A Distributed Framework for Traffic Flow Management in the Presence of Unmanned Aircraft

Hamsa Balakrishnan and Bala Chandran
Resilient Ops, Inc.
Winchester, MA, USA

***Abstract*—The integration of unmanned aircraft systems (UAS) into the airspace system is a key challenge facing air traffic management today. An important aspect of this challenge is how to determine and manage 4-dimensional trajectories for both manned and unmanned aircraft, and how to appropriately allocate resources among different aircraft. An integrated approach requires solving the traditional Air Traffic Flow Management (ATFM) problem to balance the capacity and demand of airport and airspace resources, but at a significantly larger scale. In doing so, aircraft connectivity constraints of commercial flights must be satisfied. In addition to these and the resource capacity constraints, geofencing constraints for unmanned aircraft that keep them within or outside a certain region of the airspace, must also be incorporated.**

This paper presents a distributed implementation of an integer programming approach for solving large-scale ATFM problems in the presence of unmanned aircraft. Given desired mission plans and flight-specific operating and delay costs, the proposed approach uses column generation to determine optimal trajectories in space and time, in the presence of network and flight connectivity constraints, airport and airspace capacity constraints, and geofencing constraints. Using projected demand for the year 2030 from the United States with approximately 48,000 passenger flights and 29,000 UAS operations (on a wide range of missions) per day, we show that our implementation can find nearly-optimal trajectories for a 24-hour period in less than 4 minutes. Furthermore, a rolling horizon implementation (with 6-8 hour time windows) results in run times of less than a minute. In addition to being the largest instances of the ATFM problem solved to date, these results represent the first effort to incorporate UAS trajectories into airspace and airport resource sharing problems.

Keywords- Air Traffic Flow Management; Trajectory-Based Operations; Unmanned Aircraft Systems; Network Optimization

I. Introduction

High demand for airspace and airport resources, combined with reduced capacity during weather disruptions, result in large air traffic delays and significant economic and environmental impacts. [1] estimated that in 2007, domestic flight delays cost the United States' economy $41 billion, increased airline operating costs by $19 billion, and consumed nearly 740 million gallons of fuel. Congestion due to high traffic volume and non-extreme weather conditions are responsible for 25% of all delayed flights [2]. In the same period, enroute air traffic flow management delays in Europe cost an estimated €1.3 billion, while route extensions cost an additional €2.4 billion [3].

Air Traffic Flow Management (ATFM) is a process that strategically allocates capacity-constrained resources by assigning delays to aircraft (either on the ground or in the air), rerouting them, or by canceling them if necessary. While ATFM algorithms traditionally focused on improving the efficiency of the system by reducing delays, there is an increased interest in ensuring the fairness of the resultant solutions, the incentives for airlines to participate and report their information truthfully, and the ability of airlines to keep their proprietary information (for example, flight-specific revenue and delay cost functions) private.

Unmanned systems hold tremendous promise in applications ranging from search and rescue to agriculture, pipeline inspection, firefighting, and freight delivery. The unmanned aircraft market in the US is estimated to be of approximately $35 billion in value by the year 2035, and to result in nearly 250,000 jobs [4]. However, systems and procedures need to be developed that facilitate the integration of remotely-piloted or unmanned aerial systems (RPAS/UAS) into the airspace without negatively impacting an already capacity-constrained system. The wide array of UAS mission types in addition to the large volume of operations requires a fundamental rethinking of how aircraft trajectories are planned and managed, while maintaining safety, efficiency, and an equitable distribution of airspace resources.

Trajectory-based operations (TBO) have the potential to greatly improve the efficiency of the ATM system. The underlying concept is the use of a 4D trajectory (a set of points in space-time) in order to describe the most likely path of an aircraft. This trajectory can then be used to coordinate decisions between different facilities, agents, and time-scales. Prior literature on technology as well as ATM policy has typically focused on how to ensure tactical safety once strategic aircraft trajectories have been determined [5, 6]. However, to the best of our knowledge, few studies have focused on how the trajectories could be determined and managed, and resources appropriately allocated, across a large number of manned and unmanned aircraft. The question of how to strategically generate trajectories for a large number of aircraft, and manage airspace resources in the presence of manned and unmanned aircraft, is the focus of this paper.

The proposed system, named DRIFT (Distributed Resilient Framework for TBO), is a distributed system that consists of algorithms and an information-sharing framework that would enable autonomous trajectory planning by manned and unmanned aircraft, while optimizing system-wide objectives such as safety, efficiency, and equity. DRIFT provides a framework under which aircraft and Air Traffic Control (ATC) can iteratively exchange trajectory intent and congestion feedback to develop trajectories that are efficient and equitable, while preserving an aircraft's autonomy in generating its own trajectories based on its internal objective tradeoffs.

A. Design principles

We propose an framework for the large-scale generation and management of trajectories for both manned aircraft and UAS, guided by the following key design principles:

1) *Autonomy:* In order to enable each aircraft to autonomously determine its own trajectory based on its internal time/cost tradeoffs, DRIFT deploys a distributed architecture in which the role of the centralized authority (for example, the ANSP) is to facilitate, and not dictate, the efficient use of system resources. The framework also allows for flexibility in the level of autonomy. For example, in the short-term (as in the current system), the distributed implementation will primarily lead to a computational benefit, by enabling fast run times. In the medium-term, the autonomy may be at the level of the air carriers, in which each airline (or UAS operator) will compute and transmit a set of desired trajectories to the ANSP. Finally, in the long-term, full autonomy may be achieved, and individual aircraft will determine their own trajectories based on their internal cost-revenue tradeoffs.
2) *Safety through constraint satisfaction:* ATM operations are subject to a large range of constraints, including airspace sector and airport capacity constraints, and in the case of UAS, geofencing constraints that regulate access to certain areas of the airspace at certain times. Depending on weather, workload, and other factors, these constraints are likely to be dynamic, that is, vary with time. The proposed algorithms therefore need to ensure that aggregate flow/density constraints are satisfied at various system resources at all times.
3) *Efficiency:* While allowing aircraft operators to autonomously determine trajectories based on their internal tradeoffs, it is desirable to maximize resource utilization, and to generate global outcomes that optimize system-wide metrics.
4) *Scalability:* The National Airspace System (NAS) in the US currently serves approximately 29,000 commercial flights (air carriers and air taxi operations) per day [7], which determines the number of trajectories that need to be determined. However, the projected growth in passenger demand and the introduction of UAS are expected to result in a dramatic increase in the number of flights, to nearly 80,000 flights per day, by the year 2030 [8, 9]. In addition, the dynamic and unpredictable nature of UAS demand implies that the underlying optimization problems may need to be solved quickly in order to be able to re-plan when new demand arises, or when conditions change. The proposed DRIFT architecture and algorithms therefore need to scale to solve the very large-scale optimization problems in a computationally tractable manner.
5) *Robustness to information inaccuracies and uncertainties:* As in any distributed architecture in which different agents exchange information, communication delays and uncertainty may cause inaccurate or obsolete information to be transmitted. It is therefore desirable for the proposed framework to be robust to reasonable levels of imperfect information and trajectory uncertainty.
6) *Equity and incentives:* Since flights are operated by a number of aircraft operators with competing interests, we are interested in algorithms that can ensure an equitable distribution of resources among the different users. While the scope of this paper is restricted to the optimization framework, future research will address the design of mechanisms that can enable equity, create incentives for participation and truthful reporting of information by aircraft operators, and monitor and enforce conformance.

