

Article

Wind-Aware RRT* with Neural Energy Refinement for Energy-Efficient Urban Air Mobility

Farhad Bagheri ¹, Mohammadali Amiri Atashgah ^{1,*} and Morteza Ebrahimi ²¹ School of Aerospace Engineering, University of Tehran, Tehran 1439957131, Iran; farhad.bagheri@ut.ac.ir² School of Intelligent Systems, University of Tehran, Tehran 1439957131, Iran; mo.ebrahimi@ut.ac.ir

* Correspondence: atashgah@ut.ac.ir; Tel.: +98-2186093190

Abstract

Urban air mobility depends on small aerial vehicles threading through dense, wind-swept cities, yet most sampling-based planners treat the urban wind field as noise to reject rather than structure to exploit—and pay for it in flight energy. We take the opposite view. Behind every building lies a sheltered wake where the air slows and aerodynamic drag drops, and this work turns that physical fact into a planning principle. We present an energy-aware, wind-shadow-aware framework that routes a single quadrotor, at the planning level, through these low-wind corridors. The wind model couples a power-law shear profile with Ekman directional veer and a frozen-turbulence gust component, grounding the planner in realistic boundary-layer physics. A feed-forward neural energy surrogate, trained to approximate a cost field that aggregates wind exposure and obstacle clearance, then guides a two-stage refinement—energy-aware, collision-checked shortcutting followed by Laplacian and energy-guided smoothing—so that every accepted change stays collision-free. Against classical sampling-based baselines (RRT, goal-biased RRT, Informed RRT*, and BIT*) over a 50-run Monte-Carlo study, evaluated with multi-criteria metrics and Pareto-dominance analysis, the framework characterizes how wind-shadow-aware routing balances route energy against smoothness and clearance, offering a reproducible, wind-informed basis for flying robots navigation.

Keywords: sampling-based motion planning; wind-shadow; wind-aware path planning

1. Introduction

Urban air mobility (UAM) has moved past the concept stage. It is now an active engineering problem, with aircraft expected to carry people, deliver goods, inspect structures, and respond to emergencies over crowded cities [1,2]. Small flying robots and electric vertical-take-off-and-landing platforms are the leading candidates for these short city hops [3,4]. The catch is that the same low-altitude, cluttered airspace that makes UAM appealing is also where the trouble lives. Buildings crowd the route, usable corridors are narrow, and the wind grows strongly heterogeneous where the atmospheric boundary layer meets the built environment [5]. A workable deployment therefore depends on motion planning that stays collision-free, runs fast enough to use, and treats the aerodynamic cost of urban airflow as a first-class concern.

City wind behaves nothing like wind over open ground. Buildings reshape the local pressure field. They funnel and speed up flow through tight gaps—the canyon effect—

Academic Editor: Massimiliano Caramia

Received: 1 July 2026

Revised: 27 July 2026

Accepted: 30 July 2026

Published: 6 August 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

while leaving separation, recirculation, and slow, sheltered wakes behind their leeward faces [5,6]. Computational Fluid Dynamics (CFD) studies of urban canyons back this up: mean velocity, turbulence intensity, and flow direction all shift with how far apart the buildings sit, how tall they are, and which way the incoming flow is pointed [7,8]. That variability is not cosmetic. It carves the airspace into patches of very different wind speed, and some of those patches are genuinely dangerous for a small aircraft [9]. Newer simulation pipelines now pair real building geometry with CFD or large-eddy simulation to recreate the wakes a quadrotor would actually fly through [10,11]. Even so, the reviews are blunt that we still do not fully understand how UAM interacts with this kind of complex urban airflow [5,12]. One consequence is well documented yet barely used by planners. Just downwind of a building sits a pocket of slowed air—a wind shadow—where drag drops noticeably. Routing a vehicle through those sheltered pockets is a direct, physically grounded way to save energy. Yet most planners, when they consider wind at all, treat it as a nuisance to suppress instead of a spatial structure worth using.

Energy is the constraint that decides what UAM can actually do. Multicopter burns power just to stay up, so its modest battery sets a hard ceiling on how far it can go and how much it can carry [13,14]. Researchers have built power-consumption models that tie rotor thrust, vehicle dynamics, and aerodynamic drag to the instantaneous electrical draw [15,16], then folded those models into trajectory optimization to spend as little energy as possible [13,17]. Bring wind into the picture and the planner can use it—leaning away from headwinds and riding tailwinds to cut real flight cost [18]. Centralized UAM planners that fold in navigation and coverage requirements show how tightly city airspace boxes in these routes. Other work leans on graph search—Voronoi decomposition, A*, Dijkstra—to attach wind-dependent costs to edges across an urban network, and studies in lower-altitude UAM airspace confirm that respecting wind dynamics produces more energy-efficient paths than wind-agnostic baselines [18]. The problem is that graph search does not cope well with fine three-dimensional resolution and forces you into coarse network abstractions. That pushes the field toward sampling-based planners, which work straight in the continuous configuration space.

Among continuous planners, the Rapidly-exploring Random Tree (RRT) family is the one everyone reaches for. It mixes random exploration with probabilistic completeness and handles high-dimensional spaces without much fuss [19,20]. Its optimal cousin, RRT*, adds a rewiring step that drives path cost down, and it has become the family's workhorse [19,21]. From there the variants pile up. Informed-RRT* confines its sampling to an ellipsoidal admissible subset the moment a first solution appears [21,22], and bidirectional and goal-biased schemes chase faster convergence from other angles [23]. Recent surveys count dozens of RRT* tweaks from 2021 to 2024 and notice a pattern: almost all of them retune the expansion strategy or the path-retrieval step, and hardly any touch the sampling distribution itself [20,24]. That is the opening. The sampling distribution is exactly where you could pour in prior knowledge of the environment—the shape of the wind field, say, or third-party and ground-risk cost maps—to steer new nodes toward the regions you want. Learning a sampling distribution has already paid off in generic motion-planning settings [25]. What is missing is a wind-physics-informed bias for urban flight, one that pulls node altitudes and positions toward calm, wake-sheltered air. Filling that gap is the first goal of this paper.

Alongside classical search, learning-based planning has grown into a real alternative. Motion Planning Networks (MP-Net) push obstacle geometry through an encoder into a latent space, then let a planning network predict the next state toward the goal step by step; pairing that neural policy with a classical planner restores the formal guarantees [26,27]. Diffusion-based planners take a different route, treating trajectory generation as iterative denoising, and they hold up well in high-dimensional problems [28,29]. Surveys

track how fast learning and sampling-based planning are merging [30,31], including residual convolutional designs aimed at real-time path generation. One idea keeps coming back across this work: train a neural surrogate to mimic an expensive objective, then use it to nudge a trajectory toward cheaper regions for a fraction of the usual evaluation cost [25,32]. For urban UAM the obvious objective is an energy-cost field—wind exposure, obstacle clearance, and altitude rolled together. Each piece is costly to evaluate on its own, but a small regression network trained with the Levenberg–Marquardt method can learn the whole thing [33]. A surrogate like that can then drive energy-aware shortcutting and smoothing while every change it proposes still gets collision-checked. You get the speed of learning without giving up the safety of classical validation. Building that refiner is the second goal of this work.

Planning is only half the story; a large body of work tackles wind through control and data-driven estimation instead. Robust and adaptive controllers, online tuning, and Deep Reinforcement Learning (DRL) policies have all been shown to fend off wind shear, turbulence, and gusts on a single quadrotor, trimming tracking error along the way [34,35]. DRL in particular shows up everywhere in single-vehicle navigation and obstacle avoidance through cluttered or unfamiliar spaces [36–38]. Some formulations go further and tie wind rejection to low-power flight under safety constraints [17], while control-barrier functions have been fused with learning to keep a vehicle safe through aggressive maneuvers [39]. There is also the estimation side: onboard sensors, moving-horizon estimators, or Gaussian-process regression let a vehicle figure out the disturbance acting on it in the moment [40]. And the Artificial Potential Field (APF) method is still a favorite for wind-aware local planning and tracking, with newer variants reshaping the attractive and repulsive forces to hold up better against gusts over the city [41–43]. Recent collaborative active visual Simultaneous Localization and Mapping (SLAM) frameworks have demonstrated effective real-time navigation of multiple quadrotors in unknown outdoor environments by coupling mapping, localization, and path planning [44]. Integrating such active sensing and mapping capabilities with the proposed wind-biased RRT* and neural energy refinement would enable the present energy-aware planner to operate under localization uncertainty and partial maps, while still exploiting urban wind-shadow corridors for reduced aerodynamic cost.

All of this confirms two things—the urban wind problem is serious, and handling wind explicitly pays off. But it lives mostly at the control or estimation layer. It leaves the harder geometric question untouched: how should the structure of the wind shape the route itself, and in particular, how can a planner deliberately spend time in building wind shadows to save energy?

Step back and three strands run through this literature, rarely touching one another. One studies the physics of urban wind and building wakes in detail. Other builds energy-aware, sampling-based planners but seldom changes the sampling distribution to reflect the wind. The third refines trajectories with learning, yet rarely tailors that learning to an urban energy-cost field. This paper sets out to tie the three together. The core idea is simple: stop treating the wind field as a disturbance and start treating it as a planning input in its own right, letting its shear, veer, and wake structure decide where samples land while a learned energy model shapes how each route gets refined.

So, this paper builds an integrated, energy-aware and wind-shadow-aware framework for a single quadrotor crossing a cluttered, windy city. We work at the planning level, without closed-loop tracking control, and the framework rests on two contributions that carry equal weight and feed each other. The first is a wind-biased RRT* planner. Instead of sampling uniformly, it pulls samples toward low-wind altitudes and toward the sheltered wakes downwind of buildings. The wind model behind that bias is deliberately realistic—a power-law vertical shear profile, Ekman directional veer, and a frozen-

turbulence gust component together—so the sampling reflects real boundary-layer physics while RRT* keeps its probabilistic completeness. The second contribution is an energy-aware neural refinement stage. Here a feed-forward neural energy surrogate—trained to approximate a cost field over wind exposure, obstacle clearance, and altitude—guides a multi-stage cleanup: energy-aware, collision-checked shortcutting first, then Laplacian and energy-guided smoothing, with every accepted change re-checked to stay collision-free. The two halves act as one pipeline. The biased sampler seeds routes in aerodynamically favorable regions, and the learned refiner polishes them for energy and smoothness. To see how well it works, we test the framework against classical sampling-based baselines—RRT, goal-biased RRT, Informed RRT*, and BIT*—across a 50-run Monte-Carlo study, scoring each run on length, smoothness, energy, and risk and reading the trade-offs through Pareto-dominance analysis.

The rest of the paper is organized as follows. Section 2 presents the modeling and solution framework, including the mathematical formulation of the urban environment, wind-field representation, and the proposed optimization problem and introduces the proposed method, consisting of the wind-biased RRT* planner and the energy-aware neural refinement strategy for trajectory optimization. Section 3 presents the simulation results, including Monte Carlo experiments and multi-criteria performance evaluation. Section 4 discusses the obtained results and highlights the limitations of the proposed approach, followed by conclusions in Section 5.

2. Modeling and Solution

2.1. Workspace Urban Wind Field Modeling

The physical flight domain is defined as an axis-aligned bounding box $\Omega \subset \mathbb{R}^3$ expressed as

$$\Omega = [0, L_x] \times [0, L_y] \times [0, L_z] \quad (1)$$

where the default operational dimensions are set to $L_x = L_y = 1200$ m and $L_z = 220$ m. The simulated urban environment (Figure 1) consists of numerous buildings with diverse geometric characteristics, providing a realistic representation of a dense metropolitan area. Furthermore, multiple suspended no-fly zones (NFZ), illustrated in red, are distributed throughout the workspace. These aerial restricted regions are located above ground level and are treated as forbidden airspaces that must be avoided by the path-planning algorithm.

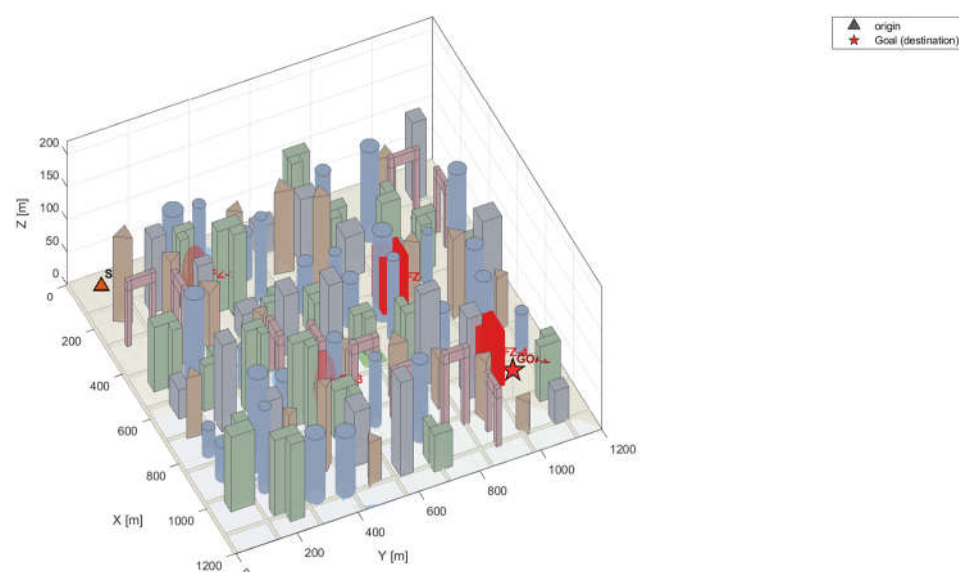


Figure 1. Working Space of algorithm with complex map and No-Flying-Zones (Red obstacles).

The obstacle-free navigation airspace is defined as

$$\Omega_{free} = \Omega \setminus \bigcup_j O_j \quad (2)$$

where $O_j \subset \Omega$ denotes the compact spatial volume occupied by the j -th urban structure or building.

