Tackling Airspace Congestion: A Scalable and Robust Framework for End-to-End UAS Traffic Management

Minjun Sung, Sambhu H. Karumanchi, Christophe J. Hiltebrandt-McIntosh,
Hunmin Kim, and Naira Hovakimyan

***Abstract*— We present an end-to-end air traffic management framework for Unmanned Aircraft Systems (UAS) operations that is both scalable and robust. Our approach involves defining congestion in the airspace and developing a congestion-based cost map that can mitigate potential congestion while adhering to regulations and guidelines. Each UAS operation leverages a recursively updated cost map in solving the path planning problem, providing scalability of the framework. Additionally, we introduce a time-critical controller to enhance the robustness of mission execution. Empirical evidence confirms the feasibility and effectiveness of our method, achieving significant reductions in both cumulative and maximum levels of airspace congestion.**

I. Introduction

With the advent of commercial drones for emergent markets such as delivery, inspection, or surveillance, the use of aerial vehicles is expected to dramatically rise in the near future [1]. In this new environment, novel methods for air traffic management are necessary to navigate the complexities of airspace both in and around cities [2].

NASA and the Federal Aviation Administration (FAA) have presented a conceptual framework for the Unmanned Aerial System's Traffic Management (UTM) in their *Concept of Operations* brief, recognizing the need for a universal traffic management solution to address the growing number of low altitude aerial traffic management issues faced by operators [3]. However, while the brief outlines the conceptual desires, the technical specifications are left up to designers.

The path planning problem has been the subject of prolonged research [4]–[6]. In the UTM context, granting planning authority to individual UAS operators is desirable for scalability, however, it presents a challenge to the UTM in potentially congested airspace. Current safety measures, such as control barrier functions [7], may encounter deadlocks and oscillations in dense environments [8]. More comprehensive methods - such as Model Predictive Control (MPC) [9], HJ Reachability analysis [10], and game theoretic methods [11], [12] - incorporate predictive estimates to avoid collisions and congestion. However, when faced with a large number of operators, these methods can struggle to scale effectively.

This work is supported by NASA (NNH20ZEA001N-ULI and 21-USRCCycle2-0001).

Minjun Sung, Sambhu H. Karumanchi, and Naira Hovakimyan are with the Department of Mechanical Science and Engineering; Christophe J. Hiltebrandt-McIntosh is with the Department of Aerospace Engineering, University of Illinois at Urbana-Champaign, USA; {mjsung2,nhovakim,shk9,cjh11}@illinois.edu.

Hunmin Kim is with the Department of Electrical and Computer Engineering, Mercer University, USA; kim_h@mercer.edu.

In this vein, a UTM design must be capable of accommodating a large number of operators while also addressing the potential challenges posed by operators planning their routes independently. Furthermore, the UTM must also be able to handle real-time updates of environment data. In this paper, we present a scalable and robust end-to-end UTM framework that seamlessly addresses the challenges of path planning and control in dense and dynamic environments.

A. Literature Review

The authors in [13] proposed a heuristic to enable a UAS traffic flow through independent *service providers*. This work primarily focuses on the coordination between the service providers in order to obtain a feasible trajectory that traverses through multiple service providers. On the contrary, our work rather focuses on the potential *congestion* problem within the range of a single service provider. The authors in [14] incorporated high-level operational requirements provided by the FAA into a mathematical formulation in order to plan a trajectory for UAS operations. This work focuses on the physical requirements, such as speed limits and avoiding obstacles to find a trajectory that fulfills the mission requirement. While our work also attempts to mathematically encode FAA guidelines, we propose a broader framework that can also capture more ambiguous qualitative guidelines.

Traffic routing algorithms that aim to avoid congestion have been widely studied for ground vehicles [15]. However, similar efforts for air traffic have been relatively limited, despite the recognized importance of addressing congestion issues in the airspace [16], [17]. Previous research has explored UAS path planning in cluttered environments [18]. However, there is a lack of comprehensive research systematically addressing congestion reduction at the UTM level.

B. Statement of contributions

In this paper,

1) we propose a scalable end-to-end framework for UTM that leverages a novel definition of airspace congestion;
2) we formulate an optimization problem for each mission based on the latest congestion-based cost map that is recursively updated, enabling each UAS operation to mitigate airspace congestion;
3) we propose a feasible algorithm for integrating a time-critical controller to enhance robustness;
4) we empirically validate the efficacy of the proposed framework in reducing both the cumulative and maxi-

mum level of airspace congestion with limited use of communication bandwidth.