B. Contributions of this paper

Capacity-demand imbalances in the current ATM environment are addressed by ATFM processes. The TBO concept helps synchronize resource management and coordination across multiple spatial and temporal scales by determining trajectories in space and time for all flights. In the absence of UAS, approximately 20,000-30,000 trajectories per day would need to be managed by ATFM in the US, under current demand levels. In addition, air carrier service in the US currently serves more than 370 airports, and the airspace is divided into nearly 400 high-altitude sectors. However, the two largest instances of the ATFM problem that have been solved to date have involved 6,745 flights, 30 airports and 145 sectors for an 8-hr window with 15-min trajectory discretization (Bertsimas et al., 2011: run-time of 10 min [10]), and, more recently, 17,500 flights, 370 airports and 375 sectors for a 24-hr window with 5-min trajectory discretization (Balakrishnan and Chandran, 2014: run-time of 5 min [11]). However, the introduction of UAS into the ATM environment will dramatically increase the number of flights [8, 9]. An TBO framework that manages trajectories in an integrated manned would therefore need to be able to manage a very large number of trajectories in a computationally tractable manner, allow aircraft operators (including UAS operators) to autonomously generate trajectories based on their internal tradeoffs, and allow diverse classes of aircraft with differing constraints to share airspace and airport resources.

This paper presents a distributed implementation of an integer programming approach that was recently proposed for large-scale ATFM [11]. While the prior implementation was on a laptop with 8 virtual cores, this paper demonstrates that the approach scales to deployment on the cloud (AWS). We show that this approach can be used to incorporate geofencing constraints that are required for the management of UAS trajectories in a TBO environment. Using datasets of projected operations in 2030 and real-world sector definitions, we show that our framework can determine trajectories to within 0.1%

of optimal for 77,000 flights (manned and UAS), with 2,400 airports and 955 airspace sectors, for a 24-hour window with 1-min trajectory discretization, with a computational time of less than 4 min. We also show that this framework can seamlessly incorporate UAS and associated geofencing constraints, and facilitate efficient resource sharing, while allowing aircraft operators to autonomously determine their trajectories. In addition to being the largest instances of the ATFM problem solved to date, these results represent the first effort to incorporate UAS trajectories into airspace and airport resource sharing problems.

II. Integrating UAS into ATFM

Prior studies on UAS integration into the airspace have primarily focused on tactical collision detection and avoidance, once strategic trajectories have been determined [5, 6, 12, 13]. The strategic handling of imbalances between capacity and demand for airspace resources has been the purview of Air Traffic Flow Management (ATFM) algorithms. Starting with the first mathematical formulation of the flow management problem [14], there has been much research in developing computational techniques to address this problem [15, 16, 17, 10, 18, 19, 20]. However, the ability to solve truly nation-scale (in the case of the United States) or continent-scale (in the case of Europe) instances has been a long-standing challenge. We recently proposed a parallelizable, column generation algorithm to determine optimal trajectories in the presence of network and flight connectivity constraints, as well as airport and airspace capacity constraints [11]. The proposed approach was shown to be fast enough to solve realistic nation-scale instances of 24-hour duration (with ∼17,500 flights, 370 airports and 375 sectors, at a time-discretization of 5 minutes) in less than 5 min. The focus of this paper is to understand whether this method can scale to handle the significantly larger challenge of including unmanned operations and the constraints associated with them (e.g., geofencing), while still handling the constraints of the traditional ATFM problem.

A. Notation

We represent the airspace system as a node-link network along which aircraft are routed. The network model consists of the following components (Fig. 1):

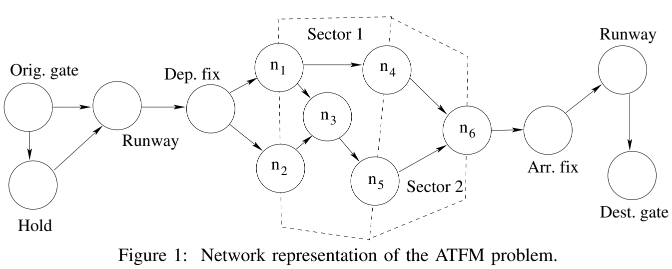

Figure 1: Network representation of the ATFM problem.

1) *Node:* A node can be either a physical location corresponding to a region in the airspace, or a decision point (e.g., hold at gate vs. pushback for taxi).
2) *Arc:* An arc is a directed segment that connects two nodes. It is associated with a minimum transit time (> 0), a maximum transit time, and a cost as a function of the transit time.

Given this network, a *trajectory* or "4D-trajectory" is a sequence of node-time combinations that represent the path of an aircraft. It implicitly specifies the arcs along the path and the transit times on those arcs. In traveling from an origin airport to a destination, a flight traverses through several *sectors*, or contiguous regions of airspace. An arc in the network is required to be fully contained within a sector: a sector is therefore a collection of arcs. Nodes at the sector boundaries ensure that arcs do not cross sectors. A node could however be present inside a sector. A single physical aircraft (or *tail*) can operate several consecutive flights during a time-horizon. Therefore, for a given tail, the destination airport of a flight must be the origin airport of the successor/connecting flight. Each flight is in turn associated with a set of arcs that form the network along which that aircraft can be routed from its origin to destination. The parameters of the arcs (minimum and maximum transit times and cost) can be flight-dependent.

