A Lane-Based Approach for Large-Scale Strategic Conflict Management for UAS Service Suppliers

David Sacharny and Thomas C. Henderson
School of Computing
The University of Utah
Salt Lake City, Utah
Email: sacharny@cs.utah.edu

***Abstract*—This paper provides a detailed airspace capacity analysis for the Federal Aviation Administration's Unmanned Aircraft System (UAS) Traffic Management (UTM) concept of operations. Prior work has addressed aspects of this problem under specific assumptions about individual behavior of operators (human and autonomous) and the structure of the airspace, however recent discussions held between NASA and industry stakeholders indicate that cooperation will be necessary to minimize the need for tactical collision avoidance. This problem is referred to as Strategic Conflict Management and it imposes constraints on the system that can become computationally intractable. In this paper, an airspace structure inspired by roadway roundabouts, and a computationally tractable trajectory scheduling algorithm for UAS Service Suppliers (USS) are presented to solve this problem.**

I. Introduction

During a working-group discussion with the Utah Department of Transportation (UDOT), Division of Aeronautics, the prospective airspace regulator expressed a desire that all Unmanned Aerial Systems (UASs) operating in Class G uncontrolled airspace for unmanned package delivery and air-taxi services [1] should be constrained to the airspace above roads. The reasoning was two-fold: the roads are public space, and people on the road would be protected within their cars should a UAS fall to the ground. Industry stakeholders at this meeting expressed their concern that such a requirement would unnecessarily constrain and complicate their operations. Furthermore, it was not a requirement in any other region where stakeholders were considering a deployment. In this paper we analyze the relationship between airspace capacity and the structure imposed by such a requirement. A brief summary of related applications and research is presented first to demonstrate the complexity of this problem, then we reveal a lane-based design featuring roundabouts, and a capacity analysis that represents a baseline for comparison to other methods of airspace organization.

What may not have been clear for the previously mentioned industry stakeholders is that the majority of motion planning algorithms that exist today rely on some form of discretization, and the airway-over roads concept is simply one form of this. For example, some popular methods for discretization in motion planning research are cell-decomposition or probabilistic sampling such as Rapidly Exploring Random Trees (RRT) [2], [3]. The algorithms that don't rely on discretization either assume a functional representation of trajectory (e.g., a spline) or are tactical because they apply to controls directly. The decisions related to discretization are vital in determining the effectiveness and complexity of a motion planning problem. For instance, in the RRT algorithm the line connecting sampled locations must be discretely sampled to determine if any conflicts exist. If the sample resolution is too fine, then computation resources suffer. If the sample resolution is too coarse, then there is the possibility that a conflict exists that would not be discovered until it was too late.

In the layered [4] and "full mix" [5] airspace designs, UASs are afforded the maximum amount of freedom and must rely on tactical collision avoidance to maintain safe separation. Tactical collision avoidance must be computed during flight and includes a time constraint, therefore it is important to consider the time and space complexity for solving this problem. While many heuristic methods have been developed, see for example [6], there still remains the possibility that the number of conflicts may overwhelm the algorithms (see [7] for an analysis of cascading effects of conflict resolution). There has been a large amount of research into quantifying the risk of conflict in this type of system (e.g., [5], [7]–[10]), indicating that there are numerous risk factors that an operator would need to consider in order to reduce the risk of collision. Lane-based airways were analyzed in [11], however the UAS operations were not deconflicted pre-flight and instead were simulated much like car-following models (e.g., [12]). Recently, a report published by NASA detailing negotiations among stakeholders regarding requirements for USSs described the following overarching requirement for operations within the UTM system: "A UTM Operation should be free of 4-D intersection with all other known UTM Operations prior to departure and this should be known as Strategic Deconfliction within UTM" [13]. Furthermore, they discuss the requirement that any scheme for strategic deconfliction must be mandated by the airspace regulator.

Strategic deconfliction, or strategic conflict management, refers to the first of three layers of conflict management defined by the International Civil Aviation Organization (ICAO), "achieved through the airspace organization and management, demand and capacity balancing, and traffic synchronization" [14]. The next layers are applied in order of the shrinking conflict horizon, they are *tactical* in nature and termed "separation provision" and "collision avoidance." Broadly speaking, strategic conflict management deals with planning collision free paths, which in the most general case of planning for multiple agents is PSPACE-hard [3]. Even the more narrow problem of tuning velocity profiles is NP-hard [15]. In this paper we consider the simpler, but more realistic scenario given the UTM architecture, of scheduling UASs in real-time within lanes, reducing the configuration space of the UAS to a single dimension. The result is a practical, computationally tractable algorithm for strategic conflict management. The experimental section of this paper considers the capacity constraints imposed by this system, which enables airspace regulators to make informed decisions about how to address the demand from users.

A. Contributions

In this paper, we present an airspace structure inspired by roadway roundabouts, and a computationally tractable trajectory scheduling algorithm for UAS Service Suppliers (USS) within this structure. A capacity analysis follows the description of the airway structure to provide a baseline for further research. Prior research into the capacity of airspaces does not simultaneously consider the complexity of planning the operations, however both concepts must be considered together since the airspace regulator is expected to manage both (as mentioned in the introduction). The following section presents research from various application areas that are directly, or indirectly, related to collision-free trajectory planning and capacity analysis.

B. Related Work

The central problem is strategic conflict management for many aircraft (autonomous or human controlled) in a large area (e.g., 1-100 mi^2). This problem shares characteristics with many analogous application areas, as well as theoretical work in discrete mathematics (see [16] in the context of scheduling) and topology (see the chapter on configuration spaces in [2]).

