Scheduling of Urban Air Mobility Services with Limited Landing Capacity and Uncertain Travel Times

Qinshuang Wei, Gustav Nilsson, and Samuel Coogan

***Abstract*— Urban air mobility, in which air transportation is used for relatively short trips within a city or region, is emerging as a possible component in future transportation networks. In this paper, we study the problem of how to schedule urban air mobility trips when travel times are uncertain. Unlike in ground transportation, urban air mobility scheduling has to take into account that there is limited landing capacity at each destination, and for safety reasons, it must be guaranteed that an air vehicle will be able to land before it can be allowed to take off. We first present a network model for an on-demand urban air mobility service with uncertain travel times and limited landing capacity at nodes. For the practically relevant special case of one final destination and many origins, we give necessary and sufficient conditions for a feasible schedule to exist for a given demand of flights. Next, we present a mixed integer program for obtaining an optimal schedule in this case. The paper concludes with a numerical study for a previously proposed urban air network in the city of Atlanta, Georgia.**

I. Introduction

There is growing interest in utilizing urban airspace for transportation of people and goods. Both commercial mobility-on-demand operators [1] and state funded research institutes, such as NASA [2], are exploring such urban air mobility (UAM) solutions in cities and surrounding regions. Studies such as [3]–[8] propose various approaches to allow urban air vehicles (UAVs) to travel safely and efficiently through cities. These proposed ideas cover a wide range of possibilities such as allowing UAVs to land at *vertistops* or *vertiports* installed on roofs of existing buildings or within cloverleaf exchanges on freeways. Several simulation tools [9]–[11] have also been developed, many of them based on how commercial airline traffic is managed today, while others focus on safe coordination of unmanned drones.

In this paper, we study scheduling for UAM networks that accounts for uncertainty in travel time and limited landing capacity. These two aspects have not been considered together in any other UAM solutions. We focus on the practically relevant class of *star-branch* networks consisting of a main destination node, multiple leaf origin nodes, and possibly intermediate nodes between leaf nodes and the destination node. This model captures, for example, travel from exurbs to a main city with possible stops at additional suburbs along the way and is the configuration for all four UAM networks (metro areas of Austin, Atlanta, Boston, and San Francisco) considered the INRIX report [6].

We assume given a demand of flights from origin nodes with arrival deadlines at the destination and consider the problem of scheduling flight departures to ensure that all flights arrive no later than the specified deadline and that there is always an available landing spot at the destination and intermediate nodes when each flight arrives. The main contributions are as follows. First, we formalize an extensible UAM network model accounting for uncertain travel times and limited landing capacity; this model is easily extended and modified for future research. Second, we present necessary conditions for the existence of a feasible schedule for the UAM network. When there are no intermediate nodes, we also show that this condition is sufficient for feasibility. Third, we present a mixed integer program to compute an optimal schedule for the UAM network. We demonstrate our approach on a case study in Atlanta, Georgia based on the example considered in [6].

In the transportation scheduling literature, prior works have considered uncertain travel times or limitations on parking capacity separately. Particularly, [12] investigates how the flow of UAVs depends on the congestion level and finds through simulations similarities with ground highway traffic with high traffic density. However, different from today's mobility-on-demand services on the ground, the availability of landing spots at vertistops or vertiports will be limited for UAM networks. Since UAVs will have limited power storage, safety concerns will dictate that each UAV is guaranteed an available landing spot upon arrival. In addition, UAM travel is particularly vulnerable to uncertain travel times caused by the relatively short travel distances and high variability of factors such as weather. Moreover, for UAM solutions, it is likely that the UAV will only stay on the ground for a short amount of time (e.g., several minutes) to unload and load passengers, emphasizing the importance of timely operation of the whole system.

Uncertainty in routing problems has also been studied before for ground transportation. The paper [13] provides a literature review of such problems, and examples of more recent work are presented in [14], which studies computation of minimum-cost paths through a time-varying network and considers several classes of waiting policies. In [15], a theoretical basis for optimal routing in transportation networks with highly varying traffic conditions is provided, where the goal is to maximize the probability of arriving on time at a destination given a departure time and a time budget.

As mentioned earlier, the availability of landing spots upon

The authors are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, 30332, USA. {qinshuang, gustav.nilsson, sam.coogan}@gatech.edu. S. Coogan is also with the School of Civil and Environmental Engineering, Georgia Institute of Technology. This work was supported in part by the NSF under grant #1749357 and in part by the NASA University Leadership Initiative (ULI) under grant number 80NSSC20M0161.

arrival is critical for safe operation of a UAM network. While the parking availability problem is not usual in ground transportation, it can be critical for truck scheduling, where the drivers are usually required by law to park and rest after a specified amount of driving time. This problem has been addressed in [16], where the authors consider deterministic travel time between different locations and the truck drivers only have access to parking spots during specific time windows, but space limitations at the parking spots are not considered. In [17] the authors solve a similar problem with time-dependent travel times, but do not take the availability of parking spots into account.

Scheduling problems have also been well-studied in the real-time systems community, e.g., in [18], [19], where jobs often are assumed to arrive with a fixed periodicity and in some models have an uncertainty in their processing time. Our fundamental limits are similar in nature and compatible with those found in these works, but our results are tailored to applications in UAM networks. For example, for finite demands, we consider scheduling to achieve prescribed deadlines without excessively early departure times.