B. Operational constraints on trajectories

In determining the optimal 4D-trajectories, there are a range of operational constraints that need to be satisfied. These constraints may differ depending on the class of aircraft (for example, manned or unmanned), and also by type of aircraft (for example, the desired altitude of the UAS mission).

1) Airport and airspace capacity constraints: These constraints represent limits on flows in the air traffic network.

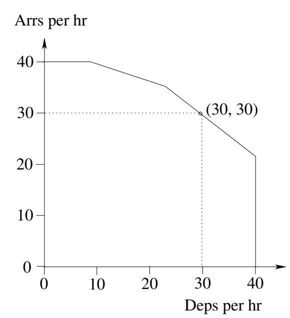

Figure 2: An airport runway capacity envelope with four segments.

Airport capacity constraints limit the flow through an airport at any time. The airport throughput constraint at any time is represented by a capacity envelope composed of segments that specify the tradeoff between arrival and departure operations. For example, the envelope shown in Figure 2 stipulates that if the runway is in a "departures-only" or "arrivals-only" operating mode, the limit on the number of operations is 40 per hour. However, in the case of mixed (arrival and departure) operations, the throughput may be higher. For instance, the runway is capable of handling 30 arrivals and 30 departures per hour. Since envelopes are typically convex [21], the individual segments can be modeled as independent constraints, the intersection of which forms the capacity envelope. Depending on the airport of operation and the class of UAS, these constraints may impact unmanned aircraft as well. The DRIFT framework is capable of including UAS within the airport capacity constraints.

Airspace capacity constraints limit the number of aircraft that can be in a sector at any time, and are driven by the geometry of the sector as well as air traffic controller workload [22]. Departure queue management strategies that limit the number of aircraft on the airport surface can be enforced using a "surface" sector or a taxi arc. Airspace throughput constraints, such as miles-in-trail and minutes-in-trail constraints that stipulate the minimum spacing between two aircraft [22], can be represented by node throughput constraints similar to the airport capacity constraints. Such sector capacity and node throughput constraints can be used to manage UAS trajectories, both for high-altitude traffic that is impacted by the enroute constraints faced by manned flights, as well as low-altitude (or UTM [23]) operations that involve interactions among UAS.

2) Flight connectivity constraints: Airline operations see high levels of flight connectivity, i.e., the same aircraft being used on multiple legs in sequence. In fact, only about 6% of flights on a typical day in the US have no connection; a typical aircraft performs 4–6 flights in a day [11].

Fig. 3 shows the causes of flight delays in the US and Europe in 2015. We see that in both systems, the late arrival of the aircraft (from its preceding leg) is the largest cause of delay, about 40%. We additionally note that in Europe, 5% of delays are due to crew and passenger connectivity. It is likely to be similar in the US, where such delays are included under the category of "air carrier delays". Due to such high levels of connectivity, it is important to account for the downstream impacts of ATFM actions.

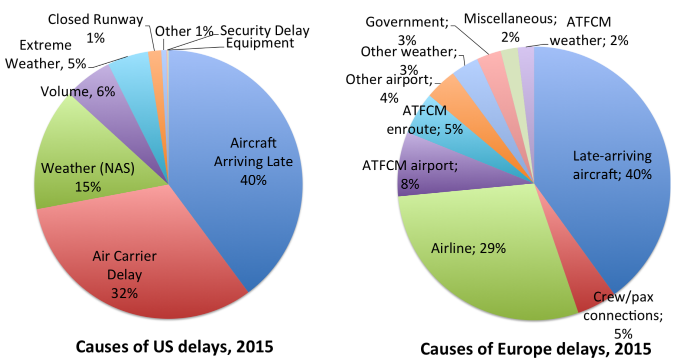

Figure 3: Causes of flight delays, [left] in the US [2], and [right] in Europe [24].

The connectivity constraints also imply that solving the ATFM problem by considering a small geographical region or using a naïve rolling horizon approach would result in a significant loss of efficiency. It is therefore desirable to develop scalable optimization approaches that can handle problems at a nationwide scale over time horizons spanning an entire day.

3) Geofencing constraints: One approach that has been proposed for integrating UAS into the airspace system is *geo-fencing* [23]. A geofence is a boundary that encloses an area that UAS trajectories must avoid. Geofences may be static or dynamic (time-varying), and could be used by Air Navigation Service Providers (ANSPs) to better manage the airspace. For example, only manned aircraft may be allowed in a certain region of the airspace during a certain period of time, or alternatively, only certain classes of unmanned aircraft. It is expected that the geofence definitions will then be communicated to airspace users and UAS operators [25]. In this paper, we illustrate how such geofencing constraints can be incorporated into ATFM, and how the DRIFT algorithms can be used to determine trajectories that conform to the geofence boundaries.

4) Length of planning horizon: As mentioned earlier, high levels of flight connectivity in passenger or cargo operations can limit the efficacy of decomposing the problem into smaller planning horizons, by resulting in very sub-optimal solutions. The nature of UAS demand further motivates the need for solving the ATFM problem for long planning horizons. Some UAS missions, such as ones for agricultural or aquatic surveillance, and infrastructure inspection, are long duration missions, ranging from many hours to a day. In order to accommodate trajectories of these lengths, it is necessary to solve the ATFM problem for long planning horizons. In addition, since UAS demand is likely to be unscheduled, it is important to be able to re-optimize operations for the remainder of a day when new demand materializes. In other words, there is a need for computational approaches that can solve the nation-scale problem for planning horizons of 24-hour length, in a few minutes.

III. Problem description and solution approach

Time is discretized in our mathematical formulation of the problem, implying that all transit times in the network are integer multiples of the time period. Similarly, all operations occur at a set of periodic epochs. The problem can then be stated as follows:

Given a set of flights (and the associated aircraft/tails operating each of them), airport and airspace capacity constraints, and geofence constraints, identify a 4D-trajectory for each aircraft that maximizes the system-wide benefit (revenue plus cancellation penalty) minus costs (operating costs plus delay costs), and that obeys operational and capacity constraints for all time periods.

A. Mathematical formulation

In the discussion that follows, the term *network* refers to the network representation of the ATFM problem, shown in Fig. 1. First, we introduce the following notation.