II. Problem Definition

A. UTM Background

The primary entities comprising the UTM system are Air Navigation Service Providers (ANSP), UAS operators, and System-wide Data Service Providers (SDSP) [19]. The ANSP is responsible for 1) defining and updating airspace constraints, 2) real-time airspace management, and 3) exchanging data with other entities to facilitate smooth operations. UAS operators, in accordance with regulations and guidelines, have the obligation to ensure safety, which includes responsibilities such as avoiding collisions with other vehicles. Lastly, the SDSP transmits external data, including meteorological data, construction activity, and others to the ANSP for updating the spatial and temporal constraints.

Given the expected scale of UAS operations and different computation capabilities among UAVs, it is crucial to minimize data transmissions between the UTM entities in order to facilitate seamless communication. This is particularly critical in uncontrolled airspace, where UAS share airspace with other users, such as gliders, balloons, and parachutists [19].

While the ANSP is responsible for providing relevant information to UAS operators to ensure safe operations, it is ultimately the operator's responsibility to adhere to regulations and guidelines. However, as described previously, typical collision avoidance methods alone are not sufficient to reduce congestion in UAS traffic. As such, UAS operators should not exclusively depend on collision avoidance methods but should adopt a supplementary planning strategy to anticipate and avoid congestion.

In addition to congestion-aware path planning, it is crucial for UAS operators to execute their plans with robustness. Specifically, they must stay close to their reported paths in the presence of uncertainty during flight. Deviations from the planned trajectory can be amplified through cascaded system layers, potentially compromising the UTM system.

B. Problem Statement

The purpose of this paper is to develop a UTM framework that fulfills specific technical requirements including

1) limited communication bandwidth between entities to ensure scalability;
2) reduced maximum and cumulative congestion levels to ensure safe and efficient operations;
3) robustness of UAS operations against environmental uncertainty to conform to its plan.

III. Methods

A. Overview

To address the specifications stated in Section II-B, the following subsections will be covered in this section.

1) **(Scalable UTM Framework)** The formal definition of congestion and the specification of information flow among entities, including the logistics of both the ANSP and the UAS, will be discussed.
2) **(Congestion-aware UAS Path Planner)** A congestion-based cost map that is compatible with the UTM framework will be presented, which can be used to formulate an optimization problem for path planning of UAS operators.
3) **(Time-critical UAS controller)** A low-level controller that assists in coordinating UAS operations, enabling them to collaboratively adhere to their respective reported paths, will be discussed.

B. Scalable UTM Framework

1) Domain Discretization: We define a set of discrete time instances $\mathcal{T} = \{t_0, t_0 + dt, \cdots, t_\infty\} \subset \mathbb{R}$, where the time interval dt, initial time t_0, and terminal time t_∞ can be chosen or regularly updated by the ANSP. Additionally, the fixed 2 dimensional spatial domain $\mathcal{X}$ is discretized and enumerated as $\mathcal{X} = \{1, \cdots, n_x\} \times \{1, \cdots, n_y\} \subset \mathbb{N} \times \mathbb{N}$, where n_x and n_y are also chosen by the ANSP. Finally, we define the spatio-temporal artifact, referred to as the *airspace*, as $\Omega = \mathcal{T} \times \mathcal{X}$. It can be visualized as a 3-dimensional array where each element is a tuple denoted by (t, x) representing time and planar position. In this work we chose a 2-dimensional spatial domain for visualization purposes; its extension to 3-dimensional space is straightforward.

Remark 1. *Each element $x \in \mathcal{X}$ does not represent a single point in space but rather a region capable of accommodating multiple vehicles simultaneously. The point-wise representation often adopted in collision avoidance problems is a special case of our representation when $n_x, n_y \to \infty$.*

2) Capacity, Demand, and Congestion: In ground traffic, *capacity* is defined as the maximum number of vehicles a roadway can accommodate within a given time frame, while *demand* refers to the number of vehicles attempting to use the roadway during that same period [20]. However, defining capacity for airspace is challenging, as it does not have well-defined physical limits. In the following, we propose an extension of the concepts of capacity and demand to the airspace in a way that can also incorporate the flight guidelines established by the FAA.

