Leveraging Computational Fluid Dynamics in UAV Motion Planning

Huang, Yunshen; Greiff, Marcus; Vinod, Abraham P.; Di Cairano, Stefano

TR2024-050 May 09, 2024

Abstract

We propose a motion planner for quadrotor unmanned aerial vehicles (UAVs) in windy environments, where the motion is defined by a sequence of Bézier curves in the flat output space of the UAV. The real-time implementable planner incorporates wind information from high-fidelity computational fluid dynamics simulations performed offline and utilizes convexity properties of Bézier curves to enable real-time implementations. For this purpose, we: (i) identify a model for the UAV-wind interaction; (ii) use the OpenFoam software to compute a model of the wind speeds subject to world geometry and boundary conditions; (iii) describe a method for regressing this wind model into a more compact representation; and finally (iv) demonstrate how this representation is amenable to minimum-snap motion planning of quad-rotor UAVs in realistic environments. We validate our approach using simulations and hardware experiments, and show a significant improvement in the thrust used by the UAV in presence of strong winds.

American Control Conference (ACC) 2024

Mitsubishi Electric Research Laboratories, Inc.
201 Broadway, Cambridge, Massachusetts 02139

Leveraging Computational Fluid Dynamics in UAV Motion Planning

Yunshen Huang, Marcus Greiff, Abraham Vinod*, and Stefano Di Cairano

***Abstract*— We propose a motion planner for quadrotor unmanned aerial vehicles (UAVs) in windy environments, where the motion is defined by a sequence of Bézier curves in the flat output space of the UAV. The real-time implementable planner incorporates wind information from high-fidelity computational fluid dynamics simulations performed offline and utilizes convexity properties of Bézier curves to enable real-time implementations. For this purpose, we: (i) identify a model for the UAV-wind interaction; (ii) use the OpenFoam software to compute a model of the wind speeds subject to world geometry and boundary conditions; (iii) describe a method for regressing this wind model into a more compact representation; and finally (iv) demonstrate how this representation is amenable to minimum-snap motion planning of quad-rotor UAVs in realistic environments. We validate our approach using simulations and hardware experiments, and show a significant improvement in the thrust used by the UAV in presence of strong winds.**

I. Introduction

Emerging technologies such as the wind-LiDAR in [1] promise instantaneous estimates of spatial wind speed vector fields. It is therefore relevant to ask how to best leverage such information to aid UAV motion planning. This question comes with several challenges, and needs to be addressed both at the level of the motion planner and at the level of the underlying controllers. The proposed approach builds upon existing motion planning methods with minimum-snap objectives over motions defined as parametric curves, such as the polynomial planning in [2]–[5], and the related Bézier approaches in [6]–[8]. In these methods, quadratic programs (QPs) or mixed-integer QPs (MIQPs) are formulated with objectives in the total variation of the higher-order derivatives of the motion. By ensuring smoothness of the planned trajectories, such approaches guarantee motion feasibility with respect to the full nonlinear UAV dynamics [9]. However, in the context of wind disturbances, there are significant drawbacks with the aforementioned methods:

- The flexibility of the objective is limited, as snap (the fourth derivative of motion) is a geometric property that is not influenced by disturbances [2]–[4], [6]–[8].
- If considering wind, the corresponding flatness equations [2], [9] of UAVs are well defined only if the disturbance is sufficiently smooth in time.

Exogenous forces caused by wind or gravity contribute significantly to the total energy consumed by a UAV while flying [10] and should therefore be incorporated into the objective. However, additional care is required to formulate tractable optimization problems which retain the favorable computational properties of the minimum snap planning [3], [4], [6], [7]. To this end, we discuss how high-fidelity computational fluid dynamics (CFD) simulations, such as the flow depicted in Fig. 1, can be incorporated in the above mentioned approaches by simply augmenting the objective function. The approach is not limited to a specific planner, but can be used with the planning methods in [5]–[8].