$\mathscr{F}$	Set of flights.
$\mathscr{L}$	Set of tails.
$\mathscr{S}$	Set of sectors.
$\mathscr{T}$	Set of time periods in the time horizon.
$\mathscr{N}$	Set of nodes in the network.
$\mathscr{A}$	Set of arcs in the network.

s_a Sector containing arc a.

$\mathscr{R}$ Set of all feasible 4D-trajectories, where a 4D-trajectory specifies the routing of a contiguous set of flights starting with the tail's origin. A trajectory could contain only the 4D-trajectory for an aircraft until the first cancellation; all flights with no trajectory (the first cancelled flight and all subsequent connections) are considered cancelled.

$\mathscr{R}_L(\ell)$ The set of feasible 4D-trajectories for tail ℓ. Feasible trajectories satisfy both routing constraints on the different flights, as well as maximum/minimum travel times on any arc, and minimum turnaround time constraints for consecutive flights of the same tail.

$P(a,t)$ Number of aircraft on arc $a \in \mathscr{A}$ at time $t \in \mathscr{T}$.

$Q(a,t)$ Number of aircraft that enter arc $a \in \mathscr{A}$ at time $t \in \mathscr{T}$.

$\mathscr{J}(n,t)$ The set of segments in the capacity envelope of node n at time t.

ρ_r Benefit (revenue plus cancellation penalties) minus costs of trajectory $r \in \mathscr{R}$.

$B_{s,t}$ Capacity of sector s at time t.

$D_{n,t,j}$ Right-hand side of the linear constraint for segment j of the capacity envelope for node n at time t.

$a_{r,t}$ Arc that an aircraft following trajectory r is on at time t.

$\sigma_{a,t,j}$ Coefficient of arc a in the linear constraint representing segment j of the capacity envelope for the head node of arc a at time t

$\mathscr{R}_S(s,t)$ Set of trajectories that use sector s at time t.

$\mathscr{R}_N(n,t,j)$ Set of trajectories that leave node n at time t and use the resource represented by segment j of the capacity envelope at that time.

Sector capacity constraints limit the total number of aircraft in a sector. They can be written as

$$\sum_{a \in A_S(s)} P(a,t) \leq B_{s,t} \qquad \forall s \in \mathscr{S}, t \in \mathscr{T} \qquad (1)$$

where $B_{s,t}$ is the capacity of sector s at time t.

A trajectory $r \in \mathscr{R}$ is said to *intersect* with a sector capacity constraint for sector s at time t if the trajectory results in an aircraft being present in sector s at time t. A trajectory can intersect with at most one sector capacity constraint at any time (in the deterministic case). The set of trajectories that intersect with a capacity constraint representing sector s at time t is denoted by $\mathscr{R}_S(s,t)$.

Node capacity constraints limit the throughput of aircraft through a node (for example, an airport runway or a metering fix). As mentioned earlier, runway capacity constraints are typically expressed in terms of a capacity envelope (Fig. 2). Although the node capacity envelope at a certain time could consist of multiple segments, we treat each segment as an independent constraint, as the intersection of all segments at a certain time defines the envelope as a whole. Therefore, when we refer to a node constraint, we refer to one segment $j \in \mathscr{J}(n,t)$ of the envelope of a node n at time t. The node throughput constraint is linear, and is written as

$$\sum_{a \in A_N(s)} \sigma_{a,t,j} Q(a,t) \leq D_{n,t,j} \quad \forall n \in \mathscr{N}, t \in \mathscr{T}, j \in \mathscr{J}(n,t) \quad (2)$$

where σ and D are constants that define the shape of the segment.

A trajectory $r \in \mathscr{R}$ is said to *intersect* with a node capacity constraint for node n at time t if the trajectory results in an aircraft entering an outgoing arc of node n at time t (i.e., the aircraft passes through node n at time t). The set of trajectories that intersect with a node capacity constraint representing envelope segment j of node n at time t is denoted by $\mathscr{R}_N(n,t,j)$.

The traditional formulation of the TFMP has capacity and operational constraints, with binary decision variables that indicate whether a flight has reached a sector by a certain time period. By contrast, our formulation has only capacity constraints, and all the other constraints (minimum and maximum transit times, flight connectivity, turnaround times, etc.) are absorbed into the definition of the variable. This formulation results in fewer constraints but an exponentially greater number of variables. The decision variables are defined as follows:

$$x_r = \begin{cases} 1 & \text{if trajectory r is chosen, and} \\ 0 & \text{otherwise} \end{cases} \qquad \forall r \in \mathscr{R} \quad (3)$$

The Master Problem can now be stated as follows:

$$\begin{aligned}
\text{maximize} \quad z \quad &= \quad \sum_{r \in \mathscr{R}} \rho_r \, x_r && (4)\\
\text{s.t.} \quad \sum_{r \in \mathscr{R}_L(\ell)} x_r \quad &\leq \quad 1, \quad \forall \ell \in \mathscr{L} && (5)\\
\sum_{r \in \mathscr{R}_S(s,t)} x_r \quad &\leq \quad B_{s,t}, \quad \forall s \in \mathscr{S}, t \in \mathscr{T} && (6)\\
\sum_{r \in \mathscr{R}_N(n,t,j)} \sigma_{a_{r,t},t,j} \, x_r \quad &\leq \quad D_{n,t,j}, \; \forall n \in \mathscr{N}, t \in \mathscr{T}, j \in \mathscr{J}(n,t) && (7)\\
x_r \quad &\in \quad \{0,1\}, \; \forall r \in \mathscr{R} && (8)
\end{aligned}$$

Objective (4) maximizes the total benefit minus cost of all trajectories selected. Constraint (5) states that at most one trajectory may be selected for each tail. Constraints (6) and (7) are the sector and node capacity constraints respectively.

B. Solution approach

The problem as formulated can be efficiently decomposed into a set of parallelizable sub-problems, with an easy-to-solve master problem that coordinates between the sub-problems, using column generation. A comprehensive discussion of the algorithms, mathematical proofs, and experiments on synthetic airspace sectorization data can be found in [11]. Fig. 4 shows a schematic of the solution process. The master problem transmits the pricing signal, which comprises of the *dual prices* associated with the different resource capacity constraints. In particular, the dual prices $\lambda_{s,t}$ and $\mu_{n,t,j}$ correspond to the dual variables for constraints (6) and (7) respectively, and signal congestion at the sectors and nodes. The dual prices help guide the sub-problems (solved by the distributed nodes, potentially one for each tail) to an optimal solution. The dual price π_ℓ denotes the dual price for constraint (5), namely, the marginal cost of including that particular trajectory.