1) The Air Traffic Flow Management Problem: A natural problem model comes from research into the *Air Traffic Flow Management Problem* (TFMP) [17]. In this model the airspace is partitioned into sectors that are controlled by regional regulators who provide separation services. The sectors are characterized by capacities that represent the maximum number of aircraft that may be in a sector at any time, and depend on factors such as weather. TFMP implements two control strategies to ensure that sector capacity constraints are not violated: ground-holding and speed adjustment. Ground holding shifts the entire flight in time by delaying the departure of an aircraft. Speed adjustment is applied to each sector in flight and represents an "air delay." Optimal ground-hold time and speeds for every planned flight are calculated, but each operation does not deviate spatially (this is called the *Air Traffic Flow Management Rerouting Problem* (TFMRP) [17]).

Rios and Lohn [18] compare techniques for finding a solution to the Bertsimas and Stock-Patterson (BSP) model: binary integer programming, genetic algorithms, and simulated annealing. They also compare a greedy scheduler that schedules flights on a "first-come, first-serve basis by finding the first available departure time for each flight in turn that will not violate sector capacities when combined with previously scheduled flights." The greedy scheduler is so named because it is locally optimal for the flight in question, but it does not guarantee globally optimal solutions. Solutions to the BSP model provide time intervals during which a flight must enter each segment, and the solution is guaranteed to minimize total delay. Although the size of the problem formulation is bounded by a linear relationship between the number of intervals, the number of flights, and the number of sectors, the integer linear programming formulation suggests that there is no known time-polynomial algorithm to solve it [19].

Despite its non-deterministic features, this representation is appealing because it supports the goals of strategic conflict management, namely "airspace organization" via sectors, and "demand and capacity balancing, and traffic synchronization" via ground-holding and speed adjustment. What it lacks is an explicit representation for the intersection of routes in 4-D space. The primary issue is that the sectors are large and there is no way to tell if routes intersect. One way to adapt this representation is to shrink the size of each sector such that capacity is fixed to one aircraft per sector. Bertsimas and Patterson explored this assumption and determined that the computational complexity of the TFMP is NP-hard [17]. Also, reducing the size of the sectors dramatically increases the space complexity.

2) The Job-Shop Scheduling Problem: By shrinking sector capacities to one, the TFMP can be reformulated as a scheduling problem (see [19], [20] for a definition of the general scheduling problem, and [21] and [22] for an overview of the job shop scheduling problem). Bertsimas and Patterson [17] reformulate the problem as follows: for each job create an aircraft and for each processor associate a sector (sectors include airports). Each job is composed of tasks that represent a flight segment (time spent in a sector). A solution to this formation is a total ordering of sectors for every job, and a list of flight times for each task such that the total delay is minimized, and all flights are performed by a deadline. This formulation guarantees that no aircraft will occupy the same sector at the same time and therefore satisfies the non-intersection requirement.

There are however, several practical issues with this formulation. To begin with, the job-shop scheduling problem is NP-hard – this makes it a poor choice for USSs that may need to contest with tens of thousands of "jobs." Furthermore, it is not clear what the sector size should be, given the variation of UAS sizes expected to utilize the airspace. Too large a sector could result in an unreasonable amount of tactical separation maneuvers, while too small a sector could become computationally intractable.

To account for uncertain speeds, the scheduling model can incorporate probabilistic durations. This formulation still suffers from the time complexity as before (and likely worse if the durations are not assumed to be independent random variables) [23].

3) The Multi-Robot Motion Planning Problem: Strategic deconfliction may be cast as a multi-robot motion planning problem. The key concept for any motion planning problem is the *configuration space*, which combines the kinematic constraints of the robot and the environment. Multiple robots may be combined into a conceptual "composite robot" [24] and the motions are planned in a joint configuration space. Centralized, or coupled, algorithms provide a path for every robot, while decentralized, or decoupled, algorithms usually provide solutions for a subset of the robots. In [24], the authors decompose a multiple robot planning problem into partitions of robots that are planned together. While this approach does reduce the complexity of the joint configuration space, it does not guarantee a reduced complexity of the problem because each partition can still be very complex. Other approaches, such as incremental coordination [25], combine the centralized and decentralized algorithms into a single iteration.

Multi-robot motion planning is also a more natural representation because solution methods, such as *rapidly-exploring random trees*, can incorporate dynamic constraints and uncertainty directly. The desire for optimality, however, results in a worse-case time complexity comparable to the job-shop scheduling problem.

The two-phase decoupled approach [25] involves first computing a path for each robot individually while ignoring other robots, then operations are applied to the resulting path set to avoid collisions. The advantage of this approach is that the "search space explored by the decoupled planner has lower dimensionality than the joint configuration space explored by the centralized planner" [25]. The drawback is that it is an incomplete algorithm, meaning it is not guaranteed to find a solution even if one existed by considering the system as a whole. This approach resembles the greedy, first-come, first-serve algorithm by Rios in the sense that previously planned paths are considered as static obstacles and each new flight is delayed until the capacity constraints are met [26].

4) The Traffic Assignment Problem: The traffic assignment problem (TAP) is a sub-problem in the transportation planning process that models the route-choice behavior of travelers given a set of possible routes [27]. This problem is mentioned here because prior research such as [9] measure the performance of the airway system by simulating origin-destination data from population centers. When determining the capacity of a particular network configuration, the traffic assignment problem should be considered separately because its benefit is mainly to predict the demand on the system.