To facilitate collision-checking and environmental querying, the geometry is discretized into an occupancy map indicator function $M(p)$ sampled on a uniform grid with a spatial resolution of $\Delta m = 0.5$ m.

$$M(p) = \begin{cases} 1, & \text{if } p \in \bigcup_j O_j \\ 0, & \text{otherwise} \end{cases} \quad \forall p \in \Omega \quad (3)$$

The continuous spatial wind velocity vector $\mathbf{w}(p) \in R^3$ is constructed modularly at any coordinate $p = [x, y, z]^T \in \Omega_{free}$. The total velocity field is initialized from the atmospheric boundary layer framework and systematically incorporates vertical wind shear, directional turning, stochastic turbulence, localized gusts, and structural building wakes (Figure 2).

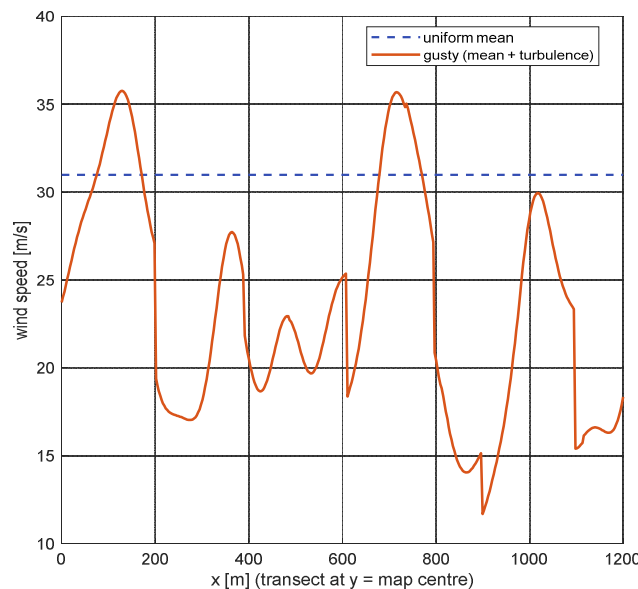


Figure 2. Uniform wind condition versus gusty wind.

Near the ground surface, the mean horizontal wind speed profile $U(z)$ is primarily governed by a constant turbulent shear stress τ_0 within the atmospheric constant-flux layer. By defining the characteristic friction velocity as

$$u_* = \sqrt{\tau_0 / \rho_a} \quad (4)$$

where ρ_a is the ambient air density and adopting Prandtl's mixing length hypothesis $l_m = \kappa z$ where $\kappa \approx 0.41$ represents the von Kármán constant), the mechanical stress balance equation

$$\tau_0 = \rho_a l_m^2 (dU/dz)^2 \quad (5)$$

yields the fundamental differential gradient:

$$\frac{dU}{dz} = \frac{u_*}{\kappa z} \Rightarrow U(z) = \frac{u_*}{\kappa} \ln\left(\frac{z}{z_0}\right) \quad (6)$$

which represents the classic logarithmic wind profile, where z_0 denotes the aerodynamic roughness length of the terrain.

For high-level engineering applications over complex and highly obstructed urban topologies, this logarithmic law is robustly approximated using an empirical power-law profile anchored at a specific reference height z_{ref} .

By formulating the profile as

$$U(z) \approx U_{\text{ref}}(z/z_{\text{ref}})^{\alpha} \quad (7)$$

And matching its logarithmic derivative slope precisely at z_{ref} , the dimensionless shear exponent α is mathematically identified as:

$$\alpha = \left(\frac{d \ln U}{d \ln z} \right)_{z_{\text{ref}}} = \frac{1}{\ln(z_{\text{ref}}/z_0)} \quad (8)$$

The equation above demonstrates that aerodynamically rougher urban landscapes (characterized by larger z_0 values) inherently yield a larger shear exponent α .

To accurately replicate a dense, built-up metropolitan environment, we adopt a design exponent of $\alpha \approx 0.30$ throughout this study [45].

To implement a computationally efficient yet physically accurate representation of the atmospheric boundary layer, the continuous horizontal wind speed magnitude $V(z)$ at a given altitude z is governed by an empirical power-law profile:

$$V(z) = V_0 \left(\frac{\max(z, z_{\min})}{z_{\text{ref}}} \right)^{\alpha} \quad (9)$$

where V_0 represents the baseline reference wind speed measured at a standard reference altitude of $z_{\text{ref}} = 10$ m, which scales dynamically according to the predefined environmental wind intensity level. To eliminate the mathematical singularity and infinite velocity gradients inherent to traditional boundary layer formulations as $z \rightarrow 0$, a lower boundary clamp is introduced at $z_{\min} = 0.5$ m. The dimensionless shear parameter ($\alpha \approx 0.30$) represents the atmospheric roughness coefficient corresponding to highly built-up urban topologies.

The vertical wind shear profile, which characterizes the localized wind velocity change relative to flight altitude, is analytically derived by differentiating the velocity profile with respect to z for all regions above the boundary clamp ($z > z_{\min}$):

$$\frac{dV}{dz} = \frac{\alpha V_0 z^{\alpha-1}}{z_{\text{ref}}^{\alpha}} = \frac{\alpha V(z)}{z} \quad (10)$$

The resulting differential gradient highlights that the vertical shear is highly concentrated near the ground surface and decays progressively with increasing altitude. This spatial gradient serves as a critical physical driver within the motion planning framework, naturally rewarding trajectories that dynamically adjust flight altitudes to minimize aerodynamic resistance.

Above the surface layer, the steady horizontal momentum balance is governed by the equilibrium among the Coriolis force, the pressure-gradient force, and the turbulent friction modeled via the eddy viscosity coefficient K . For the horizontal wind components (u, v) aligned with a geostrophic wind vector $(u_g, 0)$ and a Coriolis parameter f_c , this mechanical equilibrium is expressed as:

$$f_c v = -K \frac{d^2 u}{dz^2}, \quad -f_c (u - u_g) = -K \frac{d^2 v}{dz^2} \quad (11)$$

The classical analytical solution to this system characterizes the Ekman spiral, which dynamically rotates the horizontal wind vector as a function of altitude:

$$u(z) = u_g \left(1 - e^{-z/\delta_e} \cos\left(\frac{z}{\delta_e}\right) \right), \quad v(z) = u_g e^{-z/\delta_e} \sin\left(\frac{z}{\delta_e}\right), \quad \delta_e = \sqrt{\frac{2K}{f_c}} \quad (12)$$

where δ_e denotes the characteristic Ekman boundary layer depth. Consequently, the wind heading angle

$$\theta_w(z) = \text{atan2}(v, u) \quad (13)$$

exhibits a monotonic directional turning with increasing altitude. To maintain computational tractability over the thin low-altitude urban band investigated in this work, this vertical directional veer is linearized about the surface wind heading θ_0 using the leading term of a first-order Taylor expansion:

$$\theta_w(z) = \theta_0 + \gamma z, \quad \gamma \approx 0.18^\circ \text{m}^{-1} \quad (14)$$

where γ represents the constant directional turning rate. This linear approximation accurately captures the physical directional veer of the lower atmosphere while minimizing the online computational overhead during the node expansion phase of the planning pipeline.

The localized atmospheric turbulence intensity $T_i(z)$ is defined as the dimensionless ratio of the velocity standard deviation (the root-mean-square of turbulent velocity fluctuations, σ_u) to the local mean wind speed magnitude $V(z)$:

$$T_i(z) = \frac{\sigma_u(z)}{V(z)} \quad (15)$$

Within the constant-flux surface layer, the turbulent fluctuation magnitude scales proportionally with the friction velocity such that $\sigma_u \approx Au_*$, where $A \approx 2.5$ is an empirical scaling constant. Substituting the logarithmic wind law into this relationship reveals that the theoretical turbulence intensity is inversely proportional to the altitude logarithm, $T_i \propto 1/\ln(z/z_0)$, indicating a monotonic decrease with height.

To ensure numerical stability and consistent scaling throughout the workspace grid, the simulation environment adopts a robust, surface-anchored monotonic decay model:

$$T_i(z) = T_{i,0} \left(\frac{z_{\text{ref}}}{\max(z, z_{\text{ref}})} \right)^\beta \quad (16)$$

where $\beta \approx 0.25$ denotes the empirical intensity decay exponent. The baseline surface turbulence intensity $T_{i,0}$ is dynamically coupled to a user-defined operational turbulence level $T_{\text{lev}} \in \{1, \dots, 10\}$ via a linear mapping function:

$$T_{i,0} = 0.04 + \frac{T_{\text{lev}} - 1}{9} \cdot 0.26 \quad (17)$$

This formulation maps the discrete turbulence levels linearly onto a continuous surface intensity bounds where $T_{i,0} \in [0.04, 0.30]$. This parameterization enables the motion planner to evaluate the stability risk of the path across a wide spectrum of meteorological conditions, ranging from calm urban environments to highly chaotic, boundary-layer conditions.

Following Taylor's frozen-turbulence hypothesis, atmospheric turbulence is treated as a rigid spatial pattern advected past the moving vehicle at the mean wind velocity. Under this assumption, the temporal gust profile experienced along a flight trajectory is mathematically equivalent to a static, spatially varying field. We synthesize this multi-dimensional field $f_s(p)$ as a finite superposition of deterministic Fourier modes governed by randomized yet fixed wavevectors $\mathbf{k}_j \in \mathbb{R}^3$ and uniform phases $\phi_j \in [0, 2\pi)$:

$$f_s(p) = \frac{1}{\sqrt{N_g}} \sum_{j=1}^{N_g} \sin(\mathbf{k}_j^T p + \phi_j) \quad (18)$$

where $N_g = 8$ represents the discrete number of spectral modes. This algebraic formulation guarantees two critical statistical and numerical properties:

By applying the triangle inequality alongside the native trigonometric bound $|\sin(\cdot)| \leq 1$, the maximum absolute amplitude of the synthesized field is strictly bounded:

$$|f_s(p)| \leq \frac{1}{\sqrt{N_g}} \sum_{j=1}^{N_g} 1 = \sqrt{N_g} \quad (19)$$

Following a unit-weight normalization step, the continuous spatial domain of the field is mapped precisely to lie within the stable numerical range $[-1, 1]$.

By assuming independent, uniformly distributed phases, the expectation operators evaluate to $\mathbb{E}[\sin(\cdot)] = 0$ and $\mathbb{E}[\sin^2(\cdot)] = 1/2$. This yields the following localized ensemble metrics:

$$\mathbb{E}[f_s] = 0, \quad \text{Var}[f_s] = \frac{1}{N_g} \sum_{j=1}^{N_g} \frac{1}{2} = \frac{1}{2} \quad (20)$$

The resulting field is statistically stationary and perfectly reproducible, remaining frozen throughout a single simulation run because $\mathbf{k}_j$ and ϕ_j are locked by the initial pseudo-random seed. To drive independent heading jitter, a second decoupled spatial field $f_d(p)$ is generated using identical structural properties. The aggregate instantaneous wind speed $\tilde{V}(p)$ and gusted wind heading $\tilde{\theta}_w(p)$ evaluated at any coordinate $p \in \Omega_{\text{free}}$ are formulated as:

$$\tilde{V}(p) = \max\left(V(z)\left(1 + A_g f_s(p)\right), 0\right), \quad \tilde{\theta}_w(p) = \theta_w(z) + J_g f_d(p) \quad (21)$$

where $A_g \approx 0.35$ represents the fractional gust amplitude scale capturing peak velocity fluctuations, and $J_g \approx 18^\circ$ defines the maximum angular heading-jitter standard deviation. The nonlinear $\max(\cdot, 0)$ operator enforces a strict physical lower bound, preventing negative airspeed values during aggressive downburst cycles.

Behind a bluff body immersed in a fluid flow, the aerodynamic wake exhibits a localized velocity deficit characterized by three distinct physical properties: (i) strict confinement to the leeward (lee) side of the structure, (ii) lateral diffusive spreading following a Gaussian distribution, and (iii) progressive kinetic energy recovery further downstream. To mathematically formalize this phenomenon within the workspace grid, the relative position vector of any given coordinate $p \in \Omega_{\text{free}}$ relative to the geometric center of a building $c \in \Omega_{\text{obs}}$ is projected onto the local horizontal unit wind direction vector:

$$\hat{\mathbf{d}} = [\cos \theta_w, \sin \theta_w, 0]^T \quad (22)$$

This yields the localized downwind along-wind coordinate ξ and crosswind coordinate η :

$$\xi = (p - c)^T \hat{\mathbf{d}}, \quad \eta = |(p - c) - \xi \hat{\mathbf{d}}| \quad (23)$$

Following classical self-similar far-wake theory, the crosswind velocity profile is modeled as a Gaussian deficit distribution with a characteristic wake half-width σ_w , while the downwind fluid relaxation is governed by a monotonic downstream recovery function that depends on a characteristic recovery length λ . Because the fluid deficit is strictly a leeward phenomenon, this interaction is modulated by an indicator function $I_{[\xi > 0]}$, which ensures the wake is active exclusively in the downwind domain. Combining these continuous spatial modifiers, the multi-building wind attenuation is formulated via a multiplicative wake scaling factor $s_w(p)$:

$$s_w(p) = 1 - (1 - s_{\min}) \cdot I_{[\xi > 0]} \cdot \exp\left(-\frac{\eta^2}{2\sigma_w^2}\right) \cdot (1 - e^{-\xi/\lambda}) \quad (24)$$

where $s_{\min} \approx 0.4$ represents the strict lower bound of the velocity deficit corresponding to maximum flow attenuation within the deep wake core, and σ_w scales proportionally with the characteristic cross-stream dimension of the building. The resulting scaling factor is strictly bounded such that $s_w(p) \in [s_{\min}, 1]$.

By applying this multiplicative factor directly to the instantaneous wind velocity, the framework accurately synthesizes localized “wind shadows” behind urban obstacles, establishing low-velocity flight corridors that the motion planner can proactively exploit for energy optimization (Figure 3).