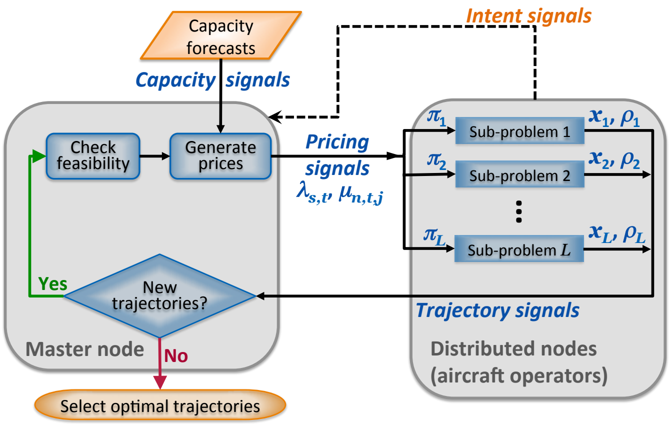

Figure 4: Schematic of the distributed solution process for very large-scale ATFM [11].

C. Interpretation of architecture

The proposed architecture of DRIFT can be interpreted as follows. Fig. 5 shows a high-level depiction of the architecture.

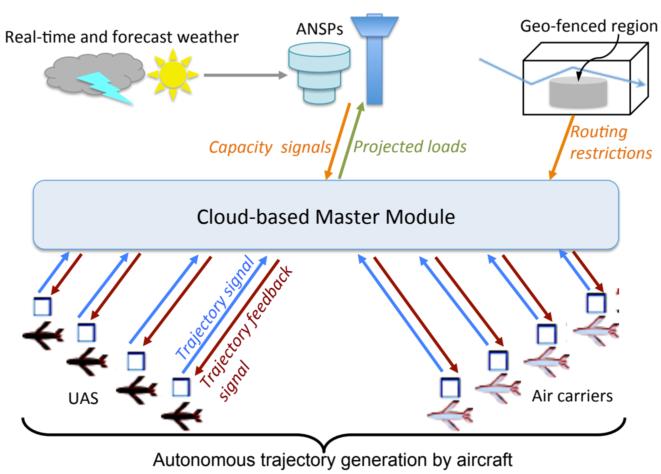

Figure 5: High-level depiction of the DRIFT architecture.

1) Each aircraft shows "intent" by expressing the desire to fly a certain flight by sharing the origin, destination, preferred departure and arrival times (and in the case of UAS, the type of mission, i.e., grid, point-to-point, etc.). This is known as the *intent signal*.
2) DRIFT broadcasts the "cost" of using each airspace resource at each point in time for each aircraft class; the cost is a function of projected capacity, projected loading, and the fraction of resource consumed by each aircraft type. The term "cost" does not imply a financial transaction, an indication of how congested that particular region of airspace is. This is known as the *pricing signal*. By comparing the cost of the required resources to the revenue of flying a particular trajectory, an aircraft operator can decide whether or not the trajectory is worth flying.

   Since the pricing signal can be made to vary by aircraft class, it can be used to implement geofencing constraints. For example, suppose a certain region of airspace needs to be enforced as a "keep-out" geofence for low-altitude UAS, that is, low-altitude UAS are not allowed access to this region. Then, low-altitude UAS see an effectively infinite price for the geo-fenced sector, and their trajectories can therefore not enter that sector. By contrast, manned and high-altitude UAS will continue to see prices for that sector, as calculated by considering the demand for that resource.
3) Each aircraft may autonomously determine its optimal 4-D trajectory (sequence of resources to use and times at which to use them) as well as the "benefit" of this trajectory to it. At each iteration, the aircraft communicates its desired trajectory and the resulting benefit to DRIFT. This is known as the *trajectory signal*. In the short- and medium-term, an air carrier or aircraft operator would determine and communicate the trajectory signals of all its aircraft to the master node.
4) Upon receiving the trajectory signal from the aircraft, DRIFT assesses the trajectory for feasibility (whether there is capacity available on resources along the proposed trajectory). If the trajectory is feasible, DRIFT adds the trajectory (i.e., the column), and returns a confirmation (*trajectory feedback signal*) to the aircraft.
5) DRIFT re-computes the optimal trajectories and prices every time the capacity of a resource changes, or when an aircraft successfully reserves capacity on a resource.

1) Types of signals: We further elaborate on the information that is exchanged:

1) **Capacity signal.** A key input to allocation resources is the expected capacity of each resource in the system over time. DRIFT will have default values for each resource (similar to MAP values for sectors, or runway capacities for airports), which are updated by resource operators based on expected weather conditions or other dynamic constraints caused by geo-fencing.
2) **Intent signal.** The intent signal is the start of an aircraft's interaction with DRIFT. While it is expected that some fraction of UAS demand will not show intent (similar to pop-ups in todays Air Traffic environment), the system cannot operate as a purely on-demand tactical system in the interests of safety. One key feature of DRIFT is that it is able to rapidly re-compute all signals, and therefore can adjust to resource overloads from pop-ups quickly (within a few minutes). In order to deal with pop-up demand, DRIFT will reserve a portion of capacity on each resource for unexpected demand. DRIFT will have defaults for the size of the buffer, which may be dynamically updated by the resource operators as part of the capacity signal. It is envisioned that non-scheduled

flights will still be managed by a system that resembles the current emergency COA procedures.

3) **Pricing signal.** The prices of resources are also a by-product of the optimization process, obtained as dual prices of the resources at various times. A special property of the prices is that they are zero for unconstrained resources (i.e., resources that are not at capacity) while they may be positive for constrained resources. Additionally, the prices can be adjusted such that geo-fenced regions have extremely high cost (ensuring that no aircraft may use it) or a high cost only for certain classes of aircraft (if the fenced region is off-limits only to certain classes).
4) **Trajectory signal.** Given the resource prices published and the cost/revenue functions for each flight, each aircraft computes an optimal trajectory. This computation is equivalent to solving a longest-path problem on the NAS network. The aircraft l then transmits the desired trajectory, x_l along with the corresponding benefit, ρ_l. We note that the aircraft operators get to maintain a certain level of privacy, since they do not have to share their internal delay cost and revenue functions, but instead only need to share the net benefit of the entire trajectory.
5) **Trajectory feedback signal.** The trajectory feedback signal is a mechanism to ensure that an aircraft is only allocated a feasible trajectory. It can also be used to ensure that the final trajectory flown by the aircraft is consistent with its intent, that is, for conformance monitoring.