The remainder of this paper is organized as follows: In Section II we present the UAM network model. In Section III, we introduce the concept of schedule feasibility in the limit and present necessary and sufficient conditions for feasibility. In Section IV, we present a mixed integer optimization problem to obtain optimal schedules. We then demonstrate in Section V the proposed solution for a realistic scenario of the city of Atlanta proposed in the recent INRIX report [6].

We let $\mathbb{N}$ denote the natural numbers without zero while $\mathbb{N}_0$ the natural numbers with zero, and $\mathbb{R}$ the reals while $\mathbb{R}^+$ the positive reals. For a finite set $\mathcal{A}$, we let $\mathbb{R}^{\mathcal{A}}$, denote the set of vectors indexed by the elements in $\mathcal{A}$.

II. Problem Formulation

We model an urban air mobility (UAM) network using a special class of directed graph $\mathcal{G}_* = (\mathcal{V}_*, \mathcal{E}_*)$, where the set of nodes $\mathcal{V}_*$ consists of: a central node v_0; L leaf nodes v_l for $l = 1, 2, \ldots, L$; and, for each leaf node v_l, a set of intermediate nodes between v_l and the central node v_0 denoted $v_{l,k}$ for $k = 1, 2, \ldots, K_l$ for $K_l \in \mathbb{N}_0$. If $K_l = 0$, then there are no intermediate nodes between v_l and v_0. Then $\mathcal{E}_* = \{(v_{l,k}, v_{l,k-1}) \mid 1 \leq l \leq L \text{ and } 1 \leq k \leq K_l + 1\}$ is the set of links for the network, where we let $v_{l,0} = v_0$ and $v_{l,K_l+1} = v_l$ for all l. We define any graph that satisfies the conditions above as a *star-branch graph*. An example star-branch graph is shown in Fig. 1 and serves as the case study below, where the central node is labeled ATL, the leaf nodes are labeled BUF, KEN, and ALP, and the intermediate notes are labeled a, b, and c.

In a star-branch graph, for each leaf node v_l, there exists a unique set of connected links $\{e_{l,k}\}_{k=1}^{K_l+1}$ along the path to the central node v_0, where we let $e_{l,k} = (v_{l,k}, v_{l,k-1})$. We define *branch* l as the pair $(\{e_{l,k}\}_{k=1}^{K_l+1}, \{v_{l,k}\}_{k=0}^{K_l+1})$.

Each node has a capacity, determined by how many UAVs the node can handle simultaneously. For a node $v \in \mathcal{V}_*$ we

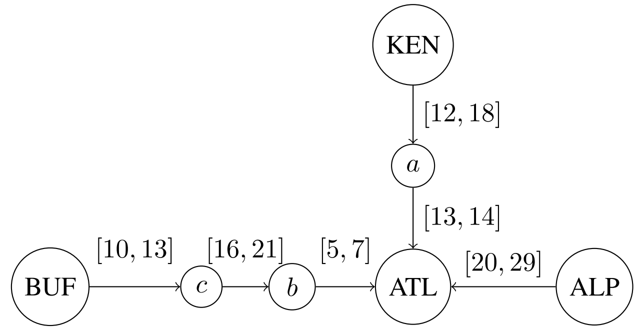

Fig. 1. A star-branch graph for a UAM network of Atlanta (ATL) with three exurbs: Alpharetta (ALP), Kennesaw (KEN) and Buford (BUF). We consider Atlanta as the central node, and there are $0, 1, 2$ intermediate nodes between Atlanta node and the three leaf nodes "ALP", "KEN" and "BUF", respectively. The time interval need for traveling through each link is labeled beside the corresponding link.

denote its capacity as $C_v \in \mathbb{N}_0$, that is, there are C_v *parking slots* at node v where each parking slot allows at most one UAV to stay at any time. We denote the vector of capacities $C \in \mathbb{N}_0^{|\mathcal{V}_*|}$. Nodes are sometimes called *vertistops*, and are called *vertiports* if they have larger capacity. We assume that, due to operational reasons, the UAVs are only allowed to travel along the links specified in $\mathcal{E}_*$.

Since the travel time depends on external factors such as weather conditions, we assume that the travel time for each link is not exact, but rather bounded by a time interval. That is, for each link $i \in \mathcal{E}_*$, we let $\overline{x}_i \in \mathbb{R}^+$ denote the maximum travel time for the link and $\underline{x}_i \in \mathbb{R}^+$ denote the minimum travel time. It is assumed that $\overline{x}_i \geq \underline{x}_i > 0$ for all $i \in \mathcal{E}_*$.

Definition 1 (Urban Air Mobility (UAM) Network): An urban air mobility (UAM) network $\mathcal{N}$ is a tuple $\mathcal{N} = (\mathcal{G}_*, C, \underline{x}, \overline{x})$ where $\mathcal{G}_*, C, \underline{x}, \overline{x}$ are as defined above.