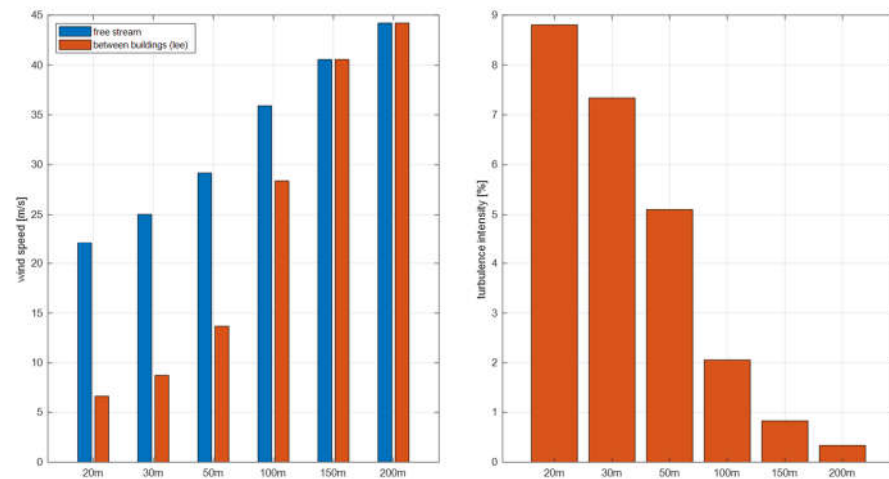


Figure 3. Lee ward wind speed reduction and turbulence changes with increasing altitude.

2.2. RRT* and Wind-Shear-Biased RRT* Planning

The RRT* algorithm was introduced specifically to overcome the fundamental limitation of traditional RRT, which is its inability to guarantee path optimality. The core idea behind RRT* is to preserve the powerful exploratory capabilities of the original RRT while introducing an incremental optimization mechanism that systematically rewires the tree structure to find lower-cost paths. In the RRT* framework, a cost function is defined for each node $q \in V$, representing the cumulative path cost from the initial starting configuration to that specific node. In its most basic formulation, this objective function is defined strictly as the Euclidean path length:

$$Cost(q) = \sum_{i=1}^k \|q_i - q_{i-1}\| \quad (25)$$

where $\{q_0, q_1, \dots, q_k\}$ represents the unique path within the tree structure connecting q_{start} to the current node q . Upon the generation of a new candidate node q_{new} , rather than simply connecting it to the nearest existing node, the algorithm evaluates a set of neighboring nodes located within a specified hyper-spherical neighbor radius:

$$Q_{near} = \{q \in V \mid \|q - q_{new}\| \leq r_n\} \quad (26)$$

In this approach, it is proven that the neighbor radius must be selected as a function of the current iteration count n and the spatial dimensionality d to mathematically guarantee asymptotic optimality:

$$r(n) = \gamma \left[\frac{\log(n)}{n} \right]^{\frac{1}{d}} \quad (27)$$

where n represents the total number of samples generated, d denotes the dimensionality of the configuration space, and γ is a positive tuning parameter configured to be strictly greater than a critical threshold determined by the volume of the free workspace airspace.

Inspired by random geometric graph theory, this scaling relationship serves as a cornerstone for proving the asymptotic optimality of RRT*, ensuring that the tree remains appropriately connected as the sample size grows. Specifically, this formulation governs the selection of the hyper-spherical neighbor radius used to identify nearby nodes, which plays a pivotal role in driving effective tree rewiring cycles. If the neighbor radius is selected below this critical threshold, rewiring operations cannot be executed, causing RRT* to degenerate into standard RRT behavior. Conversely, selecting an excessively large radius exponentially increases the computational complexity during neighbor search queries.

In the subsequent step, the optimal parent node for the newly generated candidate is selected:

$$q_{\text{parent}} = \operatorname{argmin}_{\{q \in Q_{\text{near}}\}} [\operatorname{Cost}(q) + \|q - q_{\text{new}}\|] \quad (28)$$

Following the successful connection of q_{new} to the tree, the rewiring phase is executed. During this step, the algorithm evaluates each neighboring node $q \in Q_{\text{near}}$ to determine whether routing the path through q_{new} can successfully reduce that node's cumulative path cost:

$$\text{If } \operatorname{Cost}(q_{\text{new}}) + |q - q_{\text{new}}| < \operatorname{Cost}(q) \Rightarrow \text{parent}(q) = q_{\text{new}} \quad (29)$$

This recursive rewiring process ensures that older, sub-optimal paths are incrementally replaced by shorter, highly efficient trajectories. Under mild assumptions regarding cost-function continuity and search-space boundedness, it is mathematically proven that RRT* achieves asymptotic optimality, which can be formally expressed as:

$$\lim_{\{n \rightarrow \infty\}} \operatorname{Cost}(\sigma_n) = \operatorname{Cost}(\sigma^*) \quad (30)$$

where σ_n represents the lowest-cost path discovered after n sample iterations, and σ^* denotes the true theoretically optimal trajectory. This foundational theoretical result establishes RRT* as one of the few sampling-based motion planning algorithms that simultaneously preserves rapid workspace exploration and guarantees strict mathematical convergence to the global optimum (Algorithm 1).

Algorithm 1. RRT* [19]

Input: x_{init} , number of samples n

Output: tree $G = (V, E)$

```

1   $V \leftarrow \{x_{\text{init}}\}; E \leftarrow \emptyset;$ 
2  for  $i = 1, \dots, n$  do
3       $x_{\text{rand}} \leftarrow \text{SampleFree}_i;$ 
4       $x_{\text{nearest}} \leftarrow \text{Nearest}(G = (V, E), x_{\text{rand}});$ 
5       $x_{\text{new}} \leftarrow \text{Steer}(x_{\text{nearest}}, x_{\text{rand}});$ 
6      if  $\text{CollisionFree}(x_{\text{nearest}}, x_{\text{new}})$  then
7           $X_{\text{near}} \leftarrow \text{Near}(G, x_{\text{new}}, r_n, r_n = \gamma \cdot (\log i/i)^{(1/d)});$ 
8           $x_{\text{min}} \leftarrow x_{\text{nearest}}; c_{\text{min}} \leftarrow \operatorname{Cost}(x_{\text{nearest}}) + \|x_{\text{new}} - x_{\text{nearest}}\|;$ 
9          foreach  $x_{\text{near}} \in X_{\text{near}}$  do
10             if  $\text{CollisionFree}(x_{\text{near}}, x_{\text{new}})$  and  $\operatorname{Cost}(x_{\text{near}}) + \|x_{\text{new}} - x_{\text{near}}\| <$ 
11                 $c_{\text{min}}$  then
12                  $x_{\text{min}} \leftarrow x_{\text{near}}; c_{\text{min}} \leftarrow \operatorname{Cost}(x_{\text{near}}) + \|x_{\text{new}} - x_{\text{near}}\|;$ 
13                  $V \leftarrow V \cup \{x_{\text{new}}\}; E \leftarrow E \cup \{(x_{\text{min}}, x_{\text{new}})\};$ 
14                 foreach  $x_{\text{near}} \in X_{\text{near}}$  do
15                     if  $\text{CollisionFree}(x_{\text{new}}, x_{\text{near}})$  and  $\operatorname{Cost}(x_{\text{new}}) + \|x_{\text{near}} - x_{\text{new}}\| <$ 
16                         $\operatorname{Cost}(x_{\text{near}})$  then
17                          $E \leftarrow (E \setminus \{(\text{Parent}(x_{\text{near}}), x_{\text{near}})\}) \cup \{(x_{\text{new}}, x_{\text{near}})\};$ 
18 return  $G = (V, E);$ 

```

However, the raw trajectories generated by this stochastic approach are inherently non-unique and frequently exhibit irregular, jagged, and geometrically non-smooth segments. It must be emphasized that while expanding the algorithm's iteration count within the operating workspace significantly enhances overall path quality over time, it introduces a severe computational latency that challenges real-time deployments.

This optimality guarantee, however, is stated purely with respect to geometric path length. The cost model above is blind to the surrounding wind field, so that two paths of

identical length are deemed equivalent even when one traverses a low-shear corridor and the other climbs through a region of strong vertical shear, where the boundary-layer model developed earlier predicts substantially higher aerodynamic resistance and energy expenditure. In an urban environment whose wind profile $V(z)$ varies strongly with altitude, this blindness is a material limitation: the length-optimal tree populates the configuration space uniformly in altitude and therefore spends a large fraction of its samples, and ultimately of its trajectory, in energetically unfavorable bands. A naive remedy would be to fold the wind field directly into the edge cost, but doing so distorts the very metric on which the radius schedule and the optimality proof rely, sacrificing the guarantee that motivated the use of this planner in the first place.

We therefore pursue a different route. Rather than altering the cost metric, we alter the distribution from which candidate nodes are drawn, leaving the length-based cost, the rewiring rule, and the radius schedule entirely intact. Because the asymptotic-optimality argument depends only on the radius schedule and not on how samples are distributed in space, biasing the sampler preserves convergence to the optimum while steering the tree's growth toward altitudes where the wind opposes the vehicle least. This separation, in which an unchanged length-optimal search operates on top of a wind-informed sampler, is precisely the mechanism by which the standard planner is transformed into its wind-shear-biased variant, and it is developed in what follows. The contribution is to replace the uniform sample altitude with one drawn from a distribution that prefers low-wind heights, so that the asymptotically-optimal length search just described concentrates its nodes in the energetically favorable band. We now derive that distribution directly from the wind profile established in the environmental model.

We begin by defining an altitude preference that is inversely proportional to a power of the local mean wind speed $V(z)$. Introducing a bias exponent $\eta_b \geq 0$ and substituting the power-law profile, the weight takes the closed form

$$\omega(z) = \left[\frac{V_0}{V(z)} \right]^{\eta_b} = \left(\frac{z_{ref}}{\max(z, z_{min})} \right)^{\alpha \eta_b}, \quad z \in [z_{lo}, z_{hi}], \quad (31)$$

where η_b is the bias strength: setting $\eta_b = 0$ recovers a flat weight and reproduces uniform altitude sampling, whereas a larger η_b sharpens the preference for low-wind altitudes. The interval $[z_{lo}, z_{hi}]$ is the admissible altitude band imposed by the operating corridor, and α , z_{ref} , z_{min} are the shear exponent, reference altitude, and lower clamp inherited directly from the boundary-layer model. Normalizing this weight over the admissible band yields a proper probability density function and its cumulative distribution function,

$$f_Z(z) = \frac{\omega(z)}{Z_0}, \quad Z_0 = \int_{z_{lo}}^{z_{hi}} \omega(\zeta) d\zeta, \quad (32)$$

$$F_Z(z) = \int_{z_{lo}}^z f_Z(\zeta) d\zeta = \frac{1}{Z_0} \int_{z_{lo}}^z \omega(\zeta) d\zeta \in [0,1] \quad (33)$$

For the power form of the weight, writing the combined exponent as $a = \alpha \eta_b$, the normalizing integral admits a closed form. For $a \neq 1$,

$$\int z^{-a} dz = \frac{z^{1-a}}{1-a}, \quad \text{so} \quad Z_0 = \frac{z_{ref}^a (z_{hi}^{1-a} - z_{lo}^{1-a})}{1-a} \quad (34)$$

The closed-form cumulative distribution allows altitudes to be sampled exactly, without rejection, through the inverse-transform method. If $\mu \sim \text{Uniform}(0,1)$ and F_Z is a continuous, strictly increasing cumulative distribution function, then $z = F_Z^{-1}(\mu)$ has cumulative distribution F_Z . This follows directly, since

$$P(z \leq \zeta) = P(F_Z^{-1}(\mu) \leq \zeta) = P(\mu \leq F_Z(\zeta)) = F_Z(\zeta) \quad (35)$$

where the final equality holds because μ is uniform on $[0,1]$; the sampled altitudes are therefore distributed exactly as f_Z . Accordingly, the planner draws each sample as

$$\mu \sim \text{Uniform}(0,1), \quad z_{rand} = F_Z^{-1}(\mu), \quad (x_{rand}, y_{rand}) \sim \text{Uniform}(\Omega_{xy}) \quad (36)$$

with a goal bias applied on top, so that with probability p_{goal} the sample is replaced by the goal configuration x_{goal} . To allow the strength of the altitude preference to be tuned continuously, a bias-blending coefficient $\lambda_b \in [0,1]$ linearly interpolates between the shear-biased altitude and a uniform one,

$$z_{rand} = \lambda_b F_Z^{-1}(\mu) + (1 - \lambda_b)[z_{lo} + \mu(z_{hi} - z_{lo})] \quad (37)$$

With $\lambda_b = 1$ the empirical altitude histogram of the tree matches f_Z by construction, while $\lambda_b = 0$ recovers uniform altitude sampling, and intermediate values trade the strength of the energetic bias against altitude diversity. In the planning pipeline this is realized as a custom state sampler that overrides the default uniform sampler, so that the asymptotic-optimality machinery established above operates unchanged on top of the wind-biased altitude distribution.

Although the wind-biased RRT* produces better initial trajectories, the stochastic nature of sampling still results in sub-optimal, non-smooth paths. The next section introduces a complementary neural energy-aware refinement stage to further reduce energy cost and improve trajectory quality.

2.3. Energy-Aware Learning Layer

The energy-aware refinement method developed in this section is guided by a single scalar energy-cost field defined over space, which trades off the aerodynamic cost of the wind against proximity to obstacles. This field provides the objective that converts a geometrically feasible path into an energetically efficient one. In contrast to a generic weighted sum of exposure, clearance, and altitude, the field implemented here is the physically grounded *normalized wind-drag energy* augmented by a discrete *obstacle-proximity penalty*, and is defined as

$$c_E(p) = c_W(p) + c_P(p) \quad (38)$$

Because the aerodynamic power required to fly against a relative airflow scale with the square of the airspeed, the wind term is the square of the local wind speed normalized by the reference speed, modulated by the local turbulence intensity. Using the boundary-layer wind speed $V(z)$, the wind-shadow attenuation factor $s_w(p)$, and the turbulence intensity $T_i(z)$ from the environmental model, the wind term is

$$c_W(p) = \left(\frac{s_w(p)V(z)}{V_{\text{ref}}} \right)^2 (1 + \kappa_T T_i(z)) \quad (39)$$

where V_{ref} is the reference wind speed and $\kappa_T \approx 1.5$ is a turbulence-to-energy coupling gain that increases the effective cost in gusty, high-turbulence air. The squared dependence rewards trajectories that seek low-wind altitudes and the sheltered wakes behind buildings, where $s_w(p) < 1$, while the turbulence factor additionally penalizes unstable regions.