Definition 1. *(Airspace capacity) Airspace capacity is a mapping $q : \Omega \to \mathbb{Z}_{\geq 0}$ that specifies the number of aerial vehicles each $(t, x) \in \Omega$ can accommodate.*

Capacity in airspace is determined by various factors such as regulations, guidelines, local constraints, and more. To account for these factors and translate them into airspace capacity, we suggest dividing the airspace into three distinct subsets, or *zones*: prohibited zones $\mathcal{R}$, restricted zones $\mathcal{Y}$, and permissible zones $\mathcal{G}$. UAS operations must circumvent the prohibited zones and traverse the restricted zones to the minimal extent possible while adhering to routes that traverse the permissible zones in completing their mission. Here, $\Omega = \mathcal{R} \cup \mathcal{Y} \cup \mathcal{G}$ and $\mathcal{R}, \mathcal{Y}, \mathcal{G}$ are mutually exclusive. We use the terms **R**ed, **Y**ellow, and **G**reen zones interchangeably with $\mathcal{R}, \mathcal{Y}, \mathcal{G}$, respectively.

Remark 2. *The categorization of airspace given the environmental information is a process that is determined by the ANSP and/or SDSP. As this process is subject to policy and regulation, we assume that a reasonable and safe categorization has been established for our UTM framework.*

Having $\mathcal{R}, \mathcal{Y}, \mathcal{G}$ well established and provided to the ANSP, the airspace capacity q can be specified as

$$q(t,x) = \begin{cases} 0 & \text{for} \quad (t,x) \in \mathcal{R} \\ \rho(t,x) & \text{for} \quad (t,x) \in \mathcal{Y} \\ q_{max} & \text{for} \quad (t,x) \in \mathcal{G} \end{cases}, \tag{1}$$

where $q_{\max}$ is the maximum allowable airspace capacity and $\rho : \Omega \to \{0, 1, \cdots, q_{\max}\}$ is a ANSP-defined function.

Let us denote the trajectory of each UAS as $\pi_i = \{(t_i^j, x_i^j)\}_{j=0}^{g} \subset \Omega$, where i belongs to the set $\{1, \cdots, N\}$, N being the total number of UAS operations reported to take flight. The superscript j denotes the j^{th} element of the set when the elements are ordered in the increasing order of t_i such that $j = 0$ and $j = g$ correspond to the initial and terminal conditions, respectively. The demand of the airspace $d(t,x)$ is represented by

$$d(t,x) = \sum_{i=1}^{N} \mathbb{1}_{\pi_i}(t,x), \tag{2}$$

where $\mathbb{1}_{\pi}(\cdot)$ is an indicator function for π such that

$$\mathbb{1}_{\pi}(t,x) = \begin{cases} 1 & \text{if } (t,x) \in \pi \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

Finally, we define airspace congestion as follows:

Definition 2. *(Airspace congestion) The airspace congestion is defined by a suitably chosen mapping $\kappa(d(\cdot,\cdot), q(\cdot,\cdot)) \mapsto [0,1)$ based on demand and capacity at each (t,x).*

An example of an airspace congestion κ, which we adopt in this work is:

$$\kappa(d(t,x), q(t,x)) = \tanh\left(\frac{d(t,x) + \epsilon_1}{q(t,x) + \epsilon_2}\right), \tag{4}$$

where $1 > \epsilon_1 \gg \epsilon_2 > 0$ renders $\kappa(d(t,x), q(t,x)) \simeq 1$ when $q(t,x) = 0$, or in other words, $(t,x) \in \mathcal{R}$. Note $\kappa(d(t,x), q(t,x)) \geq 0$ for all $(t,x) \in \Omega$. Moreover, κ is monotonically increasing in d and decreasing in q, which is a desired property. These properties will be utilized in the subsequent subsection to construct a cost map.

It should be emphasized that three-dimensional array representation of Ω directly allows for the calculation of (1)-(4) for all $(t,x) \in \Omega$ through a simple array operation. The respective outcome of this calculation will be denoted as $q(\Omega)$, $d(\Omega)$, $\mathbb{1}_{\pi_i}(\Omega)$, and $\kappa(\Omega)$ for brevity.

3) UTM Logistics: Algorithm 1 provides a concrete description of the logistics of the ANSP.

The *Abort* signal is used to indicate whether the UAS operation is reporting a mission abortion. In such cases, the indicator of the reported path will be subtracted from the demand instead of being added by default. The decision to abort a mission will be made online when the UAS finds the scheduled mission to be infeasible. The threshold for mission abortion will be discussed in Section III-D.

It is important to note that the data transmission to and from the ANSP in Algorithm 1 involves only an array of size $|\Omega|$. This means that the required bandwidth for data transmission will only scale linearly with the number of UAS operation queries at each time instant.

