PREPRINT

Potential Flow Theory based Guidance Algorithm for 3D Obstacle Avoidance in Cluttered Urban Environments

M. Kürşat Yurt* , Zeynep Bilgin† , and İlkay Yavrucuk ‡
Technical University of Munich, Munich, Germany, 85748

Murat Bronz §
ENAC, University of Toulouse, Toulouse, France, 31055

In this paper, an existing guidance method based on potential flow theory is enhanced through the incorporation of 3D models of obstacles, thereby allowing for the optimal utilization of available airspace in urban air mobility applications. The proposed 3D guidance method addresses the limitations of the previous 2D approach, enabling navigation around complex obstacles such as tunnels and torus-like structures. The effectiveness of the proposed algorithm is demonstrated through scaled hardware experiments conducted at The Toulouse Occitanie Drone Flight Area in France. The experimental results demonstrate the successful real-time guidance and collision avoidance capabilities of the proposed algorithm in cluttered environments for multiple aerial vehicles, with an effective utilization of vertical and horizontal space. This improvement makes the proposed 3D guidance algorithm a suitable candidate for urban air mobility operations. Supplementary Videos: `https://www.youtube.com/watch?v=A-gMgseRrsk&list=PLnodu_kCLOJ7B6ERxjkFYDNH2yd0ujseq`

I. Nomenclature

d	=	Euclidean distance
v	=	Flow velocity induced by potential element
v_x, v_y and v_z	=	Flow velocity components
N	=	Number of vehicles
v_{vehicle}	=	Induced velocity by vehicle
v_{target}	=	Induced velocity by target
ϕ	=	Potential Function
$\xi - \eta$	=	Local coordinate system axes
σ	=	Source strength
Subscript	=	**Meaning**
∞	=	Free stream
i and j	=	Panel number

II. Introduction

Ground traffic congestion is a growing problem in densely populated cities and urban areas. The 2019 Global Traffic Scorecard reports that, on average, American drivers lose 99 hours a year due to traffic congestion, costing them nearly $ 88 billion in 2019[1]. One promising solution to the challenges of urban mobility is expanding urban transportation to urban airspace. Shifting a portion of urban traffic from the ground to the air can reduce carbon emissions and traffic congestion while also presenting a business opportunity as the global urban air mobility (UAM) market is projected to grow from USD 4.21 billion in 2024 to USD 14,64 billion by 2032 [2]. This innovative mobility solution draws the attention of companies and governments alike. However, this mobility solution is not without its

*PhD Candidate, Chair of Rotorcraft and Vertical Flight, kursat.yurt@tum.de
†PhD Candidate, Chair of Rotorcraft and Vertical Flight, zeynep.bilgin@tum.de
‡Professor, Chair of Rotorcraft and Vertical Flight , ilkay.yavrucuk@tum.de
§Assistant Professor, ENAC Lab - OPTIM - Dynamic Systems, murat.bronz@enac.fr, AIAA Member

complexities. The introduction of urban air mobility in densely populated cities indicates aerial vehicles, varying in design, size, maneuverability, automation level, and communication capabilities, competing for the same limited urban airspace that was previously unregulated. Ensuring collision avoidance between urban aircraft is essential for UAM traffic safety, which is crucial for enabling large-scale UAM operations. Incorporating vehicles capable of generating collision-free paths enhances urban airspace's capacity and reduces the efforts required for urban air traffic management [3]. Therefore, collision-free guidance in urban airspace is an important issue that needs to be addressed. The guidance problem in urban airspace involves multiple vehicles operating within the same limited space. Additionally, operations may need to occur at lower than usual altitudes which requires complex architectural structures to be considered. Consequently, autonomous navigation of aerial vehicles in urban airspace becomes a complex issue involving multiple agents in a cluttered environment. Recent studies have addressed path planning problems in urban environments [4–7].

A classical path planning approach for autonomous vehicles is the artificial potential field method where an artificial potential field is defined to guide the vehicle towards the goal position while avoiding obstacles [8–13]. One major disadvantage of artificial potential fields is that they may have a local minima that can trap the vehicle before it can reach its destination. Furthermore, in urban air mobility applications, obstacles can have complex shapes; thus, classical potential field methods using circular representations may become inefficient. To overcome this restriction, a local path planning approach called the Collision Cone approach has been proposed in the literature for collision avoidance in dynamic environments with arbitrarily shaped obstacles [14–16]. This method searches for a safe avoidance maneuver by considering the velocity of the obstacles without explicitly building a map or considering long-term goals. Alternatively, a global path planning approach that uses harmonic potential fields can be extended to arbitrarily shaped obstacles. Harmonic potential fields do not suffer from the local minima limitation and produce smooth trajectories in static and dynamic environments [17–22]. One specific harmonic potential field that can be used in guidance is the potential field of an irrotational flow of an ideal fluid [13, 23–25]. Flow field around any arbitrarily shaped obstacle can be defined using a numerical method, namely the panel method borrowed from the fluid mechanics domain [25, 26]. Panel method was previously used for motion planning for both ground robots [18, 27, 28] and aerial vehicles [29–31]. Harmonic potential field-based guidance is also considered for urban air mobility applications both for autonomous guidance methods and as an advisory service to be used for air traffic control [32–36].

Text

This paper focuses on further improving the potential flow theory-based guidance algorithm proposed in [30], named Potential Guidance Flow (PGFlow), by adding 3D models of obstacles. Having 3D obstacle models can fully utilize available air space and enable additional avoidance maneuvers in the third dimension. Previously, ascent and descent maneuvers were studied using 3D flow elements together with a 2D cross-section of obstacles [36]. This approach is limited in its ability to utilize the entire available space for gates and torus-like obstacles. In contrast, when 3D models of the obstacles are used, potential flow through tunnels or complex torus-like 3D obstacles can be calculated.

Conducting experiments is an essential component of evaluating the performance of any guidance method. This paper presents the results of scaled hardware experiments conducted to demonstrate the effectiveness of the proposed 3D PGFlow method. The experiments were conducted at the Toulouse Occitanie Drone Flight Area in France. This facility offers the unique opportunity to conduct experiments with micro-drones and embedded systems in one of the most hi-tech instrument-equipped indoor flight zones in Europe [37].

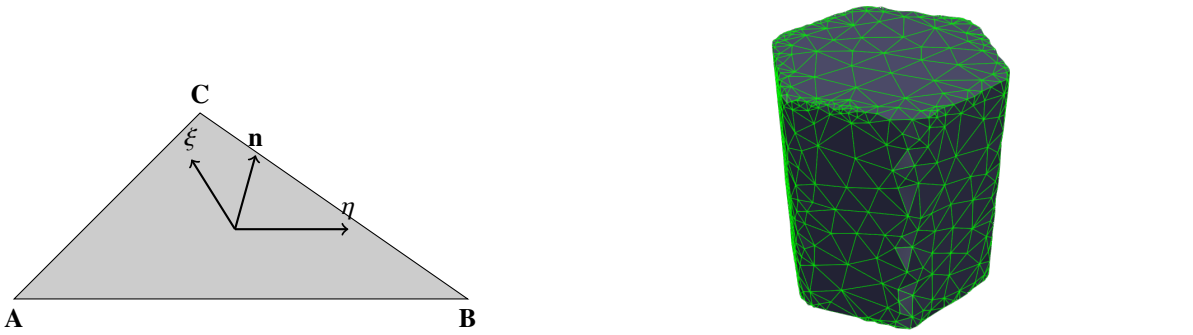
III. Methodology

In this paper, the guidance algorithm previously proposed in [30] is further extended for 3D applications. The proposed guidance algorithm, named Potential Guidance Flow (PGFlow), utilizes the potential flow theory from the aerodynamics domain. The incompressible, irrotational flow of a fluid around an arbitrarily shaped submerged body can be modeled by a harmonic potential function [25, 26]. It is a fundamental principle of fluid dynamics that flow cannot enter into solid objects. Consequently, following the fluid streamlines provides a means of ensuring that collisions are avoided and that the path taken is smooth and collision-free.