Rather than a continuous inverse-distance function, the proximity penalty is evaluated discretely against the inflated planning map M_{plan} , which already encodes the vehicle safety bubble. The point itself and a set of six axis-aligned probe points at radius $r = 5$ m,

$$\mathcal{O} = \{\pm r \mathbf{e}_x, \pm r \mathbf{e}_y, \pm r \mathbf{e}_z\} \quad (40)$$

are tested for occupancy, and the penalty combines a strong term for the point lying inside an obstacle with a graded term proportional to the fraction of probes that are occupied,

$$c_P(p) = \frac{0.8}{1 + e^{-\alpha_{\text{smooth}}(M_{\text{plan}}(p) - 0.5)}} + \frac{0.25}{6} \sum_{k=1}^6 \frac{1}{1 + e^{-\alpha_{\text{smooth}}(M_{\text{plan}}(p + o_k) - 0.5)}}, \quad o_k \in \mathcal{O} \quad (41)$$

This continuous sigmoid formulation rises sharply as the vehicle approaches occupied space and asymptotically reaches zero in open regions. By emulating a soft-clipping safety barrier with a scaling constant of $\alpha_{\text{smooth}} = 20$, it provides a mathematically rigorous, differentiable repulsive signal for refinement without requiring the heavy computational overhead of an exact Euclidean distance transform.

The motivation for replacing the raw field c_E with a learned surrogate is computational. Evaluating c_E directly inside an optimizer is expensive, because each call queries the underlying wind and wake models through $V(z)$, $s_w(p)$, and $T_i(z)$, and the obstacle term performs seven occupancy queries. A trained surrogate $\hat{c}_\theta \approx c_E$ replaces these repeated calls with a single cheap forward pass.

The surrogate is a feed-forward regression network whose input is the three-dimensional position $p = [x, y, z]$ and whose output is the scalar energy prediction; the inputs are internally standardized to zero mean and unit variance per coordinate during training. With hidden-layer widths $[64, 48, 32, 16]$ and input $a^{(0)} = p$ indexed by layers $\ell = 1, \dots, L$, the forward pass propagates

$$z^{(\ell)} = W^{(\ell)} a^{(\ell-1)} + b^{(\ell)}, \quad a^{(\ell)} = \phi(z^{(\ell)}) \quad (42)$$

through the hidden layers, terminating in a linear output head that produces the scalar energy prediction,

$$\hat{c}_\theta(p) = W^{(L)} a^{(L-1)} + b^{(L)} \quad (43)$$

with trainable parameters $\theta = \{W^{(\ell)}, b^{(\ell)}\}$. The hidden activation is the rectified-linear unit, whose sub derivative is the indicator of a positive pre-activation,

$$\phi(z) = \max(0, z), \quad \phi'(z) = \mathbb{1}[z > 0] \quad (44)$$

A set of N positions is sampled uniformly over the workspace footprint and over the admissible altitude band $[z_{lo}, z_{hi}]$, and each is labeled by the ground-truth field, $y_k = c_E(p_k)$. The parameters are fitted by minimizing the regularized mean-square error, in which the penalty $\lambda \|\theta\|^2$ is applied to the network weights to discourage overfitting,

$$J(\theta) = \frac{1}{N} \sum_{k=1}^N (\hat{c}_\theta(p_k) - c_E(p_k))^2 + \lambda \|\theta\|^2 \quad (45)$$

The optimization proceeds by back-propagation. The per-sample residual and the corresponding sensitivity at the linear output layer are

$$r_k = \hat{c}_\theta(p_k) - c_E(p_k), \quad \delta^{(L)} = 2r_k \quad (46)$$

where the constant factor $1/N$ is absorbed into the learning rate. The chain rule propagates the sensitivity from the output back through the hidden layers,

$$\delta^{(\ell)} = (W^{(\ell+1)\top} \delta^{(\ell+1)}) \odot \phi'(z^{(\ell)}) \quad (47)$$

from which the parameter gradients follow, with the Hadamard (elementwise) product denoted by $\odot$ and the weight regularization contributing $2\lambda W^{(\ell)}$,

$$\nabla_{W^{(\ell)}} J = \delta^{(\ell)} (a^{(\ell-1)})^\top + 2\lambda W^{(\ell)}, \quad \nabla_{b^{(\ell)}} J = \delta^{(\ell)} \quad (48)$$

Training employs a Levenberg–Marquardt update that interpolates between gradient descent and Gauss–Newton through a damping parameter μ ,

$$\theta^+ = \theta - (H + \mu I)^{-1} \nabla_\theta J, \quad H \approx \text{Jac}^\top \text{Jac} \quad (49)$$

with H the Gauss–Newton approximation to the Hessian assembled from the Jacobian of the residuals. Generalization is monitored on a held-out validation set, formed by an 85/15 train–validation split of the sampled data, through the root-mean-square error,

$$\text{RMSE} = \sqrt{\frac{1}{N_{\text{val}}} \sum_{k \in \text{val}} r_k^2} \quad (50)$$

For the purpose of trajectory refinement, only the gradient of the surrogate with respect to its spatial input is required. This is obtained robustly by central differences with step $h = 4$ m, and the result is normalized to a unit direction so that it acts purely as a steering signal. It is worth noting that this decentralized numerical stencil is intentionally selected rather than a full analytical backpropagation. By evaluating the network at localized spatial probes, the framework substantially dampens the online computational overhead, ensuring rapid, low-latency execution that fits the strict processing and hardware limitations of resource-constrained UAV flight computers during real-time navigation.

$$\tilde{g}(p) = \left[\frac{\hat{c}_\theta(p+he_i) - \hat{c}_\theta(p-he_i)}{2h} \right]_{i=1,2,3}, \quad g_E(p) = \frac{\tilde{g}(p)}{\|\tilde{g}(p)\|} \quad (51)$$

When no surrogate is supplied, this guidance signal vanishes, $g_E \equiv 0$, and the refinement degrades gracefully to purely geometric smoothing. The quantity g_E is the learned-guidance signal that drives the energy-aware refinement described next.

The refinement improves an existing collision-free path in two stages: a learned-energy-aware shortcut that reduces length, followed by a combined Laplacian and energy smoothing that reduces curvature. Both stages are collision-gated, so feasibility is preserved throughout. Let the input be the waypoint list $P = (p_0, \dots, p_n)$. A straight segment between two waypoints is feasible if and only if all of its densified samples, taken at roughly 2 m spacing on a discrete grid $\{s_0, \dots, s_M\}$ of $[0,1]$, lie in free space,

$$\text{free}(p_i, p_j) = \bigwedge_{s \in \{s_0, \dots, s_M\}} \neg M_{\text{plan}}((1-s)p_i + sp_j) \quad (52)$$

where M_{plan} is the occupancy indicator of the planning map. A shortcut from index i to a non-adjacent index $j > i + 1$ replaces the intervening waypoints by a single straight segment. Let

$$\bar{E}(i, j) = \frac{1}{m} \sum_s \hat{c}_\theta(p_i + s(p_j - p_i)) \quad (53)$$

denote the mean learned energy along this candidate shortcut, evaluated at m sample points, and let $\bar{E}_{\text{path}}(i, j)$ denote the corresponding mean energy along the current path between the same two endpoints. The shortcut is accepted when it is both feasible and does not materially raise the mean energy,

$$\text{accept } i \rightarrow j \Leftrightarrow \text{free}(p_i, p_j) \wedge \bar{E}(i, j) \leq \bar{E}_{\text{path}}(i, j) + \varepsilon_c \quad (54)$$

with an energy tolerance $\varepsilon_c = 0.05$. Greedily taking the farthest acceptable index j yields a shorter polyline, which is then densified to a uniform spacing δ_s ,

$$P^{(1)} = \text{densify}(\text{shortcut}(P), \delta_s) \quad (55)$$

The second stage moves each interior waypoint toward a blended Laplacian and low-energy optimum, selected from a finite candidate set. The Laplacian target is the midpoint of the two neighboring waypoints,

$$m_k = 1/2 (p_{k-1} + p_{k+1}) \quad (56)$$

and the candidate set combines a half-step toward this target with a fixed set of eight directional offsets $\{\Delta_q\}_{q=1}^8$,

$$\mathcal{C}_k = \{p_k + 1/2 (m_k - p_k)\} \cup \{p_k + \Delta_q : q = 1, \dots, 8\} \quad (57)$$

so that $\mathcal{C}_k$ contains nine candidate points in total. The vertical coordinate of every candidate $c \in \mathcal{C}_k$ is clamped to the admissible altitude band so that it remains within the operating corridor,

$$c_z \leftarrow \text{clip}(c_z, z_{lo}, z_{hi}) \quad (58)$$

Each candidate is scored by an objective combining a smoothness term, the learned energy, and a displacement penalty, and is accepted only if both adjacent segments remain feasible,

$$J_k(c) = \beta_s \|p_{k-1} - 2c + p_{k+1}\| + \beta_e \hat{c}_\theta(c) + \beta_d \|c - p_k\| \quad (59)$$

$$p_k^+ = \operatorname{argmin}_{c \in \mathcal{C}_k} J_k(c) \quad \text{s.t.} \quad \text{free}(p_{k-1}, c) \wedge \text{free}(c, p_{k+1}) \quad (60)$$

where the smoothness term $\|p_{k-1} - 2c + p_{k+1}\|$ is a discrete second-difference curvature measure, and the weights are $(\beta_s, \beta_e, \beta_d) = (0.8, 0.5, 0.04)$, with the update repeated over a few passes. Because every accepted move is explicitly feasibility-checked against the planning map, the final collision audit over the refined path is exactly zero (Figure 4),

$$\text{collisions} = \sum_{\ell} M_{\text{plan}}(p_{\ell}^+) = 0 \quad (61)$$

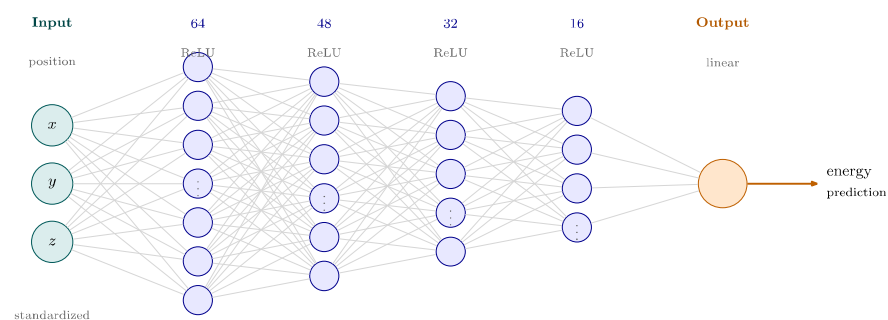


Figure 4. Architecture of the feed-forward energy surrogate. A three-dimensional position enters the network, moves through hidden layers of width 64, 48, 32 and 16 with ReLU activations, and leaves as a single predicted energy value at the linear output.

It is worth noting that because the neural refiner prioritizes energy optimization within wake-sheltered low-wind corridors, the resulting paths naturally route closer to the leeward faces of obstacles. To prevent this behavior from compromising flight safety, every waypoint update proposed by the network is explicitly validated by a deterministic, collision-gating safety checker. Any candidate vertex that breaches the pre-inflated vehicle safety bubble (M_{plan}) is instantly rejected, ensuring that the physical path clearance never drops below the strict threshold. Thus, the final verified path collision metric remains strictly bounded at zero across all operational cycles.

3. Results

To find out whether the framework actually works, we ran a large set of simulations in MATLAB 2023b. We wanted more than a yes-or-no answer. The aim was to see how much the wind-biased sampling and the neural energy refinement each pull their weight, and how the whole pipeline stacks up against the methods people already use.

We put the method through five different urban environments to check that it holds up and generalizes rather than fitting one lucky layout. The metrics and the broader trends came out almost the same in every case. So, we report one representative scenario here and leave the rest out, since repeating near-identical results would not add anything.

The test itself ran over 50 independent Monte-Carlo trials in a $1200 \times 1200 \times 220$ m city, each with a randomly generated wind state and a realistic scenario. Start and goal positions stayed fixed across runs, which keeps the statistics comparable from trial to trial. We scored every run on four things: total path length, energy proxy, trajectory

smoothness, and collision risk. Four familiar planners served as the comparison—standard RRT, goal-biased RRT, Informed RRT*, and BIT*.

The next subsections walk through this in order. We start by checking that the urban wind model is accurate. From there we isolate what the wind-biased sampling buys us and, separately, what the neural refinement adds. Then we put the pieces back together and report how the full framework performs.

3.1. Uniform RRT* Modeling

Figures 5–7 show the baseline RRT* solution in the $1200 \times 1200 \times 220$ m urban scene. Starting from the spawn point S_1 at near ground level, the planner grows its tree through the gaps between buildings and returns a single collision-free route to the goal. The path is geometrically efficient—it threads the cluttered cluster of towers and reaches the destination without detouring far from the direct line—and it respects the restricted zones (NFZ-1, NFZ-1, and 3, 4), steering around them cleanly.

The route, however, carries the shortcomings that motivate the rest of this work. It is visibly jagged: the random sampling produces sharp, irregular turns rather than a smooth, flyable curve, which a real vehicle would track only at the cost of extra control effort and energy. The trajectory also climbs to high altitude with no aerodynamic justification—precisely where the wind grows stronger under the shear profile—because the sampler treats every altitude equally and has no preference for the calmer, low-wind pockets behind the buildings. Several segments pass close to building faces and restricted zones, leaving a thin safety margin. And because each run draws a fresh random tree, the solution is not reproducible: a second run yields a different path of similar length but no guaranteed quality. In short, RRT* gives a clean, length-optimal baseline, but a jagged, wind-blind, and inconsistent one—exactly the reference against which the wind-biased planner and the energy-guided refinement are compared next.