To model the demand of UAV flights in a UAM network $\mathcal{N} = (\mathcal{G}_*, C, \underline{x}, \overline{x})$, we assume that every flight originates from a leaf node and is ultimately destined for the central node v_0 and stops at intermediate nodes along the way. Therefore, a *demand* is a pair (o, f) where $o \in \{v_1, \ldots, v_L\}$ is the origin leaf node and $f \in \mathbb{R}^+$ is the latest time the UAV must arrive (i.e., land) at the destination v_0, i.e., its deadline. Once a UAV has landed at node $v_{l,k}$, it is assumed to block the landing spot for a fixed waiting time $w_{l,k} \in \mathbb{R}^+$, and thus the available capacity for other UAV landing at that node decreases by one during this time. For ease of exposition, we assume the waiting time is uniform at all intermediate nodes and possibly different at the destination node. Thus, we take $w_{l,k} = w_I$ if $k > 0$ and $w_{l,k} = w$ if $k = 0$ for some fixed $w_I > 0$, $w > 0$. The arrival time of the demand is specified instead of the departure time because it is assumed more important for the users to get to their destination on time, rather than to depart at a specific time. The challenge, defined below, is to determine departure times for all demands to ensure UAVs arrive by the specified time.

A set of demands for a UAM network is denoted $\mathcal{D} = \{(o_j, f_j)\}_{j \in \mathcal{J}}$ where $\mathcal{J}$ is a finite or countably infinite index set. We assume throughout that the deadlines f_j

are uniformly lower bounded, but they need not be upper bounded; in particular, $\mathcal{D}$ might be an infinite collection of demands corresponding to, e.g., regular daily or weekly flights planned into the future. Whenever $\mathcal{D}$ is an infinite set, the deadlines f_j must be unbounded. To coordinate the operation of the UAVs, a centralized scheduler assigns to each demand a journey consisting of a sequence of edges from origin to destination and a departure time. Formally, for each $j \in \mathcal{J}$, a *journey* for demand $(o_j, f_j) \in \mathcal{D}$ with $o_j = v_{l_j}$ for $l_j \in \{1, \ldots, L\}$ is a pair $(\{e_{l_j,k}\}_{k=1}^{K_{l_j}+1}, \delta_j)$. For the star-branch networks considered here, the sequence of edges $\{e_{l_j,k}\}_{k=1}^{K_{l_j}+1}$ is uniquely determined by the leaf node $o_j = v_{l_j}$ for the demand, but the departure time $\delta_j \in \mathbb{R}$ is a decision variable of the scheduler. For safety reasons, a UAV must be able to land immediately upon arrival at an intermediate node or the destination. The latest arrival time at some intermediate node $k_I \in \{1, 2, \ldots, K_{l_j} + 1\}$ along the journey, denoted a_{j,k_I}, is then given by

$$a_{j,k_I} = \delta_j + \sum_{k=1}^{k_I} \overline{x}_{e_{l_j,k}} + (k_I - 1)w_I \,, \tag{1}$$

i.e., a_{j,k_I} is the departure time plus the upper bound of the time interval for the first k_I links with the time spent at each intermediate node. Further, the time interval that the UAV will potentially block a landing spot at node k_I is given by

$$\mathcal{M}_{j,k_I} = \left[\delta_j + \sum_{k=1}^{k_I} \underline{x}_{e_{l_j,k}} + (k_I - 1)w_I, a_{j,k_I} + w_{l_j,k_I}\right] .$$

We let $a_j = a_{j,K_{l_j}+1}$ and $\mathcal{M}_j = \mathcal{M}_{j,K_{l_j}+1}$. The task, then, is to assign to each demand a journey such that capacity constraints and arrival deadlines are satisfied. Given a set of demands $\mathcal{D}$, a corresponding set of journeys is denoted $\mathcal{S}$ and called a schedule provided that these constraints are satisfied. Given a finite or countably infinite set of demands, it is always possible to construct a schedule, but if the demands are infinite, depending on the arrival time requirements, the departure times may need to be scheduled arbitrarily far in advance. When this is not the case, we say the schedule and demands are feasible. The notions are formally defined next.

Definition 2: Given a set of demands $\mathcal{D} = \{(o_j, f_j)\}_{j\in\mathcal{J}}$, a corresponding set of journeys $\mathcal{S} = \{(\{e_{l_j,k}\}_{k=1}^{K_{l_j}+1}, \delta_j)\}_{j\in\mathcal{J}}$ is a *schedule* for $\mathcal{D}$ if $a_j \leq f_j$ for each $j \in \mathcal{J}$; the number of UAVs at a node never exceeds capacity, i.e., for all $v \in \mathcal{V}$ and all $t \geq 0$,

$$\sum_{j:d_j=v} \mathbf{1}\left(t; \mathcal{M}_j\right) \leq C_v$$

where the notation $\mathbf{1}(\cdot;\cdot)$ is an indicator such that $\mathbf{1}(t; [a, b]) = 1$ if $t \in [a, b]$ and $\mathbf{1}(t; [a, b]) = 0$ otherwise; and, as a mild technical requirement, in any finite time interval, only a finite number of UAVs depart. A schedule is *feasible* if there exists some $\underline{t} \in \mathbb{R}$ such that $\delta_j \geq \underline{t}$ for all $j \in \mathcal{J}$, and a collection of demands $\mathcal{D}$ is *feasible* if there exists a feasible schedule for $\mathcal{D}$.