A. Potential Flow Theory Preliminaries

Potential flow theory is used in aerodynamics to describe the flow of an incompressible, irrotational, ideal fluid [25, 26]. The flow is defined by a scalar field called the velocity potential that satisfies the Laplace's Equation. The flow field around simple shapes such as circles and ellipses can be solved analytically [25, 26]. To model the flow around arbitrarily shaped bodies, numerical techniques called panel methods are employed. The approach involves partitioning the surface of an object into discrete boundary elements known as 'panels'. This is the origin of the term 'panel method'. The

method utilizes the superposition of each panel to compute the flow velocity field around the object [25, 26, 38].



(a) A Triangular Panel and its Local Coordinate System. **(b) Sample Surface Triangulation for a Hexagonal Building.**

Fig. 1 Computational Representation of Obstacles.

Following [25, 38], let $\mathcal{D}$ be a region in $\mathbb{R}^3$ containing an ideal fluid and let $\boldsymbol{P} = (x, y, z)^T$ be an arbitrary point in $\mathcal{D}$. A plane source element in the $\xi - \eta$ (or $x - y$) plane is illustrated in Figure 1a. The velocity induced at point $\boldsymbol{P}$ is given by Equation 1 for a unit source density σ.

$$
\begin{aligned}
v_x &= -\frac{\partial \phi}{\partial x} \iint\limits_A \frac{(x-\xi)d\xi d\eta}{d^3} \\
v_y &= -\frac{\partial \phi}{\partial y} \iint\limits_A \frac{(y-\eta)d\xi d\eta}{d^3} \\
v_z &= -\frac{\partial \phi}{\partial z} \iint\limits_A \frac{z d\xi d\eta}{d^3}
\end{aligned}
\tag{1}
$$

Where ϕ is the potential function, v_x, v_y, and v_z are the induced velocity components, and d is the Euclidian distance of a point from the origin of the local coordinate system. The exact integrals of Equation 1 are provided by [38] and used in this paper. For each boundary element, the coordinates are first transformed to the panel local coordinates $\xi - \eta$, and the induced velocity is calculated and transformed to the inertial coordinates.

The target points and vehicles are represented by point sink/source elements. For the element intensity of σ, the induced velocity at point $\boldsymbol{P}$ is given in Equation 2.

$$
\begin{aligned}
v_x &= -\frac{\sigma(x-x_0)}{4\pi[(x-x_0)^2+(y-y_0)^2+(z-z_0)^2]^{\frac{3}{2}}} \\
v_y &= -\frac{\sigma(y-y_0)}{4\pi[(x-x_0)^2+(y-y_0)^2+(z-z_0)^2]^{\frac{3}{2}}} \\
v_z &= -\frac{\sigma(z-z_0)}{4\pi[(x-x_0)^2+(y-y_0)^2+(z-z_0)^2]^{\frac{3}{2}}}
\end{aligned}
\tag{2}
$$

where $(x_0, y_0, z_0)^T$ is the element location, and v_x, v_y, and v_z are the induced velocity components in inertial coordinate system.

Initially, the strengths of the obstacle panel are unknown. In order to obtain the unknown panel strengths, a Dirichlet boundary condition is applied at the center location of each panel, such that the velocity component normal to the obstacle surface must be zero. This problem can be reduced to a system of linear equations, as given in Equation 3.

$$
\begin{pmatrix}
a_{11} & a_{12} & \dots & a_{1i} \\
a_{21} & a_{22} & \dots & a_{2i} \\
\vdots & & \ddots & \vdots \\
a_{i1} & a_{i2} & \dots & a_{ii}
\end{pmatrix}
\begin{pmatrix}
\sigma_1 \\ \sigma_2 \\ \vdots \\ \sigma_i
\end{pmatrix}
=
\begin{pmatrix}
RHS_1 \\ RHS_2 \\ \vdots \\ RHS_i
\end{pmatrix}
\tag{3}
$$

The coefficients are calculated by taking the dot product of the velocity induced per unit strength by each panel j on the other panel i, v_{ij} and panel normal vector n_i as given in Equation 4.

$$a_{ij} = v_{ij} \cdot n_i \quad \text{for} \quad \sigma_j = 1.0 \tag{4}$$

The right-hand side vector is given in Equation 5, where v_∞ is freestream velocity, and n_i is the normal vector of the ith panel.

$$RHS_i = -v_\infty \cdot n_i \tag{5}$$

B. Potential Flow Theory Application in Guidance Problem

Potential flow theory provides a fluid potential field around submerged bodies. This potential field does not have any local minima by definition [17, 18, 25]. Furthermore, the flow does not penetrate the obstacle surfaces due to the boundary condition forced by the panel method equations. These two inherent properties make potential flow theory a promising candidate for guidance applications. To better fit the needs of the guidance problem, the right-hand side vector given in Equation 5 is further modified. The modified RHS vector is presented in Equation 6.

$$RHS_i = -\left(\sum_{j=1}^{N_{\text{vehicle}}} v_{\text{vehicle}} + v_\infty + v_{\text{target}} \right) \cdot n_i \tag{6}$$

Here, in addition to the free stream velocity v_∞, the velocity induced by the point sink element representing the goal v_{target} and the point source elements representing the vehicles v_{vehicle} in the flight arena are also included. The scalar value is once more obtained by taking the dot product with the panel normal vector n_i and the total induced velocity.

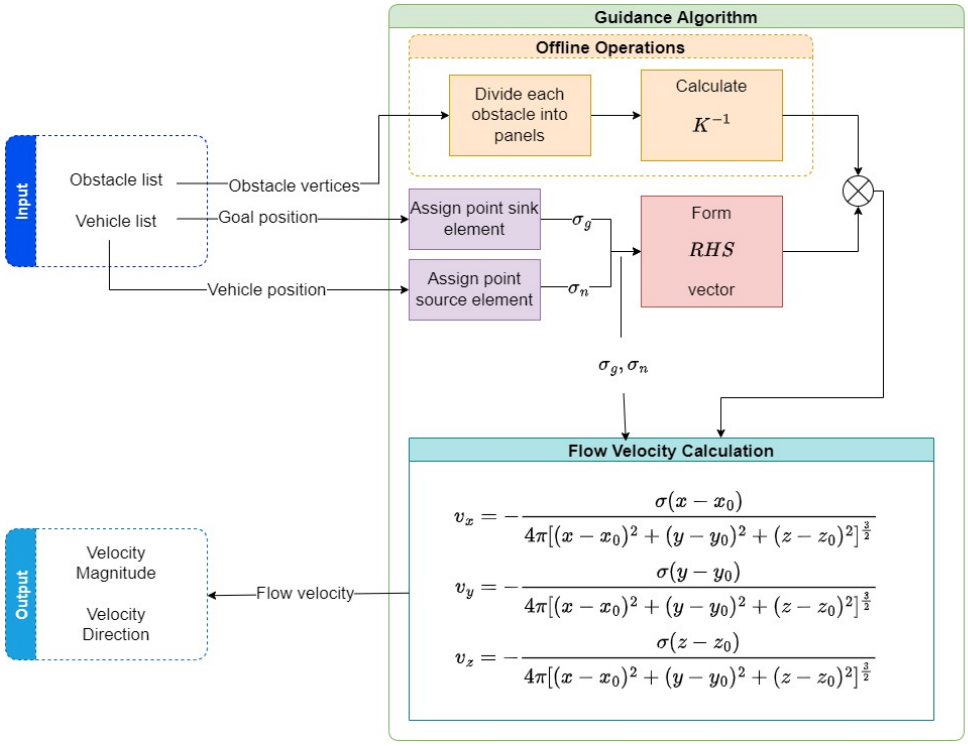


Fig. 2 Block diagram for Potential Guidance Flow algorithm.

