

$$\sum_{(i,j)\in A'} \ell_{ij} \cdot x^k_{ij} \leq L_k \qquad \forall k \in K \tag{1e}$$

$$\sum_{(i,j)\in A} c_{ij} \cdot z_{ij} \leq B \tag{1f}$$

$$x_{ij}^k \in \{0,1\} \qquad \forall (i,j) \in A',\ \forall k \in K \tag{1g}$$

$$f_{ij} \geq 0 \qquad \forall (i,j) \in A' \tag{1h}$$

$$z_{ij} \in \{0,1\} \qquad \forall (i,j) \in A \tag{1i}$$

Eq. (1a) is the objective function; it measures the total travel time experienced by vehicles when routing on the given paths in the selected network considerate of congestion effects. Constraints (1b) are the flow conservation constraints which choose a path in the network from each origin to its corresponding destination. As long as travel times are positive, these constraints will always result in a simple path being chosen because the objective is to minimize total travel time. Constraints (1c) calculate the flow on each arc. Constraints (1d) are the design forcing constraints; they ensure that every arc used in an OD-pair's path is open. Constraints (1e) ensure that the time it takes each vehicle to traverse its path satisfies the corresponding static travel time limit. Constraint (1f) ensures the selected network's cost does not exceed the budget. Constraints (1g), (1h), and (1i) are the variable bound constraints.

3.3. Piecewise linear approximation

Past work on CSO traffic assignment (see Section 2.2) introduced and solved models with similar features to (UTM-TT). The authors of these works used techniques such as the Franke-Wolfe algorithm, SOCP, and piecewise linear (PWL) approximation to obtain good solutions for the models. The nonlinearity of the (UTM-TT) model stems from the BPR function appearing in the objective. As observed by others, the BPR function is convex and therefore amenable to PWL approximation techniques. We note two particularly relevant examples of using PWL functions to handle congestion terms; for a general overview of using PWL functions to solve mixed integer non-linear programs see Geißler et al. (2011). Avella et al. (2019) use PWL congestion functions within a network design problem, citing the benefit of providing a formulation that is suited to commercial mixed integer linear programming solvers. Angelelli et al. (2021, 2020) present a piecewise linearization of the CSO model to capture flow-dependent travel times within the constraints. They analyze the quality of the approximation, and investigate various options for discretization technique. We choose to follow these authors in part to leverage the strength commercial solvers. Moreover, the sparser traffic flows and size of our instances allow us to choose a set of sample points for the approximation that is relatively dense in the domain of the problem.

To approximate the objective function with a PWL function, we introduce additional variables, constants, and parameters to the formulation. We follow the notation used in Angelelli et al. (2020). For each arc $(i,j) \in A'$:

- Total travel time variable σ_{ij} denotes the approximation of the total travel time experienced by traffic on the arc.
- Tunable parameter n_{ij} denotes the number of pieces used in the PWL approximation.
- Tunable parameter U_{ij} denotes the maximum amount of flow that may be assigned to the arc.
- Constant breakpoint b_{ij}^s for $s \in \{0,\dots,n_{ij}\}$ denotes the sth breakpoint used in the PWL approximation. Necessarily $b_{ij}^0 = 0$ and $b_{ij}^{n_{ij}} = U_{ij}$. Usually we will use evenly spaced breakpoints.
- Constant $\Delta_{ij}^s := b_{ij}^s - b_{ij}^{s-1}$ for $s \in \{1,\dots,n_{ij}\}$ gives the length of the sth interval.
- Arc-flow variable λ_{ij}^s for $s \in \{1,\dots,n_{ij}\}$ denotes the amount of flow assigned to the sth interval.
- Constant C_{ij}^s for $s \in \{0,\dots,n_{ij}\}$ gives the value of $x \to x \cdot t_{ij}(x)$ evaluated at b_{ij}^s.

The resulting model, which will be denoted (UTM-TT-PWL), is as follows.

$$\underset{z,x,f,\lambda,\sigma}{\text{minimize}} \quad \sum_{(i,j)\in A'} \sigma_{ij} \tag{2a}$$

$$\text{subject to} \quad \text{(1b), (1c), (1d), (1e), (1f), (1g), (1h), (1i)}$$

$$\sum_{s\in\{1,\dots,n_{ij}\}} \lambda_{ij}^s = f_{ij} \qquad \forall (i,j) \in A' \tag{2b}$$

$$\sigma_{ij} = \sum_{s\in\{1,\dots,n_{ij}\}} \left(\frac{C_{ij}^s - C_{ij}^{s-1}}{\Delta_{ij}^s} \right) \lambda_{ij}^s \qquad \forall (i,j) \in A' \tag{2c}$$

$$0 \leq \lambda_{ij}^s \leq \Delta_{ij}^s \qquad \forall\ (i,j) \in A',\ \forall s \in \{1,\dots,n_{ij}\} \tag{2d}$$

$$\sigma_{ij} \geq 0 \qquad \forall (i,j) \in A' \tag{2e}$$

Eq. (2a) is the objective function which sums the approximated total travel time experienced for each arc. Constraints (2b) decompose the arc-wise flow for the purposes of PWL approximation. Constraints (2c) compute the arc-wise contribution to the objective function using the PWL approximation. Constraints (2d) and (2e) are the additional variable bound constraints. As the congestion function is convex, the interval arc-flow variables will be assigned flow in the correct order; i.e. in any optimal solution $(z^*, x^*, f^*, \lambda^*, \sigma^*)$, if $(\lambda^*)_{ij}^s < \Delta_{ij}^s$ for $(i,j) \in A'$ and $s \in \{1,\dots,n_{ij}\}$, then $(\lambda^*)_{ij}^{s'} = 0$ for all $s' \in \{s+1,\dots,n_{ij}\}$.

4. Congestion functions for UAVs

Congestion functions (a.k.a. volume-day, travel time, or link capacity functions) have been used since the 1950s for road traffic modeling. Various mathematical forms for congestion functions were proposed early on. See Branston (1976) for a survey of early congestion function proposals and Gürel (2011, 2.2) for a summary of common mathematical requirements for congestion functions. The BPR function developed by the United States Bureau of Public Roads in the early 1960s (Bureau of Public Roads, 1964) continues to be used by practitioners and in Operations Research studies today. The parameterized form of the BPR function, which maps flow to travel time, is

$$t(f) = t_{FF}\left(1 + \alpha\left(\frac{f}{\kappa}\right)^{\beta}\right) \tag{3}$$

where the parameters are t_{FF} - the road's *free flow travel time*, κ - road "capacity", and $\alpha \geq 0,\ \beta \geq 1$ which influence the functional form. The free flow travel time is the amount of time it takes a vehicle to traverse the road when there is no congestion. As discussed in Section 2, the "capacity" gives an estimate of the critical point where the volume of traffic begins to significantly impede travel time. The values for α and β recommended by the BPR engineers (Bureau of Public Roads, 1964) were $\alpha := 0.15$ and $\beta := 4$, which were computed by a linear regression based on highway data. These parameter values remain a common choice in modeling today. Various researchers have worked to improve the parameter choices, often by calibrating the function to match observed traffic patterns in a specific region; see, for example, Casey et al. (2020), Mtoi and Moses (2014), Šarić and Lovrić (2021) and Wong and Wong (2016).

Using observed traffic patterns to calibrate a congestion function for UAV traffic above a road will not be possible until data is gathered from early UTM implementations. Accordingly, we take a more theoretical approach and use the common choice for parameters (α, β) to take on values $(0.15, 4)$. By increasingly penalizing flows over capacity κ, this form of congestion function captures the trade-off between relieving a highly congested arc and opening a shorter route for a UAV assigned a long path. Since our modeling objective is to provide a realistic high level assessment of network capability, we need to choose an appropriate capacity value κ to reflect the critical point for the trade-off. Unlike automobile traffic, safety regulations require UAVs to maintain strict separation standards, indicating the vertical and lateral distance required between UAVs in the air. In essence, each UAV is modeled as occupying an ellipsoid of 3D volume, and intersections of these ellipsoids are forbidden. Assuming traffic respects these standards, we will use simulation to compute capacity numbers which penalize the flow values that would result in UAVs waiting long times for clear airspace.

Before describing the simulation, we discuss the remaining BPR function parameter - free-flow travel time t_{FF}. For automobile traffic, this value is usually determined by the speed limit and road characteristics because most vehicles maintain a speed near the posted (or socially accepted) limit. Road vehicles may differ in acceleration/deceleration speeds and ability to maneuver to change lanes. Models considering these differences often modify the total flow on the road using the concept of an "equivalent passenger car unit" (ECU), which specifies a vehicle type's contribution to total flow relative to a passenger car. Less agile vehicles (ex. trucks) may be assigned an ECU above one and more agile vehicles (ex. motorcycles) may be assigned an ECU below one. Whether specific speed limits will apply within UTMs is, as of yet, undetermined. It is expected that UAVs operating within the managed airspace will be required to meet minimum operating standards. Above that standard, however, UAV speeds may vary greatly depending on the vehicle type, as well as the payload onboard. In this paper, we make the simplifying assumption that all UAVs travel at the same constant speed and therefore use free flow travel times directly proportional to road length.

As we are restricting to flight above roads in segmented layers of vertical airspace, there are only two UAV separation standards to consider: horizontal and vertical. Along a single segment of road, these separation standards have straightforward interpretations. The *horizontal distance* (H) indicates that a UAV entering a section of road at a fixed altitude must wait until the UAV "ahead of it" is H units of distance away before beginning its traversal. The UAV must continue to maintain a distance of H from the preceding vehicle while it traverses the road. The next UAV to enter must do the same. The *vertical distance* (V) indicates the distance between the vertically segmented layers of airspace. The number of layers of airspace available is given by the flight ceiling less the flight floor divided by V. Each of these layers must be assigned a fixed direction of traversal. A typical UAV flight ceiling is 400ft; specific values are dictated by the corresponding regulatory authority. For example, flight ceilings for UAVs in the United States can be obtained from the FAA Unmanned Aerial System Facility Map (UASFM) grid cells (FAA, 2023). A typical UAV flight floor is 100ft; specific values depend on the underlying ground terrain.

The horizontal distance, vertical distance, flight ceiling, flight floor, average UAV speed, and a *lane change penalty*, can be used to determine an approximate capacity for the BPR function. First, the flight ceiling, flight floor and vertical distance are used to determine the *number of lanes* available for traversal (in a fixed direction). Then, the horizontal distance and average UAV speed are used to determine the *clearance time*: how long it takes a UAV to clear the horizontal distance. We simulate f flights per hour arriving according to a Poisson process and then determine the average waiting plus traversal time of a flight. When there is only one lane, this is a simple queuing process. However, when there are multiple lanes, we assume that the vehicles arrive at a given lane (i.e. vertical layer) uniformly at random and will traverse at that layer unless its queue is too long. A *lane change penalty* is added to the travel time of a UAV when it changes lane to model the time it takes to switch altitude. These lane changes are unique to UAV flight; car traffic on a highway does not experience the same disutility in terms of speed and battery drain when switching lanes. For $f \in \{0, 1, \dots\}$ flights per hour, we can obtain a simulated travel time $t(f)$. We find the value of f for which the travel time $t(f)$ is 1.15 times the free flow time t_{FF} and use it as the BPR capacity for the given parameters. Pseudocode for one simulation is provided in Algorithm 1; the simulation can be repeated sufficiently many times to account for the random Poisson arrival times and random arrival lanes.