Algorithm 1: Logistics of the ANSP

```
Initialize T, X, Ω and q(Ω), d(Ω)
t = t_0
while True do
    if SDSP update exists then
        Input: R, G, Y
        Update q(Ω) as in (1)
    if Any UAS reports at time t then
        Input: π_i, Abort=False
        if Abort then
            d(Ω) ← d(Ω) − 1_{π_i}(Ω)
        else
            d(Ω) ← d(Ω) + 1_{π_i}(Ω)
        Update κ(Ω) as in (4)
        Output: κ(Ω)
    t ← t + dt
```

C. Congestion-aware UAS Path Planner

Artificial potential field (APF) is a path planning method that borrows the concept of potential field from physics, which explains the movement of objects by the two kinds of forces: attraction and repulsion. Attractive and repulsive potentials are combined and then the gradient is taken to derive the resulting force acting on the agent. Discretizing the configuration space using some cell decomposition methods, one can convert the APF problem into a discrete APF (DAPF) problem. DAPF problem is typically solved by assigning an infinite repulsive force to cells in the obstacle region to avoid collisions [21].

We extend the DAPF method by augmenting the time variable to the domain in order to transform $\kappa(\Omega)$ into a cost map $\mathcal{C}(\Omega)$ without separating the obstacle region and free space. Formally, $\mathcal{C}(t,x) = \mathcal{C}_{att}(x) + \mathcal{C}_{rep}(t,x)$, where

$$\begin{aligned} &\mathcal{C}_{att}(x) = k_{att}\|x - x^g\|, \\ &\mathcal{C}_{rep}(t,x) = \\ &\qquad \sum_{\tau \in \mathcal{B}_t} \sum_{\chi \in \mathcal{B}_x} \gamma^{\frac{|t-\tau|}{dt}} \frac{k_{rep}}{\|x - \chi\|^2 + \delta} \left(\frac{\kappa(\tau,\chi)}{1 - \kappa(\tau,\chi)}\right). \end{aligned} \tag{5}$$

Here, $\mathcal{B}_t := \{t - mdt, t - (m-1)dt, \cdots, t, \cdots, t + mdt\}$, $\mathcal{B}_x := \{\chi \in \mathcal{X} : \|x - \chi\|_2^2 \leq \lambda\}$, where $m \in \mathbb{N}$. Moreover, $\lambda, k_{att}, k_{rep}, \delta \in \mathbb{R}_{\geq 0}$ are tuning parameters, and $\gamma \in (0,1)$ is a discounting factor.

Our formulation, which accounts not only for spatial but also temporal proximity of congestion, is referred to as Repulsive Force Through Time (RFTT). This deviates from the typical cost setup for the DAPF problem, which

focuses only on avoiding obstacles. Specifically, our use of the parameter δ ensures that the cost will diverge only when $\kappa = 1$, indicating a red zone that must be strictly avoided. Otherwise, the cost will remain finite, allowing for movement through congested regions, albeit with increasing penalties.

In addition, the discount term accounts for temporal uncertainty, acknowledging the possibility of actual congestion having temporal variance. Such consideration enhances the robustness of the cost map, reducing its sensitivity to operational uncertainty during execution. This will be further strengthened in the upcoming subsection.

Having $\mathcal{C}(\Omega)$ available, each UAS operator can plan their paths π_i by solving the following optimization problem:

$$\begin{aligned} &\min_{\pi_i, t_i^g} \sum_{j=0}^{g} \mathcal{C}(\pi_i^j) \\ \text{subject to} \quad &\pi_i^0 = (t_i^0, x_i^0), \quad \pi_i^g = (t_i^g, x_i^g) \\ &t_i^{j+1} = t_i^j + dt, \quad \|x_i^{j+1} - x_i^j\| \le l_i, \end{aligned} \tag{6}$$

where π_i^j indicates the j^{th} element of π_i, and l_i is the maximum distance (measured in $\mathcal{X}$) each UAS can travel per time interval. Each UAS operator can choose from a broad range of algorithms to obtain an optimal or suboptimal solution to this optimization problem [22], [23].

When a UAS operator cannot perform a computationally complex optimization algorithm, a suboptimal solution can be readily obtained by solving

$$\begin{aligned} x_i^{j+1} = &\arg\min_x \mathcal{C}(t_i^j + dt, x) \\ &\text{subject to } \|x - x_i^j\| \le l_i. \end{aligned} \tag{7}$$

A major drawback of this method is that it can create a locally minimum cost. This can be prevented by adopting methods in [21] and related references therein.

Our UTM framework achieves scalability by inducing each UAS operator to sequentially solve the optimization problem, utilizing a recursively updated congestion map that includes the decisions up to the previous operator. Thus, the computational complexity is kept constant regardless of the number of operations in the traffic. Moreover, this approach provides flexibility for operators to select algorithms that align with their computational capacity. In essence, the ANSP manages the UAS operations queue and updates the airspace congestion, while the path optimization task is delegated to each UAS operator.