IV. Computational experiments

A. Datasets

1) Manned aircraft demand: The data for manned aircraft was obtained from a run of the FAA's System-Wide Analysis Capability (SWAC) [8], representing a typical day of flights in the year 2030. The forecasted demand in SWAC accounts for demand changes by origin/destination pair, fleet mix changes, as well as equipage improvements.

The dataset contains flight data for approximately 47,900 passenger flights operating between approximately 2,400 airports, and represent all flights with an origin or destination in the United States. There are approximately 27,500 unique aircraft IDs in the data set (implying that each unique aircraft flights nearly two legs on average). The dataset also contains flight paths for each flight, which were merged with the sector definition files to obtain the en route and terminal resources required by each flight.

2) UAS demand: The UAS demand set for 2030 was generated using the assumptions described in prior work [9]. The dataset consists of approximately 29,000 flights (which includes both high-altitude and low-altitude operations). The flights span various UAS types including Predators, ScanEagles, Ravens, etc. The missions operated span a wide range of applications, including communications, agriculture, fish spotting, cargo delivery, etc. In terms of the desired trajectories, the four key mission types in the data set are:

1) Point-to-point, in which the aircraft travels directly from the origin to the destination;
2) Polygon, in which the UAS performs a polygonal flight path around an area of interest;
3) Random, in which the UAS randomly traverses a region of airspace;
4) Grid, in which the UAS traverses a region along a structured grid.

While there are more than 4,000 airports in this data set, most origin/destinations are smaller airports (that do not carry commercial traffic); many are military airports as the data set includes military UAS as well. A key assumption made by the dataset is that all operations begin and end at an FAA-recognized airport, i.e., it does not contain aircraft launched by hand or catapult. The airports used do not belong to Class B, C, or D airspace (and hence there is little terminal-area interaction with commercial aircraft). We did not consider alternate shorter routing options for unmanned aircraft, since their mission trajectories are likely to be convoluted by design (for example, a UAS may need to conduct surveillance by remaining in a certain area for several hours).

The airspace sectorization was assumed to be similar to the current one. The implementation considered 955 sectors, at both low and high altitudes. The sector capacities were assumed to be the same as current values, while the airport capacity envelopes were assumed to correspond to the values with planned 2030 improvements. A snapshot of the 2030 traffic implemented with the sector boundaries is shown in Fig. 6.

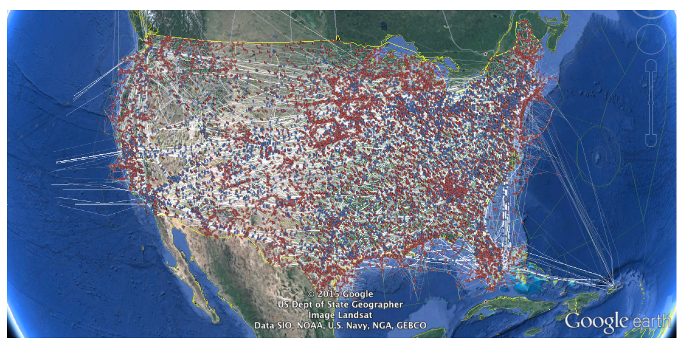

Figure 6: Snapshot of traffic of unmanned (red) and manned (blue) traffic for the year 2030.

B. Pricing signals

As seen in Sec. III the prices of resources at different times are given by the dual prices. The price of a resource is zero if it is not at capacity, but may be positive if a resource is constrained, that is, demand for it exceeds the capacity at that time. Since the constraints and demand (flight schedules and UAS mission times) are dynamic, the prices of resources also vary with time. Fig. 7 shows examples of pricing signals for two resources, an airport and an airspace sector, as functions of time for a portion of the day.

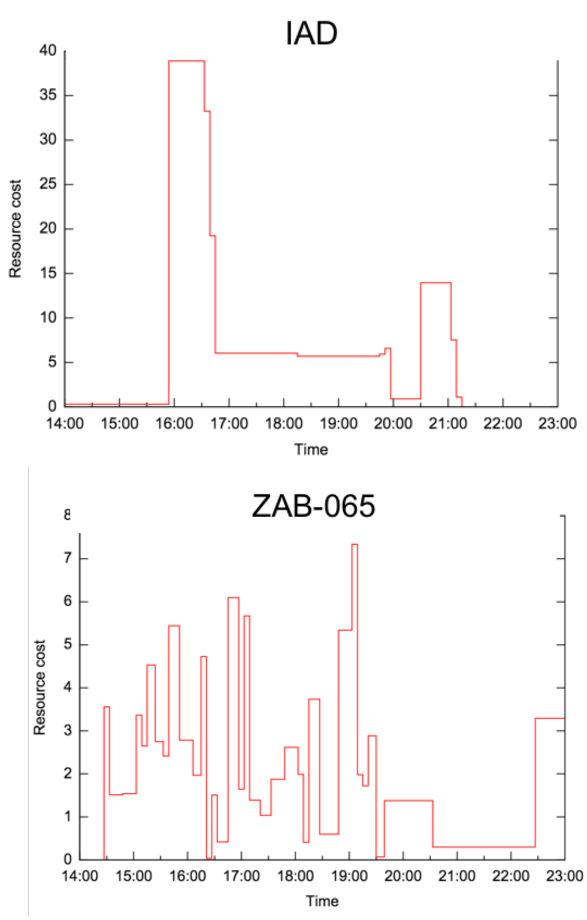

Figure 7: Examples of pricing signals computed by DRIFT, for (top) an airport, IAD, and (bottom) an airspace sector. We see that the price of a resource (i.e., the dual variable corresponding to the capacity constraint) can vary with time, depending on the demand.