1. Defining Strengths of Target and Other Vehicles

The attractive velocity induced by the point sink element placed at the target position allows the vehicle to track a dynamic target or arrive at a static destination. However, the use of a constant-strength sink can result in two primary

issues. First, if the strength is insufficient, the resultant velocity vector may be influenced by repulsive velocities induced by other vehicles. Second, if the strength is excessive, it can impede the ability of the vehicles to avoid each other.
In this paper, a function of the Euclidean distance d between the vehicles and the target is used to adjust the source and sink strengths as given in Equation 7 and 8.

$$\sigma_{\text{target}}(d) = -d^3 \tag{7}$$

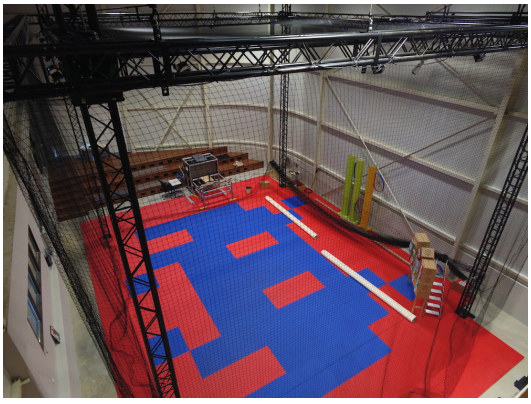
$$\sigma_{\text{vehicle}}(d) = \frac{1}{d^3} \tag{8}$$

This mechanism establishes a linear relationship between the target point and the vehicle, ensuring the attractive velocity is not overly dominant. Similarly, the repulsive velocity between vehicles decays with distance, preventing the cancellation of the target's attractive velocity by the repulsive velocity when another vehicle is near the target point. Additionally, it prevents excessive repulsive velocity from obstacles when another vehicle is close to the obstacles but far from the vehicle.

2. Calculating the Guidance Flow Velocity
The guidance flow velocity is calculated in three steps for each vehicle. In the first step, the target sink strength is determined using Equation 7. The velocity induced by the sink element is calculated. In the second step, using Equation 8 for each vehicle, a source strength is assigned, and their induced velocity is summed. In the last step, using the target point and all the vehicles, the right-hand-side vector is calculated using Equation 6. The strength of each surface panel is calculated by solving the system Equation 3.
Finally, the velocity induced by the buildings is superposed to other velocities obtained in the previous steps. A block diagram for the guidance algorithm and flow velocity calculation is presented in Figure 2. The output of the guidance algorithm is the magnitude and the direction of the flow velocity. This velocity is normalized and used in an automatic flight controller as a drone direction vector.

C. Flight Test Setup

The flight tests are conducted at the Toulouse Occitanie Drone Flight Arena of ENAC, which consists of an 8 m × 8 m × 8 m flight volume and is equipped with a high-precision 16-camera Optitrack system for the real-time tracking of quadrotor motion. The flight arena is shown in Figure 3a. Obstacles with various shapes and heights are built and inserted into the flight area to model a scaled urban environment. The scaled urban environments created for experiments 3 and 4 are depicted in Figure 4.

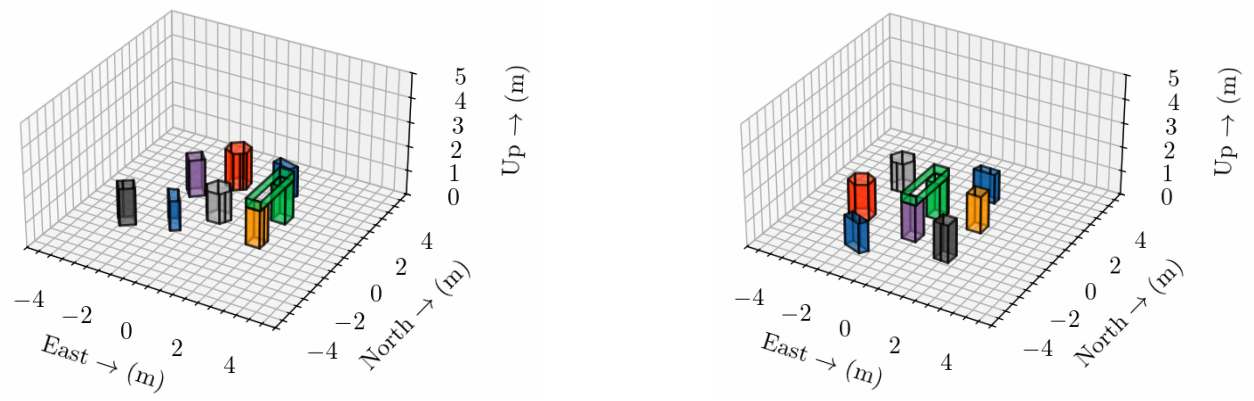


(a) The Toulouse Occitanie Drone Flight Arena.

(b) DJI Tello EDU Drone.

Fig. 3 Experimental Setup.

The DJI Tello EDU quadrotor system shown in Figure 3b is used in the flight test. Quadrotors are operated through the ground station with a 2.4 Ghz 802.11n Wi-Fi connection. The maximum achievable speed of the drone is eight meters per second. The drone weighs 87 grams and is powered by a battery. During the flight test, the position of the vehicles is tracked and updated with a 100Hz refresh rate by the Optitrack system. The reference velocity vectors are calculated at



(a) Experiment 3. **(b) Experiment 4.**

Fig. 4 Scaled Urban Environments from Experiments.

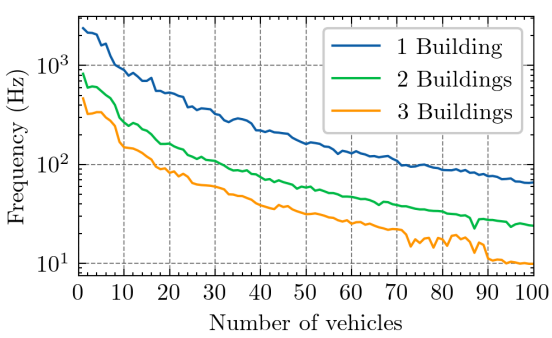
40 Hz and sent as input to the drones. The DJI-Tello quadrotors accept translational velocity as an input, and the internal controller tries to follow the given command by its internal velocity estimation from a downward-looking optic-flow camera system.