Solutions to TAP result in aggregate measures, "a macroscopic description or prediction of the traffic volume" [27]. The relationship between volume of travelers and their average travel-time are modeled by *link performance* functions [27]. Queuing models also play a role in the development of link performance functions.

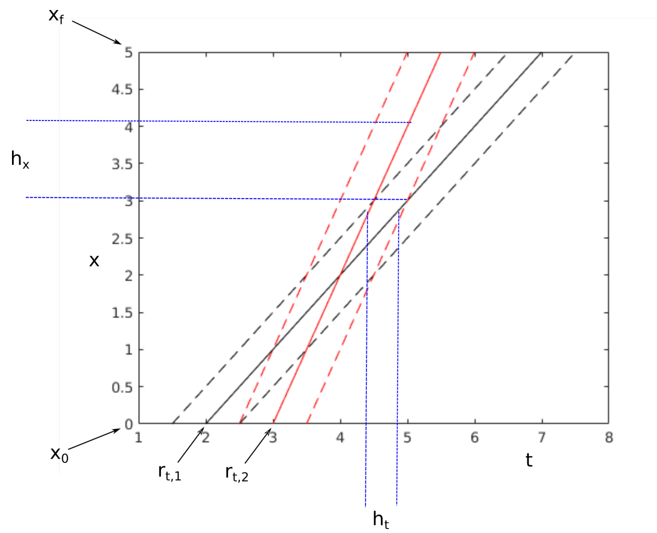

Fig. 1. Time-space diagram for two UASs in a lane. The x-xis is time and the y-axis is distance along the lane. h_t is the time headway (distance between UASs in time in lane), and h_x is the space headway (distance between UASs in lane). Note that h_t and h_x are are linearly related due to the constant speed. The two trajectories in this scenario intersect at $t = 4$ and $x = 2$, however they violate space-headway before then.

5) The Optimization Problem: The FAA expects tens of thousands of UASs to utilize the airspace in close proximity, therefore the problem model composition is important to ensure that safety requirements are met. There are two ways in general to represent the safety requirements, as a constraint and as an objective function. The objective is to maximize the separation (or headway) between UASs. Assuming the solution is optimal, the question of whether it meets the safety requirement is determined by a threshold, e.g., "the minimum separation is at least 10 meters," or "the minimum separation is at least 10 meters with 99.9% probability." In this paper we only consider the constraint model and cast the objective as a function of the time between desired release times and scheduled release times (a more complete description can be found in the section titled *Scheduling Algorithm*).

II. Lane-Based Formulation

Figure 1 shows a representative time-space diagram for two UASs in a lane; a model borrowed from ground traffic engineering. An airway lane constrains the trajectory of the UAS to the center-line of the airway, referred to as the longitudinal direction of the aircraft trajectory in prior research (e.g. [7]). The vertical and lateral directions are assumed to be under control to remain inside the lane. Uncertain altitude and lateral movements should be compensated for in the design of the width and height of the airway; this is a subject of ongoing research. Also, a constant velocity is assumed within a segment; this constraint will also be relaxed in subsequent research.

An important point is that the lane-based formulation can apply in all cases where the trajectory is reduced to a single dimension. It is not required that lanes follow the road network on the ground; lanes may be organically created by operators and reused by other operators. The critical aspect of this formulation is that there are no crossing-conflicts; this concept is described in detail in the *Airway Design* subsection below.

A. System Constraints

Under these constraints, the kinematic motion of the UASs may be described as follows:

$$uas_1 : x^1 = v_g^1(t^1 - r_t^1) \quad (1)$$
$$uas_2 : x^2 = v_g^2(t^2 - r_t^2) \quad (2)$$

where x^y is the longitudinal position (meters) within an airway segment for uas_y, v_g^y is the ground speed (meters/second), t^y is the time along the segment (seconds), and r_t^y is the release time, i.e., the time at which the UAS begins its trajectory across the segment. Also note from Figure 1 that x_0 and x_f represent the start and end of the segment, so that $x_f - x_0 = length(segment)$. The time headway (distance between UASs in time) and the space headway (distance between UASs in space, sometimes referred to as *spacing*) are given by h_t and h_x, respectively.

The error bars in Figure 1 represent the required spacing between UASs, also known as *well-clear* in the UAS literature. The vertical distance from a point on the line to the error bar is the *well-clear* and is denoted h_x^y for uas_y. Due to the linear nature of the problem, h_t^y and h_x^y are related by:

$$h_t^y = \frac{h_x^y}{v_{g,y}} \quad (3)$$

This equation mirrors the relationship between density (or occupancy in space), speed, and flow (or occupancy in time) described in the *Highway Capacity Manual* [28]. This is important because it connects the concepts of road capacity, well known in road-traffic engineering, to airway capacity, which is explored in the following sections.

The separation constraints for any two UASs may be described as follows:

$$h_x = |x^1 - x^2| > max(h_x^1, h_x^2), \forall x^y : x^y \in [x_0^y, x_f^y] \quad (4)$$
$$h_t = |t^1 - t^2| > max(h_t^1, h_t^2), \forall t \quad (5)$$

Since h_t^y and h_x^y are linearly related, it suffices to consider only one constraint. These separation constraints are more general than the one considered in [7] to describe the capacity analysis in a foundational way. UAS operators may prescribe a required headway as needed by their vehicle and other operational considerations.