Sample simulation results are provided in Fig. 1 for clearance time $t_{\text{clear}} = 0.1$ min, lane change penalty $t_{\text{switch}} = 0.1$ min and free flow travel time $t_{\text{FF}} = 11$ min. Fig. 1 shows simulated travel times for one, two and three lanes. The resulting BPR arc capacities are 48, 104 and 163 for one, two, and three lanes, respectively. For the parameters illustrated here with a small lane change penalty,

Algorithm 1 Simulating Travel Time as a Function of Number of Flights

```
Inputs:  - $f$: the number of flights per hour
         - $n_{\text{lanes}}$: flight ceiling less flight floor, divided by $2V$
         - $t_{\text{clear}}$: time it takes a vehicle to traverse the horizontal separation standard
         - $t_{\text{switch}}$: lane change penalty
         - $t_{\text{FF}}$: free flow travel time
Outputs: $t(f)$: the simulated average travel time when $f$ flights arrive in an hour
1:  $t_{\text{entry}} \leftarrow [0.0 \text{ for } i \in \{1,\dots,f\}]$
2:  $t_{\text{current}} \leftarrow 0.0$
3:  $t_{\ell} \leftarrow [0.0 \text{ for } i \in \{1,\dots,n_{\text{lanes}}\}]$
4:  for $i \in \{1,\dots,f\}$ do
5:      $t_{\text{wait}}$ is a randomly drawn wait time for a Poisson process with $f$ arrivals per hour
6:      $t_{\text{current}} \leftarrow t_{\text{current}} + t_{\text{wait}}$
7:      $a$ is drawn uniformly at random from $\{1,\dots,n_{\text{lanes}}\}$
8:      if $(t_{\ell}[a] \leq t_{\text{current}})$ then
9:          $t_{\ell}[a] = t_{\text{current}} + t_{\text{clear}}$
10:         $t_{\text{entry}}[i] \leftarrow t_{\ell}[a] - t_{\text{current}}$
11:     else if $(t_{\ell}[a] \leq t_{\text{current}} + t_{\text{switch}})$ then
12:         $t_{\ell}[a] = t_{\ell}[a] + t_{\text{clear}}$
13:         $t_{\text{entry}}[i] \leftarrow t_{\ell}[a] - t_{\text{current}}$
14:     else
15:         $a^* = \text{argmin}\{t_{\ell}(j) \text{ for } j \in \{1,\dots,n_{\text{lanes}}\}\}$
16:         if $(a^* == a)$ or $(t_{\ell}[a^*] + t_{\text{switch}} > t_{\ell}[a]))$ then
17:             $t_{\ell}[a] = t_{\ell}[a] + t_{\text{clear}}$
18:             $t_{\text{entry}}[i] \leftarrow t_{\ell}[a] - t_{\text{current}}$
19:         else
20:             if $(t_{\ell}[a^*] < t_{\text{current}})$ then
21:                 $t_{\ell}[a^*] \leftarrow t_{\text{current}} + t_{\text{clear}} + t_{\text{switch}}$
22:             else
23:                 $t_{\ell}[a^*] \leftarrow t_{\ell}[a^*] + t_{\text{clear}} + t_{\text{switch}}$
24:             end if
25:             $t_{\text{entry}}[i] \leftarrow t_{\ell}[a^*] - t_{\text{current}}$
26:         end if
27:     end if
28: end for
29: $t(f) \leftarrow t_{\text{FF}} - t_{\text{clear}} + \text{avg}(t_{\text{entry}})$
```

the capacity of the arc increases superlinearly with the number of lanes. As the lane change penalty grows, the incoming UAV increasingly prefers to wait to traverse the altitude at which it arrives and thus lanes begin to operate as separate queues, each subject to the standard waiting time distributions. These simulations show projected travel times under a specific assumption on how traffic will be managed at the entry to a road segment. In particular, the framework assumes that the lane change penalty is a fixed cost and thus that the time it takes to change lanes is not proportional to the number of UAVs waiting to use the airspace above the road.

The simulation method may be extended to allow a queue-length dependent lane change penalty by modifying Algorithm 1. One option is to restrict the lane switches so they may only occur between the arrival lane and adjacent lanes. (i.e change line 15. to $a^* := \text{argmin}\{t_{\ell}(a+1), t_{\ell}(a-1)\}$ if $a \in \{2,\dots,n_{\text{lanes}}-1\}$; $a^* = 2$ if $a = 1$; $a^* = n_{\text{lanes}} - 1$ if $a = n_{\text{lanes}}$). This restriction prevents a UAV from altitude switching through a congested lane to arrive at an open lane. This change is equivalent to restricting lane switches so that they may only occur if all lanes between the starting lane and ending lane are available for the duration of the lane switch. Under this restriction, the time the nearest lane is free is always the first available time to switch lanes. Another approach would be to assess the lane change penalty at all lanes between the entering lane and the lane chosen (i.e. prior to line 25 set $t_{\ell}[a] = t_{\ell}[a] + t_{\text{switch}}$ for all lanes between lane a and a^*). With this change, the lane change penalty contributes to congestion along those lanes without necessarily ensuring there is true clearance. In Fig. 2, these two adjustments are compared against the results of Algorithm 1 for the case of four levels of altitude.

Even when adjusting the simulation for congestion caused by altitude switching, the queuing behavior and management of traffic at intersections is simplified in our model. This aligns with our choice of using a static traffic assignment model with congestion corresponding to oversubscription of the airspace above a road. The simulations determine only one BPR parameter – capacity – and thus the congestion experienced entering an arc is amortized into the travel time function for traversing the arc. Future operational and tactical models will need to consider traffic management in a more nuanced way. Two directions to explore are (1) using a queuing model and (2) extending models to include variables that represent the lane in which (i.e. altitude at which) a UAV will fly

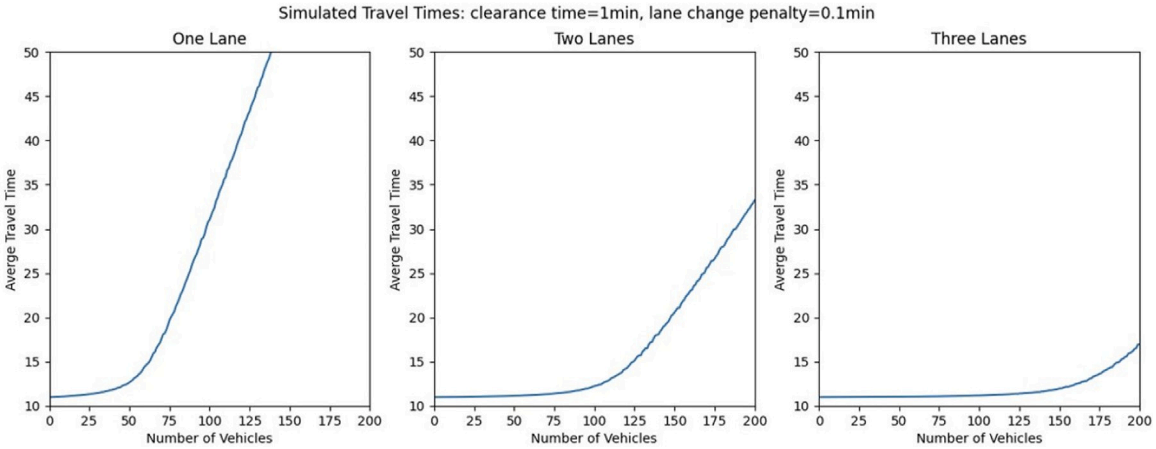

Fig. 1. Results of simulated travel times for parameters as indicated and a free flow travel time of 11 min.

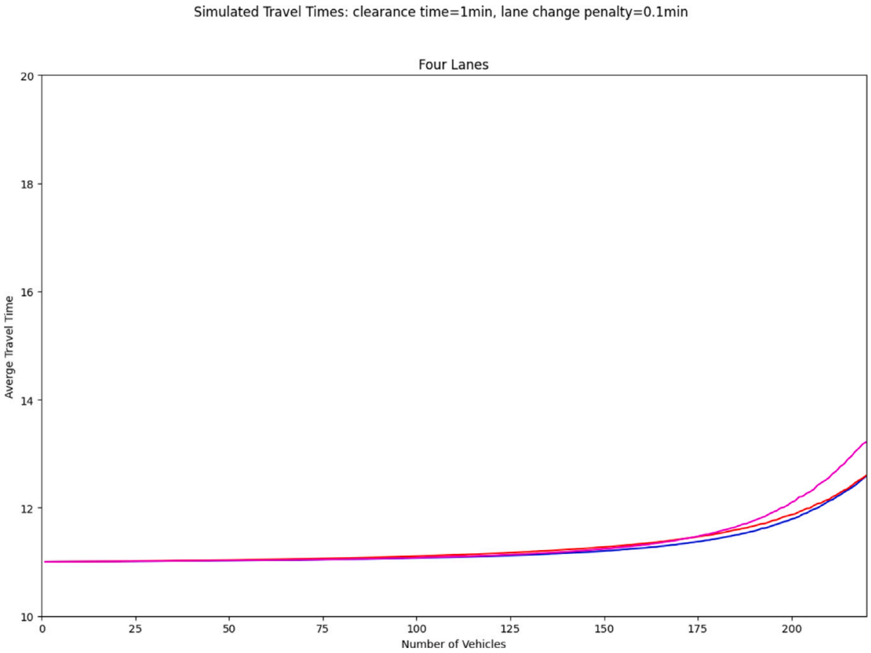

Fig. 2. Results of simulated travel times for parameters as indicated and a free flow travel time of 11 min. The bottom blue and red lines are the results of Algorithm 1 and the first modification that allows lane changes only to adjacent lanes. The top magenta line is the result of assessing a lane change penalty to all lanes between the arrival lane and the lane of traversal.

for each segment of its route. We expect the models to be informed by the development of the rules of the roads and intersection policies in early UTMs.

In the next section, numerical experiments are performed in which we fix arc capacities consistent with the network instances being solved; the simulation here provides a simple link back to the physical constraints imposed by separation standards, available lanes, and lane change costs. As different parameter sets may lead to the same capacity, the link is not one to one. These links are important as UTM providers determine the ‘‘rules of the road’’ both in early UTM development and at maturity. Although this section introduced separation standards as rules regarding how far apart UAVs may fly, in practice the separation standards may reflect a regulatory requirement for UAVs to respect the airspace reservations awarded to approved flight requests. In this case, the clearance time would be dictated by the length of the timeslots allocated by the airspace reservation system. UTM providers determining the timeslot sizes will need to evaluate their impact on the UTM network’s ability to serve demand.

5. Numerical experiments

This section first discusses a process for arriving at practical test instances. By following the process, we generate test instances for Chicago City Center. We first experiment with a specific instance class, using the network design model to analyze design tradeoffs.

Then, we investigate whether these managerial insights hold or differ for other instance classes. Lastly, we discuss model runtimes and accuracy. All models described in this section were implemented using Gurobi (v10.0.1) via the Python API and instances were run on an Apple M1 Pro Version macOS Ventura 13.2 with 16 GB of memory. Each optimization problem solved is given a runtime limit of three hours.

5.1. Generating and sizing instances

First, we discuss our approach to generating suitable test instances. We develop a process by which we generate a UAV suitable network from the network of roads. Then, we suggest a random demand generation procedure to select origin–destinations and set traffic volumes. We suggest three demand patterns to reflect different UAV uses. Finally, we describe how we determine arc congestion functions and budget by solving minimum distance and minimum cost models.

5.1.1. Capturing the road network

The first step in generating an instance is obtaining an appropriate base road network. Although we tested our models with the standard transportation research networks from Transportation Networks for Research Core Team (2022), we found these networks were not ideal for extending to UAV flight. The geographic regions covered were often too large and the aggregate roads too sparse because they represented major arteries. In the early stages of UTM implementation, a UTM provider may make a strategic decision regarding which area of a city to service first because opening up airspace in an entire metropolitan area could be both costly and risky. We expect the service area covered to align with delivery UAV flight capabilities and accordingly suggest a test instance whose geographic size is around 10–15 km by 10–15 km. In an area of this size, a typical delivery UAV can fly point-to-point from most origins to most destinations and back without recharging. Operations Research work relating to UAV flight assumes similar ranges. For example, the authors of Hong et al. (2018) study locating UAV recharging stations and consider two possible ranges of five miles and ten miles. However, they remark that determining a realistic flight range for delivery UAVs was a challenge.

Having narrowed in on a geographic size for our test instances, we turn to the problem of identifying candidate roads. OSMNX (Boeing, 2017) is a Python library that can download OpenStreetMap (Open Street Map Contributors, 2022c) data in graph format. We use it to obtain the underlying road network and filter based on the "highway" tag to restrict the set of roads considered. For example, the base network for the Chicago instance was obtained by restricting candidate roads to those with a "highway" tag of "motorway", "motorway_link", "primary", or "secondary". According to the OSM guidelines for tagging highways in Illinois (Open Street Map Contributors, 2022a) (and the US more generally (Open Street Map Contributors, 2022b)), these candidate roads include interstate highways and expressways with similar characteristics, main surface streets in urban areas, and the roads in urban areas that extend long distances through multiple neighborhoods and have high volumes. These roads are the best candidates for UAV traffic because flying down major streets provides access to the busiest areas of the city and is preferential to flying on residential roads from a privacy and societal acceptance perspective. Note that the choice of "highway" tag may depend on the features of the area being mapped; in the suburbs, it may be appropriate to use additional tags. Besides providing the topology - i.e. the relationships between nodes (intersections) and arcs (roads) - the OSM data also provides street geometries by representing roads with a sequence of line segments that capture any turns or bends. Therefore, we obtain arcs lengths that reflect the true driving distance along the road, and thus the horizontal distance a UAV will travel if it flies above the road.