If $\mathcal{D}$ is a finite collection of demands, then there always exists a feasible schedule since departure times may be scheduled as early as needed to satisfy capacity constraints and desired arrival times. However, it is still desirable to optimize the schedule so that departure times are as late as possible subject to these constraints; this is the topic of Section IV. When $\mathcal{D}$ is an infinite collection of demands, then $\mathcal{D}$ may or may not be feasible. We assume that the number of deadlines set in the time interval $[\tau, \tau + T]$ will be finite if $T < \infty$ and the average number of deadlines in $[\tau, \tau + T]$ remains constant for $T \to \infty$, for any $\tau \in \mathbb{R}$. Fundamental limitations on the feasibility of $\mathcal{D}$ is the subject of the following Section III.

III. Fundamental Limits

In this section, we present fundamental limits for scheduling demands in a UAM network. We first consider a star-branch network with no intermediate nodes, i.e., a star network. We call this a *local network* because it can be interpreted as a local portion of a larger network where we study only incoming flights to a particular node. In this case, $\mathcal{V} = \{v_0, v_1 \ldots, v_L\}$ and there are L edges $\mathcal{E} = \{(v_i, v_0) \mid 1 \leq i \leq L\}$. Hence each journey will only consist of one link. We label edge (v_i, v_0) simply as edge i. We generalize our analysis to general star-branch networks in Section III-B.

Suppose a set of demands $\mathcal{D} = \{(o_j, f_j)\}_{j\in\mathcal{J}}$ is given for the local network, then a schedule for $\mathcal{D}$ will be on the form $\mathcal{S} = (e_j, \delta_j)_{j\in\mathcal{J}}$. This since in the local network, each journey will only consist of one link, i.e., $e_j = i$ if and only if $o_j = v_i$, $i \in \{1, \ldots, L\}$.

We introduce a cumulative departure function in time interval $[t_1, t_2]$ as $\Delta_v(t_1, t_2) : \mathbb{R}^2 \to \mathbb{N}$ for all $v \in \mathcal{V}\backslash\{v_0\}$ so that $\Delta_v(t_1, t_2)$ is the number of UAVs departing from node v in the time interval $[t_1, t_2]$, i.e.,

$$\Delta_v(t_1, t_2) = |\{j \in \mathcal{J} \mid o_j = v \text{ and } \delta_j \in [t_1, t_2]\}| \,.$$

Given the cumulative departure function, we define the long-term average departure rate r_v at node $v \in \mathcal{V}\backslash\{v_0\}$ as

$$r_v := \lim_{T\to+\infty} \frac{1}{T} \Delta_v(\tau, \tau + T) \,, \quad \forall\tau \in \mathbb{R}.$$

In the same manner, we introduce a cumulative arrival function $\Gamma_v(t_1, t_2) : \mathbb{R}^2 \to \mathbb{N}$ as the cumulative number of UAVs that depart from origin v and arrive at the destination in the time interval $[t_1, t_2]$, i.e., $\Gamma_v(t_1, t_2) = |\{j \in \mathcal{J} \mid o_j = v \text{ and vehicle arrives in } [t_1, t_2]\}|$.

For any schedule, the average departure and the arrival rates must be equal, as stated next.

Lemma 1: Consider a local UAM network $\mathcal{N}$ with a set of demands $\mathcal{D}$. For any feasible schedule $\mathcal{S}$ and for all nodes $v \in \mathcal{V} \setminus \{v_0\}$, it holds that the average arrival rate from the node equals the average departure rate from the node, i.e.,

$$r_v = \lim_{T\to+\infty} \frac{1}{T} \Gamma_v(\tau, \tau+T) \,, \quad \forall v \in \mathcal{V}\backslash\{v_0\} \,, \forall\tau \in \mathbb{R} \,. \tag{2}$$

The proof of Lemma 1 is straightforward follows directly from the conservation of the total number of UAVs in the network. In the same manner, it follows that for any schedule, the arrival rate must satisfy

$$r_v = \lim_{T\to+\infty} \frac{1}{T} \sum_{j\in\mathcal{J}|o_j=v} \mathbf{1}\left(f_j; [\tau, \tau+T]\right), \quad \forall v \in \mathcal{V}\setminus\{v_0\}, \forall \tau \in \mathbb{R}. \tag{3}$$

A. Necessary and Sufficient Condition for a Feasible Schedule in a Local Network

In the following theorem, we obtain a necessary and sufficient condition for the existence of a feasible schedule for a local network when the set of demands $\mathcal{D}$ is infinitely large, so that we can say immediately if there exists any feasible schedule. This will hence provide fundamental limits for how large demands a local network can handle.

Theorem 1: Consider a local UAM network $\mathcal{N}$ with $C_{v_0} \geq 1$. An infinite set of demands $\mathcal{D}$ is feasible if and only if

$$\sum_{1\leq i\leq L} r_{v_i} \cdot (\overline{x}_i - \underline{x}_i + w) \leq C_{v_0}, \tag{4}$$

where r_{v_i} is as given in (2) for all $v_i \in \mathcal{V}\setminus\{v_0\}$[1].