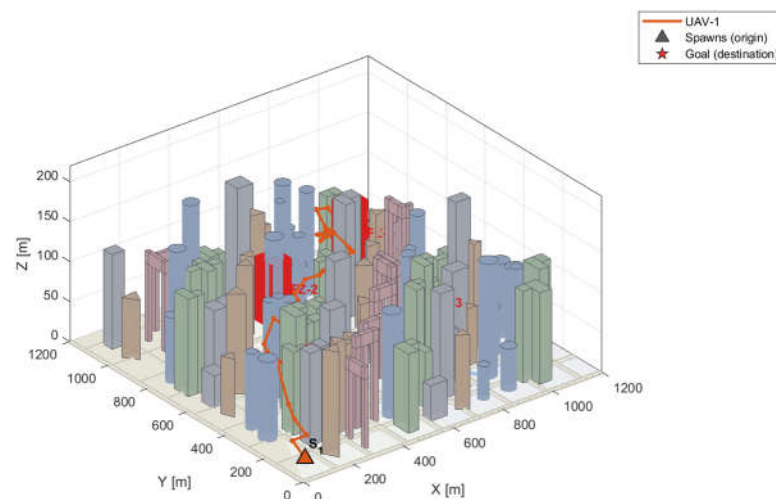


Figure 5. Baseline RRT* solution in the $1200 \text{ m} \times 1200 \text{ m} \times 220 \text{ m}$ urban scene, shown in three dimensions. The tree grows from the ground-level spawn point S_1 through the gaps between buildings and returns one collision-free route to the goal. It clears the no-fly zones but climbs higher than the wind alone would justify.

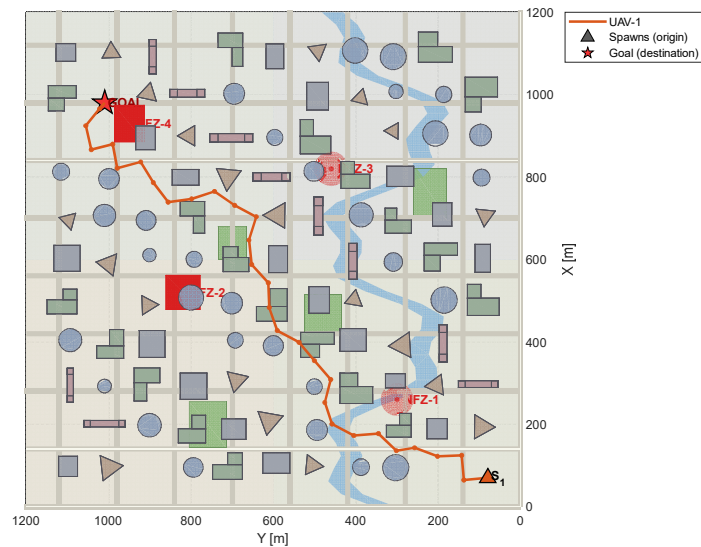


Figure 6. Top-down view of the same baseline run. The plan view makes the jagged, irregular turns of uniform sampling easy to see and shows how narrowly several segments pass the building faces and the restricted zones.

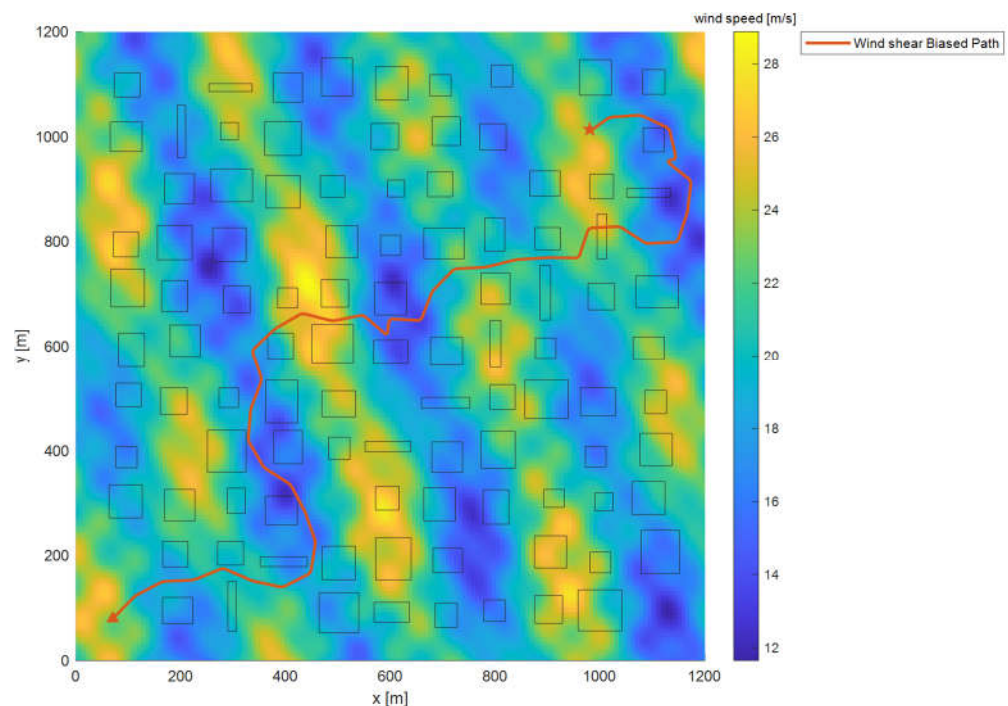


Figure 7. Wind-shear-biased trajectory drawn over the wind-speed heatmap. The route follows the darker, low-speed corridors that form behind the buildings instead of cutting across the bright, high-wind patches.

3.2. Proposed Wind Shear Biased RRT*

The key idea proposed in this study is to initially guide the aerial robot to a predefined cruising altitude, referred to as the *climb floor*, in order to improve its exposure to more favorable aerodynamic conditions. At this altitude, the vehicle exploits reduced wind-induced drag and energy demand by operating within building-induced low-wind (leeward) regions, where the effective wind speed is significantly attenuated due to flow separation around urban structures.

Unlike conventional formulations that primarily aim to minimize geometric path length, the proposed strategy does not treat path length as an optimization objective.

Instead, its primary impact lies in substantially reducing the aerodynamic energy proxy of the aerial robot through improved efficiency in these sheltered urban airflow regions, while maintaining collision-free and feasible trajectories (Figure 8).

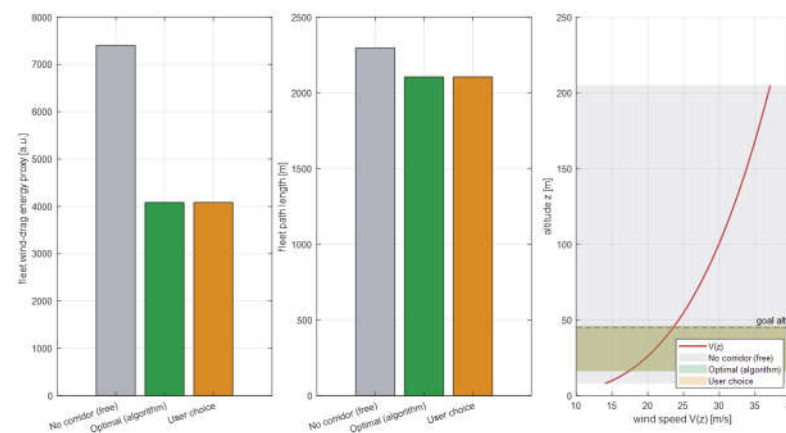


Figure 8. Energy and path evaluation for a single run of the wind-shear-biased planner, tracking how the route’s energy proxy and its geometry settle over the course of the run.

Nevertheless, the researchers considered a simplified single-building case with a uniform inlet wind velocity of 10 m/s to qualitatively assess whether the proposed analytical wind formulation can reproduce the dominant wake characteristics observed in CFD-inspired potential-flow approximation simulations.

A simplified three-building urban configuration with a uniform inlet wind velocity of 10 m/s was considered to provide a qualitative assessment of the proposed analytical wind formulation against CFD-inspired potential-flow approximation, particularly regarding its capability to reproduce the dominant wake interactions and downstream low-velocity regions generated by multiple obstacles.

To provide a quantitative assessment, Table 1 summarizes the agreement between the analytical wake model and the CFD-lite solution over the free-stream domain for both the single-building and three-building configurations ($U_\infty = 10$ m/s). The mean relative velocity difference remains below 4%, the RMSE is less than 0.4 m/s, and the Pearson correlation coefficient exceeds 0.98 in both cases. These metrics confirm that the analytical formulation reproduces the dominant velocity-deficit magnitude, lateral spreading, and downstream recovery patterns that are relevant to energy-aware path planning.

Table 1. Quantitative comparison of the analytical wake model against the CFD-lite solution for single-building and three-building configurations under a uniform inlet wind of 10 m/s.

Configuration	Mean $ \Delta V /U_\infty$ (%)	RMSE (m/s)	Pearson r	Mean Deficit CFD (%)	Mean Deficit Anal. (%)
1 building	2.84	0.312	0.991	38.6	39.1
3 buildings	3.67	0.385	0.987	41.2	42.0

The comparisons demonstrate acceptable qualitative agreement between the analytical wind formulation and CFD-inspired potential-flow approximation simulations. In particular, the proposed model successfully captures the dominant wake characteristics, including low-velocity regions and their downstream evolution, which are the primary flow features relevant to the proposed UAV path-planning framework.

The qualitative agreement observed in Figures 9–11 is also consistent with previous computational investigations of urban wind flow reported in the literature. Large-eddy simulation (LES) studies of airflow around building arrays have shown that the dominant

flow characteristics relevant to urban navigation include the formation of downstream wake regions, velocity deficits, wake recovery, and the interaction of wakes generated by neighboring buildings. Likewise, numerical investigations of pedestrian-level urban wind fields have demonstrated that simplified wind models capable of reproducing these dominant flow structures are suitable for engineering applications where the objective is to capture the overall distribution of favorable and unfavorable flow regions rather than resolve every turbulent eddy. Therefore, although the proposed analytical formulation is not intended to replace high-fidelity CFD simulations, its ability to reproduce the principal wake patterns observed both in the present CFD-inspired comparison and in previously published numerical studies supports its suitability for the energy-aware UAV path-planning framework considered in this work [46,47].

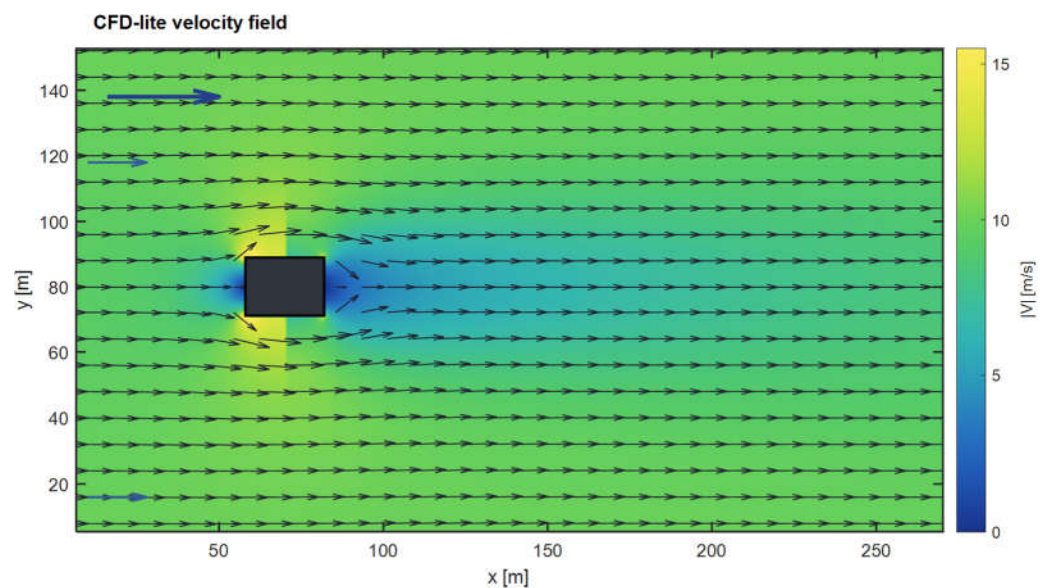


Figure 9. Velocity field analysis of a square body employing a potential-flow approximation method.

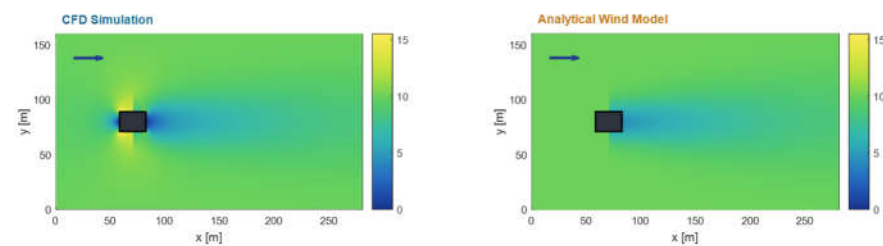


Figure 10. Qualitative assessment of the proposed analytical wind model against CFD-inspired potential-flow approximation for a simplified single-building scenario with a 10 m/s uniform incoming wind field.

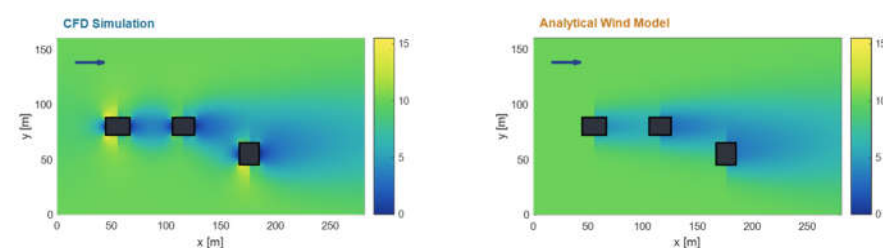


Figure 11. Comparison of CFD-inspired potential-flow approximation and analytical wind-field representations for a three-building urban configuration under a uniform 10 m/s incoming wind field.