The raw driving networks provided by OSMNX are not suited for direct input into the UTM network design model; some modifications are prudent. The driving network represents connections according to feasible (possibly one-way) driving paths which may represent specific on-the-ground or car-minded infrastructure. Some of these features are not used by UAVs in the same way. If a road passes over another (ex. a highway overpass with no on-ramp), then the road network will not contain a node representing that intersection. However, a UAV flying above the roads is not restricted in the same way. Major road intersections are often managed by segmenting traffic using turning lanes or curving ramps. However, the intersection may be managed for UAV traffic by creating vertical separation, rather than horizontal because the lane created to segment car traffic may not be appropriate for UAVs, which have greater separation requirements. The capacity provided by major expressways may correspond to separated roads — for example, when there are express lanes and local/collector lanes for a highway. These roads are represented by separate, but similar, arcs in the raw driving network. However, a UTM network may not use both these roads because simultaneous flights would violate horizontal separation standards. In light of these differences, we perform a sequence of simplifying operations to adjust the raw road network. An example of these operations is given in Fig. 3 where three main modifications were made. The highway overpasses for all roads running north to south are modified into intersections. The two-lane road represented by two parallel paths running north to south is replaced with one path. The highway running from east to west is replaced with one path. An example of simplifying an intersection is provided in Fig. 4. The simplifications made preserve distances as much as possible. A complete simplification is performed to arrive at the base network in Section 5.2.

In our implementation, the road simplification was partially automated but some steps required manual inspection. To prevent errors, code was written that could reproduce the manual sequence of steps. Helper functions were used to perform required computations such as calculating the intersection of two roads, keeping one of two parallel roads, and suppressing degree two vertices. The input graph was acquired using OSMNX and therefore the node coordinates and arc geometries were readily available. This allowed many computations to be performed in a straightforward way by leveraging Python libraries (ex. networkx, shapely). The procedure used is described below.

1. The input is a graph $G = (N, A)$ where each node $i \in N$ is described by its coordinates and each arc $(i, j) \in A$ is described by its two endpoints and its *geometry*. The *geometry* of an arc indicates how it is embedded on a flat map, and is (effectively) defined by a sequence of line segments. These steps modify G in place and return the resulting simplified graph as output.
2. After each change to G made in steps 3–5, automatically suppress every degree two vertex in the graph. Node $i \in N$ of degree two must have two incident edges — say (i, j) and (i, k). To *suppress* i means to:

   (a) remove i from N;
   (b) remove (i, j) and (i, k) from A; and
   (c) add (j, k) to A. The geometry of (j, k) is obtained by sequencing the geometries of (i, j) and (i, k).

3. The first automated simplification step is to add "missing intersections" to G corresponding to places where road traffic is vertically separated by ground infrastructure (ex. bridges, tunnels). For every pair of arcs $(i_1, j_1), (i_2, j_2) \in A$, check if their geometries intersect. If yes, then add a node at that intersection. That is, add a new node k to nodeset N, remove arcs (i_1, j_1) and (i_2, j_2) from A, and add arcs $(i_1, k), (k, j_1), (i_2, k), (k, j_2)$ with appropriate geometries.
4. The next automated simplification step is to clean *similar* roads; these are the roads that run parallel and close to each other. After the below steps, some manual adjustments are required to fix any changes in connectivity or distance distortion.

   (a) Identify pairs of arcs whose endpoints are close (less than 20 meters apart) and whose arc lengths are similar (less than 20 meters difference). Let P denote the pairs of arcs identified; add arc pair $[(i_1, j_1), (i_2, j_2)]$ to set P if

   - the Euclidean distance between nodes i_1 and i_2 is at most 20 meters,
   - the Euclidean distance between nodes j_1 and j_2 is at most 20 meters, and
   - the lengths of the geometries of arcs (i_1, j_1) and (i_2, j_2) differ by at most 20 m.

   (b) For each pair of arcs $p \in P$, either label p to correspond to a road running *north-south* or a road running *east–west*. Let $\ell(p)$ denote the label given to pair $p \in P$. A road is considered to *run east–west* if the x-coordinates of its endpoints differ more than the y-coordinates of its endpoints. Otherwise the road is considered to run *north-south*. The two arcs in the pair may not agree on a label — in this case, flag the pair of arcs for manual inspection.
   (c) For each pair of arcs, $p = [(i_1, j_1), (i_2, j_2)] \in P$:

   - If $\ell(p)$ = *east–west*: then remove (i_1, j_1) from A if it is farther south; otherwise remove (i_2, j_2) from A. The coordinates of the endpoints are used to determine which arc is further south. If i_1 is further south than i_2 but j_2 is further south than j_1, choose arbitrarily.
   - If $\ell(p)$ = *north-south* then: remove (i_1, j_1) from A if it is further east; otherwise remove (i_2, j_2) from A.

5. The remainder of the adjustments are identified by manual inspection. As such, tolerances indicated in the below steps are approximate.

   (a) Remove spurious nodes and arcs alongside highways and major intersections and simplify the adjacencies. For an example see Fig. 4.
   (b) Remove degree one nodes that are *close* to another node of the graph. For this step, node $i_1 \in N$ is considered *close* to node $i_2 \in N$ is the Euclidean distance between i_1 and i_2 is less than 50 m.
   (c) Merge two nodes that are *close* to each other by placing a new node at their midpoint and adjusting distances to neighbors. For this step, two nodes $i_1, i_2 \in N$ are considered *close* to each other if the Euclidean distance between i_1 and i_2 is less than 20 m.

This process may be automated in the future by (1) establishing deterministic procedures that reflect the manual inspections and (2) automatically verifying the resulting graph does not distort the distances in the original graph significantly. Designing a complete network reduction algorithm for this context is an avenue for future research.

5.1.2. Demand patterns

The second step in generating an instance is determining the origins and destination of flights, together with the affiliated flight volumes and path-length limits. Publicly available information about current UAV flight patterns is scarce and we therefore propose to randomly generate instances according to probable demand patterns, where a *demand pattern* specifies how flights requests are spatially distributed. We consider three different demand patterns. In the *space-uniform* (SU) demand pattern, pairs of coordinates are drawn at random within the bounding box of the area. The origin is the nearest node (geographically) to the first coordinate pair while the destination is the nearest node to the second coordinate pair. This demand pattern provides the largest geographic coverage, and may arise when the area considered is homogeneous, e.g. in terms of population features. In the *random-node-pair* (RNP) demand pattern, the origin and destination are sampled at random from all nodes. This method has the benefit of concentrating demand in areas with many major streets such as the downtown core, which is natural. However, the method is sensitive to how OSMNX chooses to place nodes, as well as the post-processing performed. In the *warehouse-customer* (WC) demand pattern, one node is presumed to be the location of a warehouse within the city. The origin of each OD-pair is the

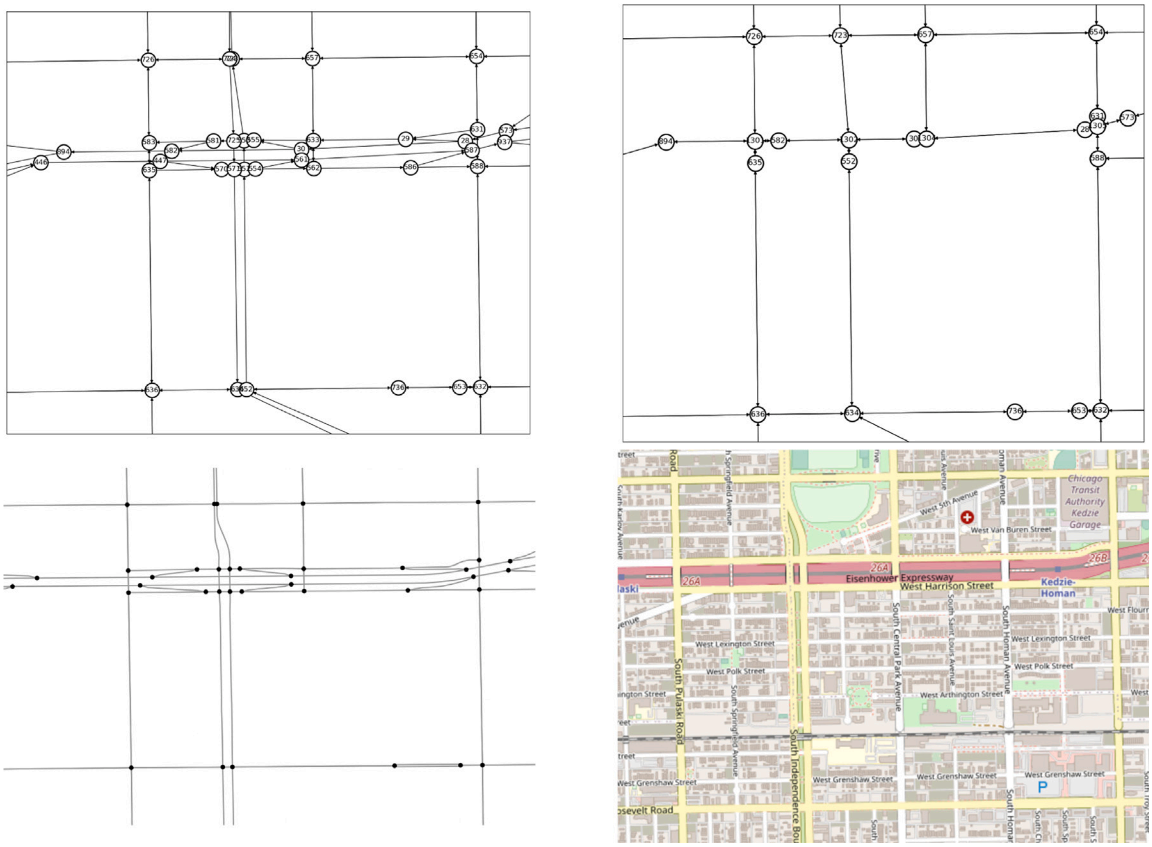

Fig. 3. A section of the road network before (top left) and after (top right) simplification, as well as the geometry of the underlying roads (bottom left), and map of corresponding city streets (bottom right).

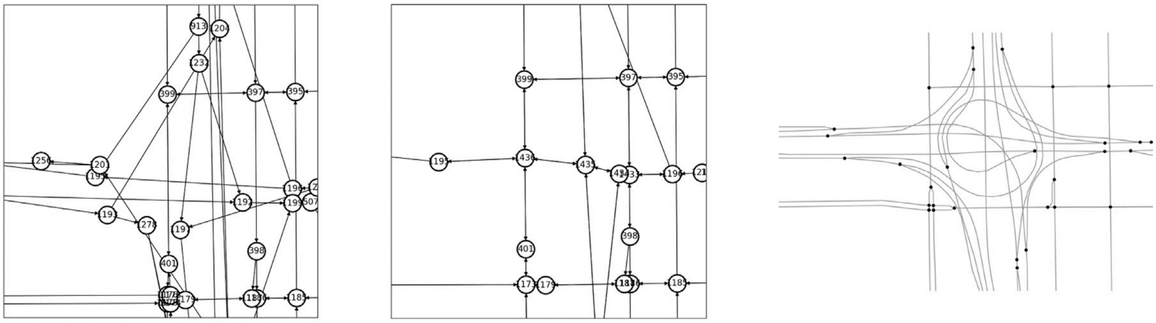

Fig. 4. An intersection before (left) and after (center) simplification, as well as the geometry of the underlying roads (right).

warehouse and the destination is a randomly selected node representing a location where a customer may request a delivery. This demand pattern is natural if a specific company intends to offer UAV delivery service, or if there is a fixed UAV launch point within the city. The demand patterns are illustrated in Fig. 5.

Although OD-pairs will be drawn at random to generate an instance, the model itself is solved for a fixed set of OD-pairs. The underlying assumption is that the random pairs are adequate to achieve good coverage (which may be analyzed), or represent a pattern that could arise when a specific city knows its needs. In Section 6, we discuss research directions related to stochastic demand scenarios or optimizing for the amount of demand served. Now, if we are generating an instance with k OD-pairs, then the origins and destinations of the OD-list are drawn without replacement from all node pairs that fit the demand pattern. The demand for each OD-pair is drawn uniformly at random from $\{1, 2, \dots, f_{max}\}$, where f_{max} is some appropriately determined upper bound. The expected total number of flight requests is $\frac{k\cdot(f_{max}+1)}{2}$.