Theorem 1 is based on the two lemmas below, which focus on the special case when $C_{v_0} = 1$. We can then extend this special case to prove Theorem 1.

Lemma 2: Consider a local UAM network $\mathcal{N}$ with $C_{v_0} = 1$. If an infinite set of demands $\mathcal{D}$ is feasible, then

$$\sum_{1\leq i\leq L} r_{v_i} \cdot (\overline{x}_i - \underline{x}_i + w) \leq 1, \tag{5}$$

where r_{v_i} is as given in (2) for all $v_i \in \mathcal{V}\setminus\{v_0\}$.

Lemma 3: Consider a local UAM network $\mathcal{N}$ with $C_{v_0} = 1$ and a countably infinite set of demands $\mathcal{D} = \{(o_j, f_j)\}_{j\in\mathcal{J}}$ with $f_j > t_0$ for all $j \in \mathcal{J} = \mathbb{N}$. If $\sum_{1\leq i\leq L} r_{v_i} \cdot (\overline{x}_i - \underline{x}_i + w) \leq 1$, then there always exists $\overline{M} \in \mathbb{R}$ such that

$$\sum_{1\leq i\leq L} \Gamma_{v_i}(t_0, t_0+T) \cdot (\overline{x}_i - \underline{x}_i + w) - f_T \leq \overline{M} \tag{6}$$

for any $T > 0$, where $f_T = \max_{\{j\in\mathcal{J}|f_j\leq T\}}\{f_j\}$ is the last deadline that needs to be achieved before T. In particular, this implies that the set of demands $\mathcal{D}$ is feasible.

B. Extension to General Star-Branch Network

We now show that the necessary condition of Theorem 1 extends to general star-branch networks with intermediate nodes between origins and the destination in the following corollary.

Corollary 1: Consider a UAM network $\mathcal{N} = (\mathcal{G}_*, C, \underline{x}, \overline{x})$ where G_* is a star-branch graph with L leaf nodes and thus L branches, $(\{e_{l,k}\}_{k=1}^{K_l+1}, \{v_{l,k}\}_{k=0}^{K_l+1})$, for all $l = 1, \ldots, L$. If a countably infinite set of demands $\mathcal{D}$ is feasible, then

$$r_{v_l} \leq \hat{r}_{v_l} = \min_{1\leq k_I\leq K_l+1} \hat{r}_{l,k_I}, \quad \forall 1 \leq l \leq L \tag{7}$$

where $\hat{r}_{l,k_I}$ is obtained by the equation below:

$$C_{v_{l,k_I}} = \Big(\sum_{k=1}^{k_I} \overline{x}_{e_{l,k}} - \sum_{k=1}^{k_I} \underline{x}_{e_{l,k}} + w_{l,k_I}\Big) \cdot \hat{r}_{l,k_I}. \tag{8}$$

The observation that the above theoretical guarantees extend to the class of star-branch networks significantly expands the practical usefulness of these results. This is because, in many practical scenarios, a UAM network is reasonably decomposed as several independent star-branch networks. For example, during the morning commute, UAVs will travel to a central city from regional exurbs with possible stops at intermediate suburbs. This is the situation studied in the case study below. During the evening commute, reverse travel from the city to the exurbs is modeled as independent star-branch networks, each with a single branch.

IV. Optimization Problem

When the set of demands $\mathcal{D}$ is finite, as mentioned above in Section II, it is always possible to find a feasible schedule. However, some feasible schedules are less desirable if, e.g., they require the UAVs to depart too early so that the UAVs arrive too far ahead of the corresponding deadlines. Therefore, we propose an optimization program that seeks to minimize the gaps between the arrival times and the corresponding deadlines while remaining feasible in this section. We denote the order that journeys land at the destination v_0 as z, ordered according to arrival a_j, using the index $\gamma \in \{1, \ldots, |\mathcal{J}|\}$. For each $j \in \mathcal{J}$, z_γ^j is an indicator for whether the journey j is scheduled as the γ-th arrival or not: $z_\gamma^j = 1$ means that the assigned UAV is the γ-th flight entering v_0 while $z_\gamma^j = 0$ if not. In particular, consider a star-branch network $\mathcal{N}_* = (\mathcal{G}_*, C, \underline{x}, \overline{x})$ and the optimization program

$$\min_{\delta_j, a_j, z_\gamma^j} \sum_{j\in\mathcal{J}} (f_j - \delta_j)$$

$$a_j \leq f_j \qquad \forall j \in \mathcal{J} \tag{9}$$

$$a_j = \delta_j + \sum_{k=1|v_l=o_j}^{K_l+1} \overline{x}_{e_{l,k}} + K_l w_I \quad \forall j \in \mathcal{J} \tag{10}$$

$$\sum_{\gamma=1}^{|\mathcal{J}|} z_\gamma^j = 1 \qquad \forall j \in \mathcal{J} \tag{11}$$

$$\sum_{j\in\mathcal{J}} z_\gamma^j = 1 \qquad \forall 1 \leq \gamma \leq |\mathcal{J}| \tag{12}$$