D. Scalability of The Algorithm

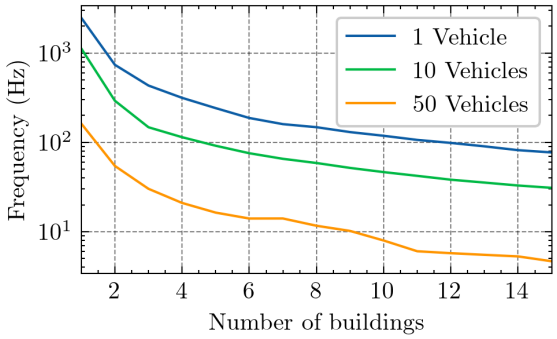
In contrast to the method employed in [30, 36], where buildings are represented using 2D cross-sections and corresponding 2D flow elements, the introduction of 3D obstacle models significantly enhances the complexity of flow calculations. The transition from two-dimensional to three-dimensional representation necessitates an increase in the number of panels, leading to a more detailed and accurate representation of obstacles. Consequently, this dimensional expansion results in a substantial increase in computational effort. Specifically, the calculations for the right-hand side vector now require an additional multiplication for each element, resulting in a 50% increase in computational load. Additionally, the matrix dimensions expand due to the greater number of panels needed to accurately model the obstacle in three dimensions. The computational load for matrix-vector multiplication scales as $O(n^2)$, where n is the number of elements in the matrices used in the algorithm. This scaling implies that the computational effort grows quadratically with the number of elements, further emphasizing the significant increase in computational resources required for 3D models compared to 2D approximations. The algorithm is transferred from the original Python code* to a new C++ code to handle this computational load. The C++ backend provides a Python interface, enabling the use of existing frameworks such as the motion capture and drone communication modules. This configuration offers users the flexibility to implement further enhancements and prototype new solutions without requiring any compiled software beyond the core library.

Two different scalability tests were conducted to evaluate the computational performance of the developed 3D guidance method and software. The tests were performed on a desktop PC equipped with an Intel(R) Core(TM) i7-12700K processor with 8 cores. In the initial study, the number of buildings was maintained at a constant level, while the number of vehicles was increased from zero to one hundred. In the subsequent study, the number of buildings was increased while the number of vehicles remained unchanged. Figure 5a illustrates that an increase in the number of vehicles does not significantly impact the algorithm's performance. In all cases with up to 50 drones, the algorithm operates above 30 Hz, which is satisfactory for real-life applications due to the communication rate. In consideration of a limit of 24 Hz as a requisite, cases with one and two buildings are capable of accommodating up to one hundred drones. However, when the number of buildings is increased to three, the performance of the algorithm slows to 24 Hz for 65 drones and drops to 10 Hz for 100 drones. Although the calculations between vehicles are relatively inexpensive, each additional calculation between vehicles and buildings requires a costly matrix-vector multiplication. Consequently, the number of buildings significantly affects performance, as shown in Figure 5b. As anticipated, the addition of each building significantly affects the calculation speed. While scenarios involving 1 and 10 vehicles run faster than the 24 Hz threshold, the scenario with 50 vehicles quickly reaches saturation after 3 buildings. This suggests the necessity for additional optimizations or the implementation of distributed computing capabilities to effectively handle a higher number of vehicles and buildings concurrently.

*`https://github.com/enac-drones/PGFlow`



(a) Change of Maximum Calculation Frequency with Number of Vehicles.



(b) Change of Maximum Calculation Frequency with Number of Buildings.

Fig. 5 Scalability Test Results.

IV. Flight Test Results

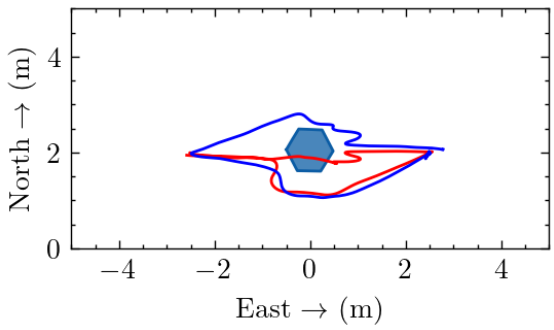
Flight tests for the proposed 3D guidance algorithm are conducted at the Toulouse Occitanie Drone Flight Arena in Toulouse, France, with Tello DJI quadrotors as shown in Figure 3.

Four different experimental scenarios with different challenges are considered in the flight test campaign to thoroughly assess the effectiveness of the proposed method. The first scenario aims to demonstrate the ability to perform 3D maneuvers of the newly proposed method in a simple setting with only one obstacle and two drones traveling to opposite sides of the arena. The second experiment demonstrates the new capability to pass through the torus-like objects - a new feature that was not possible with the previous version of the PGFlow algorithm[34, 36].

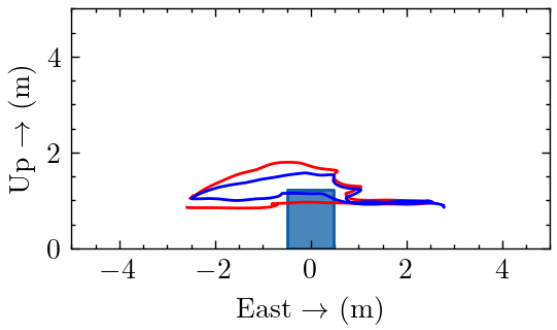
Experiments 3 and 4 are conducted in more complex settings, as shown in Figure 4. In the third flight test, the drones are tasked with navigating to the predetermined target locations in a complex environment with multiple buildings, including a gate structure introduced in experiment 2. Last but not least, in the fourth flight test scenario, the target location dynamically changes in the flight arena, and the drones follow the dynamic target in a cluttered environment.

1. Flight Test: Scenario 1

The first flight test scenario consists of a single hexagonal building and 2 drones. *Drone 0* (Red) starts its flight from $(-4E, 2N, 1U)$ to $(4E, 2N, 1U)$, while *Drone 1* (Blue) starts its flight from $(4E, 2N, 1U)$ to $(-4E, 2N, 1U)$. Upon reaching their target destinations, both drones return to their initial locations and land. The resultant trajectories of the