C. Geofencing

Geofencing is the process by which certain classes of aircraft are prevented from entering some portion of the airspace. For example, it may be desirable to prevent unmanned aircraft from entering a region around an airport for a certain time period of high density of manned operations. DRIFT manages geofencing by updating the prices of resources such that the aircraft that are not allowed to use the resources see a price of infinity, while those that are allowed to use the resource continue to see the "normal" DRIFT prices. As a result, aircraft with the higher cost cannot access the resources, since it will need an extremely high (in fact, infinite) revenue in order for the revenue to outweigh the cost. In this manner, geofencing can be applied in an aircraft-specific or aircraft-class specific fashion. Fig. 8 shows the result of a geofencing implementation, when a certain sector is closed-off to manned as well as high-altitude unmanned traffic (therefore, the sector allows only low-altitude unmanned traffic).

D. Scalability: Computation times

The trajectories were determined at a 1-min discretization, while the sector and airport capacity constraints were defined

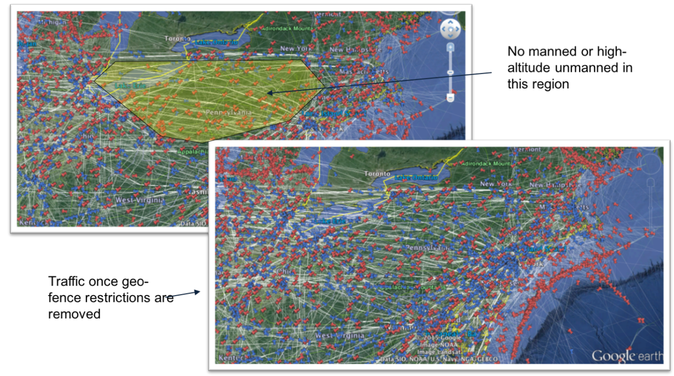

Figure 8: Example of geofence implementation using DRIFT, showing traffic unmanned (red) and manned (blue) traffic, with and without geofencing.

at a 5-min resolution, that is, were assumed to be constant over each 5-min interval. The algorithms were implemented on a Linux X-Large machine with 40 cores on AWS, reading and writing output to the cloud-storage s3 system. The algorithms were implemented in C and were optimized for speed and memory performance. The run time was found to be under 4 min in order to obtain solutions that were within 0.1% of optimal, and 7 min to be within 0.01% of optimal. The implementation demonstrates the scalability of the DRIFT architecture to take advantage of parallelism, and the ability of the algorithms to be run in real-time.

Since UAS demand is likely to be dynamic and unscheduled, it is envisioned that DRIFT will be run every 15 minutes or so, in a rolling-horizon framework. We therefore consider a rolling-horizon implementation, with a planning horizon of 8 hours (with a 2 hour overlap and freeze window). Each planning horizon has approximately 25,000 flights. We find that such an implementation further reduces the run times to be under 1 min.

For the given typical day, 50 scenarios were generated, each with varying degrees of capacity constraints and volumes. Each scenario was run to optimality. The run-time performance of the algorithm was found to be similar to the instance described above.

E. Robustness to inaccurate information

The question of how the system reacts to bad/delayed signals was investigated by simulating the behavior of the system when the prices were perturbed by a small percentage, or when the prices used were "stale", i.e., from a previous iteration. DRIFT was found to be resilient to such imperfect signals: The trajectory feedback signal is a point at which the information that is being used by the aircraft is verified for accuracy; if the data is inaccurate but still meets the latest prices, the aircraft may still fly the route, but if significant discrepancies are found in the data, a revised price signal is sent to the aircraft as part of the trajectory feedback signal. This mechanism therefore serves as a verification of the intent of the aircraft and the accuracy of its data.

F. Equity and incentives

In today's ATFM environment, the intent signals correspond to scheduled demand, and the cost of a trajectory simply corresponds to the amount of delay incurred by the flight. In this system, equity corresponds to a first-scheduled-first-served allocation of all resources. If the ATFM problem is solved to just minimize total system delays equitably, the allocation to each aircraft operator would be consistent with a Ration-By-Schedule policy [22]. However, an oft-mentioned concern [26, 27] relates to mixed-equipage scenarios, and incentivizing equipage and conformance. Similar to geofencing constraints, DRIFT's architecture is capable of placing hard constraints on certain classes of aircraft by closing access to certain resources for certain types of aircraft, through an extremely high price signal. Alternatively, "soft" constraints could be used to implement best-equipped-best-served strategies [28]: in this setting, poorly equipped aircraft would pay a higher price for flying through a resource than a better-equipped aircraft. Another potential application of the proposed approach would be to improve system performance by optimizing flight-specific costs; however, questions of truthfulness in reporting trajectory valuations would then need to be addressed before its practical adoption. Similarly, issues of conformance monitoring and enforcement would also need to be addressed. The design of market-based mechanisms for resource allocation is a topic that we intend to investigate further.

V. Conclusions

This paper presented a distributed implementation on AWS of a solution approach to very-large scale ATFM, in the presence of unmanned aircraft. The approach determines optimal trajectories in space and time in the presence of network and flight connectivity constraints (for commercial air carriers), airport and airspace capacity constraints, and geofencing constraints. Using projected US demand for the year 2030 with approximately 48,000 passenger flights and 29,000 UAS operations per day, we showed that our implementation can find solutions to within 0.1% of optimal for a 24-hour period in less than 4 minutes. Furthermore, a rolling horizon implementation (with 6-8 hour time windows) results in run times of less than a minute. The UAS demand spanned a range of missions and altitudes, ranging from low-altitude cargo delivery to high-altitude surveillance for fishing applications. Geofencing, a key requirement for the integration of UAS into the airspace system, was also demonstrated within the proposed framework. These results show that the proposed solution approach to ATFM can be extended to manage imbalances between capacity and demand, even in the presence of unmanned systems.

Acknowledgments

This work was supported in part by NASA under a Small Business Innovative Research (SBIR) grant. The authors thank Metron Aviation for help with airspace and demand data, as well as for helpful discussions.

References

[1] Joint Economic Committee, US Senate, "Your Flight has Been Delayed Again: Flight Delays Cost Passengers, Airlines, and the US Economy Billions," 2008.

[2] Bureau of Transportation Statistics, "Airline On-Time Statistics and Delay Causes," 2016. [Online]. Available: http://www.transtats.bts.gov/

[3] EUROCONTROL Performance Review Commission, "Performance Review Report: An assessment of Air Traffic Management in Europe during the Calendar Year 2007," 2008.

[4] John A. Volpe National Transportation Systems Center, U.S. Department of Transportation, "Unmanned Aircraft System (UAS) Service Demand 2015-2035: Literature Review and Projections of Future Usage, Version 0.1," 2013, Technical Report No. DOT-VNTSC-DoD-13-01.