Consider the case where uas_2 is already scheduled and now a USS is presented with uas_1 to schedule. Since v_g^1 is considered constant, r_t^1 (the release time for uas_1) is the only decision variable. Let $h_{t,max} = max(h_x^1, h_x^2)$ and $r_t^1 < r_t^2$; we can describe the first position at which *well-clear* is violated by the following equation,

$$x_v(v_g^1 - v_g^2) + v_g^1 v_g^2(r_t^2 - r_t^1 - h_{t,max}) = 0 \quad (6)$$

where x_v is the position along the segment where a violation first occurs. When the velocities are equal, then this equation reduces to the simple relationship,

$$r_t^1 = r_t^2 - h_{t,max} \quad (7)$$

The corresponding constraint for planning purposes is then,

$$r_t^1 < r_t^2 - h_{t,max} \quad (8)$$

This assumption of uniformity of velocities is assumed in the experimental section to make network capacity constraints more visible. In the general case, however, when $v_g^1 > v_g^2$, then x_v is negative for all $r_t^1 < r_t^2$ and therefore the only constraint is the same as Eq. 8. When $v_g^1 < v_g^2$, then the violation point may lie within the segment (this is the case in Figure 1). The constraint is therefore:

$$r_t^1 < r_t^2 - h_{t,max} - \frac{x_f}{m}, \quad m = \frac{-v_g^1 v_g^2}{v_g^1 - v_g^2} \quad (9)$$

B. Scheduling Algorithm

The algorithm that we propose for this system is a greedy scheduler (Algorithm 1):

Require: $r_d, r_e, r_l, path, v_g$
$r_d \leftarrow$ desired release time
$r_e \leftarrow$ earliest release time
$r_l \leftarrow$ latest release time
$path \leftarrow$ requested segment ids
$v_g \leftarrow$ speed
$seats \leftarrow$ available time slots
$l_s \leftarrow 0$ {The segment length}
for each $segment$ in $path$ **do**
  $seats_{segment} \leftarrow$ seats on segment at $t \in [r_e, r_l] + \frac{l_s}{v_g}$
  $seats \leftarrow seats_{segment} \mid seats$ {Binary OR}
  $l_s \leftarrow$ segment length
end for
$r_t \leftarrow$ open seat closest to r_d
return r_t

Algorithm 1: Greedy-Scheduler Algorithm

It is called "greedy" because the scheduler only considers the currently requested operation and minimizes the distance between the scheduled and desired release time. In other words it is locally optimal with respect to the desired release time. It is not globally optimal, in the sense that there may have been a better solution if all operations were considered simultaneously. In the UTM system, where operations are scheduled online and desired release times are unknown to the scheduler until the request is made, a globally optimal algorithm may not exist. To see why, this problem may be cast in terms of what Pinedo would describe as an online job-shop scheduling problem with no-wait constraints [20]. Specifically, this is an online-over-time problem because the scheduler "does not know at any point in time during the process how many more jobs are going to be released in the future and what their release dates are going to be" [20]. It is also classified as *clairvoyent* because all relevant

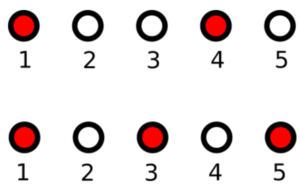

Fig. 2. Two Ways to Fill Five Seats with Spacing Equal to One

information, such as speed, are available to the scheduler. It may be possible that a USS knows when its operations will be requested, however it is still true that it will not know when another USS's operations will be requested (at least not in the currently envisioned UTM system). The no-wait constraint refers to the fact that, in the scenarios considered in this system, UASs cannot wait (park or hover) between successive segments. The problem of minimizing maximum lateness (a measure of the worst violation of due-dates), for a single machine with requested release dates (in Pinedo's nomenclature $1|r_j|L_{max}$), is NP-hard [20]. A polynomial-time online algorithm therefore represents an approximation of the optimal algorithm.

This algorithm applies equally well to homogeneous and heterogeneous velocities, however only the homogeneous setup is considered here. The heterogeneous version of this algorithm applies additional time-headway as required by the term $\frac{x_f}{m}$ in Eq. 9.

1) Occupancy and Utilization: To analyze the expected segment utilization and capacity in a real-world scenario, given Algorithm 1, we can begin by assuming that scheduling requests are probabilistic and uniformly distributed across a discretized time-interval, for example 5-second time slots over a 12 hour period. We would like to know what the expected time-occupancy (percentage of time slots filled) is once no more UASs can be scheduled without violating their time-headway requirements. This problem was solved by a number of authors, in particular by Page [29], and Freedman and Shepp [16], who contextualized the problem as *An Unfriendly Seating Arrangement*. Both problems are discrete versions of Renyi's parking problem [30]. To help visualize the problem, we analyze the greedy scheduler applied to the unfriendly seating arrangement described by Freedman and Shepp.

Consider the problem of filling five seats (representing time slots for the scheduler) in Figure 2, where individuals cannot be seated next to each other. The required headway in this instance can be considered as one seat. Let occupancy be defined as the ratio of occupied seats to total seats. As each individual arrives, one at a time, if they are seated in the remaining valid seats according to a uniform distribution, then the probability of filling 2 seats, and therefore an occupancy (the ratio of filled to unfilled seats, denoted E_n) of $\frac{2}{5}$, is $p(E_n = \frac{2}{5}) = \frac{8}{15}$. The probability of filling 3 seats, and therefore an occupancy of $\frac{3}{5}$, is $p(E_n = \frac{3}{5}) = \frac{7}{15}$. The expected occupancy is $\mathbb{E}(E_n) = 0.493$ or 49.3%.