(a) Top View

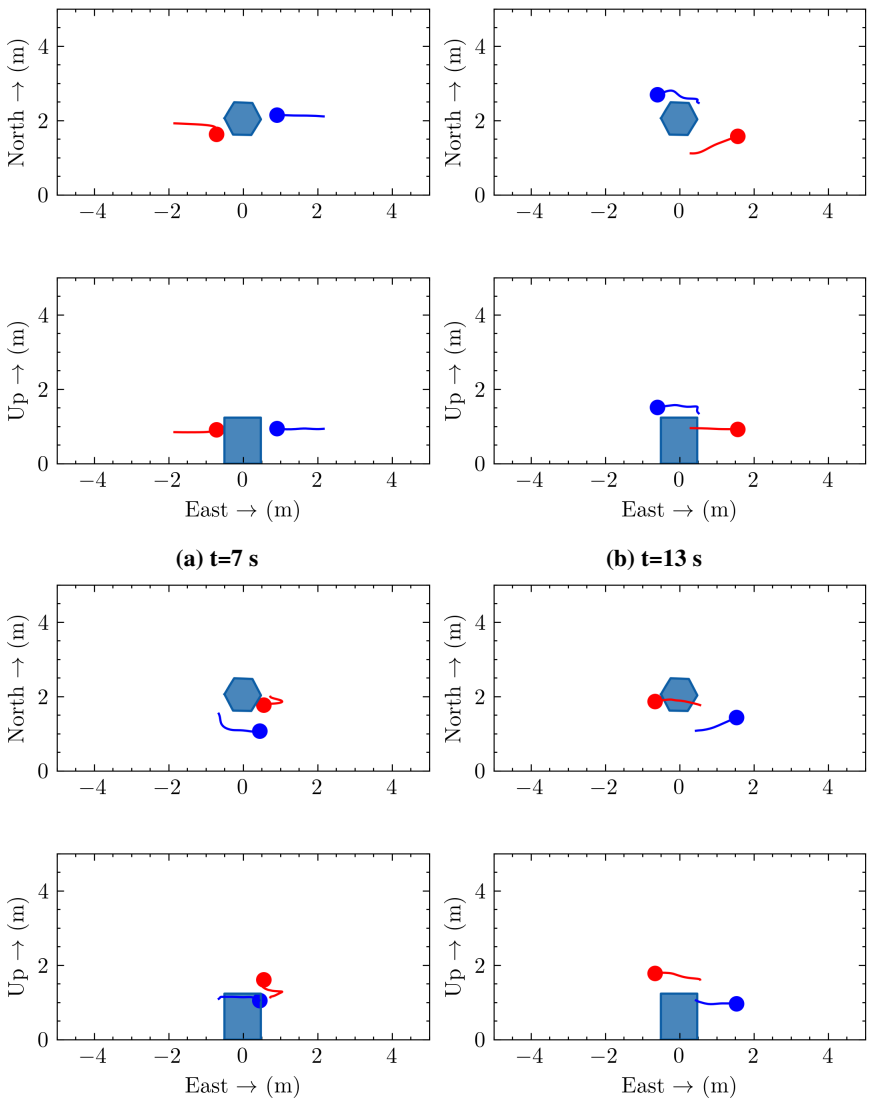


(b) Side View

Fig. 6 Resultant flight trajectories of scenario 1.

drones are given in Figure 6 in the top view and side view, clearly showing that the drones maintained a safe distance from each other.

In Figure 7, sequential positions and flight trajectories of the drones are presented. In the seventh second of the flight, the drones fly at the same altitude and approach the obstacle (see Figure 7a). Drone 1 performs an evasive maneuver



(a) t=7 s **(b) t=13 s**

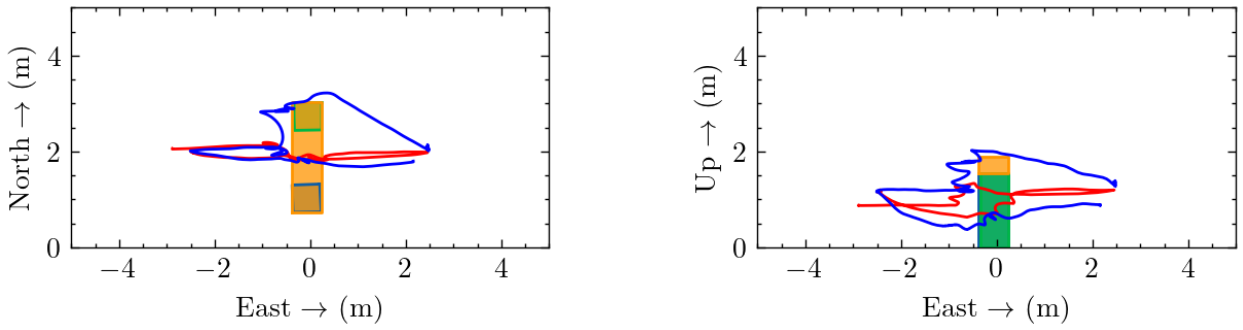
(c) t=25 s **(d) t=28 s**

Fig. 7 Flight trajectories of scenario 1.

to the south, while Drone 0 chooses a trajectory from the north and ascends to avoid the building (see Figure 7b). This maneuver was not possible with the previous version of the method presented in [30, 36] when only the 2D cross-sections of the buildings were considered. Similarly, on the return flight, Drone 1 follows a trajectory south of the building, and Drone 0 flies over the building, as depicted in Figure 7c and 7d.
The experiment demonstrates the collision-free guidance capability of the proposed method in three-dimensional space. Furthermore, drones perform avoidance maneuvers over buildings that were previously not feasible when the buildings were modeled as two-dimensional cross-sections, indicating that the proposed algorithm utilizes the available airspace in a more efficient manner.

2. Flight Test: Scenario 2

In the second flight test experiment, a gate-like structure is constructed and placed within the indoor flight arena. Similar to the previous scenario, *Drone 0* (Red) starts its flight from $(-4E, 2N, 1U)$ to $(4E, 2N, 1U)$, while *Drone 1* (Blue) starts its flight from $(4E, 2N, 1U)$ to $(-4E, 2N, 1U)$. The drones are tasked with flying to the opposite sides of the flight arena, returning to their initial positions, and landing. The resultant paths of the drones are presented in Figure 8.



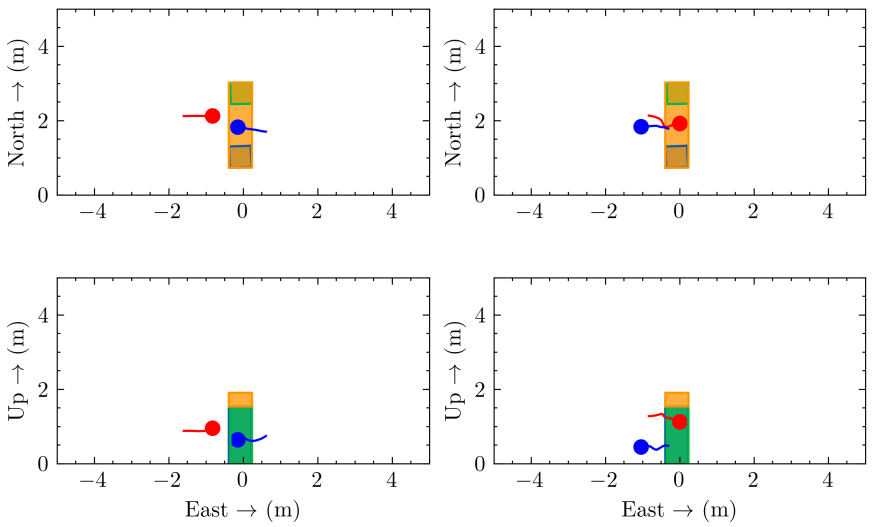
(a) Top View **(b) Side View**

Fig. 8 Resultant flight trajectories of scenario 2.

In Figure 9, sequential positions and flight trajectories of the drones are presented. On the 10th second of the flight, *Drone 1* flies inside the gate. When *Drone 0* reaches the gate, *Drone 1* dives and *Drone 0* ascends, as shown in Figures 9a and 9b. With this maneuver, both drones effectively use the available space within the gate. On the return flight, *Drone 0* flies through the gate, while *Drone 1* performs an escape maneuver to the north, climbs, flies over the gate, and reaches its destination, as shown in Figures 9c and 9d. In the absence of a three-dimensional model of the gate-like structure, the area beneath the gate is inaccessible to the guidance algorithm. Thus, this experiment demonstrates how the proposed guidance algorithm expands the available airspace for guidance maneuvers.