D. UAS Operation: Time Critical Execution

Our UTM framework assumes that each UAS is capable of following its reported plan. However, a major risk of this assumption is that each UAS operation path is planned on top of previously planned operations, and if any one of the operations deviates from its reported path, the actual congestion will become erroneous accordingly. While Algorithm 1 allows for flexibility in aborting and re-planning capability, it is still important to ensure that each UAS operation stays as close as possible to its reported path within a certain tolerance. To address this issue, we introduce a time-critical feedback control law.

Once the control points, or $\{x_i^0, \cdots, x_i^g\}$ in our problem, are determined, a set of Bézier curves lying in the convex hull of the control points can be obtained via methods like the *de Casteljau* algorithm [24]. These curves can be considered as actual spatial trajectories that each vehicle may follow.

Denote the trajectory as

$$p_i(t_\mu) : [0,1] \to \mathbb{R}^2, \ \ i \in \{1, 2, \cdots, N\},$$

where t_μ represents the mission time that parameterizes a trajectory p_i. Note that t_μ is different from actual time such that $p_i(0) = x_i^0$ and $p_i(1) = x_i^g$. In this setup, dynamic constraints regarding maximum speed and acceleration can be written as

$$\begin{aligned} &\|\frac{dp_i(t_\mu)}{dt_\mu}\| \le v_{\mu_i,\max} < v_{i,\max}, \\ &\|\frac{d^2p_i(t_\mu)}{dt_\mu^2}\| \le a_{\mu_i,\max} < a_{i,\max}, \ \ i \in \{1, 2, \cdots N\}. \end{aligned} \tag{8}$$

Here $v_{i,\max}, a_{i,\max}$ are the maximum speed and acceleration that the i^{th} vehicle can follow, and $v_{\mu_i,\max}, a_{\mu_i,\max}$ are conservative design values.

Remark 3. *We assume that the trajectory avoids all red zones. If no such trajectory exists, one may add intermediate destinations between control points until there is one.*

Let the *virtual time* be represented as

$$\zeta_i : [0, \infty) \to [0, 1], \ \ i \in \{1, 2, \cdots N\}.$$

Here $[0, 1]$ represents the domain of the parameter t_μ that parameterizes the trajectory p_i. Therefore, the desired position of the i^{th} vehicle at actual time t can be expressed as $p_i(\zeta_i(t))$, where $\zeta_i(t)$ represents the progress of the vehicle along its initial planned trajectory.

Consequently, we can determine the pace of the mission for the i^{th} vehicle by calculating $\dot{\zeta}_i(t)$. If $\dot{\zeta}_i(t) = 1$, it indicates that the command speed coincides with the desired speed chosen during the offline planning. If $\dot{\zeta}_i(t) > 1$ or $\dot{\zeta}_i(t) < 1$, it means that the mission is executed faster or slower than planned, respectively.

Dynamic constraints written with respect to virtual time (8) can be rewritten in terms of actual time t as follows:

$$\begin{aligned} &\|\frac{dp_i(\zeta_i(t))}{dt}\| \le v_{i,\max}\dot{\zeta}_i(t), \\ &\|\frac{dp_i(\zeta_i(t))}{dt}\| \le a_{i,\max}\dot{\zeta}_i^2(t) + v_{\mu_i,\max}\ddot{\zeta}_i(t). \end{aligned}$$

At this point, we can satisfy the dynamic constraints by properly controlling $\dot{\zeta}_i(t)$ and $\ddot{\zeta}_i(t)$.

Time criticality is achieved at time t if each vehicle is progressing at the desired mission rate $\dot{\zeta}_{\mu_i} > 0$, or formally

$$\dot{\zeta}_i(t) = \dot{\zeta}_{\mu_i}(t), \ i \in \{1, 2, \cdots, N\}.$$

We use the time-criticality formulation to decide the mission abortion of the i^{th} agent, given by

$$\|\zeta_i(t) - \zeta_{\mu_i}(t)\| > \theta, \tag{9}$$

where θ is a threshold parameter set by the user.