[5] M. Perez-Batlle, E. Pastor, X. Prats, P. Royo, and R. Cuadrado, "Maintaining separation between airliners and RPAS in non-segregated airspace," in *Tenth USA/Europe Air Traffic Management Research and Development Seminar (ATM2013)*, 2013.

[6] Z. Sunberg, M. Kochenderfer, and M. Pavone, "Optimized and Trusted Collision Avoidance for Unmanned Aerial Vehicles using Approximate Dynamic Programming," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2016.

[7] Federal Aviation Administration (FAA), "OPSNET website," 2014, http://aspm.faa.gov/.

[8] K. Noonan, "FAA's System-Wide Analysis Capability," in *Integrated Communications, Navigation and Surveillance Conference (ICNS)*, 2011.

[9] F. Wieland, R. Sharma, and D. Watulak, "Predicting future unmanned aerial system flights," in *AIAA Aviation Technology, Integration, and Operations Conference*, 2013.

[10] D. Bertsimas, G. Lulli, and A. Odoni, "An integer optimization approach to large-scale air traffic flow management," *Operations Research*, vol. 59, no. 1, pp. 211–227, January-February 2011.

[11] H. Balakrishnan and B. G. Chandran, "Optimal large-scale air traffic flow management," 2014, web.mit.edu/hamsa/www/pubs/BalakrishnanChandran_ATFM.pdf.

[12] D. Alejo, R. Conde, J. Cobano, and A. Ollero, "Multi-UAV collision avoidance with separation assurance under uncertainties," in *2009 IEEE International Conference on Mechatronics*, 2009.

[13] S. Graham, W. Chen, J. D. Luca, J. Kay, M. Deschenes, N. Weingarten, V. Raska, and X. Lee, "Multiple intruder autonomous avoidance flight test," in *Infotech@Aerospace Conference*, 2011.

[14] A. R. Odoni, "The flow management problem in air traffic control," in *Flow Control of Congested Networks*, A. R. Odoni, L. Bianco, and G. Szego, Eds. Berlin: Springer-Verlag, 1987, pp. 269–288.

[15] D. Bertsimas and S. Stock Patterson, "The air traffic flow

management problem with enroute capacities," *Operations Research*, vol. 46, no. 3, pp. 406–422, May-June 1998.

[16] ——, "The traffic flow management rerouting problem in air traffic control: A dynamic network flow approach," *Transportation Science*, vol. 34, no. 3, pp. 239–255, August 2000.

[17] G. Lulli and A. Odoni, "The European Air Traffic Flow Management Problem," *Transportation Science*, vol. 41, no. 4, pp. 431–443, November 2007.

[18] O. Richard, S. Constans, and R. Fondacci, "Computing 4D near-optimal trajectories for dynamic air traffic flow management with column generation and branch-and-price," *Transportation Planning and Technology*, vol. 34, no. 5, pp. 389–411, 2011.

[19] H. D. Sherali, R. W. Staats, and A. A. Trani, "An Airspace Planning and Collaborative Decision-Making Model: Part I – Probabilistic Conflicts, Workload, and Equity Considerations," *Transportation Science*, vol. 37, no. 4, pp. 434–456, November 2003.

[20] ——, "An Airspace-Planning and Collaborative Decision-Making Model: Part II – Cost Model, Data Considerations, and Computations," *Transportation Science*, vol. 40, no. 2, pp. 147–164, 2006.

[21] E. P. Gilbo, "Airport capacity: Representation, estimation, optimization," *IEEE Transactions on Control Systems Technology*, vol. 1, no. 3, pp. 144–154, September 1993.

[22] T. W. M. Vossen, R. Hoffman, and A. Mukherjee, "Air Traffic Flow Management," in *Quantitative Problem Solving Methods in the Airline Industry*, ser. International Series in Operations Research & Management Science, C. Barnhart and B. C. Smith, Eds. Springer Science+Business Media, LLC, 2012, vol. 169, pp. 385–453.

[23] P. Kopardekar, J. Rios, T. Prevot, M. Johnson, J. Jung, and J. E. R. III, "Unmanned Aircraft System Traffic Management (UTM) Concept of Operations," in *AIAA Aviation Technology, Integration, and Operations Conference*, 2016.

[24] EUROCONTROL, Central Office of Delay Analysis (CODA), "Delay analysis in 2015: Annual report," 2016.

[25] S. D'Souza, A. Ishihara, and B. Nikaido, "Feasibility of varying geo-fence around an unmanned aircraft operation based on vehicle performance and wind," in *IEEE Digital Avionics Systems Conference*, 2016.

[26] NASA Advisory Council, Aeronautics Committee, UAS Subcommittee, "Unmanned Aircraft Systems (UAS) Integration in the National Airspace System (NAS) Project Subcommittee Final Report," 2012.

[27] ——, "The UAS ARC Activities and NASA Project Alignment," 2013.

[28] A. M. Churchill, M. O. Ball, A. D. Donaldson, and R. J. Hansman, "Integrating best-equipped best-served principles in ground delay programs," in *Ninth USA/Europe Air Traffic Management Research and Development Seminar (ATM2011)*, 2011.

Author Biographies

Hamsa Balakrishnan is the co-founder and Chief Scientist of Resilient Ops, Inc., and an Associate Professor of Aeronautics and Astronautics at the Massachusetts Institute of Technology. Her research is in the design, analysis, and implementation of control and optimization algorithms for large-scale cyber-physical infrastructures, with an emphasis on air transportation systems. She received her PhD in Aeronautics and Astronautics from Stanford University. She is a recipient of the Donald P. Eckman Award from the American Automatic Control Council (2014), the Lawrence Sperry Award from the American Institute of Aeronautics and Astronautics (2012), the CNA Award for Operational Analysis (2012), the Best Paper Award at the FAA/Eurocontrol Air Traffic Management R&D Seminar (2011), and an NSF CAREER Award (2008).

Bala Chandran is the co-founder and CEO of Resilient Ops, Inc. He received his PhD in Industrial Engineering and Operations Research from the University of California at Berkeley in 2007, and a M.S. from the R.H. Smith School of Business at the University of Maryland, College Park, in 2002. Prior to founding Resilient Ops, he spent 7 years as an Operations Research Consultant.