3. Flight Test: Scenario 3

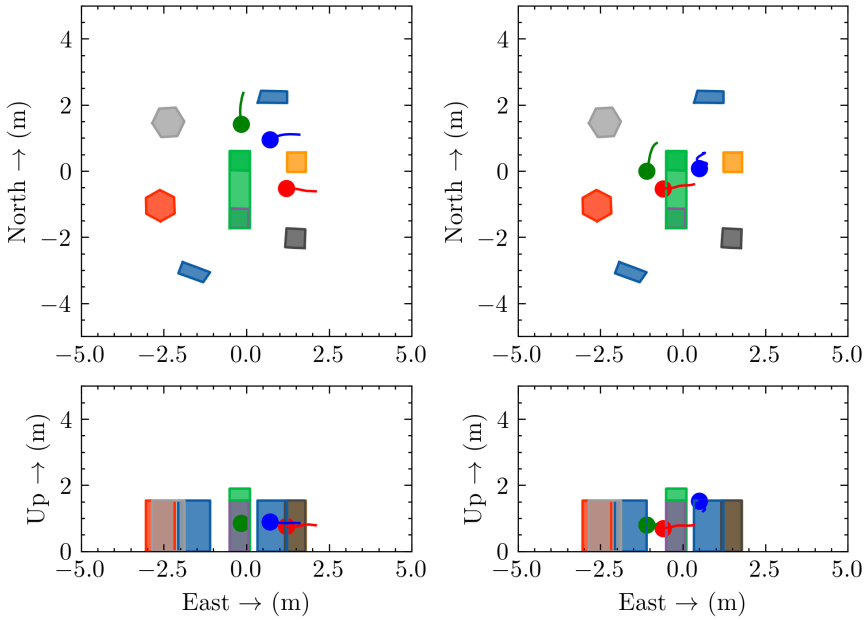
In the third scenario, the number of drones is increased to three, and the number of obstacles is increased to six. *Drone 0* (Red) initiates its flight from the southeast of the arena, traversing the northwest quadrant before returning to its point of origin. Similarly, *Drone 1* (Blue) commences its journey from the northeast, traversing the southwest quadrant and returning. Meanwhile, *Drone 2* (Green) patrols the southern and northern regions of the flight area.



(a) t = 10s **(b) t = 16s**

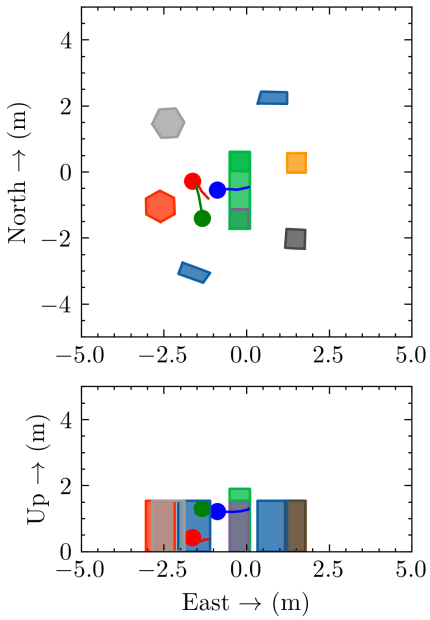
(c) t = 34s **(d) t = 43**

Fig. 9 Flight trajectories of scenario 2.

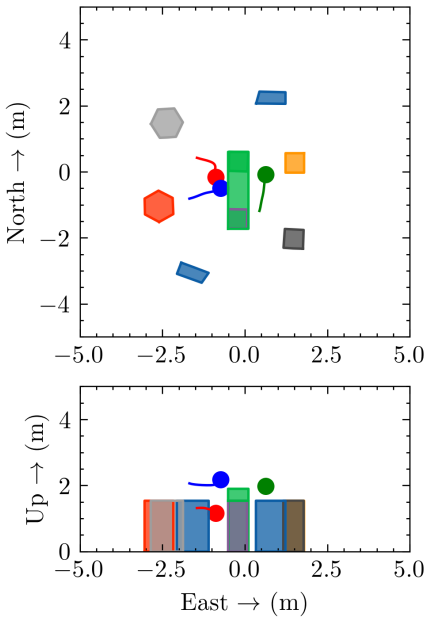


(a) t = 11s

(b) t = 17s



(c) t = 25s



(d) t = 50s

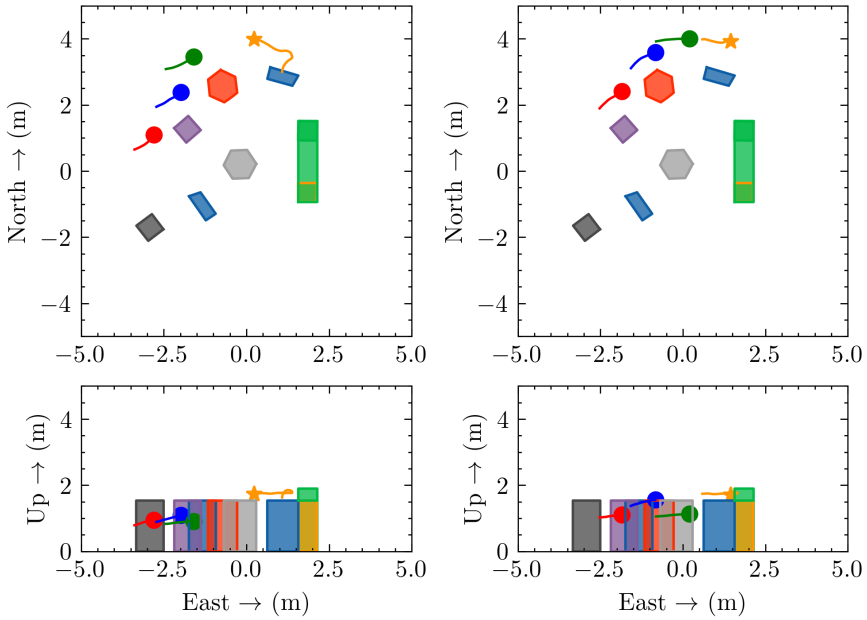
Fig. 10 Flight trajectories of scenario 3

The flight trajectories during the different flight phases are illustrated in Figure 10. Figure 10a illustrates the eleventh second of the flight, where all the drones are flying at approximately the same altitude and approaching their targets. *Drone 0* (Red) continues through the gate structure, while *Drone 1* (Blue) climbs and moves south to avoid *Drone 2* (Green). *Drone 2* (Green), avoiding obstacles and *Drone 1* (Blue), moves westward without changing altitude. At the seventeenth second of the flight, *Drone 0* (Red) descends to avoid *Drone 2* (Green) as it continues toward its target point, as shown in Figure 10b. Concurrently, *Drone 2* (Green) climbs to avoid *Drone 0* (Red). By the twenty-fifth second, all the drones have clear paths to their goals, avoiding each other and the obstacles around them (see Figure 10c). On the fiftieth second, *Drone 0* (Red) dives and flies through the gate, while *Drone 1* (Blue) climbs and passes over the gate, shown in Figure 10d.
This experiment demonstrates the performance of the proposed 3D PGFlow algorithm in a cluttered environment with multiple vehicles. The guidance algorithm successfully guides all vehicles through collision-free paths simultaneously in real-time.

4. Flight Test: Scenario 4

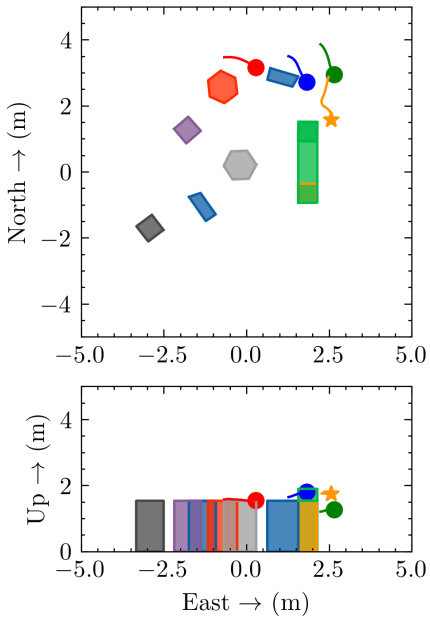
Fig. 11 Drones from Experiments.