For each OD-pair, we require a corresponding path-length limit. As in Jahn et al. (2005), we consider path-length deviations which are used to compute sets of path-length limits. An instance is said to be solved for *path-length deviation* $\delta\%$ if the path length

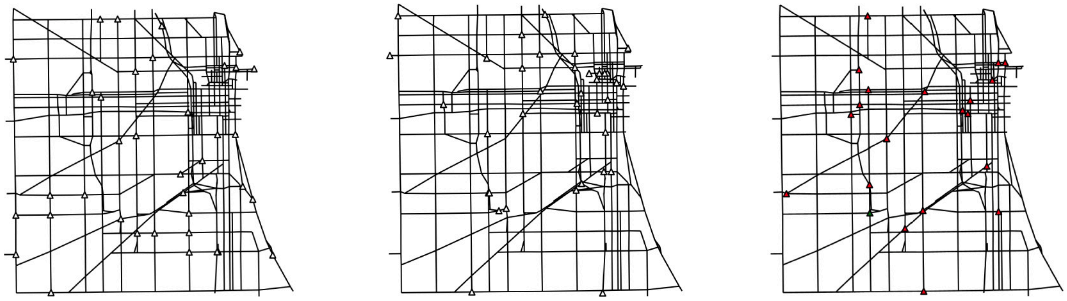

Fig. 5. The SU (left) and RNP (center) demand pattern with both origins and destinations indicated by triangles. The WC (right) demand pattern with the warehouse indicated by a green triangle and the delivery points indicated by red triangles.

limit for each OD-pair is equal to $(100+\delta)\%$ times the length of the shortest path on the road network between origin and destination. Since we chose an area size in which UAVs can fly between any two points while respecting battery life, path length deviations up to 25% yield limits within a reasonable range. For close together OD-pairs, the limit may seem overly restrictive. However, unduly long routes may not be acceptable to UAV pilots because long flight times introduce unnecessary air risk. Note that we measure path-length deviation with respect to the shortest path on the existing road network, which is almost always longer than the Euclidean shortest path, and thus introduces further deviation from the greedy choice.

5.1.3. Remaining data and summary

Arc congestion functions and a budget are the last two pieces of data required to complete an instance. As per Section 4, we will use the BPR function with standard α and β. The road length provided by OSMNX can be used as both the mapping cost for a road segment, as well as the free flow travel time following the constant speed assumption. Hence, it remains to determine an appropriate budget and arc capacities. As discussed in Section 4, known separation standards, UAV speeds and flight ceilings may be used to determine arc capacities. These standards are still in development, and are not yet publicly available. However, it should be readily apparent that the budget, path-length limits, demand volume, and arc capacities must be consistent and aligned to obtain practically interesting instances. For example, if every network that provides each OD-pair a path within its path length limit (1e) is above the budget (1f), then the model is infeasible. Similarly, the total demand, budget, and arc capacities must be aligned so that there exists a network within budget for which most arcs do not have flow significantly above capacity.

Pre-analysis can ensure that the inputs to the (UTM-TT) model are consistent and aligned. The idea is to set parameters in succession to produce reasonable instances. We choose to fix demand first, and compute appropriate arc capacities last. Similar techniques may back out consistent inputs by fixing budget or arc capacity parameters first. Now, given a base road network and OD-list together with a path length deviation, we can compute a minimum cost network satisfying the corresponding path length limits by solving

$$\underset{z,\,x}{\text{minimize}} \quad \sum_{(i,j)\in A} c_{ij} z_{ij} \tag{4a}$$

$$\text{subject to} \quad \text{(1b), (1d), (1e), (1g), (1i).}$$

Denote this model by (UTM-Cost). Its optimal value provides the cost of the cheapest network which connects every OD-pair by a short enough path. The corresponding (UTM-TT) model is feasible only if the budget is at least the optimal value of (UTM-Cost). After determining a reasonable budget, we solve

$$\underset{z,\,x}{\text{minimize}} \quad \sum_{(i,j)\in A} \ell_{ij} f_{ij} \tag{5a}$$

$$\text{subject to} \quad \text{(1b), (1c), (1d), (1e), (1f), (1g), (1h), (1i).}$$

Denote this model by (UTM-Dist). The model minimizes the total distance traveled when flights are routed with no regard for congestion effects. By analyzing the arc usage in the model results, we can determine the capacity level at which arcs start to be oversubscribed and use it to determine interesting arc capacities. Note that (UTM-Cost) and (UTM-Dist) provide insights into candidate networks and their capability. For example, when the total flight volume is low and therefore congestion effects are negligible, the (UTM-Dist) model results mimic the (UTM-TT) model. Varying the flight volume, the (UTM-Dist) model gives an indication of the throughput level at which bottlenecks occur and congestion effects should be modeled.

To summarize, instance generation requires the following inputs: **city**, **demand pattern**, number of OD pairs **k**, max number of flights between an OD-pair $\mathbf{f_{max}}$, and path length deviation δ. The instance generation procedure is as follows.

1. Obtain base road network for **city** from OSMNX.
2. Generate **k** OD-pairs according to the **demand pattern**. Assign each OD-pair a demand uniformly at random from $\{1, \ldots, \mathbf{f_{max}}\}$.
3. Calculate path length limits corresponding to path length deviation δ.

Table 1
Instance characteristics for fifty OD-pair RNP instances. The average shortest path distance is obtained by taking the average over all flights.

Instance #	Number of flights	Average shortest path distance (km)
1	280	6.4
2	288	5.7
3	275	6.3
4	275	5.3
5	322	6.4

Table 2
Summary statistics reported for UTM network design models.

Heading	Description	Applies to
avg	Average taken over five instances in the instance class	All
std	Standard deviation taken over five instances in the instance class	All
dmd	Demand pattern considered (SU, RNP, or WC)	All
pld	Path length deviation considered	All
num ods	Number of OD-pairs in the instance class	All
bd	Indicates the budget deviation from the minimum budget from (UTM-Cost)	(UTM-Dist), (UTM-TT-PWL)
cap	Arc capacity considered	(UTM-TT-PWL)
cost	Total length of roads in the chosen network in kilometers	All
num arcs	Number of arcs in the chosen network	All
avg pld	Weighted average of the realized path length deviation over all OD-pairs	(UTM-Dist), (UTM-TT-PWL)
num on sp	Number of OD-pairs provided a path equal to the length of their shortest path	(UTM-Dist), (UTM-TT-PWL)
max flow	Amount of demand on the arc which carries the most demand	(UTM-Dist), (UTM-TT-PWL)
travel dist	The total distance traveled by all flights (i.e. the value of (5a)) in kilometers	(UTM-Dist), (UTM-TT-PWL)
travel time	The total travel time of all flights (i.e. the value of (1a))	(UTM-TT-PWL)

4. Use the (UTM-Cost) model to determine an appropriate budget range.
5. Use the (UTM-Dist) model to determine an appropriate arc capacity range.

Instances with the same city, demand pattern and number of OD pairs will be said to belong to the same *instance class*.

5.2. Case study of Chicago city center RNP fifty OD-pair instance class

We analyze UTM networks for an area of Chicago consisting of the downtown core and adjacent neighborhoods and focus first on instances with fifty OD-pairs drawn according to the RNP demand pattern. The characteristics of each instance are summarized in Table 1. We measure the impact of budget, congestion, and path length limits on the network design across the five instances drawn from the instance class. Table 2 describes the abbreviations and conventions we adopt to report model results. For runtime analysis, see Appendix A.4. Here we report only timeouts and PWL approximation parameters as applicable because they inform the validity of the analysis.

The area of Chicago analyzed is bounded by: the I50 on the west, the Lake Michigan harbour on the east, the I64 on the north, and 47th Street on the south. The precise bounding box has corners with (longitude, latitude) given by $(-87.581972, 41.806857)$ and $(-87.750543, 41.914249)$. The resulting area is 14 km by 12 km. The region considered and initial set of roads downloaded are illustrated in Fig. 6. The simplifications described in Section 5.1 are performed on the base network. Other straightforward simplifications are made to reduce the problem size: in particular, some nearby parallel roads are removed from the downtown core to eliminate similar solutions. After all simplification, the raw driving network with 1283 nodes and 2735 (directed) arcs is reduced to a graph with 500 nodes and 863 edges. The final network is illustrated in Fig. 7.

To assess appropriate budget ranges, (UTM-Cost) is solved for all five instances and aggregated results are reported in Table 3. There are a total of 424.8 km of roads that may be mapped. For path length deviations of 10% and 20%, the minimum budget network for all instances is around 100 km. The cheapest network that routes every OD-pair on a shortest path between its origin and destination is calculated using path length deviation 0%. This optimal value gives the maximum sensible budget for the (UTM-Dist) model, which can do no better than shortest path routing. The (UTM-TT) model budget could reasonably be set higher as mapping more roads may alleviate congestion. For the fifty OD-pair RNP instance class, however, we found the rate of objective value improvement diminished as budget increased and therefore present results to a maximum budget of 150% the minimum cost network. In particular, we first vary the budget in increments of about 10 km by considering budgets equal to $110\%, 120\%, \dots, 150\%$ of the optimal value of the corresponding (UTM-Cost) model. These instances will be said to have *budget deviation* $10\%, 20\%, \dots, 50\%$ respectively.

To assess appropriate arc capacities, (UTM-Dist) is solved for all five instances and aggregated results are reported in Table 4. As the budget increases, the number of arcs included in the network and the number of OD-pairs routed on their shortest path increases while the average path length deviation decreases. These relationships do not just hold on aggregate; each can be observed within a given instance. Such a simple monotonic relationship does not hold for the maximum arc flow; one possible explanation is that as more arcs are mapped, more OD-pairs are able to connect to a shared short route through an area. Based on the (UTM-Dist) results,

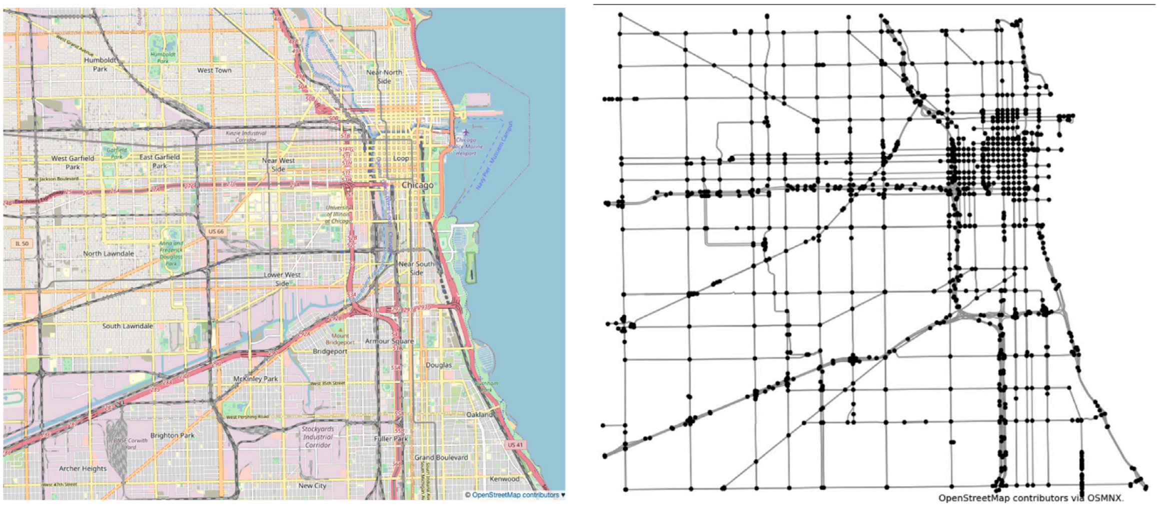

Fig. 6. A map of the area of Chicago considered on the left, and the corresponding major roads obtained from OSMNX on the right. The data was obtained on 7 November 2022.