With the greedy scheduler, filled seats tend to cluster together since it finds the closest valid seat to the desired one, resulting in a higher expected occupancy. Consider the case where the first seat is taken in Figure 2. For the uniform scheduler in the unfriendly seating arrangement problem, the remaining valid seats (3, 4, and 5) are equally likely, $p(K = 3) = p(K = 4) = p(K = 5) = \frac{1}{3}$. For the greedy scheduler, if the next desired seat is 1 or 2 or 3, then the scheduled seat is 3, i.e., $p(K = 3) = \frac{3}{5}$, $p(K = 4) = p(K = 5) = \frac{1}{5}$. The occupancy probabilities for the greedy scheduler in this case are $p(E_n = \frac{2}{5}) = \frac{12}{25}$ and $p(E_n = \frac{3}{5}) = \frac{13}{25}$. The expected occupancy is $\mathbb{E}(E_n) = 0.504$ or 50.4%.

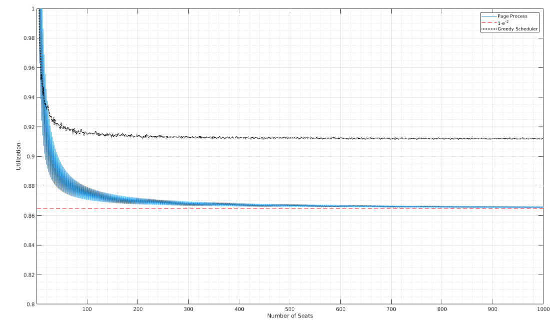

Fig. 3. Greedy Scheduler versus Uniform Scheduler (Page Process)

The expected occupancy for the uniform scheduler, as the number of seats (denoted n) approaches infinity, is approximately $\mathbb{E}(E_n) = \frac{1-e^{-2}}{2} = 0.4323$ [29]. Figure 3 shows a comparison of the Page process versus the greedy scheduler using the measure of *utilization*. Utilization takes into account the maximum possible occupancy for each value of n:

$$U_n = \frac{E_n}{E_{n,max}} \tag{10}$$

where, $E_{n,max} = \frac{ceil(n/2)}{n}$. The expected utilization for the greedy scheduler was evaluated empirically in Figure 3, where 1000 trials were run for each number of seats (from 1 to 1000), and found to be $\mathbb{E}(U_n) = 0.9122$ as $n \to \inf$.

2) Flow and Density: The scheduler assigns time slots for each route request, therefore the expected flow ($\mathbb{E}(F)$), in vehicles-per-second ($\frac{v}{s}$), is determined by,

$$\mathbb{E}(F) = \frac{\mathbb{E}(E_n)}{t_{slot}} \tag{11}$$

where t_{slot} is the sampling period in seconds, the discretization of time. For example, if $\mathbb{E}(E_n) = 0.4561$ (the expected value for the greedy scheduler) and $t_{slot} = 1.25$, then the expected flow is, $\mathbb{E}(F) = 0.365\ \frac{v}{s}$.

The expected density, in vehicles per meter ($\frac{v}{m}$) is calculated as follows,

$$\mathbb{E}(D) = \frac{\mathbb{E}(F)}{v_g} \tag{12}$$

where v_g is the homogeneous ground speed in meters per second ($\frac{m}{s}$). If we assume the ground speed is 10 $\frac{m}{s}$, then the expected density of vehicles in the previous example is, $\mathbb{E}(D) = 0.0365\ \frac{v}{m}$.

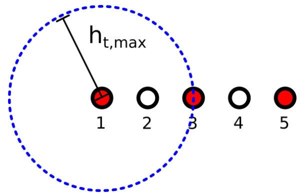

Fig. 4. Relationship Between Time-Headway and Seat Spacing

3) Discretization: The expected occupancy and utilization for the greedy scheduler shown in Figure 3 represents the expected *maximum* for those metrics assuming there is at least one "seat" in between each filled one. This assumption may be built into the discretization of time for the scheduler by assuming homogeneous time-headway and a sampling period is defined as follows,

$$t_{slot} = \frac{h_{t,max}}{2} \tag{13}$$

Figure 4 demonstrates this relationship.

4) Lane Capacity: If the greedy scheduler is employed and the discretization of time is as described in the previous section, then the expected maximum flow, and hence capacity, for a single lane is approximately,

$$\mathbb{E}(F_{max}) = \frac{0.9122}{h_{t,max}} = \frac{0.9122 v_g}{h_{x,max}} \tag{14}$$

For example, if the maximum required space-headway is $25m$ and the homogeneous ground speed is $10\frac{m}{s}$, then $\mathbb{E}(F_{max}) = 0.365$. The corresponding density is $\mathbb{E}(D_{max}) = 0.0365$. For a lane segment that stretches $100m$, at the expected maximum density there will be approximately 3.64 vehicles per lane at any given time. If the airspace regulator determined that all flights must begin within a 12-hour period each day, then a single lane may be expected to support approximately 15,768 vehicles per day.

C. Complexity

The input to the greedy scheduler are the open and filled seats for each segment along a route. If the total number of seats in a single schedule is n, and the number of segments in a route is s, then the worse-case is that the scheduler will traverse every slot in a schedule; hence the complexity is $\mathcal{O}(s+n)$. Notice that because the segment schedules (the seats) are represented as a binary string and OR'd together, as shown in Algorithm 1, the scheduler only needs to consider a single binary string of length n when looking for an open seat.