In the fourth scenario, instead of traveling to pre-specified goal positions, the drones follow a dynamic target that moves during flight. The moving target does not follow a pre-set trajectory but rather moves randomly within the flight arena. The target position is known and updated at every control input.
The flight begins with a stationary target and three drones. The drones *Drone 0 and Drone 2* (Red and Green) are launched from the top of buildings, and *Drone 1* (Blue) is launched from the ground. They all follow the target point indicated by the yellow star in Figure 12. At the eighth second of the flight, *Drone 0 and Drone 2* (Red and Green) follow the target without changing the altitude, and *Drone 1* (Blue) ascends. The target point is displaced behind the obstacle east of the flight arena. The drones at high altitudes follow the target by flying over the obstacle. In contrast, the drone at a lower altitude prefers to fly inside the gate-like obstacle, as illustrated in Figure 12b. Then, the target is displaced to the south of the arena. The drones immediately found a collision-free trajectory input toward the target point and followed the new trajectory inputs as shown in Figure 12c. Furthermore, when the target is displaced again to the northwest of the flight arena, in the opposite direction of the drone location. The drones immediately made a turn maneuver as 'U' shaped trajectories shown in Figure 12d.
One of the strengths of the proposed guidance algorithm is its ability to adapt to dynamic environments. This experiment demonstrates the capability of the guidance algorithm to handle a dynamically changing target position. Even with a randomly moving target in a cluttered environment, the proposed algorithm can guide multiple vehicles without collision with obstacles or with each other in a cluttered environment in real-time.

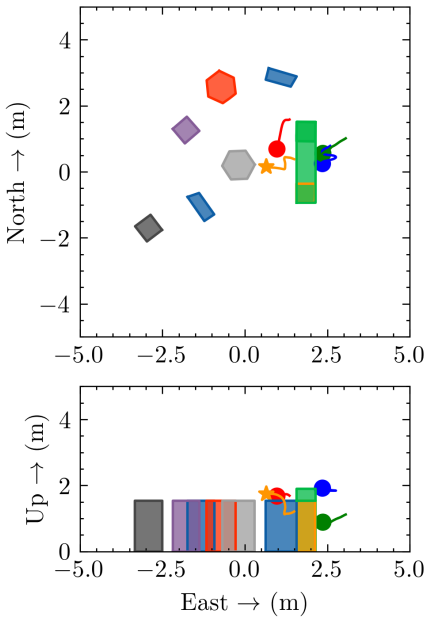


(a) t = 8s

(b) t = 14s



(c) t = 23s



(d) t = 32s

Fig. 12 Flight trajectories of scenario 4

V. Conclusion

In this paper, the previously proposed Potential Guidance Flow (PGFlow) algorithm [30] has been further improved by incorporating 3-Dimensional models of obstacles. Using 3D obstacle models enhances the PGFlow method's capability and efficiency of navigation in complex urban air spaces. This improvement is especially useful for urban air mobility (UAM) applications, where the effective utilization of vertical and horizontal space can increase the efficiency of operations and air space capacity. This paper addresses the limitations of the previous method presented in [30, 36] by introducing 3D obstacle models and effectively using the available airspace.

The effectiveness of the proposed method has been demonstrated through hardware experiments in a scaled urban environment built in the Toulouse Occitanie Drone Flight Area. Indoor flight tests show that the 3D PGFlow algorithm can successfully navigate around arbitrarily shaped obstacles, including tunnels and torus-like objects, which were previously challenging with the 2D PGFlow approach.

One major challenge encountered in this study was the increased computational load of the algorithm due to the increased number of elements introduced by the 3D obstacle models. This challenge was overcome by mitigating the core calculation library to an optimized C++ code and providing Python bindings for usability. While C++ allows for more efficient handling of the extensive mathematical operations and large data structures required by the 3D PGFlow algorithm, the Python interface facilitates integration with various modules, such as motion capture systems and drone communication protocols, enabling seamless operation within established workflows. Scalability tests have been conducted to evaluate the computational performance of the proposed 3D PGFlow method and software. The tests indicate that increasing the number of vehicles does not significantly reduce the algorithm's performance. On the other hand, the effect of increasing the number of buildings is greater due to the costly matrix-vector multiplication required for each additional building.

To conclude, the experimental results suggest that the 3D PGFlow method is capable of providing effective obstacle avoidance and collision avoidance capabilities in cluttered environments for multiple vehicles in real time. The guidance algorithm's extension to the third dimension allows for more effective use of the available air space and maneuvers around complex architectural structures. These properties make the 3D PGFlow algorithm a promising candidate for urban air mobility applications where multiple vehicles are expected to operate in limited airspace.

Acknowledgments

The authors would like to thank Cédric Boutet and Laurent Lacaze for constructing a scaled urban environment at ENAC, and Xavier Paris for the indoor flight arena technical support.

References

[1] INRIX, Inc., "Congestion Costs Each American Nearly 100 hours, $1,400 A Year," , Mar. 2020. URL https://www.prnewswire.com/news-releases/inrix-congestion-costs-each-american-nearly-100-hours-1-400-a-year-301020632.html.

[2] Fortune Business Insights, "Urban Air Mobility (UAM) Market Size, Share & Industry Analysis, By Vehicle Type, By Range, By Operation, By End User, Regional Forecast, 2024-2032," , Jun. 2024. URL https://www.fortunebusinessinsights.com/urban-air-mobility-uam-market-106344, report ID: FBI106344.

[3] Bauranov, A., and Rakas, J., "Designing airspace for urban air mobility: A review of concepts and approaches," *Progress in Aerospace Sciences*, Vol. 125, 2021, p. 100726. https://doi.org/https://doi.org/10.1016/j.paerosci.2021.100726, URL https://www.sciencedirect.com/science/article/pii/S0376042121000312.

[4] Ahamed, S., Prakash, S., Shuvrangshu, J., and Debasish, G., "UAS Flight Path Planning using Numerical Potential Fields in Dense Non-segregated Airspace," *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2021, pp. 1010–1019. https://doi.org/10.1109/ICUAS51884.2021.9476791.

[5] Zhang, N., Zhang, M., and Low, K. H., "3D path planning and real-time collision resolution of multirotor drone operations in complex urban low-altitude airspace," *Transportation Research Part C: Emerging Technologies*, Vol. 129, 2021, p. 103123. https://doi.org/https://doi.org/10.1016/j.trc.2021.103123, URL https://www.sciencedirect.com/science/article/pii/S0968090X2100142X.

[6] Dai, W., Pang, B., and Low, K. H., "Conflict-free four-dimensional path planning for urban air mobility considering airspace occupancy," *Aerospace Science and Technology*, Vol. 119, 2021, p. 107154. https://doi.org/https://doi.org/10.1016/j.ast.2021.107154, URL https://www.sciencedirect.com/science/article/pii/S1270963821006647.

[7] Huang, S., and Low, K. H., "A Path Planning Algorithm for Smooth Trajectories of Unmanned Aerial Vehicles via Potential Fields," *2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, 2018, pp. 1677–1684. https://doi.org/10.1109/ICARCV.2018.8581198.

[8] Orozco-Rosas, U., Montiel, O., and Sepúlveda, R., "Mobile robot path planning using membrane evolutionary artificial potential field," *Applied Soft Computing Journal*, 2019, pp. 236–251. https://doi.org/10.1016/j.asoc.2019.01.036.

[9] Lifen, L., Ruoxin, S., Shuandao, L., and Jiang, W., "Path planning for UAVS based on improved artificial potential field method through changing the repulsive potential function," *2016 IEEE Chinese Guidance, Navigation and Control Conference (CGNCC)*, 2016, pp. 2011–2015. https://doi.org/10.1109/CGNCC.2016.7829099.