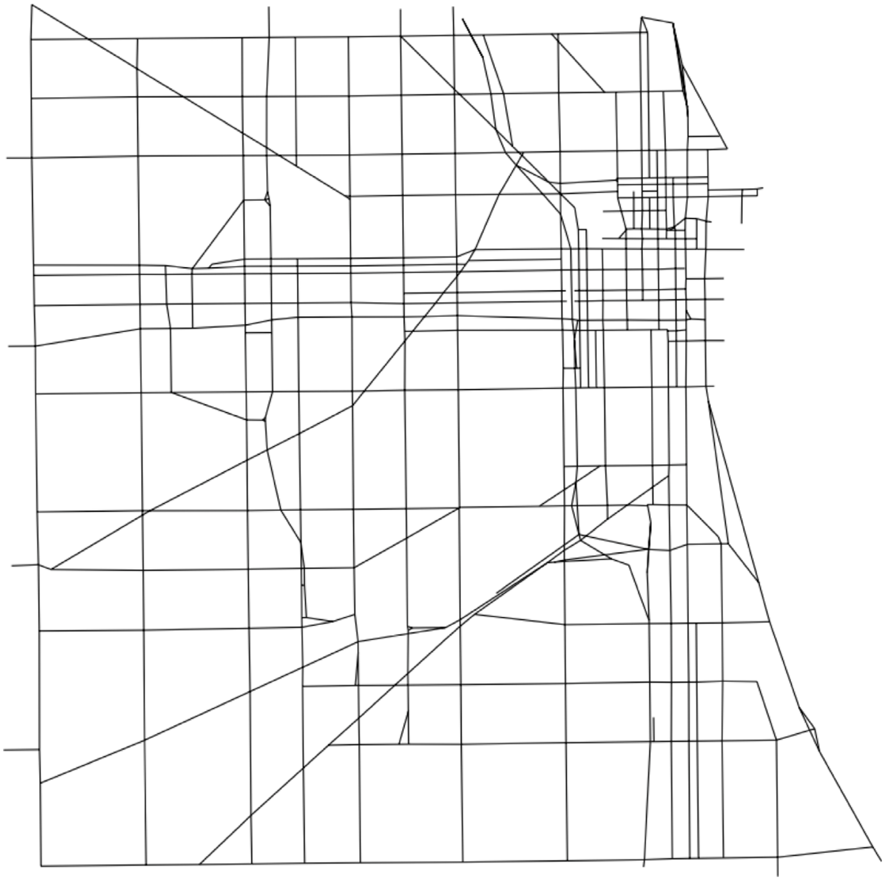

Fig. 7. Final Chicago Road Network after UAV-specific adjustments and simplification.

we choose to analyze the (UTM-TT) model when the BPR arc capacities are equal to 30. At this level, we expect incorporating congestion should force the model away from overloading any arc without being too restrictive. Note that using Section 4, we can back out possible sets of corresponding separation standards and lane numbers.

Table 3
Minimum cost networks satisfying path length constraints for fifty OD-pair RNP instances.

dmd	pld	cost		num arcs	
		avg	std	avg	std
rnp	0	170.85	11.73	394.60	12.58
rnp	10	106.30	3.25	252.80	12.40
rnp	20	92.26	3.79	227.20	6.53

Table 4
Minimum travel distance with path length and budget constraints for fifty OD-pair RNP instances.

pld	bd	cost		num arcs		travel dist		avg pld		num on sp		max flow	
		avg	std	avg	std	avg	std	avg	std	avg	std	avg	std
10	10	116.89	3.55	273.80	10.38	1774.52	232.86	1.65	0.25	14.40	1.82	36.80	1.79
10	20	127.50	3.91	296.20	15.02	1755.44	229.09	0.71	0.13	21.80	3.42	33.80	2.17
10	30	138.09	4.25	318.60	12.84	1748.12	228.79	0.33	0.14	25.40	4.93	35.00	4.00
10	40	148.69	4.60	342.60	17.70	1744.13	228.04	0.14	0.09	30.80	6.46	33.20	5.72
10	50	158.76	5.17	364.40	18.01	1742.32	227.63	0.04	0.04	39.00	9.92	31.80	4.32
20	10	101.47	4.17	246.00	10.00	1814.97	233.46	3.66	0.54	9.00	2.12	44.40	9.40
20	20	110.67	4.53	265.00	12.59	1782.24	230.45	2.05	0.37	12.60	2.61	34.80	4.02
20	30	119.90	4.90	280.60	13.22	1762.80	227.62	1.15	0.23	19.00	2.83	33.60	3.71
20	40	129.08	5.27	301.40	10.11	1753.11	226.833	1.64	0.22	21.80	3.56	33.80	2.17
20	50	138.30	5.64	319.60	11.97	1747.85	227.15	0.33	0.14	25.40	5.03	35.20	4.09

Table 5
Minimum travel time with path length and budget constraints for fifty OD-pair RNP instances.

pld	bd	cost		num arcs		travel dist		avg pld		num on sp		max flow		travel time	
		avg	std	avg	std	avg	std	avg	std	avg	std	avg	std	avg	std
10	10	116.91	3.57	276.80	8.93	1785.30	233.29	2.03	0.32	13.20	3.11	29.00	3.39	1809.74	246.34
10	20	127.52	3.91	298.20	9.71	1763.44	230.34	1.14	0.21	17.00	4.64	27.00	2.74	1781.54	239.10
10	30	138.15	4.19	321.60	15.50	1754.59	230.16	0.64	0.22	23.00	5.39	26.00	1.22	1768.61	236.61
10	40	148.78	4.54	345.40	17.66	1749.16	228.92	0.43	0.08	26.80	5.89	26.00	2.00	1761.74	235.46
10	50	159.37	4.94	371.00	21.05	1746.23	228.63	0.26	0.08	29.80	5.76	26.00	1.41	1757.83	234.44
20	10	101.47	4.17	247.00	11.73	1830.34	235.40	4.39	1.03	8.80	2.68	31.80	3.27	1871.47	250.64
20	20	110.68	4.55	268.40	11.59	1790.33	230.98	2.37	0.42	11.20	2.28	30.00	3.32	1818.54	242.89
20	30	119.92	4.93	285.80	9.81	1772.20	228.90	1.56	0.33	14.80	4.15	27.40	4.51	1792.24	237.77
20	40	129.13	5.32	303.40	10.78	1760.90	228.11	1.00	0.19	19.20	3.49	26.40	1.67	1776.38	235.07
20	50	138.38	5.69	323.00	10.98	1754.12	228.23	0.62	0.13	22.00	5.61	25.60	1.34	1767.41	234.35

First, we evaluate the impact of budget on networks designed by the (UTM-TT) model. To do so, (UTM-TT-PWL) is solved with a maximum arc capacity of 50 and fifty pieces in the PWL congestion functions. Since the flow on each arc is restricted to be integer and provided the true optimal solution does not route over 50 units of demand on any one arc, (UTM-TT-PWL) models (UTM-TT) exactly. The underlying optimization problems were solved to optimality, except for two instances with budget deviation 10% and path length deviation 20% with mipgaps of 1.3% and 0.8% at three hours and one instance with budget deviation 10% and path length deviation 20% with a mipgap of 0.1% at three hours. The results are summarized in Table 5. Observe that the improvement in total travel time as a function of budget is more pronounced for low budget deviations. That is to say, the value obtained from slightly increasing the mapping budget is not linear and the largest impact occurs when raising the budget slightly above the cost of the cheapest feasible network. Another interesting take-away from these instances is that the travel time minimizing model is able to find a network that has small total travel distance and simultaneously avoids arcs that carry large amounts of flow. Comparing the average path length deviation and number on shortest path between (UTM-Dist) in Table 4 and (UTM-TT) in Table 5 for the same budget and path length deviation, we see that the total travel time is improved at the expense of routing fewer OD-pairs on their shortest paths. This should be expected as shortest paths for different OD-pairs may share a short route through a specific area, and congestion may be relieved by choosing an alternative path for one such OD-pair. In all these instances, at least 44 of the 50 OD-pairs are assigned a path with a travel time within 1.2 times the travel time of the network path between the origin and destination with the fastest travel time.

Next, we evaluate the impact of path length deviation on the network designed by the (UTM-TT) model. We choose our budget based on the minimum cost network for path length deviation of 10%; in particular, we use a budget of 110% of the corresponding minimum cost network. We continue to use an arc capacity of 30, max arc capacity of 50 and fifty pieces in the PWL congestion functions. For this budget with path length deviation 5%, the resulting model is infeasible for all five instances. The remainder of the problems are solved to optimality and the results are summarized in Table 6. Observe that at this budget level, the networks designed share very similar summary statistics; this observation hold true even analyzing within an instance. It is interesting to note that the average path length deviation decreases as the allowable path length deviation increases. The result may seem counter-intuitive when phrased this way. However it is natural when considering the increasingly less restrictive optimization problems being solved;

Table 6
Minimum travel time fixing budget and varying path length limit for fifty OD-pair RNP instances.

pld	cost		num arcs		travel dist		avg pld		num on sp		max flow		travel time	
	avg	std	avg	std	avg	std	avg	std	avg	std	avg	std	avg	std
10	116.91	3.57	276.80	8.93	1785.30	233.29	2.03	0.32	13.20	3.11	29.00	3.39	1809.74	246.34
15	116.92	3.59	277.00	11.20	1780.25	235.09	1.86	0.32	13.20	3.56	28.20	3.35	1801.97	246.70
20	116.91	3.57	280.60	12.42	1779.58	236.84	1.80	0.45	13.20	3.96	28.20	3.35	1801.12	246.78
25	116.91	3.57	279.80	15.16	1778.95	236.76	1.83	0.45	13.40	3.21	29.20	3.96	1800.78	247.03

Table 7
Minimum travel time fixing budget and varying arc capacity for fifty OD-pair RNP instances.

cap	cost		num arcs		travel dist		avg pld		num on sp		max flow		travel time	
	avg	std	avg	std	avg	std	avg	std	avg	std	avg	std	avg	std
25	116.92	3.57	275.00	11.05	1790.96	237.67	2.36	0.40	12.00	2.12	28.20	3.35	1833.47	258.17
30	116.91	3.57	276.80	8.93	1785.30	233.29	2.03	0.32	13.20	3.11	29.00	3.39	1809.74	246.34
35	116.91	3.57	274.80	7.76	1780.36	232.27	1.86	0.22	13.80	2.95	30.60	3.78	1797.32	240.01
40	116.91	3.59	275.20	11.19	1776.70	232.63	1.75	0.16	14.40	2.51	32.00	3.00	1789.14	236.79
45	116.92	3.59	275.20	11.88	1775.73	232.60	1.71	0.18	14.00	1.87	32.80	3.63	1784.39	235.22

the ability to deviate further from the shortest paths to the disadvantage of certain OD-pairs can only lead to system improvement which comes from routing more distant OD pairs on their shortest path.

Lastly we evaluate the impact of arc capacity on the networks designed by the (UTM-TT) model. Setting the budget to 110% of the minimum and the path length deviation to 10%, (UTM-TT-PWL) is solved for arc capacities of $25, 30, \dots, 45$. We use a maximum arc capacity of 50 and fifty pieces in the PWL congestion functions. All except three instances solved to optimality within three hours; of the instances that timed out, the maximum MIP gap was 0.2%. The results are summarized in Table 7. The results show that for lower arc capacities, the solution deviates from the minimum travel distance solution to route more flow on congestion-avoiding paths. One natural question to ask here is: are the different solutions a result of choosing different networks at different arc capacity levels, or simply representative of a different choice of routing? Comparing the arc capacity 25 solutions to the arc capacity 35 solutions, there is an average of 29.2 different arcs in the lower capacity solution. Comparing the arc capacity 25 solutions to the arc capacity 45 solutions, there is an average of 37.0 different arcs in the lower capacity solution. Three of the instances select the same network for arc capacity 40 and 45, showing that the arc capacity impact on network designs levels off when congestion is low.

Lastly, we narrow in on one instance and compare the networks designed. Fig. 8 includes an image of the nodes appearing in some OD-pair for the instance (left), and depicts the minimum cost network (middle) and minimum distance network for budget deviation 10% (right). The networks chosen are somewhat similar — with more budget small adjustments can be made to route more OD-pairs on shorter paths. Fig. 9 depicts the solutions of the (UTM-TT) model with arc capacity 30, max capacity 50 and budget deviations 10% (left), 20% (center), and 30% (right). The minimum distance and minimum travel time network choices for budget deviation 10% differ as expected; when congestion is considered, more arcs in the downtown core area are opened as opposed to outer roads. As the budget increases within the (UTM-TT) model, the larger budget allows the model to both alleviate oversubscribed arcs and open additional outer roads used only by a few OD-pairs. Fig. 10 depicts the solutions of the (UTM-TT) model with budget deviation 10%, max capacity 50 and arc capacities 25 (left), 35 (center), and 45 (right). Again we see that the importance placed on congestion via arc capacity choice impacts the trade-off between mapping underused outer roads and relieving congestion on city center roads.

Many of the networks chosen by the models have a topology consisting of a backbone grid in the central region complemented by access paths to outlying nodes. This topology arises in other network design problems and there are two driving forces prompting it to arise here. Firstly, absent congestion effects, the path length limits are sufficient to induce multiple ways to cross the backbone region. Secondly, considering congestion effects leads to opening alternative routes that divert flow from shared short paths. A cursory glance at the networks designed show they have good survivability. Roughly, this means that there are multiple paths from origin to destination and therefore failure of an arc or node does not result in complete network failure. Determining survivability in the UTM context also requires evaluating the length of the paths that remain after a node or arc is removed; see Section 6 for further discussion.