$$\sum_{j\in\mathcal{J}} z_\gamma^j \cdot a_j + w \leq \sum_{j\in\mathcal{J}} z_{\gamma+C_{v_0}}^j \cdot \Big[\delta_j + \sum_{k=1|v_l=o_j}^{K_l+1} \underline{x}_{e_{l,k}} + K_l w_I\Big] \quad \forall 1 \leq \gamma \leq |\mathcal{J}| \tag{13}$$

$$\sum_{j|o_j=v_l} z_\gamma^j \cdot a_j \leq \sum_{j|o_j=v_l} z_{\gamma+1}^j \cdot (a_j - \frac{1}{\hat{r}_{v_l}}) \quad \forall 1 \leq \gamma \leq |\mathcal{J}|, \forall 1 \leq l \leq L \tag{14}$$

$$\delta_j,\, a_j \in \mathbb{R},\, z_\gamma^j \in \{0,1\},\, \forall j \in \mathcal{J},\, \forall 1 \leq \gamma \leq |\mathcal{J}|. \tag{15}$$

In the optimization problem above, the objective function minimizes the sum of the difference between the deadline

[1] Due to space limitation, the full proofs of Theorem 1, Lemma 2 and 3 and Corollary 1 are omitted.

and the departure time of each trip. Constraint (9) requires each UAV to arrive by the deadline of the corresponding trip. The constraint (10) defines the latest arrival time for the journey j as the departure time δ_j plus the longest time it may need to travel along the journey. As depicted in (11), each journey is assigned exactly one order index to enter the destination and (12) shows that each order index must be occupied by one journey. Since all journeys stay at destination v_0 for a fixed time w, we implicitly assign the parking slots to the journeys by their order index γ cyclically. The constraint (13) says for each parking slot, the earliest time that the UAV $(\gamma + C_{v_0})$ can enter the parking slot must be later than the latest time that UAV γ leaves the slot (the latest time that a UAV leaves the slot is the latest time that the UAV will arrive at the slot plus the time it will stay at the node). Similarly, the constraint (14) guarantees the rate that the UAVs entering each branch l will not exceed its maximal allowed average departure rate $\hat{r}_{v_l}$. Lastly, (15) provides the range for all the variables. Notice that time 0 is arbitrarily fixed so that, in particular, $\delta_j < 0$ and $a_j < 0$ are possible.

Even though the constraints (13) and (14) contain the multiplication of two decision variables, we can reformulate them into a set of affine constraints because z_γ^j is binary and a_j is bounded for any $1 \leq \gamma \leq \mathcal{J}$ and $j \in \mathcal{J}$. Hence, the problem can be transformed into a mixed-integer linear program (MILP). Although the optimization problem after transformation is still non-convex, it can be efficiently solved using solvers such as Gurobi [20] or Cplex [21].

V. Numerical Case Studies

A recent report by INRIX suggests that a UAM local network traveling to the city of Atlanta from three exurbs, Alpharetta, Kennesaw and Buford, has the potential to offer significant time savings compared to ground transportation during peak travel times [6], and we construct our case study from data reported in [6]. Table I reports plausible travel time intervals for UAVs to travel from the three exurbs to Atlanta and the corresponding time savings versus peak hours. The fourth column of Table I is taken from [6], and the third column is inferred from the fourth column and free flow travel times obtained via Google Maps. We further assume there is an intermediate vertistop a at a suburb between Kennesaw and Atlanta, and intermediate vertistops b and c at suburbs between Buford and Atlanta, as shown in Fig. 1. The corresponding travel time intervals are labeled beside the links. Conforming to the local network model in Section III, we take Atlanta as node number 0 and the exurbs as numbered in Table I. This is therefore a star-branch network with 3 branches, where $v_{2,1} = a$, $v_{3,1} = b$ and $v_{3,2} = c$.

We now present the case study with demand sets $\mathcal{D} = \{(o_j, f_j)\}_{j \in \mathcal{J}}$ defined subsequently. Note that the destination for all demands is Atlanta, node v_0. We consider a time horizon of three hours ($T = 180$ minutes) and assume there are two landing spots in Atlanta, but only one landing spot at vertistops a, b and c, i.e., $C_{v_0} = 2$, $C_a = C_b = C_c = 1$. Each UAV stays at the intermediate vertistops along its path for

TABLE I

Time Needed for UAVs and Cars to Travel to Atlanta from Different Origins in Rush Hours

Node	Origin	UAV Travel Time (min)	UAV Savings Versus Car Travel Time (min)
1	Alpharetta	$[\underline{x}_1, \overline{x}_1] = [20, 29]$	-41
2	Kennesaw	$[\underline{x}_2, \overline{x}_2] = [25, 32]$	-19
3	Buford	$[\underline{x}_3, \overline{x}_3] = [31, 41]$	-32