Whenever the mission is not aborted, we use the following feedback control law (for a double integrator system):

$$u(t) = \ddot{\zeta}_i(t) = -b(\dot{\zeta}_i(t) - \dot{\zeta}_{\mu_i}(t)) + \ddot{\zeta}_{\mu_i}(t) - a \sum_{\eta \in \mathcal{N}_i} (\zeta_i(t) - \zeta_\eta(t)) - \bar{\alpha}(t), \quad (10)$$

where $\zeta_i(0) = \zeta_{\mu_i}(0) = t_i^0$, and $\dot{\zeta}_i(0) = \dot{\zeta}_i(0) = 1$. Moreover, a, b are positive feedback gains, and

$$\bar{\alpha}(t) = \frac{\dot{p}_i(\zeta(t))^\top (p_i(\zeta(t) - p_i(t)))}{\|\dot{p}_i(\zeta(t))\| + \delta_v},$$

where $\delta_v \in \mathbb{R}_{\geq 0}$, relates the virtual time $\zeta_i(t)$ to the actual position of the vehicle.

The term $a \sum_{\eta \in \mathcal{N}_i} (\zeta_i(t) - \zeta_\eta(t))$ facilitates the communication and coordination among UAS within their local neighborhood $\mathcal{N}_i$, adjusting the acceleration to align with the *relative* path plans. This coordination mechanism ensures that if one mission experiences a delay, other agents synchronize with the mission, mitigating the impact of a delay by distributing the effects at a manageable level. Furthermore, operations deviating significantly from their original paths are aborted based on the threshold (9), limiting the influence of a perturbation on neighboring vehicles.

IV. Simulation and Results

This section presents simulation scenarios to evaluate the UTM framework's effectiveness in mitigating congestion. We assess the RFTT scheme's efficacy in a simple environment and use satellite imagery of a real city to conduct a multi-agent simulation in ROS, allowing for a more accurate representation of real-world scenarios.

For the RFTT experiment described next, we use the parameters outlined in Table I. Here, $l \equiv l_i \ \forall i \in \{1, \ldots, N\}$. Moreover, we inject noise $\omega \sim \mathcal{N}(0, 0.1)$ into each control channel of the vehicle. Then, we designate each control point as an intermediate destination to ensure the avoidance of red zones. Each drone adheres to the physical constraints[1].

TABLE I: Simulation parameters.

t_0	0	n_y	30	k_{att}	2.5	a	0	γ	0.8
t_∞	500	ϵ_1	0.1	k_{rep}	2	b	1	δ	1
n_x	30	ϵ_2	10^{-6}	λ	10	l	2	δ_v	0.1

We assess the performance of RFTT in a simplified grid environment (Fig 1). In this scenario, we set $\rho(t, x) \equiv 1$, $q_{\max} = 2$, and $N = 4$. The top row of the evaluation highlights the limitations of optimizing paths when the temporal aspect of congestion is not taken into account. The red agent follows a convoluted path, while the orange agent spends unnecessary time in a yellow zone. These behaviors result in higher cumulative and maximum airspace congestion, respectively. In contrast, the bottom row demonstrates the effectiveness of RFTT in reducing congestion.

[1] Detailed in https://ardupilot.org/copter/docs/auto-mode.html

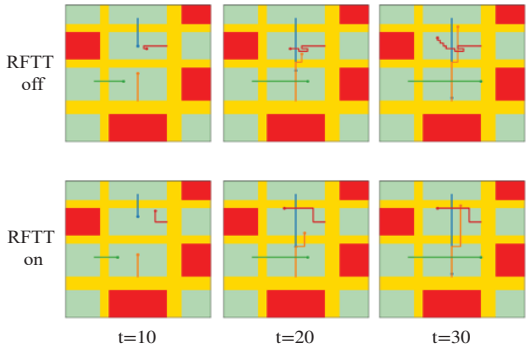

Fig. 1: Path planning comparison with and without RFTT scheme. Both scenarios followed (7) for the optimization. Different regions are represented by color codes: red for $\mathcal{R}$, yellow for $\mathcal{Y}$, and green for $\mathcal{G}$ zones.

The advantages of our RFTT scheme in mitigating congestion become increasingly evident as the number of UAS operations grows. To showcase this, we updated the parameters to $\rho(t, x) \equiv 5$, $q_{\max} = 10$, and $N = 100$. Moreover, we initialized each operation by sampling $t_i^0 \sim \mathcal{U}[0, 50]$, $x_i^0 \sim \mathcal{U}[0, 30]/\mathcal{R}$, and $x_i^g \sim \mathcal{U}[0, 30]/\mathcal{R}$, where $\mathcal{U}[r_1, r_2]$ represents a uniform distribution over the interval $[r_1, r_2]$, and $/\mathcal{R}$ indicates that we resampled whenever $x_i^{\{0,g\}} \in \mathcal{R}$. Congestion induced with and without RFTT was compared, and an additional limit on its travel distance per time interval was not imposed. As indicated by Fig. 2, the case without RFTT (RFTT-off) resulted in higher congestion levels, which persisted for a larger fraction of time compared to the case with RFTT (RFTT-on). These findings underscore the significance of incorporating congestion awareness in planning frameworks to efficiently manage airspace traffic, particularly as the number of UAS operations increases.