5.3. *Insights from other center demand patterns*

The fifty OD pair SU instance class results share many similarities with the observations made for the RNP fifty OD pair instance. As in the RNP instance, an arc capacity of 30 and max capacity of 50 are appropriate for this demand level; the detailed results of running the (UTM-Cost), (UTM-Dist) and (UTM-TT) models are provided in Appendix A.1. Here we highlight the major differences observed in the results. Firstly, the number of kilometers of road in the minimum cost network serving demand is higher across all path length deviations. Secondly, at the same path length and budget deviation, the number of OD-pairs assigned a shortest path route is greater for both the (UTM-Dist) and (UTM-TT) models. Possible explanations for the larger investment requirement are that

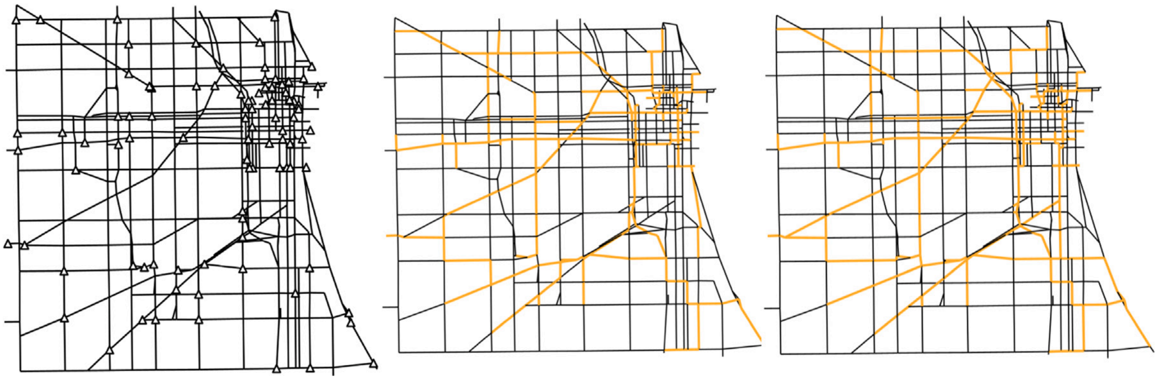

Fig. 8. Fixed fifty OD-pair RNP instance, minimum cost and minimum distance networks.

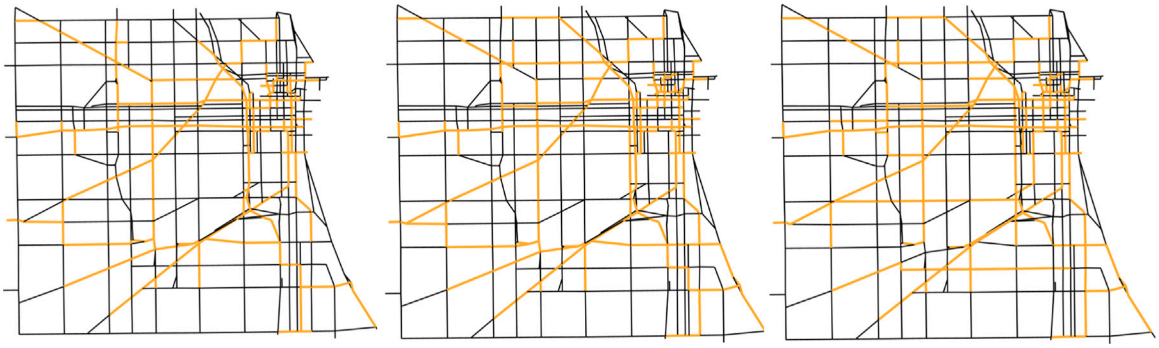

Fig. 9. Minimum travel time networks for fixed fifty OD-pair RNP instance, varying budget.

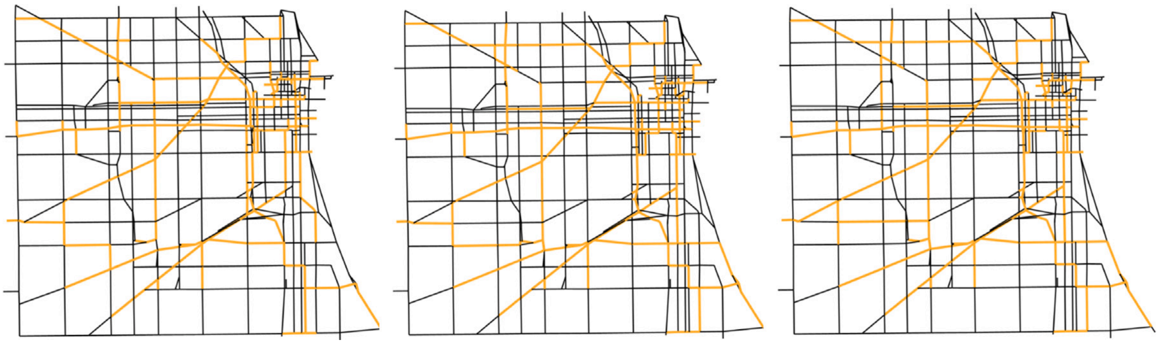

Fig. 10. Minimum travel time networks for one fifty OD-pair RNP instance varying arc capacity.

the OD-pairs are farther apart on average and there is no demand concentrated in the downtown core that allows pooling of flows. The increase in shortest path routing seems to stem from the larger budget; at the same absolute budget and path length deviation, the RNP instance routes at least as many OD-pairs on their shortest paths compared to the SU instances.

The fifty OD pair WC instance class diverges from the RNP and SU instances. To analyze the WC demand pattern, we select a warehouse location south-west of the city center, close to the Amazon warehouse's location. The main difference is that the oversubscribed roads are evident before running any model. Congestion effects near the warehouse will dominate. The detailed results of running the (UTM-Cost) and (UTM-Dist) models are provided in Appendix A.2. The minimum cost networks computed are much cheaper than the RNP and SU demand patterns. This is natural because the flights can use shared paths near the warehouse. This can be seen in the minimum distance model; the maximum flow on an arc over all instances is around 166. For a specific instance in the class, we focus on the structure of the network under various capacities. The results are summarized in Table 8. For low capacities, the network structure is complete (i.e. includes all roads) near the warehouse; as long as all demand is served,

Table 8
Minimum travel time with path length and budget constraints for fifty OD-pair WC Instance #1. A maximum capacity of 250 was used and the piecewise linear approximation used 50 pieces. The true travel time - vs. that computed by the (UTM-TT-PWL) model - is given in the last column.

pld	bd	cap	cost	num arcs	travel dist	avg pld	num on sp	travel time	true travel time
10	20	50	76.99	178	1961.93	4.02	11	2367.47	2363.74
10	20	100	76.97	183	1924.74	2.12	18	1962.12	1961.92
10	20	150	76.99	180	1889.52	0.66	23	1910.29	1910.61
10	20	200	77.01	178	1883.25	0.27	30	1893.98	1893.94
10	50	50	96.22	217	1981.63	4.96	8	2135.59	2134.21
10	50	100	96.09	215	1910.19	1.64	25	1937.47	1937.22
10	50	150	95.71	230	1887.82	0.57	33	1903.66	1903.56
10	50	200	96.10	241	1882.36	0.34	41	1890.50	1890.49

Table 9
Minimum travel time with path length and budget constraints for one hundred OD-pair RNP Instance #1. A maximum capacity of 100 was used and the piecewise linear approximation used 50 pieces.

pld	bd	cost	num arcs	travel dist	avg pld	num on sp	max flow	travel time	true travel time
10	10	160.41	382	3367.93	2.16	25	33	3444.32	3443.40
10	20	175.04	406	3338.57	1.55	30	31	3395.10	3394.33
10	30	189.63	435	3330.85	1.38	34	29	3371.82	3371.26
10	40	204.20	480	3329.53	1.35	37	29	3356.70	3360.18
10	50	218.81	506	3319.39	1.09	46	29	3352.54	3352.10

the incentive is to map roads near the warehouse to alleviate congestion. As the capacity increases, more budget is allocated to routing on shorter distance paths. The warehouse location we chose was not the geographic center of the demand which led to some interesting flight patterns. In the low capacity instances, many routes involved “flying south then north” to avoid congestion. At higher capacities, these detours were no longer required.

5.4. Insights from larger instances

The one hundred OD-pair RNP instance class is similar to the fifty OD pair instance class. The detailed results of running the (UTM-Cost) and (UTM-Dist) models are provided in Appendix A.3. The amount of extra budget needed to connect all OD pairs was about 35% more with a slightly larger increase needed for smaller path length deviations. The average total travel distances computed by the (UTM-Dist) model is a little less than double the average total travel distance for the fifty OD pair instances. The average path length deviation is also lower. This is as expected; as these models are not concerned about the number of flights along an arc, extra budget to serve demand simultaneously provides shorter routes. The (UTM-TT) model was run for these instances for path length deviation 10% and capacity of 30. Here we highlight the results for an instance that completed all settings in three hours with mipgap less than 2%.[1] One major observation is that even though the minimum distance model chooses to overload specific arcs, a same cost network and corresponding routing exists that does not overload any one arc. In fact, each arc is assigned flow less than its capacity without a hard constraint requiring this to be so. Another insight is that the large budget is not needed to alleviate congestion for this instance. The increase in budget is spent here in providing shorter paths (see Table 9).

6. Conclusion

In this work we introduced the first UAV network design model for UTM providers. The model differs from existing UAV research in that it considers the 3D nature of flights through a congestion function, assumes UAVs fly over existing streets rather than via point-to-point flight, and explicitly evaluates routings which are designed in respect of regulatory constraints and airspace design. We used simulation to develop BPR congestion function parameters consistent with routing UAVs in 3D space in respect of separation standards and along multiple corridors according to altitude.

While we used the models to design new networks, the models and methodology developed in this work apply in a network expansion context. There are many further operational research problems that arise from this point of view and in this domain. Firstly, the model studied herein may be extended to capture survivability needs and demand uncertainty. The networks designed in the case study often contained more than one short enough route from an origin to a destination; however, actively modeling multiple paths as a requirement may lead to alternative topologies. As in conventional survivable network design models, relaxing the path length limit for a “back-up path” may be appropriate; definitions of survivability should be considered with input from UTM implementers for whom safe operation is a main concern. The case study examined multiple instances drawn from each instance class; however, our objective was to show the validity of the observed results across inputs rather than design models that were

[1] Except for one instance, results were obtained for all instances for budget deviations greater than 20%. We present one instance here for clarity in reading the results.

Table 10
Instance characteristics for fifty OD-pair SU instances.

Instance #	Number of flights	Average shortest path distance (km)
0	254	7.5
1	290	7.6
2	273	7.3
3	305	8.7
4	275	7.2

Table 11
Minimum cost networks satisfying path length constraints for fifty OD-pair SU instances.

pld	cost		num arcs	
	avg	std	avg	std
0	196.99	12.361	393.80	21.16
10	125.45	6.64	244.00	6.04
20	107.81	4.50	211.80	3.56

Table 12
Minimum travel distance with path length and budget constraints for fifty OD-pair SU instances.

pld	bd	cost		num arcs		travel dist		avg pld		num on sp		max flow	
		avg	std	avg	std	avg	std	avg	std	avg	std	avg	std
10	10	137.93	7.32	267.80	8.01	2169.39	296.44	1.27	0.49	18.60	4.34	38.20	9.76
10	20	150.48	8.00	294.20	7.29	2152.17	296.52	1.58	0.28	22.80	3.96	39.60	10.21
10	30	162.82	8.66	319.60	4.98	2144.52	295.15	0.24	0.13	27.80	5.81	38.60	9.53
10	40	175.40	9.06	344.20	9.09	2141.26	294.44	0.09	0.07	35.20	6.76	36.00	10.77
10	50	186.50	7.90	374.00	5.15	2139.77	293.81	0.02	0.02	42.40	6.27	34.00	12.65
20	10	118.57	4.97	234.60	8.76	2229.69	306.94	3.69	0.97	13.20	4.09	46.80	13.03
20	20	129.34	5.42	252.40	7.77	2185.88	299.31	2.03	0.87	17.20	3.83	36.80	2.28
20	30	140.09	5.87	271.80	9.12	2163.11	295.18	1.09	0.63	19.20	4.27	39.80	10.18
20	40	150.87	6.27	293.60	13.05	2151.52	295.40	0.57	0.36	23.20	5.07	39.60	10.21
20	50	161.53	6.57	316.20	9.78	2145.11	294.83	0.29	0.23	27.80	5.36	38.00	9.90

robust to the actual realization of demand. In early stage development, UTM providers may want to design a network with good coverage regardless of the realized demand. In this case, developing stochastic models would provide valuable insights. The UTM network design problem and case study led to a high-level model with static path assignment, appropriate for the time frame of the decisions being made; however, the dynamic variant of the model gives rise to many operational questions. For example, once a UTM network consisting of 3D corridors above roads is established, which routing protocols will be best for handling flight requests in a busy urban airspace?