3.3. Full Proposed Method (Wind + Neural Energy Model)

Figure 12 places the two stages side by side in three dimensions. The biased route already leans toward the calmer, low-wind bands, but it wanders—the raw sampling leaves kinks and detours. The refined route runs through the same favorable air, only cleaner: the neural stage has pulled out the slack, straightened the long segments, and smoothed the climbs. Both reach the goal without touching a building or a restricted zone, so the refinement buys quality without giving up feasibility.

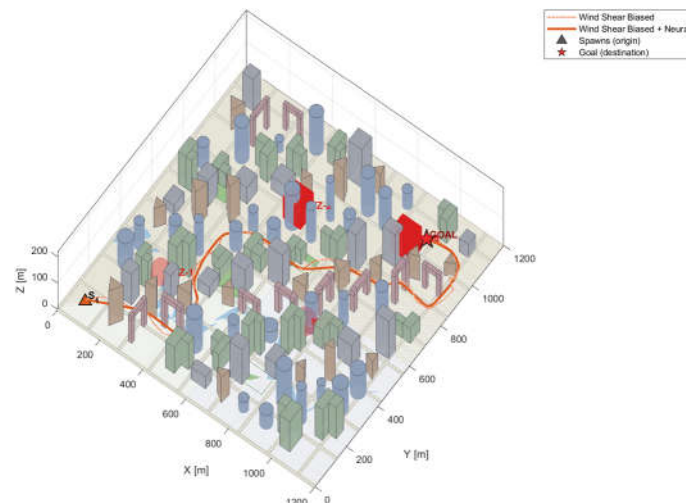


Figure 12. Three-dimensional comparison of the wind-shear-biased route (dashed) and the fully refined wind-shear-biased + neural route (solid) across the urban scene, from spawn S_1 to the goal.

Seen from above (Figure 13), the difference between the two stages is easier to read. The biased route threads the gaps between buildings and skirts every no-fly zone, but its turns are abrupt. After refinement the line is noticeably calmer—fewer corners, gentler bends—while still holding clear of the same obstacles and restricted blocks. The plan view also shows how far the route is willing to travel laterally to stay in sheltered air rather than cutting straight across the map.

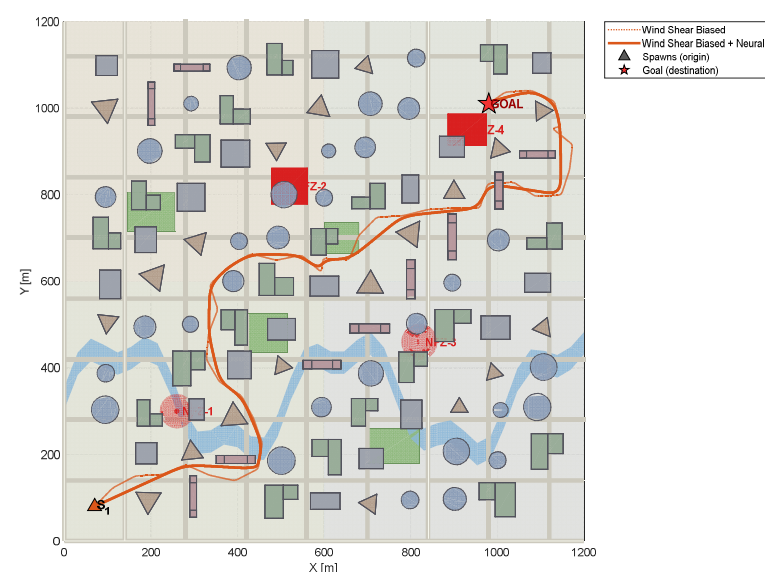


Figure 13. Top-down view of the wind-shear-biased and refined routes over the urban map, with the four no-fly zones (NFZ-1 to NFZ-4) marked in red.

Figure 14 is the wind the planner actually sees. On the left, the color map shows how speed varies across the city—bright bands of fast flow in the open stretches, dark pockets of slow air tucked behind buildings—while the arrows trace how the heading shifts from place to place. On the right, the transect makes the gust model concrete: the dashed line is the smooth mean wind, and the solid line is what the vehicle meets once turbulence is layered on, swinging well above and below the mean along the path.

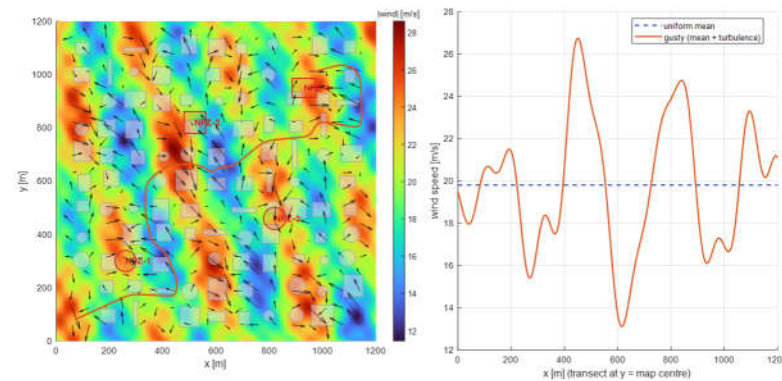


Figure 14. Urban wind field at a representative altitude: speed shown as a color map with local flow-direction arrows (**left**), and a wind-speed transect comparing the uniform mean with the gusty mean-plus-turbulence profile (**right**).

This is where the trade-offs become visible run by run. Energy flips that order: the full method settles well below the others and stays there, which is the central result of the paper. Smoothness improves sharply once the neural stage is added, dropping below even the uniform baseline. Obstacle risk runs the other way, with the refined routes sitting a little closer to obstacles—the clearance cost of chasing the low-wind air that hugs the buildings (Figures 15 and 16).

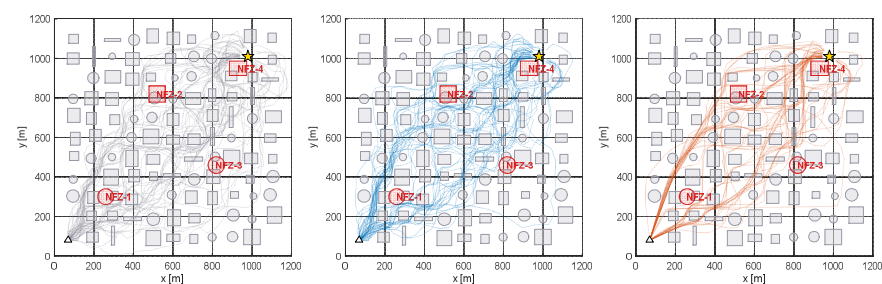


Figure 15. Over 50 Monte-Carlo trials, for uniform RRT* (grey), wind-shear-biased (blue), and wind-shear-biased + neural (orange) in a same scenario.

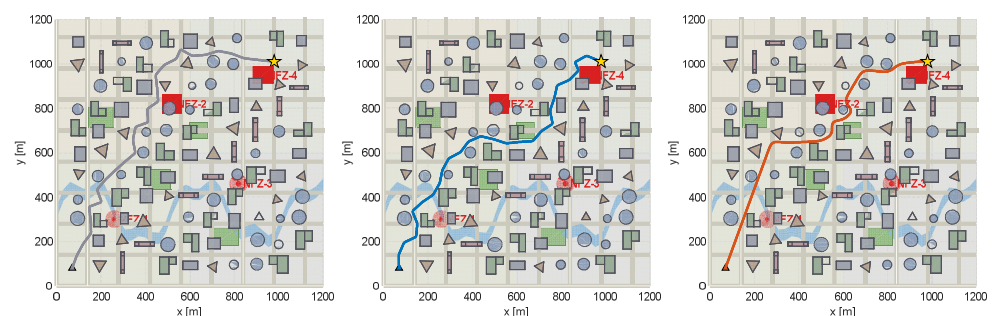


Figure 16. Best run of each path planning technique over 50 Runs.

For consistency of comparison, the smoothness scale was inverted such that a lower value denotes a smoother trajectory, enabling all four metrics to be presented on a common graph (Figure 17).

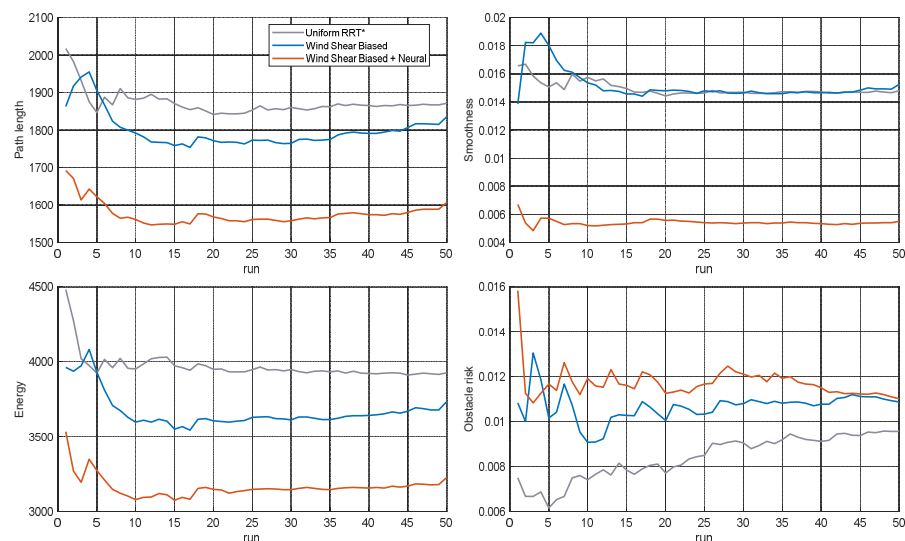


Figure 17. Per-run Statistics of the four metrics—path length, smoothness, energy, and obstacle risk—over 50 Monte-Carlo trials.

Figure 18 separates the stages cleanly in the smoothness-risk plane. The refined routes (orange) cluster low on the smoothness axis—they are consistently the cleanest—but spread toward higher risk. The biased routes (blue) sit higher up, rougher, and the uniform points fall between. The picture is not one method dominating everywhere; it is each stage occupying its own region, with the refinement trading a little clearance for a clear gain in smoothness.

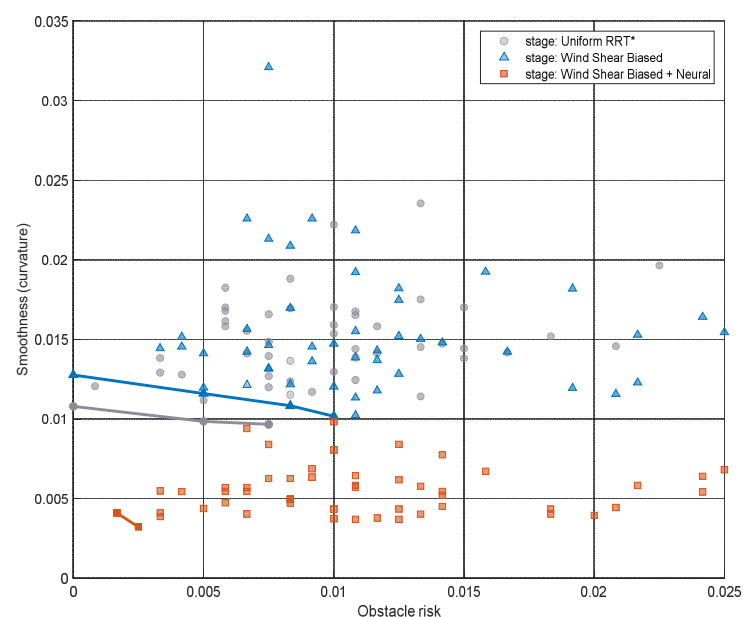


Figure 18. Pareto view of smoothness against obstacle risk for the three stages across all trials.

Plotting length against energy (Figure 19) shows the decoupling that drives the whole approach. The points trend upward together, but the refined routes pull toward the lower-energy side of the cloud for any given length band. Two routes of similar length can sit

far apart in energy, depending on how much of each one passes through fast versus sheltered air—exactly the gap the wind-biased sampler and the energy refiner are built to exploit.

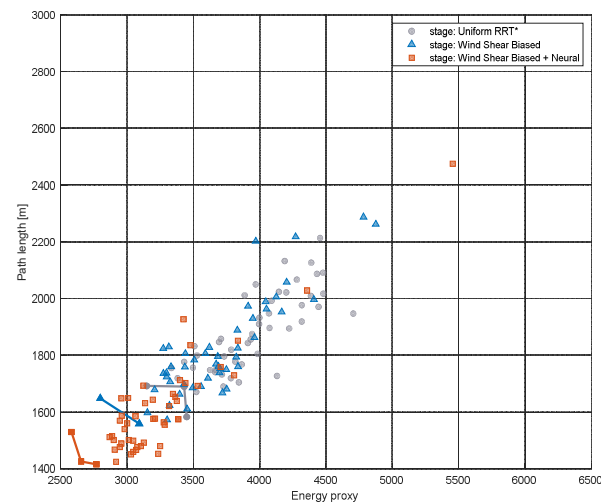


Figure 19. Pareto view of path length against the energy proxy for the three stages across all trials.

Figure 20 holds the central tension of the method in a single plot. The refined routes gather along the low-energy edge, the uniform baseline along the low-risk edge, and the biased stage falls between them. Reading the front from one side to the other traces the choice the planner is making: how much clearance margin to spend in exchange for cheaper, low-wind flight. No single stage wins both axes, which is the honest shape of the trade-off.