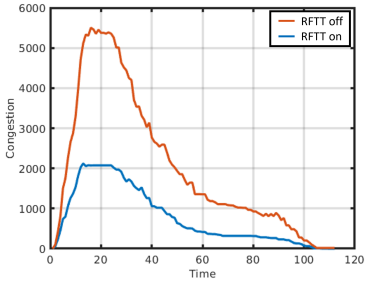

Fig. 2: Airspace congestion curve for 100 agents with random initial conditions and goals.

For a more realistic scenario, we obtained a satellite image of a portion of the city of San Carlos, California, and processed it to create a congestion map with $\mathcal{R}, \mathcal{G}, \mathcal{Y}$ zones (Figure 3). This process involves creating a 3D model of the city by comparing off-nadir satellite imagery, and then using edge detection to separate buildings from the terrain in vertical satellite images. Resulting terrain, roadway, and building models are projected vertically into polygons and color-coded based on their respective zoning to generate an airspace capacity map. Using the constructed airspace capacity, we created a congestion map for operations with

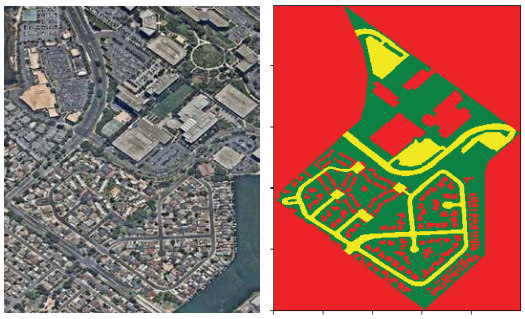

Fig. 3: (Left) Satellite image of San Carlos, CA. (Right) Permissible zones include parks and empty spaces, while pedestrian roads and vehicular routes are classified as restricted zones. Private properties and hospitals fall under the prohibited zone category, where flying over is not allowed.

$\rho(t,x) \equiv 1$, $q_{\max} = 2$, and $N = 10$. The paths taken by each UAS were visualized in Figure 4. This comprehensive demonstration illustrates the practicality and feasibility of our framework in a real-world environment.

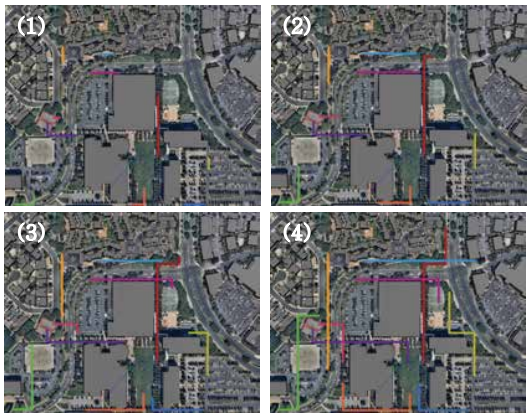

Fig. 4: Actual path following of 10 UAS operations displayed over a satellite image of San Carlos, CA.

V. Conclusion

In this study, we proposed a novel end-to-end UTM framework that addresses the challenges of effectively managing airspace. Our approach encodes guidelines and regulations in the definition of airspace capacity from which airspace congestion is defined. Then, each UAS operation plans its path based on the recursively updated cost map for scalability. Our time-critical control scheme enables UAS operators to conform to their plans and prevent congestion errors from affecting subsequent operations.

The proposed concept was validated through realistic simulations, but there are limitations and opportunities for future research. The current work lacks scenarios involving replanning and aborting, which necessitates further investigation. Moreover, we aim to analytically derive the sensitivity of the cost map to UAS trajectory deviation, an essential aspect to understand how deviations might impact the overall system through the cascaded UTM layers.

References

[1] B. Custers, "Flying to new destinations: the future of drones," in *The future of drone use*. Springer, 2016, pp. 371–386.

[2] R. Rumba and A. Nikitenko, "The wild west of drones: A review on autonomous-UAV traffic-management," in *International Conference on Unmanned Aircraft Systems*. IEEE, 2020, pp. 1317–1322.

[3] M. S. Baum, *Unmanned Aircraft Systems Traffic Management: UTM*. CRC Press, 2021.

[4] D. Jia and J. Vagners, "Parallel evolutionary algorithms for UAV path planning," in *Intelligent Systems Technical Conference*, 2004, p. 6230.