CRediT authorship contribution statement

Leanne Stuive: Writing – review & editing, Writing – original draft, Visualization, Validation, Methodology, Investigation, Formal analysis, Conceptualization. **Fatma Gzara:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization.

Acknowledgments

Map data copyrighted OpenStreetMap contributors and available from https://www.openstreetmap.org.

Appendix

A.1. SU fifty OD-pair results

See Tables 10–13.

A.2. WC fifty OD-pair results

See Tables 14–16.

A.3. RNP one hundred OD-pair results

See Tables 17 and 18.

Table 13
Minimum travel time with path length and budget constraints for fifty OD-pair SU instances.

pld	bd	cost		num arcs		travel dist		avg pld		num on sp		max flow		travel time	
		avg	std	avg	std	avg	std	avg	std	avg	std	avg	std	avg	std
10	10	137.97	7.30	268.60	7.77	2177.75	300.03	1.57	0.46	16.20	2.86	29.40	5.98	2210.95	318.34
10	20	150.49	7.95	294.40	4.04	2160.23	302.85	0.89	0.37	20.80	2.86	28.80	6.61	2184.62	312.94
10	30	163.05	8.63	321.40	5.86	2152.18	299.12	0.53	0.19	24.20	3.42	27.00	7.00	2171.40	309.31
10	40	175.39	9.02	349.60	4.98	2148.22	298.00	0.35	0.19	28.00	6.28	25.60	4.72	2164.92	305.59
10	50	188.09	9.90	377.20	7.82	2146.38	296.82	0.28	0.15	31.20	7.50	25.20	5.07	2161.57	303.55
20[a]	10[a]	118.72	5.74	233.50	5.92	2115.70	129.40	4.20	0.97	12.00	2.94	31.00	1.41	2156.19	137.84
20	20	129.34	5.39	253.00	8.66	2199.70	311.57	2.52	1.03	14.20	4.15	29.00	6.00	2237.93	335.07
20	30	140.12	5.85	272.80	9.91	2171.77	298.64	1.47	0.65	17.00	2.45	30.20	5.50	2200.65	315.31
20	40	150.92	6.30	293.80	10.80	2159.33	300.42	0.90	0.35	19.80	2.05	29.20	6.76	2183.25	312.45
20	50	161.67	6.66	317.00	9.27	2153.28	298.99	0.61	0.22	23.20	4.02	26.80	6.98	2172.12	308.51

[a] One of the pld 20 bd 10 instances timed out with no solution and we omit it from the results.

Table 14
Instance characteristics for fifty OD-pair WC instances.

Instance #	Number of flights	Average shortest path distance (km)
0	268	7.0
1	286	7.2
2	290	7.2
3	274	7.3
4	309	7.3

Table 15
Minimum cost networks satisfying path length constraints for the fifty OD-pair WC instances.

dmd	pld	cost		num arcs	
		avg	std	avg	std
wc	0	95.60	7.04	218.00	13.40
wc	10	67.27	2.29	163.40	7.13
wc	20	61.10	2.15	153.00	4.18

Table 16
Minimum travel distance with path length and budget constraints for fifty OD-pair WC instances.

pld	bd	cost		num arcs		travel dist		avg pld		num on sp		max flow	
		avg	std	avg	std	avg	std	avg	std	avg	std	avg	std
10	10	73.94	2.51	178.00	4.80	2066.67	134.07	0.67	0.32	27.40	3.36	177.20	23.50
10	20	80.52	2.63	191.40	11.74	2055.33	132.58	0.19	0.15	34.20	4.97	172.20	18.70
10	30	87.13	2.87	202.60	6.47	2051.83	133.03	0.03	0.00	42.40	2.40	169.00	12.90
10	40	92.55	4.33	214.20	9.36	2051.26	133.15	0.00	0.00	48.40	2.51	167.40	14.00
10	50	95.69	7.08	218.60	13.81	2051.21	133.14	0.00	0.00	50.00	0.00	166.80	14.20
20	10	67.15	2.35	163.80	4.32	2085.70	143.31	1.69	0.73	21.40	1.67	172.40	20.80
20	20	73.24	2.50	176.00	3.81	2065.23	139.20	0.64	0.38	28.80	2.49	174.00	18.20
20	30	79.18	2.57	188.00	7.35	2056.76	134.90	0.27	0.18	32.80	4.87	172.20	18.70
20	40	85.43	3.13	197.40	7.96	2052.23	133.12	0.05	0.05	41.40	2.61	169.00	12.90
20	50	91.14	3.56	212.40	7.89	2051.38	133.12	0.00	0.00	45.60	3.91	167.40	14.00

Table 17
Minimum cost networks satisfying path length limits for one hundred OD-pair RNP instances.

instance params		cost (km)		num arcs	
dmd	pld	avg	std	avg	std
rnp	0	233.32	17.83	538.20	31.22
rnp	10	145.53	8.74	351.40	12.66
rnp	20	122.16	6.10	308.60	13.94

A.4. Model runtime and PWL approximation accuracy

We analyze the runtime of the models, as well as the quality of the piecewise linearization in approximating (UTM-TT). Table 19 describes the additional abbreviations and conventions we adopt to report model results.

First we briefly discuss runtimes for the (UTM-Cost) and (UTM-Dist) models, which are summarized in Tables 20 and 21. Both of these models are mixed integer linear programs and there is no approximation. Our implementation applied the standard technique

Table 18
Minimum travel distance with path length and budget constraints for one hundred OD-pair RNP instances.

pld	bd	cost		num arcs		travel dist		avg pld		num on sp		max flow	
		avg	std	avg	std	avg	std	avg	std	avg	std	avg	std
10	10	160.05	9.60	377.20	17.36	3441.02	206.58	1.19	0.29	31.00	4.18	67.80	9.30
10	20	174.58	10.47	411.20	22.29	3416.28	203.03	0.54	0.13	43.20	4.82	61.60	14.40
10	30	189.08	11.29	440.20	22.05	3404.88	200.45	0.24	0.09	54.20	7.46	58.60	14.30
10	40	203.36	11.93	469.60	24.74	3399.37	198.81	0.10	0.00	64.80	7.46	55.60	15.90
10	50	218.02	13.17	504.60	19.96	3396.72	197.85	0.03	0.00	81.80	10.78	55.60	15.90
20	10	134.36	6.71	331.00	12.27	3539.21	218.00	4.03	0.67	22.20	6.06	76.20	11.65
20	20	146.57	7.33	355.00	13.49	3476.37	208.59	2.15	0.56	28.80	6.30	76.60	6.60
20	30	158.80	7.93	376.80	17.71	3439.73	203.22	1.25	0.33	32.20	4.60	66.40	8.60
20	40	170.96	8.56	404.40	20.96	3419.25	200.89	0.63	0.16	40.00	4.64	60.40	14.50
20	50	183.16	9.20	429.40	19.21	3408.11	199.67	0.33	0.11	50.00	2.83	58.20	14.60

Table 19
Runtime statistics reported for UTM network design models.

Heading	Description
nto	The number of instances that timed due to the 3 h runtime limit
nfail	The number of instances that failed to produce a solution within the 3 h runtime limit
runtime	Runtime (wall-clock time) in seconds, obtained from Gurobi
work	Gurobi-specific parameter providing a deterministic measure of the work required to solve the problem
mipgap	Absolute difference between the incumbent and dual bound, divided by the incumbent objective value, obtained from Gurobi

Table 20
Runtimes for (UTM-Cost) model.

num ods	dmd	pld	Runtime		Work		mipgap		nto
			avg	std	avg	std	avg	std	
50	rnp	0	0.01	0.00	0.01	0.00	0.00	0.00	0
50	rnp	10	196.54	213.04	454.76	499.24	0.00	0.00	0
50	rnp	20	3952.38	2005.58	11 382.01	6046.72	0.00	0.00	0
50	su	0	0.01	0.00	0.01	0.00	0.00	0.00	0
50	su	10	713.42	528.22	1416.49	1571.98	0.00	0.00	0
50	su	20	9242.41	3959.38	15 021.44	11 081.29	1.96	1.28	4
50	wc	0	0.01	0.00	0.01	0.00	0.00	0.00	0
50	wc	10	10.26	10.46	18.28	21.19	0.00	0.00	0
50	wc	20	1005.14	770.41	2632.12	2146.32	0.00	0.00	0
100	rnp	0	0.01	0.00	0.01	0.00	0.00	0.00	0
100	rnp	10	8044.23	3863.89	13 137.38	9728.25	0.51	0.85	2
100	rnp	20	11 287.31	404.67	9420.42	7657.36	4.16	1.51	5

Table 21
Runtimes for (UTM-Dist) model. The results for the fifty OD-pair instances were averaged over all pld in $\{10, 20\}$ and bd in $\{10, 20, 30, 40, 50\}$.

num ods	dmd	pld	bd	Runtime			Work		mipgap		nto
				avg	std	max	avg	std	avg	std	
50	rnp	*	*	4.57	14.07	97.27	9.79	13.21	0.00	0.00	0
50	su	*	*	12.39	44.06	292.43	33.09	135.38	0.00	0.00	0
50	wc	*	*	0.85	2.14	15.17	1.22	3.18	0.00	0.00	0
100	rnp	10	10	14.90	21.38	52.92	31.68	53.57	0.00	0.00	0
100	rnp	10	20	2.05	1.13	3.72	2.76	1.66	0.00	0.00	0
100	rnp	10	30	1.03	0.48	1.61	1.44	0.71	0.00	0.00	0
100	rnp	10	40	1.01	0.62	2.06	1.26	0.72	0.00	0.00	0
100	rnp	10	50	0.63	1.89	0.83	0.83	0.21	0.00	0.00	0
100	rnp	20	10	2868.55	4642.96	10 800.82	5470.04	11 211.56	0.05	0.1	1
100	rnp	20	20	1407.78	3075.46	6909.16	612.55	1199.48	0.00	0.00	0
100	rnp	20	30	31.48	43.34	108.53	50.97	68.38	0.00	0.00	0
100	rnp	20	40	3.38	1.26	5.49	6.15	1.95	0.00	0.00	0
100	rnp	20	50	2.36	0.81	3.70	3.75	1.28	0.00	0.00	0

of removing the x_{ij}^k variables for all arcs $(i, j) \in A'$ and OD-pairs $k \in K$ for which every path from O_k to D_k that includes (i, j) has length greater than L_k. This pre-processing step is most effective when the path length deviation is small allowing a large number of variables to be eliminated. We found that the (UTM-Cost) model may be difficult to solve, especially when the path length deviation

Table 22
Runtimes for (UTM-TT-PWL) model for the fifty OD-pair RNP instances.

num ods	dmd	pld	bd	Runtime			Work		mipgap		nto
				avg	std	max	avg	std	avg	std	
50	rnp	10	10	1286.47	1308.23	3566.21	3471.14	3562.85	0.00	0.00	0
50	rnp	10	20	401.32	308.66	783.19	1012.86	837.47	0.00	0.00	0
50	rnp	10	30	107.53	116.47	262.03	228.90	268.26	0.00	0.00	0
50	rnp	10	40	33.94	41.13	105.81	63.00	85.1	0.00	0.00	0
50	rnp	10	50	8.81	6.80	16.27	11.72	9.99	0.00	0.00	0
50	rnp	20	10	7320.99	4492.17	11 090.85	14 534.22	13 725.39	0.43	0.6	2
50	rnp	20	20	5401.41	5371.01	11 811.79	7054.74	8847.46	0.01	0.00	1
50	rnp	20	30	1883.06	1545.82	4140.85	2305.99	2644.94	0.00	0.00	0
50	rnp	20	40	418.92	354.26	914.44	596.51	642.37	0.00	0.00	0
50	rnp	20	50	87.53	72.52	191.87	189.48	185.56	0.00	0.00	0

Table 23
Runtimes for (UTM-TT-PWL) model for the fifty OD-pair SU instances.

num ods	dmd	pld	bd	Runtime			Work		mipgap		nto	nfail
				avg	std	max	avg	std	avg	std		
50	su	10	10	1288.23	2156.47	5082.13	3087.22	6302.04	0.00	0.00	0	0
50	su	10	20	1829.13	3738.66	8512.83	4785.59	10 292.20	0.00	0.00	0	0
50	su	10	30	844.73	1860.92	4173.63	2336.63	5185.04	0.00	0.00	0	0
50	su	10	40	65.83	135.62	308.40	147.73	316.61	0.00	0.00	0	0
50	su	10	50	16.61	30.20	70.60	32.76	64.85	0.00	0.00	0	0
50	su	20	10	6957.57	5282.81	11 728.33	6672.78	5213.71	0.56	0.68	2	1
50	su	20	20	6734.64	5911.20	11 625.18	2371.67	1870.72	0.3	0.67	2	0
50	su	20	30	2814.18	4999.22	11 640.74	621.58	622.02	0.07	0.1	1	0
50	su	20	40	2530.48	5012.00	11 468.55	415.94	644.37	0.07	0.1	1	0
50	su	20	50	2251.85	4481.82	10 236.97	1245.36	2707.11	0.00	0.00	0	0

is large. The (UTM-Dist) model solved relatively quickly on all 50 node instances. For the 100 node instances, the model was slowest when the budget deviation was low and for the larger path length deviation.