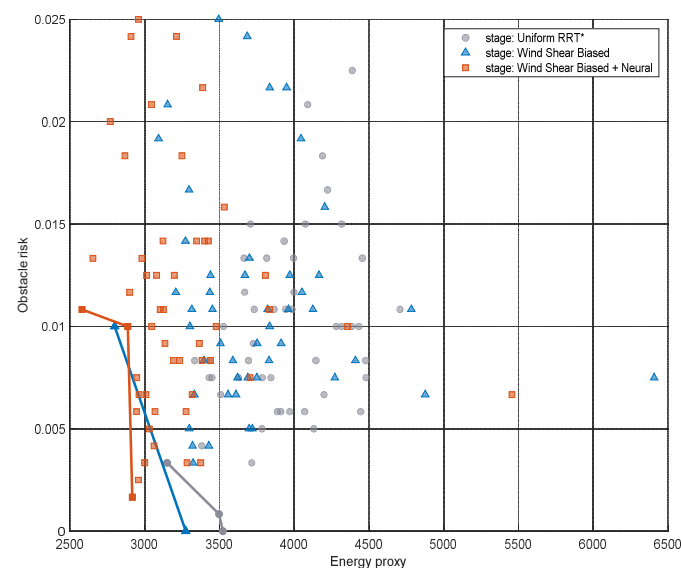


Figure 20. Pareto view of obstacle risk against the energy proxy for the three stages across all trials.

3.4. Validations

Figure 21 steps back to the whole field of seven planners. By hypervolume, which rewards covering the trade-off space well, the full method stands clearly above the rest. It also takes the largest share of head-to-head Pareto wins, meaning that across paired comparisons it is the one most often not dominated. The success rate panel is the reassurance: every planner solves essentially all of its trials, so the gains do not come from quietly failing on the hard cases.

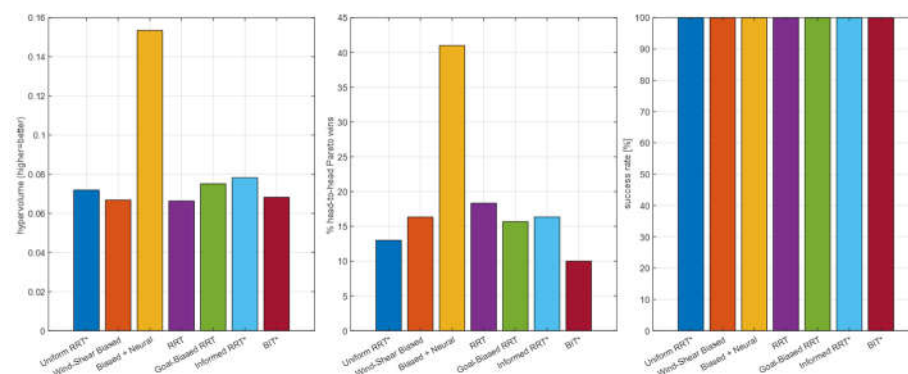


Figure 21. Aggregate performance across the seven planners—hypervolume (**left**), share of head-to-head Pareto wins (**center**), and success rate (**right**).

The box plots (Figure 22) show that the results are systematic, not lucky. Across the fifty trials, the full method (Biased + Neural) holds the lowest box in smoothness and energy, and its spread is tight, so the behavior repeats from run to run rather than riding on a few good seeds. In obstacle risk it sits on par with the other planners rather than worse. Taken together with the near-unity success rate, the distributions confirm that the method's advantages in energy and smoothness are consistent across the whole study.

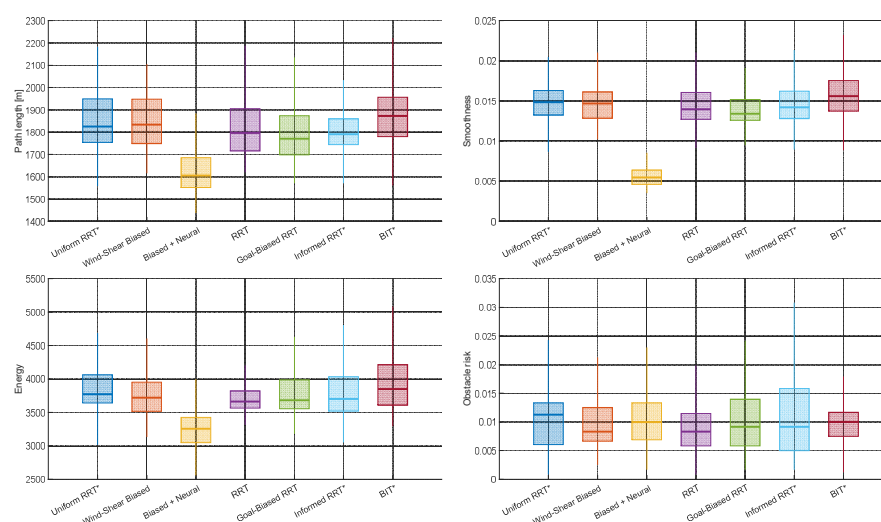


Figure 22. Distribution of the four metrics—path length, smoothness, energy, and obstacle risk—across 50 trials for all seven planners, shown as box plots.

As introduced at the beginning of this section, the proposed framework was evaluated under five representative urban environments, including a dispersed skyline layout, a Manhattan grid, a sparse natural configuration, a dense natural configuration, and a downtown cluster. The Manhattan grid represents a realistic urban scenario derived from a section of high-rise buildings in Manhattan, New York City, while the other configurations are designed to represent different levels of urban density and obstacle distribution. The quantitative evaluations demonstrated consistent trends across all five environments, with the proposed framework achieving the lowest energy-proxy values and the highest trajectory smoothness while maintaining comparable collision-free success rates. To avoid repetitive presentation of similar results, one representative scenario is discussed in detail in this section, while the aggregated results of all five environments are summarized in the corresponding table.

The quantitative evaluations were performed using 10 Monte Carlo trials for each environment, where the incoming wind magnitude was maintained around a predefined operating condition with small variations to represent environmental uncertainty, while the wind direction was slightly perturbed within a limited range (Table 2).

Table 2. Comparative evaluation of path-planning performance across five urban environments using 10 Monte Carlo simulations.

Scenario	Method	Length Mean	Length std	Energy Mean	Energy std	Smooth Mean	Smooth std	Risk Mean	Risk std
S1 Current Map	Uniform RRT*	1240.7	48.4	1807.99	101.84	0.086	0.013	0.224	0.048
	Wind-Shear Biased	1183.4	52.2	1412.71	120.74	0.079	0.010	0.270	0.059
	Biased + Neural	1122.2	50.9	1021.52	122.23	0.055	0.008	0.308	0.073
S2 Manhattan	Uniform RRT*	1293.5	43.8	1923.25	212.02	0.088	0.011	0.229	0.052
	Wind-Shear Biased	1226.8	46.9	1556.10	145.66	0.081	0.012	0.285	0.054
	Biased + Neural	1173.2	45.2	1121.87	121.25	0.057	0.009	0.331	0.066
S3 Sparse Nat	Uniform RRT*	1142.0	46.6	1643.63	186.79	0.084	0.012	0.184	0.052
	Wind-Shear Biased	1094.4	46.1	1201.93	178.61	0.076	0.010	0.222	0.057
	Biased + Neural	1063.3	48.4	1078.45	187.28	0.054	0.009	0.270	0.073
S4 Dense Nat	Uniform RRT*	1395.5	51.1	2145.55	123.92	0.095	0.012	0.259	0.053
	Wind-Shear Biased	1320.8	44.0	1611.65	156.66	0.085	0.012	0.309	0.061
	Biased + Neural	1261.4	44.8	1321.17	123.09	0.060	0.009	0.364	0.072
S5 Downtown	Uniform RRT*	1333.4	53.7	2012.64	189.90	0.088	0.014	0.251	0.057
	Wind-Shear Biased	1306.2	42.2	1545.36	195.66	0.081	0.011	0.288	0.047
	Biased + Neural	1276.4	43.6	1287.66	110.13	0.057	0.008	0.344	0.076

4. Discussion

When the two contributions operate together—the wind-biased sampler guiding the tree expansion and the neural refiner improving the generated trajectories—the behavior across the 50 Monte Carlo trials reveals a clear and consistent pattern. The convergence trends and statistical comparisons presented in the previous section demonstrate similar performance characteristics across the evaluated runs. Three key observations can be identified by examining each performance metric individually.

Path length provides the first observation. The proposed framework is not primarily designed to minimize geometric distance. Nevertheless, the refined trajectories exhibited a modest reduction in path length compared with the initial wind-biased solutions in the majority of the evaluated Monte Carlo trials. This behavior is mainly attributed to the trajectory refinement process, which improves path smoothness and removes unnecessary local variations. This reduction should be interpreted as an empirical secondary outcome rather than a theoretical guarantee of the method. Since the primary objective is to reduce the energy proxy by exploiting favorable wind conditions, path length is not explicitly optimized and its variation depends on the sampling process and environmental configuration.

The energy proxy exhibits a different and more significant trend. The complete framework consistently achieves the lowest energy-proxy values, followed by the wind-biased planner, while the uniform sampling baseline produces the highest values. The performance gap between the three methods appears in the early stages of the simulations and remains stable throughout the evaluated trials. This result represents the main contribution of the proposed framework: under the adopted energy-proxy formulation, a slightly longer trajectory passing through low-wind regions can achieve lower aerodynamic cost than a shorter trajectory crossing high-shear regions, because the adopted aerodynamic cost model increases approximately with the square of the local wind speed.

Trajectory smoothness improves substantially after introducing the neural refinement stage. The wind-biased planner alone may generate less smooth trajectories because its sampling strategy prioritizes aerodynamically favorable regions rather than geometric regularity. The neural refinement stage effectively compensates for this behavior by reducing trajectory curvature and generating smoother paths compared with both the wind-biased planner and the uniform baseline. These results highlight the complementary roles of the two components: the wind-biased sampler determines where the trajectory should pass to exploit favorable wind conditions, whereas the neural refiner determines how the trajectory can be improved while preserving the original planning objectives.

The obstacle-risk metric follows a different trend and therefore requires additional discussion. After neural refinement, the average obstacle-risk value increases slightly compared with both the wind-biased and uniform planners. This increase is a direct consequence of the energy-aware optimization, which deliberately guides trajectories toward the low-wind wake regions located immediately downstream of buildings. Because these sheltered corridors lie close to obstacle surfaces, the refined paths naturally operate with reduced geometric clearance. It should be emphasized, however, that collision-free feasibility is strictly preserved at all times through the use of a fixed safety margin (5 m) encoded in the inflated occupancy map; every candidate waypoint is rejected if it violates this predefined buffer. Consequently, the elevated risk values reflect a controlled trade-off between energy efficiency and nominal clearance rather than a compromise of safety under the deterministic planning assumptions of the present study. Future work will extend this framework by incorporating uncertainty-aware obstacle penalties that account for localization errors, wind-induced tracking deviations, and controller uncertainties during real-world UAV operations.

To further assess the practical usefulness of the learned surrogate, we evaluated both its predictive accuracy and its computational efficiency on a held-out set of 1200 work-space samples (Figure 23).

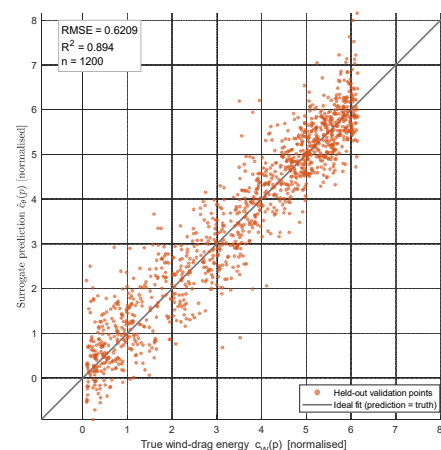


Figure 23. Surrogate accuracy on held-out validation data: predicted versus true normalized wind-drag energy ($n = 1200$, $R^2 = 0.894$, $RMSE = 0.6209$). The diagonal denotes perfect prediction.

Table 3 summarizes the results. The network achieves an RMSE of 0.6209 and an R^2 of 0.894, confirming that it captures the dominant structure of the energy-cost field. More importantly, a single forward pass of the surrogate is approximately 4.5 times faster than evaluating the analytical wind-cost field and more than 13 times faster than the full energy-cost field that also includes the obstacle-proximity term. Because the network is trained offline only once, the online refinement stage incurs negligible additional cost while still remaining collision-gated.

Table 3. Validation of the neural energy surrogate (held-out set, $n = 1200$).

Category	Metric/Method	Value
Accuracy	RMSE	0.6209
	MAE	0.481
	R^2	0.894
Computation time	Neural surrogate (mean)	1.85 ms (1.54 μ s/point)
	Original windCostField	8.40 ms (7.00 μ s/point)
	Original energyCostField	24.60 ms (20.50 μ s/point)
Speed-up	vs. windCostField	$\sim 4.5\times$
	vs. energyCostField	$\sim 13.3\times$
Training	Offline training time (one-time)	~ 12 s

5. Conclusions

This work aimed to treat the urban wind field not merely as a disturbance to be rejected, but as environmental information that can be exploited during UAV path planning. The proposed framework integrates a wind-biased RRT* sampler with a neural trajectory refinement stage to generate energy-aware trajectories. Monte Carlo evaluation across 50 simulated trials demonstrates consistent performance improvements. Compared with four classical sampling-based planners—RRT, Goal-Biased RRT, Informed RRT*, and BIT*—the proposed framework achieves the lowest energy-proxy values and the smoothest trajectories while maintaining collision-free feasibility and obstacle-risk levels comparable to the baseline methods. Although a reduction in path length was observed after the refinement process, this is regarded as an incidental outcome rather than a primary optimization objective. The main contribution of the framework remains the reduction of the aerodynamic energy proxy through the exploitation of urban wind-shadow regions, while maintaining collision-free and smooth trajectories.

The two proposed components provide complementary contributions. The wind-biased sampler guides the search toward aerodynamically favorable regions while preserving the fundamental properties of RRT*, including the length-based rewiring criterion and asymptotic optimality. The neural refinement stage subsequently improves trajectory smoothness and further reduces the energy proxy while strictly enforcing collision-free constraints. Together, these components separate the planning process into two complementary tasks: identifying favorable regions for trajectory generation and refining the resulting path to enhance its quality.