[5] A. Poty, P. Melchior, and A. Oustaloup, "Dynamic path planning for mobile robots using fractional potential field," in *International Symposium on Control, Communications and Signal Processing*. IEEE, 2004, pp. 557–561.

[6] A. Xu, C. Viriyasuthee, and I. Rekleitis, "Optimal complete terrain coverage using an unmanned aerial vehicle," in *International Conference on Robotics and Automation*, 2011, pp. 2513–2519.

[7] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," in *European Control Conference*. IEEE, 2019, pp. 3420–3431.

[8] J. S. Grover, C. Liu, and K. Sycara, "Deadlock analysis and resolution for multi-robot systems," in *International Workshop on the Algorithmic Foundations of Robotics*. Springer, 2021, pp. 294–312.

[9] J. Ji, A. Khajepour, W. W. Melek, and Y. Huang, "Path planning and tracking for vehicle collision avoidance based on model predictive control with multiconstraints," in *Transactions on Vehicular Technology*, vol. 66, no. 2. IEEE, 2016, pp. 952–964.

[10] Y. Zhou and J. S. Baras, "Reachable set approach to collision avoidance for UAVs," in *Conference on Decision and Control*. IEEE, 2015, pp. 5947–5952.

[11] Q. Yuan, S. Li, C. Wang, and G. Xie, "Cooperative-competitive game based approach to the local path planning problem of distributed multi-agent systems," in *European Control Conference*. IEEE, 2020, pp. 680–685.

[12] J. F. Fisac, E. Bronstein, E. Stefansson, D. Sadigh, S. S. Sastry, and A. D. Dragan, "Hierarchical game-theoretic planning for autonomous vehicles," in *International Conference on Robotics and Automation*. IEEE, 2019, pp. 9590–9596.

[13] C. Chin, M. Z. Li, and Y. V. Pant, "Distributed traffic flow management for uncrewed aircraft systems," in *International Conference on Intelligent Transportation Systems*. IEEE, 2022, pp. 3625–3631.

[14] Y. V. Pant, M. Z. Li, A. Rodionova, R. A. Quaye, H. Abbas, M. S. Ryerson, and R. Mangharam, "Fads: A framework for autonomous drone safety using temporal logic-based trajectory planning," *Transportation Research Part C: Emerging Technologies*, vol. 130, p. 103275, 2021.

[15] D. Vyas, R. Patel, and A. Ganatra, "Survey of distributed multipath routing protocols for traffic management," *International Journal of Computer Applications*, vol. 63, no. 17, 2013.

[16] R. She and Y. Ouyang, "Efficiency of UAV-based last-mile delivery under congestion in low-altitude air," *Transportation Research Part C: Emerging Technologies*, vol. 122, p. 102878, 2021.

[17] J. Zhou, L. Jin, X. Wang, and D. Sun, "Resilient UAV traffic congestion control using fluid queuing models," *Transactions on Intelligent Transportation Systems*, vol. 22, no. 12, pp. 7561–7572, 2020.

[18] L. Yang, J. Qi, J. Xiao, and X. Yong, "A literature review of UAV 3d path planning," in *World Congress on Intelligent Control and Automation*. IEEE, 2014, pp. 2376–2381.

[19] P. Kopardekar, J. Rios, T. Prevot, M. Johnson, J. Jung, and J. E. Robinson, "Unmanned aircraft system Traffic Management (UTM) concept of operations," in *AIAA Aviation and Aeronautics Forum*, no. ARC-E-DAA-TN32838, 2016.

[20] M. Aftabuzzaman, "Measuring traffic congestion - A critical review," in *Australasian Transport Research Forum*. ETM GROUP London, UK, 2007, pp. 1–16.

[21] A. Lazarowska, "Discrete artificial potential field approach to mobile robot path planning," *IFAC-PapersOnLine*, vol. 52, no. 8, pp. 277–282, 2019.

[22] L. A. Wolsey, *Integer programming*. John Wiley & Sons, 2020.

[23] S. Aggarwal and N. Kumar, "Path planning techniques for unmanned aerial vehicles: A review, solutions, and challenges," in *Computer Communications*, vol. 149. Elsevier, 2020, pp. 270–299.

[24] R. Choe, J. Puig-Navarro, V. Cichella, E. Xargay, and N. Hovakimyan, "Cooperative trajectory generation using Pythagorean hodograph Bézier curves," in *Journal of Guidance, Control, and Dynamics*, vol. 39, no. 8. AIAA, 2016, pp. 1744–1763.