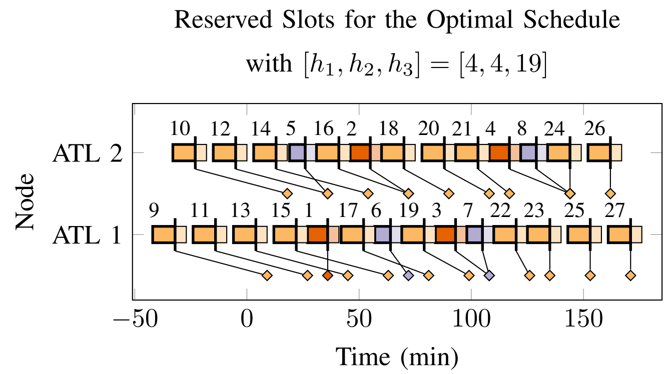

Fig. 2. The detailed optimal schedule of the second case study with $[h_1, h_2, h_3] = [4, 4, 19]$. The red, blue and orange bars represent the reserved time slots for UAVs from v_1, v_2 and v_3, respectively. We label an "ID" above each bar to track the UAVs. The diamond represents the arrival deadline of a journey; the solid-color bar represents the time interval that the UAV is scheduled to arrive. The UAV will then stay at the node for time w_I (at intermediate node) or w (at v_0) after arrival (represented by the lightly shaded bar).

$w_I = 1$ minute and at the Atlanta vertiport for $w = 5$ minutes after landing. The optimization problem (15) is solved in MATLAB using Gurobi with YALMIP toolbox to obtain a schedule $\mathcal{S} = \{(\{e_{l,k}\}_{k=1}^{K_l+1}, \delta_j)\}_{j \in \mathcal{J}}$ where $l \in \{1, 2, 3\}$ and $K_l = l - 1$. By solving the optimization problem (15), we minimize $\sum_{j \in \mathcal{J}} (f_j - \delta_j)$, the sum of the difference between the deadline and the departure time of each trip.

In the case study, we assume the number of UAVs that need to arrive at Atlanta during the three hour horizon varies across the three origins and is denoted h_1, h_2, h_3. Deadlines are set at regularly intervals, i.e., if origin v_i is tasked with sending h_i UAVs to Atlanta, the deadlines are $\left\lfloor \frac{180}{1+h_i} \cdot k + 0.5 \right\rfloor$, $k = 1, 2, \ldots, h_i$. We set the deadlines as integers for a shorter convergence time when running the algorithm. We limit the total number of UAVs under consideration, since the complexity of the non-convex optimization problem in Section IV will lead to a long computational time when there are too many UAVs. The example we present below takes around two hours for scheduling. In addition to considering a fixed three hour time window, we also consider the case when the demands are repeated indefinitely, providing a countably infinite set of demands.

Fig. 2 shows the optimal schedule computed as in Section IV for $[h_1, h_2, h_3] = [4, 4, 19]$. The "ATL 1" and "ATL 2" nodes represents the two parking slots at Atlanta node (v_0). The red bars indicate UAVs from v_1, blue for v_2 and orange for v_3. For illustrative purposes, we denote each UAV with a unique ID index, which is labeled above the bar of the

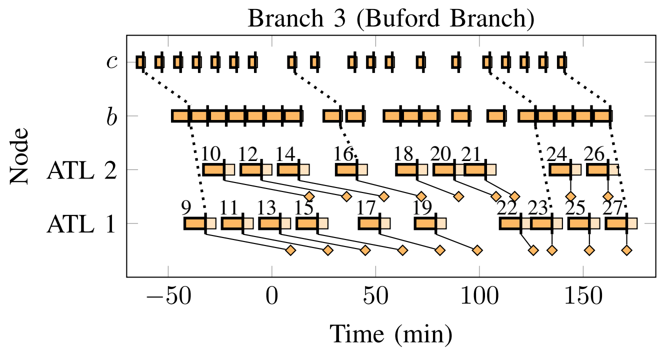

Fig. 3. The detailed optimal schedule along branch 3. UAVs traveling along branch 3 stop at intermediate nodes b, c and v_0. The dotted lines are showing that connected bars are the schedules of the same journey at different intermediate nodes and destination.

corresponding schedule. In this case, $\sum_{l=1}^{3} \frac{h_l}{180} \cdot (\overline{x}_{br_l} - \underline{x}_{br_l} + w) > 2$ so that, if we consider an infinitely repeating demand, Theorem 1 implies that this infinite set of demands is not feasible. As we can verify from the figure, no matter how we adjust the schedule while satisfying all the constraints (9)–(15), we require more parking slots or more than 180 minutes to fit in all schedules for all the journeys. As a result, when we repeat the demand in $[0, T]$ periodically, there does not exist any feasible schedule for the infinitely repeating demand. Fig. 3 shows the schedules of the UAVs along branch 3.

On the other hand, with the same total number of UAVs, if the number of departing UAVs is instead $[h_1, h_2, h_3] = [4, 19, 4]$, then $\sum_{l=1}^{3} \frac{h_l}{180} \cdot (\overline{x}_{br_l} - \underline{x}_{br_l} + w) < 2$, satisfying the necessary condition for feasibility, and it can be verified that a feasible schedule (not shown) is obtained from our proposed scheduling algorithm in this case.

VI. Conclusions

In this paper, we studied the problem of scheduling in UAM networks with uncertain travel time. One main challenge is that nodes in a UAM network, unlike in a ground transportation network, have limited parking spaces. As a result, a schedule for each UAV in the network has to be made before it takes off to ensure that a parking space is available upon arrival. We incorporated these challenges as constraints in a UAM network scheduling model for a class of star-branch networks consisting of multiple origins, a destination, and intermediate nodes between the origins and destination. We then developed necessary and sufficient conditions for a feasible schedule to exist in the special case of local star networks with no intermediate nodes between origins and the destination, and we extended the necessary condition to the general class of star-branch networks. Further, for general star-branch networks, we presented a mixed integer program to obtain an optimal and feasible schedule. We demonstrate these theoretical and computational results on a numerical case study for a UAM network in the city of Atlanta.