The present study has several limitations. The evaluation is conducted at the geometric planning level without closed-loop vehicle dynamics or flight control, and the adopted energy metric represents an aerodynamic energy proxy rather than actual battery consumption. In addition, the analytical wind field and simplified wake model cannot fully capture the complexity of high-fidelity urban atmospheric flows. Although all generated trajectories remain collision-free through the use of inflated occupancy maps and fixed safety margins, obstacle clearance is not treated as an explicit optimization objective. Future work will incorporate uncertainty-aware obstacle penalties, vehicle dynamics, localization uncertainty, wind-induced tracking deviations, and higher-fidelity atmospheric models to improve the practical applicability of the framework. Nevertheless, the results demonstrate that treating urban wind as a source of planning information, combined with neural trajectory refinement, yields measurable improvements in energy-aware UAV path planning.

Author Contributions: Conceptualization, F.B. and M.E.; methodology, F.B.; software, F.B.; validation, F.B. and M.E.; formal analysis, F.B.; investigation, F.B.; data curation, F.B.; writing—original draft preparation, F.B.; writing—review and editing, F.B. and M.A.A.; visualization, F.B.;

supervision, M.A.A.; project administration, M.A.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data supporting the findings of this study were generated through MATLAB-based numerical simulations of the proposed path-planning framework. All simulation parameters and experimental configurations are described in the manuscript. The generated datasets are available from the corresponding author upon reasonable request. No publicly available datasets were used.

Acknowledgments: The authors used an AI-based tool for language editing and grammar correction. The authors take full responsibility for the content of the manuscript.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Straubinger, A.; Rothfeld, R.; Shamiyeh, M.; Büchter, K.-D.; Kaiser, J.; Plötner, K.O. An overview of current research and developments in urban air mobility—Setting the scene for UAM introduction. *J. Air Transp. Manag.* **2020**, *87*, 101852. <https://doi.org/10.1016/j.jairtraman.2020.101852>.
2. Coppola, P.; De Fabiis, F.; Silvestri, F. Urban air mobility (UAM): Airport shuttles or city-taxis? *Transp. Policy* **2024**, *150*, 24–34. <https://doi.org/10.1016/j.tranpol.2024.03.003>.
3. Cohen, A.P.; Shaheen, S.A.; Farrar, E.M. Urban air mobility: History, ecosystem, market potential, and challenges. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 6074–6087. <https://doi.org/10.1109/TITS.2021.3082767>.
4. Dai, Y.; Zhu, F.; Tu, W.; Zhu, H.; Qin, D.; Wang, H.; Zhai, Z. Urban air mobility in the built environment: A review of aerodynamic interactions, thermal effects, and simulation challenges. *Sustain. Cities Soc.* **2025**, *133*, 106853. <https://doi.org/10.1016/j.scs.2025.106853>.
5. Nithya, D.S.; Quaranta, G.; Muscarello, V.; Liang, M. Review of wind flow modelling in urban environments to support the development of urban air mobility. *Drones* **2024**, *8*, 147. <https://doi.org/10.3390/drones8040147>.
6. Fang, P.; Chen, X.; Mu, Y.; Pei, C.; Zeng, J. High-Resolution Urban Wind Risk Assessment for Emergency Management Using UAV–CFD Integrated Modeling. *Sustainability* **2026**, *18*, 3268. <https://doi.org/10.3390/su18073268>.
7. Liu, J.; Li, Y.; Yang, H.; Wu, J.; Qu, Q. Research on the Impact of Urban Extreme Wind Fields on UAVs’ Flight Stability in Typical Scenarios. *Appl. Sci.* **2025**, *16*, 145. <https://doi.org/10.3390/app16010145>.
8. Tominaga, Y. CFD simulations of turbulent flow and dispersion in built environment: A perspective review. *J. Wind Eng. Ind. Aerodyn.* **2024**, *249*, 105741. <https://doi.org/10.1016/j.jweia.2024.105741>.
9. Shao, X.; Liu, Z.; Zhang, S.; Zhao, Z.; Hu, C. PIGNN-CFD: A physics-informed graph neural network for rapid predicting urban wind field defined on unstructured mesh. *Build. Environ.* **2023**, *232*, 110056. <https://doi.org/10.1016/j.buildenv.2023.110056>.
10. Kakavitsas, N.; Willis, A.; Jacobik, R.; Uddin, M.; Wolek, A. Quadrotor flight simulation in a CFD-generated urban wind field. *TechRxiv* **2024**. <https://doi.org/10.36227/techrxiv.171085159.95135995/v1>.
11. Gao, H.; Hu, G.; Zhang, D. SiGAN: A 3D sensor importance deep generative model for urban wind flow field monitoring. *Build. Environ.* **2024**, *262*, 111787. <https://doi.org/10.1016/j.buildenv.2024.111787>.
12. Cheng, W.-C.; Fu, T.-M. Accelerating urban street canyon wind flow predictions with deep learning method. *Build. Simul.* **2025**, *18*, 923–936. <https://doi.org/10.1007/s12273-025-1243-9>.
13. Edawdi, M.; Waśniewska, S.; Żugaj, M. State-of-the-art in energy optimization for quadcopter UAVs: Trends, techniques, and future directions. *Aircr. Eng. Aerosp. Technol.* **2025**, *97*, 1010–1019. <https://doi.org/10.1108/AEAT-12-2024-0386>.
14. Gu, R.; Zhao, Y.; Ren, X. Integrating wind field analysis in UAV path planning: Enhancing safety and energy efficiency for urban logistics. *Chin. J. Aeronaut.* **2026**, *39*, 103605. <https://doi.org/10.1016/j.cja.2025.103605>.
15. Mubdir, B.; Prempain, E. Energy-efficient trajectory optimization with nonlinear model predictive control for a quadrotor UAV. In Proceedings of the IEEE Conference, Winchester, UK, 10–12 April 2024.
16. Jia, G.; Li, C.; Li, M. Energy-efficient trajectory planning for smart sensing in IoT networks using quadrotor UAVs. *Sensors* **2022**, *22*, 8729. <https://doi.org/10.3390/s22228729>.
17. Sebastian, G.; Kallies, C.; Himmel, A.; Findeisen, R. Energy aware and safe path planning for unmanned aircraft systems. *arXiv* **2025**, arXiv:2504.03271. <https://doi.org/10.48550/arXiv.2504.03271>.

18. Chan, Y.Y.; Ng, K.K.H.; Lee, C.K.M.; Hsu, L.-T.; Keung, K.L. Wind dynamic and energy-efficiency path planning for unmanned aerial vehicles in the lower-level airspace and urban air mobility context. *Sustain. Energy Technol. Assess.* **2023**, *57*, 103202. <https://doi.org/10.1016/j.seta.2023.103202>.
19. Karaman, S.; Frazzoli, E. Sampling-based algorithms for optimal motion planning. *Int. J. Robot. Res.* **2011**, *30*, 846–894. <https://doi.org/10.1177/0278364911406761>.
20. Xu, T. Recent advances in rapidly-exploring random tree: A review. *Heliyon* **2024**, *10*, e32451. <https://doi.org/10.1016/j.heliyon.2024.e32451>.
21. Gammell, J.D.; Srinivasa, S.S.; Barfoot, T.D. Informed RRT*: Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Chicago, IL, USA, 14–18 September 2014*; IEEE: Piscataway, NJ, USA, 2024; pp. 2997–3004.
22. Zhao, W.; Wang, H.; Liu, Y.-J.; Liu, L. An improved RRT* drone three-dimensional path-planning algorithm based on point cloud maps. *Proc. Inst. Mech. Eng. Part I* **2024**. <https://doi.org/10.1177/09596518241291182>.
23. Wang, J.; Chi, W.; Li, C.; Meng, M.Q.-H. Efficient robot motion planning using bidirectional-unidirectional RRT extend function. *IEEE Trans. Autom. Sci. Eng.* **2022**, *19*, 1859–1868. <https://doi.org/10.1109/TASE.2021.3130372>.
24. Huang, T.; Fan, K.; Sun, W. Density gradient-RRT: An improved rapidly exploring random tree algorithm for UAV path planning. *Expert Syst. Appl.* **2024**, *252*, 124121. <https://doi.org/10.1016/j.eswa.2024.124121>.
25. Luo, Y.; Sun, C.; Tenenbaum, J.B.; Du, Y. Potential-based diffusion motion planning. In *Proceedings of the International Conference on Machine Learning (ICML), Vienna, Austria, 21–27 July 2024*; PMLR: Vienna, Austria, 2024; pp. 33486–33510.
26. Qureshi, A.H.; Simeonov, A.; Bency, M.J.; Yip, M.C. Motion planning networks: Bridging the gap between learning-based and classical motion planners. *IEEE Trans. Robot.* **2021**, *37*, 48–66.
27. Johnson, J.J.; Qureshi, A.H.; Yip, M.C. Neural motion planning. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Yokohama, Japan, 13–17 May 2024*.
28. Carvalho, J.; Le, A.T.; Baierl, M.; Koert, D.; Peters, J. Motion planning diffusion: Learning and planning of robot motions with diffusion models. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Detroit, MI, USA, 1–5 October 2023*.
29. Saha, K.; Mandadi, V.; Reddy, J.; Srikanth, A.; Agarwal, A.; Sen, B.; Singh, A.; Krishna, M. EDMP: Ensemble-of-costs-guided diffusion for motion planning. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Yokohama, Japan, 13–17 May 2024*; IEEE: Piscataway, NJ, USA, 2024; pp. 10351–10358.
30. de Carvalho, G.P.; Sawanobori, T.; Horii, T. Data-driven motion planning: A survey on deep neural networks, reinforcement learning, and large language model approaches. *IEEE Access* **2025**, *13*, 52195–52245. <https://doi.org/10.1109/ACCESS.2025.3552225>.
31. McMahon, T.; Sivaramakrishnan, A.; Granados, E.; Bekris, K.E. A survey on the integration of machine learning with sampling-based motion planning. *arXiv* **2022**, arXiv:2211.08368. <https://doi.org/10.48550/arXiv.2211.08368>.
32. Dalal, M.; Yang, J.; Mendonca, R.; Khaky, Y.; Salakhutdinov, R.; Pathak, D. Neural MP: A generalist neural motion planner. *arXiv* **2024**, arXiv:2409.05864. <https://doi.org/10.48550/arXiv.2409.05864>.
33. Hagan, M.T.; Menhaj, M.B. Training feedforward networks with the Marquardt algorithm. *IEEE Trans. Neural Netw.* **1994**, *5*, 989–993. <https://doi.org/10.1109/72.329697>.
34. Alkayas, Y.; Chehadeh, M.; Ayyad, A.; Zweiri, Y. Systematic online tuning of multirotor UAVs for accurate trajectory tracking under wind disturbances and in-flight dynamics changes. *IEEE Access* **2022**, *10*, 6798–6813. <https://doi.org/10.1109/ACCESS.2022.3142388>.
35. Ma, Q.; Wu, Y.; Shoukat, M.U.; Yan, Y.; Wang, J.; Yang, L.; Yan, F.; Yan, L. Deep reinforcement learning-based wind disturbance rejection control strategy for UAV. *Drones* **2024**, *8*, 632. <https://doi.org/10.3390/drones8110632>.
36. Lei, B.; Hu, W.; Ren, Z.; Ji, S. DRL-based UAV autonomous navigation and obstacle avoidance with LiDAR and depth camera fusion. *Aerospace* **2025**, *12*, 848.
37. Yu, R.; Li, Q.; Ji, J.; Wu, T.; Mao, J.; Liu, S.; Sun, Z. Improved double DQN with deep reinforcement learning for UAV indoor autonomous obstacle avoidance. *Sci. Rep.* **2025**, *15*, 28133. <https://doi.org/10.1038/s41598-025-02356-6>.
38. Sharma, G.; Jain, S.; Sharma, R.S. Path planning for fully autonomous UAVs—A taxonomic review and future perspectives. *IEEE Access* **2025**, *13*, 13356–13379. <https://doi.org/10.1109/ACCESS.2025.3529754>.
39. Liu, L.; Zhang, Y.; Zhang, L.; Kermanshahi, M. RRT-CBF based motion planning. *arXiv* **2024**, arXiv:2410.00343. <https://doi.org/10.48550/arXiv.2410.00343>.

40. Ahmed, Z.; Halefom, M.H.; Woolsey, C. Tutorial review of indirect wind estimation methods using small uncrewed air vehicles. *J. Guid. Control Dyn.* **2024**, *47*, 1145–1162.
41. Kozlov, E.; Muslimov, T. Artificial potential field method for path and trajectory planning of unmanned aerial vehicles: A review. *SSRN Electron. J.* **2025**, 1–21. <https://doi.org/10.1142/s2301385026300027>.
42. Gao, W.; Li, L.; Pang, D. Urban low-altitude UAV path planning by fusing an enhanced A* algorithm with an adaptive artificial potential field method. *Sci. Rep.* **2026**, *16*, 17892. <https://doi.org/10.1038/s41598-026-45160-6>.
43. Hao, G.; Lv, Q.; Huang, Z.; Zhao, H.; Chen, W. UAV path planning based on improved artificial potential field method. *Aerospace* **2023**, *10*, 562. <https://doi.org/10.3390/aerospace10060562>.
44. Elahian, S.; Atashgah, M.A.; Tarvirdizadeh, B. Collaborative autonomous navigation of quadrotors in unknown outdoor environments: An active visual SLAM approach. *IEEE Access* **2024**, *12*, 147115–147128. <https://doi.org/10.1109/ACCESS.2024.3473792>.
45. Cao, J.; Chen, Z.; Kong, S.; Liu, L.; Wang, R. Towards urban wind utilization: The spatial characteristics of wind energy in urban areas. *J. Clean. Prod.* **2024**, *450*, 141981.
46. Yan, Y.; Li, X.; Yang, L.; Yan, P.; Tu, J. Flow field around two tandem non-identical-height square buildings via LES. *Build. Environ.* **2020**, *183*, 107206. <https://doi.org/10.1016/j.buildenv.2021.107985>.
47. Hsieh, C.-M.; Chen, L. Numerical evaluation of pedestrian-level wind comfort around “lift-up” buildings with various unconventional configurations. *Build. Environ.* **2021**, *188*, 107429. <https://doi.org/10.1016/j.buildenv.2020.107429>.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.