D. Airway Design

To better utilize intersections, only merging or diverging conflicts should exist because crossing conflicts require that the scheduler manage nodes as well as segments. This would add additional constraints on UASs requesting time within an intersection that would be independent otherwise. Since each segment is defined by exactly one schedule that manages UAS arrivals, organizing the airspace in this way removes the need for intersection management such as the signalized intersections in [11]. In Figure 5, the node labeled "2" is an example of a diverging conflict, where incoming traffic is split into two traffic streams [31]. The node labeled "1" is an example of a merging conflict, where two traffic streams are joined into one [31]. Crossing conflicts may be eliminated by implementing a roundabout, a concept borrowed from ground traffic engineering [31]. Figure 5 displays the graph model for a roundabout, which includes unidirectional edges between eight nodes (each node represents the endpoint of a segment) in a counter-clockwise direction.

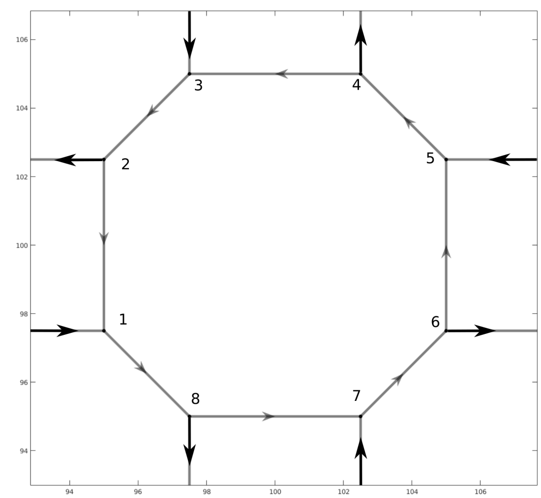

Fig. 5. Airway Roundabout

Figure 6 shows an airway model featuring several intersections, with a minimum separation of 25 meters between segment endpoints. This model may be replicated and resized to fit the underlying road network or other operational constraints.

III. Experiments

The experiments presented here serve two purposes: one is to develop an intuition regarding the relationship between airspace structure and capacity, and two is to understand the relationship between demand and reliability. To measure the capacity of the airspace structure, a large simulated demand of 20,000 scheduling requests was sampled from a uniform distribution of *land* and *launch* vertices in the network graph shown in Figure 6. *Launch* vertices are nodes where UAS may enter the network because the out-degree is exactly one. This requirement follows from the airway design described in section II-D. Likewise, *land* vertices are nodes where a UAS route may terminate because the in-degree is exactly one. The network shown in Figure 6 has a total of 48 vertices: 24 launch vertices and 24 land vertices. Figure 7 demonstrates the network capacity as it relates to the speed of vehicles, where the "mean total flow" represents the total number of vehicles scheduled during the simulation.

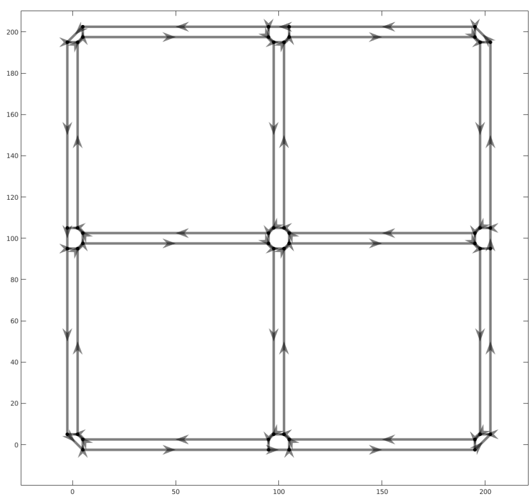

Fig. 6. Example Airway Model with Lanes

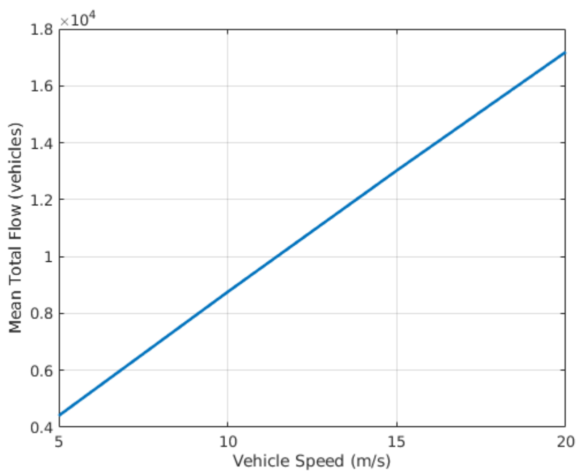

Fig. 7. Total network flow versus the speed of vehicles

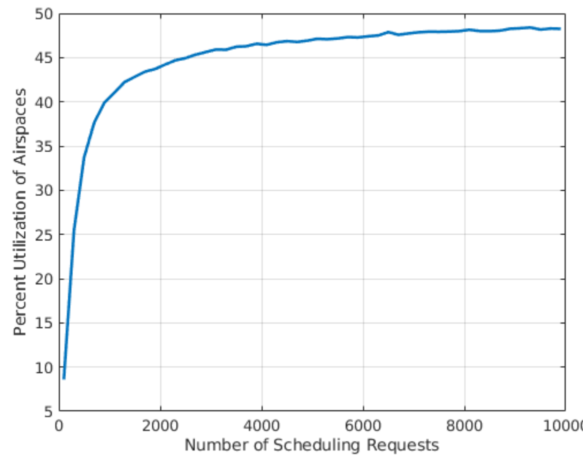

Fig. 8. Airspace utilization versus the number of scheduling requests. In this simulation the vehicle speeds were fixed to 5m/s and the headway to 25m.