The (UTM-TT-PWL) took longer to solve, but except for one SU instance, all the 50 OD-pair RNP and SU instances had good solutions at the three hour time limit. Again, runtimes were slowest for instances with a tighter budget and larger path length deviation (see Tables 22 and 23).

Data availability

Data will be made available on request.

References

Angelelli, Enrico, Morandi, Valentina, Savelsbergh, Martin, Speranza, Maria Grazia, 2021. System optimal routing of traffic flows with user constraints using linear programming. European J. Oper. Res. 293 (3), 863–879.

Angelelli, Enrico, Morandi, Valentina, Speranza, Maria Grazia, 2020. Minimizing the total travel time with limited unfairness in traffic networks. Comput. Oper. Res. 123, 105016.

Avella, Pasquale, Bernardi, Giacomo, Boccia, Maurizio, Mattia, Sara, 2019. An optimization approach for congestion control in network routing with quality of service requirements. Networks 74 (2), 124–133.

Baloch, Gohram, Gzara, Fatma, 2020. Strategic network design for parcel delivery with drones under competition. Transp. Sci. 54, 204–228.

Bauranov, Aleksandar, Rakas, Jasenka, 2021. Designing airspace for urban air mobility: A review of concepts and approaches. Prog. Aerosp. Sci. 125, 100726.

Bayram, Vedat, Tansel, Barbaros Ç., Yaman, Hande, 2015. Compromising system and user interests in shelter location and evacuation planning. Transp. Res. B 72, 146–163.

Boeing, Geoff, 2017. OSMnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks. Comput. Environ. Urban Syst. 65, 126–139.

Branston, David, 1976. Link capacity functions: A review. Transp. Res. 10 (4), 223–236.

Bureau of Public Roads, 1964. Traffic Assignment Manual for Application with a Large, High Speed Computer, vol. 2, US Department of Commerce, Bureau of Public Roads, Office of Planning, Urban.

Casey, Gerard, Zhao, Bingyu, Kumar, Krishna, Soga, Kenichi, 2020. Context-Specific Volume-Delay Curves by Combining Crowd-Sourced Traffic Data with Automated Traffic Counters (ATC): A Case Study For London. Technical Report, arXiv:2011.01964, arXiv.

Chauhan, Darshan, Unnikrishnan, Avinash, Figliozzi, Miguel, 2019. Maximum coverage capacitated facility location problem with range constrained drones. Transp. Res. C 99, 1–18.

Chin, Christopher, Gopalakrishnan, Karthik, Balakrishnan, Hamsa, Egorov, Maxim, Evans, Antony, 2021. Efficient and fair traffic flow management for on-demand air mobility. CEAS Aeronaut. J..

Chiu, Yi-Chang, Bottom, Jon, Mahut, Michael, Paz, Alexander, Balakrishna, Ramachandran, Waller, Steven, Hicks, Jim, 2011. Dynamic Traffic Assignment: A Primer (Transportation Research Circular E-C153). Transportation Research Board.

Chung, Sung Hoon, Sah, Bhawesh, Lee, Jinkun, 2020. Optimization for drone and drone-truck combined operations: A review of the state of the art and future directions. Comput. Oper. Res. 123, 105004.

Cokyasar, Taner, Dong, Wenquan, Jin, Mingzhou, Verbas, İsmail Ömer, 2021. Designing a drone delivery network with automated battery swapping machines. Comput. Oper. Res. 129, 105177.

Crainic, Teodor Gabriel, Gendreau, Michel, Gendron, Bernard (Eds.), 2021. Network Design with Applications to Transportation and Logistics. Springer.

Decker, Christopher, Chiambaretto, Paul, 2022. Economic policy choices and trade-offs for unmanned aircraft systems traffic management (UTM): Insights from europe and the United States. Transp. Res. A 157, 40–58.

Doole, Malik, Ellerbroek, Joost, Knoop, Victor L., Hoekstra, Jacco M., 2021. Constrained urban airspace design for large-scale drone-based delivery traffic. Aerospace 8 (2), 38.

Edwards, David, 2022. Flytrex receives FAA approval to double drone delivery radius to two nautical miles. URL https://roboticsandautomationnews.com/2022/07/28/flytrex-receives-faa-approval-to-double-drone-delivery-radius-to-two-nautical-miles/53615/. (Accessed 23 December 2022).

FAA, 2023. UAS facility maps. URL https://www.faa.gov/uas/commercial_operators/uas_facility_maps. (Accessed 31 March 2023).

Geißler, Björn, Martin, Alexander, Morsi, Antonio, Schewe, Lars, 2011. Using piecewise linear functions for solving MINLPs. In: Mixed Integer Nonlinear Programming. Springer, pp. 287–314.

Gürel, Sinan, 2011. A conic quadratic formulation for a class of convex congestion functions in network flow problems. European J. Oper. Res. 211 (2), 252–262.

Hong, Insu, Kuby, Michael, Murray, Alan T., 2018. A range-restricted recharging station coverage model for drone delivery service planning. Transp. Res. C 90, 198–212.

Hou, Wenjuan, Fang, Tao, Pei, Zhi, He, Qiao-Chu, 2021. Integrated design of unmanned aerial mobility network: A data-driven risk-averse approach. Int. J. Prod. Econ. 236, 108131.

Jahn, Olaf, Möhring, Rolf H., Schulz, Andreas S., 2000. Optimal routing of traffic flows with length restrictions in networks with congestion. In: Operations Research Proceedings 1999. Springer, pp. 437–442.

Jahn, Olaf, Möhring, Rolf H., Schulz, Andreas S., Stier-Moses, Nicolás E., 2005. System-optimal routing of traffic flows with user constraints in networks with congestion. Oper. Res. 53 (4), 600–616.

Jang, Dae-Sung, Ippolito, Corey A, Sankararaman, Shankar, Stepanyan, Vahram, 2017. Concepts of airspace structures and system analysis for uas traffic flows for urban areas. In: AIAA Information Systems-AIAA Infotech@ Aerospace. p. 0449.

Jiang, Tao, Geller, Jared, Ni, Daiheng, Collura, John, 2016. Unmanned aircraft system traffic management: Concept of operation and system architecture. Int. J. Transp. Sci. Technol. 5 (3), 123–135.

Kirschstein, Thomas, 2020. Comparison of energy demands of drone-based and ground-based parcel delivery services. Transp. Res. D 78, 102209.

Kopardekar, Parimal H., 2019. Unmanned autonomous systems (UAS) traffic management. In: Explore Flight: Caltrans Planning Horizons Forum. ARC-E-DAA-TN68454.

Macrina, Giusy, Di Puglia Pugliese, Luigi, Guerriero, Francesca, Laporte, Gilbert, 2020. Drone-aided routing: A literature review. Transp. Res. C 120, 102762.

Marcotte, Patrice, 1986. Network design problem with congestion effects: A case of bilevel programming. Math. Program. 34 (2), 142–162.

Mathew, Tom V., Sharma, Sushant, 2009. Capacity expansion problem for large urban transportation networks. J. Transp. Eng. 135 (7), 406–415.

Merkert, Rico, Bushell, James, 2020. Managing the drone revolution: A systematic literature review into the current use of airborne drones and future strategic directions for their effective control. J. Air Transp. Manage. 89, 101929.

Morandi, Valentina, 2021. Bridging the user equilibrium and the system optimum in static traffic assignment: how the cooperation among drivers can solve the congestion problem in city networks. arXiv preprint arXiv:2105.05804.

Moshref-Javadi, Mohammad, Winkenbach, Matthias, 2021. Applications and Research avenues for drone-based models in logistics: A classification and review. Expert Syst. Appl. 177, 114854.

Mtoi, Enock, Moses, Ren, 2014. Calibration and evaluation of link congestion functions: Applying intrinsic sensitivity of link speed as a practical consideration to heterogeneous facility types within urban network. J. Transp. Technol. 4, 141–149.

O'Brien, Ciara, 2022. Alphabet's Wing to trial drone deliveries in Ireland. URL https://www.irishtimes.com/business/2022/10/20/alphabets-wing-to-trial-drone-deliveries-in-ireland/. (Accessed 23 December 2022).

Open Street Map Contributors, 2022a. OSM Wiki: Illinois/road classification. URL https://wiki.openstreetmap.org/wiki/United_States/Road_classification. (Accessed 10 November 2022).

Open Street Map Contributors, 2022b. OSM Wiki: United States/road classification. URL https://wiki.openstreetmap.org/wiki/United_States/Road_classification. (Accessed 10 November 2022).

Open Street Map Contributors, 2022c. Planet dump retrieved from https://planet.osm.org. URL https://www.openstreetmap.org.

Otto, Alena, Agatz, Niels, Campbell, James, Golden, Bruce, Pesch, Erwin, 2018. Optimization approaches for civil applications of unmanned aerial vehicles (UAVs) or aerial drones: A survey. Networks 72 (4), 411–458.

Patriksson, Michael, 2015. The Traffic Assignment Problem: Models and Methods. Courier Dover Publications.

Perederieieva, Olga, Ehrgott, Matthias, Raith, Andrea, Wang, Judith Y.T., 2015. A framework for and empirical study of algorithms for traffic assignment. Comput. Oper. Res. 54, 90–107.

Pinto, Roberto, Lagorio, Alexandra, 2022. Point-to-point drone-based delivery network design with intermediate charging stations. Transp. Res. C 135, 103506.

Poikonen, Stefan, Campbell, James F., 2021. Future directions in drone routing research. Networks 77 (1), 116–126.

Šarić, Ammar, Lovrić, Ivan, 2021. Improved volume–delay function for two-lane rural highways with impact of road geometry and traffic-flow heterogeneity. J. Transp. Eng. A: Syst. 147 (10), 04021066.

Shavarani, Seyed Mahdi, Golabi, Mahmoud, Izbirak, Gokhan, 2019a. A capacitated biobjective location problem with uniformly distributed demands in the UAV-supported delivery operation. Int. Trans. Oper. Res. 28 (6), 3220–3243.

Shavarani, Seyed Mahdi, Mosallaeipour, Sam, Golabi, Mahmoud, Izbirak, Gokhan, 2019b. A congested capacitated multi-level fuzzy facility location problem: An efficient drone delivery system. Comput. Oper. Res. 108, 57–68.

Skydrop, 2022. SkyDrop, domino's gear up to launch commercial drone delivery trial in new zealand. URL https://www.prnewswire.com/news-releases/skydrop-dominos-gear-up-to-launch-commercial-drone-delivery-trial-in-new-zealand-301573821.html. (Accessed 23 December 2022).

Staff, Amazon, 2022. Amazon prime air prepares for drone deliveries. URL https://www.aboutamazon.com/news/transportation/amazon-prime-air-prepares-for-drone-deliveries. (Accessed 23 December 2022).

Stöcker, Claudia, Bennett, Rohan, Nex, Francesco, Gerke, Markus, Zevenbergen, J., 2017. Review of the current state of UAV regulations. Remote Sens. 9, 459.

Transportation Networks for Research Core Team, 2022. Transportation networks for research. URL https://github.com/bstabler/TransportationNetworks. (Accessed 20 November 2022).

Walmart, 2022. We're bringing the convenience of drone delivery to 4 million U.S. households in partnership with droneup. URL https://corporate.walmart.com/newsroom/2022/05/24/were-bringing-the-convenience-of-drone-delivery-to-4-million-u-s-households-in-partnership-with-droneup. (Accessed 23 December 2022).

Wardrop, John Glen, 1952. Road paper. Some theoretical aspects of road traffic research. Proc. Inst. Civ. Eng. 1 (3), 325–362.

Wong, Wai, Wong, S.C., 2016. Network topological effects on the macroscopic Bureau of Public Roads function. Transportmetrica A: Transp. Sci. 12 (3), 272–296, Publisher: Taylor & Francis.