In future work, we plan to further generalize the class of networks we consider, allowing for possibly multiple routes between origins and the destination. We also plan to investigate how dynamic scheduling can be incorporated in the model to allow replanning in the event of unforeseen disruptions such as one or more UAVs needing to reroute or land due to, e.g., adverse weather conditions.

References

[1] J. Holden and N. Goel, "Fast-forwarding to a future of on-demand urban air transportation," 2016. [Online]. Available: https://www.uber.com/elevate.pdf

[2] D. P. Thipphavong, R. Apaza, B. Barmore, V. Battiste, B. Burian, Q. Dao, M. Feary, S. Go, K. H. Goodrich, J. Homola *et al.*, "Urban air mobility airspace integration concepts and considerations," in *2018 Aviation Technology, Integration, and Operations Conference*, 2018, p. 3676.

[3] K. Balakrishnan, J. Polastre, J. Mooberry, R. Golding, and P. Sachs, "Blueprint for the sky," *The roadmap for the safe integration of autonomous aircraft. Airbus A*, vol. 3, 2018.

[4] "NextGen Independent Assessment Recommendations," *McLean, VA: The MITRE Corporation*, 2014.

[5] B. Lascara, T. Spencer, M. DeGarmo, A. Lacher, D. Maroney, and M. Guterres, "Urban air mobility landscape report: Initial examination of a new air transportation system," *McLean, VA: The MITRE Corporation*, 2018.

[6] INRIX, "Electric passenger drones could relieve housing costs and spread growth in nation's booming cities," 2019. [Online]. Available: https://inrix.com/campaigns/vtol-study/

[7] C. Al Haddad, "Identifying the factors affecting the use and adoption of urban air mobility," 2018.

[8] E. Ancel, F. M. Capristan, J. V. Foster, and R. C. Condotta, "Real-time risk assessment framework for unmanned aircraft system (UAS) traffic management (UTM)," in *17th AIAA Aviation Technology, Integration, and Operations Conference*, 2017, p. 3273.

[9] C. Bosson and T. A. Lauderdale, "Simulation evaluations of an autonomous urban air mobility network management and separation service," in *2018 Aviation Technology, Integration, and Operations Conference*, 2018, p. 3365.

[10] M. Xue, J. Rios, J. Silva, Z. Zhu, and A. K. Ishihara, "Fe3: An evaluation tool for low-altitude air traffic operations," in *2018 Aviation Technology, Integration, and Operations Conference*, 2018, p. 3848.

[11] M. A. Aiello, C. Dross, P. Rogers, L. Humphrey, and J. Hamil, "Practical application of SPARK to OpenUxAS," in *Formal Methods – The Next 30 Years*, M. H. ter Beek, A. McIver, and J. N. Oliveira, Eds. Cham: Springer International Publishing, 2019, pp. 751–761.

[12] V. Bulusu, R. Sengupta, E. R. Mueller, and M. Xue, "A throughput based capacity metric for low-altitude airspace," in *2018 Aviation Technology, Integration, and Operations Conference*, 2018, p. 3032.

[13] M. Gendreau, G. Laporte, and R. Séguin, "Stochastic vehicle routing," *European Journal of Operational Research*, vol. 88, no. 1, pp. 3–12, 1996.

[14] B. C. Dean, "Algorithms for minimum-cost paths in time-dependent networks with waiting policies," *Networks: An International Journal*, vol. 44, no. 1, pp. 41–46, 2004.

[15] S. Samaranayake, S. Blandin, and A. Bayen, "A tractable class of algorithms for reliable routing in stochastic networks," *Procedia-Social and Behavioral Sciences*, vol. 17, pp. 341–363, 2011.

[16] P. Ioannou and F. d. A. A. Vital, "Optimizing combined truck routing and parking based on parking availability prediction," No. METRANS Project 17-01, Tech. Rep., 2018.

[17] A. Kok, E. W. Hans, J. M. Schutten, and W. H. Zijm, "A dynamic programming heuristic for vehicle routing with time-dependent travel times and required breaks," *Flexible services and manufacturing journal*, vol. 22, no. 1-2, pp. 83–108, 2010.

[18] K. Albers and F. Slomka, "Efficient feasibility analysis for real-time systems with edf scheduling," in *Design, Automation and Test in Europe*. IEEE, 2005, pp. 492–497.

[19] N. Fisher and S. Baruah, "The global feasibility and schedulability of general task models on multiprocessor platforms," in *19th Euromicro Conference on Real-Time Systems (ECRTS'07)*. IEEE, 2007, pp. 51–60.

[20] Gurobi Optimization, LLC, "Gurobi optimizer reference manual," 2020. [Online]. Available: http://www.gurobi.com

[21] IBM ILOG Cplex , "V12. 1: User's manual for cplex," *International Business Machines Corporation*, vol. 46, no. 53, p. 157, 2009.