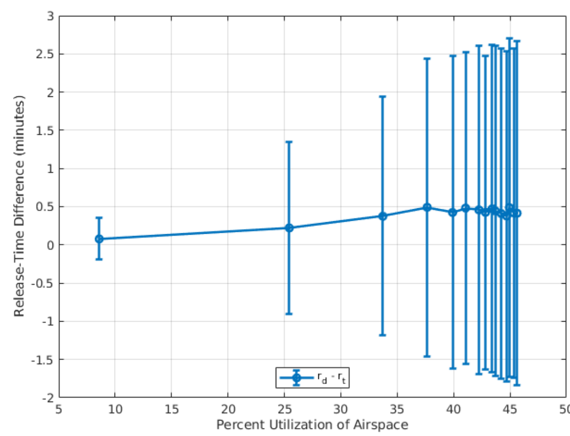

Fig. 9. Difference between desired and scheduled release time versus airspace utilization. The error bars represent the standard deviation and the round markers show the mean.

Reliability represents the variability in scheduled release times versus desired release times. To show this relationship, a simulation was run on the network in Figure 6, where a fixed speed of 5 m/s was implemented and an increasing number of scheduling requests were made. Scheduling requests represent a demand on the system and result in utilization of the airspace; Figure 8 demonstrates this relationship. The relationship between reliability and utilization is shown in Figure 9, where the mean difference between desired and scheduled release times are plotted along with error-bars representing the standard deviation in the 10 trials that were run for each data point.

IV. Discussion

The linear relationship demonstrated in Figure 7 suggests that greater capacity can be obtained by increasing the speed of vehicles on the network. This fact provides a conceptual framework for policy decisions regarding speed and capacity for airspaces. For example, it may not be possible, or desirable, to add lanes to popular corridors if the surrounding airspace is crowded; airspace regulators can consider the projected demand and operational capabilities of vehicles, then use this straightforward relationship to design fast-lanes.

The simulations presented here assumed that USS were scheduling fixed routes, i.e., they were not dynamically routing based on the current utilization of the network. As the utilization of the network grows, the probability that a series of reservations exists to accommodate a route declines. This relationship is demonstrated in Figure 8, where the utilization first grows quickly because most routes can be accommodated, but then slows as this reconciliation becomes more rare. Eventually, the utilization approaches a limit that is determined by the network structure. In this case, the value for maximum occupancy that determines the utilization was calculated by multiplying the maximum occupancy of a single segment (edge) by the number of segments in the network. The apparent 50% limit in Figure 8 stems from the bottleneck produced by the division of vertices into *land* and *launch* types. Since all traffic must pass through these nodes, they become the limiting factors.

Figure 9 demonstrates the relationship between reliability and utilization, where reliability is represented by the difference in time between the desired and scheduled release times. When the release-time difference shows high variability, such as when the network utilization is high, then a USS cannot make reliable predictions for deliveries, for example. This has repercussions for many important use-cases that are being considered for USS, such as medical delivery and emergency response. The airspace regulator should consider capping the utilization of critical corridors to ensure reliable scheduling.

V. Conclusion and Future Research

In this paper, a lane-based airspace structure and corresponding scheduling algorithm were examined in detail. A design consisting of a roundabout was also presented that enables multiple aircraft to occupy an intersection, collision-free. The metrics of capacity, utilization, and reliability are critical to consider when designing an airspace to serve a transportation demand, and the lane-based approach makes the necessary calculations palatable. The structured and organized nature of the lane-based approach is important when considering the complex requirements of users that want to deploy within the system. It enables transportation planners to reuse concepts from ground-traffic engineering, such as flow and density, as well as provide tools for determining the state of the transportation system.

Future research will include uncertainty in speed due to environmental factors and study adjustments necessary for the width and height of lanes to take into account the operational requirements of vehicles. Generally, minor variability in vehicle speeds can be compensated for by increasing headway requirements, and larger variability can be accommodated by dividing or adding lanes with speed policies.

The approach presented in this paper can also be applied to ground-based autonomous vehicles, however an assessment of the uncertainty due to non-cooperative vehicles (those not conforming to a scheduler) is needed. While the introduction included a reference to the airspace-over-roads concept, this lane-based approach functions more as a conceptual framework for designing and organizing autonomous vehicles in any situation where coordination is necessary to avoid collision.

References

[1] "Utah on its own as FAA bails out of drone regulation — Deseret News." [Online]. Available: https://www.deseretnews.com/article/900025274/utah-on-its-own-as-faa-bails-out-of-drone-regulation.html

[2] H. Choset, H. S. Lynch, Kevin M., G. Kantor, W. Burgard, L. E. Kavraki, and S. Thrun, *Principles of Robot Motion Theory, Algorithms, and Implementation*. Cambridge, Massachusetts: MIT Press, 2005.

[3] S. M. LaValle, *Planning Algorithms*. Cambridge, U.K.: Cambridge University Press, 2006.

[4] L. Sedov and V. Polishchuk, "Centralized and Distributed UTM in Layered Airspace," in *8th International Conference on Research in Air Transportation*, 2018, pp. 1–8.

[5] E. Sunil, J. Hoekstra, J. Ellerbroek, F. Bussink, D. Nieuwenhuisen, A. Vidosavljevic, and S. Kern, "Metropolis : Relating Airspace Structure and Capacity for Extreme Traffic Densities," in *11th USA/Europe Air Traffic Management Research and Development Seminar*, Lisbon, Portugal, 2015.