[10] Yingkun, Z., "Flight path planning of agriculture UAV based on improved artificial potential field method," *2018 Chinese Control And Decision Conference (CCDC)*, 2018, pp. 1526–1530. https://doi.org/10.1109/CCDC.2018.8407369.

[11] Chen, Y., Luo, G., Mei, Y., Yu, J., and Su, X., "UAV path planning using artificial potential field method updated by optimal control theory," *International Journal of Systems Science*, Vol. 47, 2016, pp. 1407 – 1420. URL https://api.semanticscholar.org/CorpusID:29757767.

[12] Liu, Y., and Zhao, Y., "A virtual-waypoint based artificial potential field method for UAV path planning," *2016 IEEE Chinese Guidance, Navigation and Control Conference (CGNCC)*, 2016, pp. 949–953. https://doi.org/10.1109/CGNCC.2016.7828913.

[13] Honglun, W., Wentao, L., Yao, P., Liang, X., and Chang, L., "Three-dimensional Path Planning for Unmanned Aerial Vehicle Based on Interfered Fluid Dynamical System," *Chinese Journal of Aeronautics*, Vol. 28, 2014. https://doi.org/10.1016/j.cja.2014.12.031.

[14] Chakravarthy, A., and Ghose, D., "Obstacle Avoidance in a Dynamic Environment: A Collision Cone Approach," *IEEE Transactions on Systems, Man, and Cybernetics—Part A: Systems and Humans*, Vol. 28, No. 5, 1998, pp. 570–574. https://doi.org/10.1109/3468.709600.

[15] Chakravarthy, A., and Ghose, D., "Generalization of the Collision Cone Approach for Motion Safety in 3-D Environments," *Autonomous Robots*, Vol. 32, No. 3, 2012, pp. 243–266. https://doi.org/10.1007/s10514-011-9270-z.

[16] Chakravarthy, A., and Ghose, D., "Collision Cones for Quadric Surfaces," *IEEE Transactions on Robotics*, Vol. 27, No. 6, 2011, p. 1166. https://doi.org/10.1109/TRO.2011.2159413.

[17] Chester, C. R., "Techniques in Partial Differential Equations," *McGraw-Hill*, 1970, pp. 72–73.

[18] Kim, J.-O., and Khosla, P., "Real-time obstacle avoidance using harmonic potential functions," *IEEE Transactions on Robotics and Automation*, Vol. 8, No. 3, 1992, pp. 338–349. https://doi.org/10.1109/70.143352.

[19] Masoud, A. A., "A harmonic potential field approach for joint planning and control of a rigid, separable nonholonomic, mobile robot," *Robotics and Autonomous Systems*, Vol. 61, No. 6, 2013, pp. 593–615. https://doi.org/https://doi.org/10.1016/j.robot.2013.02.007, URL https://www.sciencedirect.com/science/article/pii/S0921889013000432.

[20] Fahimi, F., "Autonomous robots," *Springer*, 2010, pp. 81–127.

[21] Daily, R., and Bevly, D. M., "Harmonic potential field path planning for high speed vehicles," *2008 American Control Conference*, 2008, pp. 4609–4614. https://doi.org/10.1109/ACC.2008.4587222.

[22] Garrido, S., and Moreno, L., "Robotic Navigation using Harmonic Functions and Finite Elements." 2006, pp. 94–103.

[23] Liang, X., Honglun, W., Li, D., and Liu, C., "Three-dimensional path planning for unmanned aerial vehicles based on fluid flow," 2014, pp. 1–13. https://doi.org/10.1109/AERO.2014.6836520.

[24] Lewis, R. I., *The surface vorticity method for inviscid ideal fluid flow*, Cambridge Engine Technology Series, Cambridge University Press, 1991, pp. 3–98.

[25] Katz, J., and Plotkin, A., *Singularity Elements and Influence Coefficients*, 2nd ed., Cambridge Aerospace Series, Cambridge University Press, 2001, p. 230–261. https://doi.org/10.1017/CBO9780511810329.012.

[26] Anderson, J. D., "Fundamentals of Aerodynamics," *McGraw-Hill*, 2011, pp. 319–325.

[27] Zhang, Y., and Valavanis, K. P., "A 3-D Potential panel method for robot motion planning," *Robotica*, Vol. 15, 1997, pp. 421 – 434. https://doi.org/10.1017/S0263574797000520.

[28] Velagić, J., Vuković, L., and Ibrahimović, B., "Mobile Robot Motion Framework Based on Enhanced Robust Panel Method," *International Journal of Control, Automation and Systems*, Vol. 18, No. 5, 2020, pp. 1264–1276. https://doi.org/https://doi.org/10.1007/s12555-019-0009-5.

[29] Cruz, G. C. S., and Encarnação, P. M. M., "Obstacle Avoidance for Unmanned Aerial Vehicles," *Journal of Intelligent & Robotic Systems*, Vol. 65, 2012, pp. 203–217. https://doi.org/10.1007/s10846-011-9587-z.

[30] Bilgin, Z., Yavrucuk, I., and Bronz, M., "Urban Air Mobility Guidance with Panel Method: Experimental Evaluation Under Wind Disturbances," *Journal of Guidance, Control, and Dynamics*, Vol. 47, No. 5, 2024, pp. 1–17. https://doi.org/10.2514/1.G007691.

[31] Bilgin, Z., Bronz, M., and Yavrucuk, I., "Panel Method Based Guidance for Fixed Wing Micro Aerial Vehicles," *International Micro Air Vehicle Conference and Competition*, 2022.

[32] Unal, Z., and Yavrucuk, I., "Panel Method Based Path Planning for eVTOL in Urban Environment," *Vertical Flight Society Forum 77*, 2021.

[33] Bilgin, Z., Bronz, M., and Yavrucuk, I., "Experimental Evaluation of Panel-Method-Based Path Planning for eVTOL in A Scaled Urban Environment," *Vertical Flight Society Forum 78*, 2022. https://doi.org/10.4050/F-0078-2022-17536.

[34] Bilgin, Z., Bronz, M., and Yavrucuk, I., "Experimental Evaluation of Robustness of Panel-Method-Based Path Planning for Urban Air Mobility," 2022. https://doi.org/10.2514/6.2022-3509.

[35] Bilgin, Z., Bronz, M., and Yavrucuk, I., "Automatic In Flight Conflict Resolution for Urban Air Mobility using Fluid Flow Vector Field based Guidance Algorithm," *2023 IEEE/AIAA 42nd Digital Avionics Systems Conference (DASC)*, 2023, pp. 1–7. https://doi.org/10.1109/DASC58513.2023.10311214.

[36] Bilgin, Z., Bronz, M., and Yavrucuk, I., "A Nature Inspired Guidance Method for Improving Social Acceptance of Urban Air Mobility," *49th European Rotorcraft Forum 2023 - Proceedings*, 49th European Rotorcraft Forum 2023, Deutsche Gesellschaft für Luft- und Raumfahrt - Lilienthal-Oberth e.V., Bückeburg, Germany, 2023, pp. 1–10.

[37] ENAC, "Drone Flight Arena Toulouse Occitanie," Ecole Nationale de l'Aviation Civile, Retrieved 19 October 2021, from https://www.enac.fr/en/drone-flight-arena-toulouse-occitanie-0, 2021.

[38] Hess, J. L., and Smith, A. M. O., "Calculation of Nonlifting Potential Flow About Arbitrary Three-Dimensional Bodies," *Journal of Ship Research*, Vol. 8, No. 04, 1964, p. 22–44. https://doi.org/10.5957/jsr.1964.8.4.22, URL http://dx.doi.org/10.5957/jsr.1964.8.4.22.