[6] S. Balachandran, C. Munoz, and M. C. Consiglio, "Implicitly Coordinated Detect and Avoid Capability for Safe Autonomous Operation of Small UAS," in *17th AIAA Aviation Technology, Integration, and Operations Conference*. Reston, Virginia: American Institute of Aeronautics and Astronautics, 6 2017.

[7] M. R. Jardin, "Analytical Relationships Between Conflict Counts and Air-Traffic Density," *Journal of Guidance, Control, and Dynamics*, vol. 28, no. 6, pp. 1150–1156, 2005.

[8] M. Tra, E. Sunil, J. Ellerbroek, and J. Hoekstra, "Modeling the Intrinsic Safety of Unstructured and Layered Airspace Designs," in *12th USA/Europe Air Traffic Management Research and Development Seminar*, Seattle, Washington, USA, 2017.

[9] V. Bulusu, R. Sengupta, V. Polishchuk, and L. Sedov, "Cooperative and Non-Cooperative UAS Traffic Volumes," in *2017 International Conference on Unmanned Aircraft Systems, ICUAS 2017*, Miami, FL, USA, 2017.

[10] H. Blom and G. Bakker, "Conflict probability and incrossing probability in air traffic management," in *Proceedings of the 41st IEEE Conference on Decision and Control, 2002.*, Las Vegas, NV, USA, 2002, pp. 2421–2426.

[11] D.-S. Jang, C. A. Ippolito, S. Sankararaman, and V. Stepanyan, "Concepts of Airspace Structures and System Analysis for UAS Traffic flows for Urban Areas," in *AIAA Information Systems-AIAA Infotech @ Aerospace*. Grapevine, Texas: American Institute of Aeronautics and Astronautics, 1 2017.

[12] G. Newell, "A simplified car-following theory: a lower order model," *Transportation Research Part B: Methodological*, vol. 36, no. 3, pp. 195–205, 3 2002.

[13] J. Rios, "NASA UTM Strategic Deconfliction Final Report," NASA Ames Research Center, Tech. Rep., 2018.

[14] ICAO, "Doc 9854 AN/458 - Global Air Traffic Management Operational Concept," *International Civil Aviation Organization*, p. 82, 2005.

[15] D. Alejo, J. M. Díaz-Báñez, J. A. Cobano, P. Pérez-Lantero, and A. Ollero, "The velocity assignment problem for conflict resolution with multiple aerial vehicles sharing airspace," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 69, no. 1-4, pp. 331–346, 1 2013.

[16] H. D. Friedman, D. Rothman, and J. K. Mackenzie, "Solution to: An Unfriendly Seating Arrangement (Problem 62-3)," *SIAM Review*, vol. 6, no. 2, pp. 180–182, 1964.

[17] D. Bertsimas and S. S. Patterson, "The Air Traffic Flow Management Problem with Enroute Capacities," *Operations Research*, vol. 46, no. 3, pp. 406–422, 6 1998.

[18] J. Rios and J. Lohn, "A Comparison of Optimization Approaches for Nationwide Traffic Flow Management," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*. Reston, Virigina: American Institute of Aeronautics and Astronautics, 8 2009.

[19] C. H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*. Mineola, New York: Dover Publications, 1988.

[20] M. L. Pinedo, *Scheduling: Theory, algorithms, and systems*, 5th ed. New York, NY, USA: Springer International Publishing, 2016.

[21] A. Arisha, P. Young, and M. El Baradie, "Job Shop Scheduling Problem: an Overview," in *International Conference for Flexible Automation and Intelligent Manufacturing (FAIM 01)*, Dublin, Ireland, 2001, pp. 682–693.

[22] A. Jones, L. C. Rabelo, and A. T. Sharawi, "Survey of Job Shop Scheduling Techniques," in *Wiley Encyclopedia of Electrical and Electronics Engineering*. Hoboken, NJ, USA: John Wiley & Sons, Inc., 12 1999.

[23] J. C. Beck and N. Wilson, "Proactive Algorithms for Scheduling with Probabilistic Durations," in *IJCAI International Joint Conference on Artificial Intelligence*, vol. 28, 2005, pp. 1201–1206.

[24] J. van den Berg, J. Snoeyink, M. Lin, and D. Manocha, "Centralized Path Planning for Multiple Robots: Optimal Decoupling Into Sequential Plans," in *Robotics: Science and Systems V*, 2009.

[25] M. Saha and P. Isto, "Multi-Robot Motion Planning by Incremental Coordination," in *IEEE International Conference on Intelligent Robots and Systems*, 2006, pp. 5960–5963.

[26] J. Rios and K. Ross, "Delay Optimization for Airspace Capacity Management with Runtime and Equity Considerations," in *AIAA Guidance, Navigation and Control Conference and Exhibit*, 2007, p. 10.

[27] M. Patriksson, *The Traffic Assignment Problem: Models and Methods*. Dover Publications, 2015.

[28] National Research Council (U.S.). Transportation Research Board., *Highway capacity manual.* Washington, D.C.: Transportation Research Board, National Research Council, 2000.

[29] E. S. Page, "The Distribution of Vacancies on a Line," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 21, no. 2, pp. 364–374, 1959.

[30] A. Rényi, "On a one-dimensional problem concerning random space filling." *Publications of the Mathematical Institute of the Hungarian Academy of Sciences,*, vol. 3, pp. 109–127, 1958.

[31] E. National Academies of Sciences and Medicine, *Roundabouts: An Informational Guide*, 2nd ed. Washington, D.C.: Transportation Research Board, 12 2016.