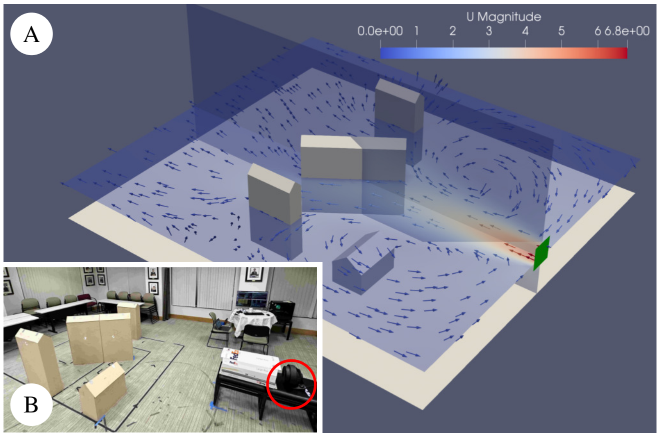

Fig. 1. (A) Two-dimensional slices of the CFD solution that is leveraged in the motion planner, here with a complex world geometry and wind boundary conditions (green). (B) Experiment setup in with a circulator fan (red circle).

A. Contributions

The contributions of this paper are threefold:

- We provide analytical expressions to aid the construction of a combined minimum snap and thrust cost as a function of the control points of time-scaled Bézier curves, and show the resulting objective to be bi-convex.
- We demonstrate how high-fidelity CFD models can be used to regress a disturbance model in an appropriate function class, specifically when the disturbance is wind acting on the translational motion of the UAV.
- We provide simulations and conduct experiments with a lightweight UAV (Crazyflie 2.1 [11]) to demonstrate the approach in an environment with complex geometries and wind fields emulating an urban landscape.

B. Outline

The mathematical preliminaries are given in Sec. II. A set of modeling assumptions are then introduced and challenged in Sec. III, describing how free-space is computed, the CFD simulator interpolation, the UAV-wind interaction and its

*Corresponding author. Email: abraham.p.vinod@ieee.org.

Y. Huang is with Washington University in St. Louis, St. Louis, MO 63130. This work was done during Huang's internship at MERL.

M. Greiff is with Toyota Research Institute, Los Altos, CA, USA, and was with Mitsubishi Electric Research Laboratories at the time of this research.

A. Vinod, and S. Di Cairano are with Mitsubishi Electric Research Labs (MERL), 02139 Cambridge, MA, USA.

experimental validation. This is followed by the definition of a the planning objectives in Sec. V. Supporting numerical and experimental results for UAV path planning are given in Sec. VI, and the conclusion in Sec. VII closes the paper.

II. Preliminaries

Vectors are written $\boldsymbol{x} \in \mathbb{R}^n$ with $[\boldsymbol{x}]_i$ denoting the i^{th} element of $\boldsymbol{x}$, and $\boldsymbol{e}_i$ denotes a unit vector where $[\boldsymbol{e}_i]_i = 1$. The identity matrix is denoted $\boldsymbol{I}_d \in \mathbb{R}^{d\times d}$ and the zero matrix is denoted $\boldsymbol{0}$. We let $\|\boldsymbol{x}\|_2 = \boldsymbol{x}^\top \boldsymbol{x}$, and take $(\mathbb{S}^n_+)\mathbb{S}^n_{++}$ to be the cone of $n \times n$ positive (semi)definite matrices. For any set of points, $\mathcal{P} = \{\boldsymbol{p}_i \in \mathbb{R}^d | i = 1, ..., n\}$, we let $\text{hull}(\mathcal{P})$ denote their convex hull, $\text{pnt}(\mathcal{P}) = \frac{1}{n}\sum_{i=1}^n \boldsymbol{p}_i \in \mathbb{R}^d$ be a point in its interior and $\text{vol}(\mathcal{P})$ denote its volume. A function $f : \mathcal{A} \mapsto \mathcal{B}$ is said to belong to $f \in \mathcal{C}^k(\mathcal{A}, \mathcal{B})$ if it is k-times continuously differentiable on $\mathcal{A}$. Let $\boldsymbol{S} : \mathbb{R}^3 \mapsto \mathbb{R}^{3\times 3}$ such that $\boldsymbol{S}(\boldsymbol{a})\boldsymbol{b} = \boldsymbol{a} \times \boldsymbol{b}$ is the cross product. Further, we let $\otimes$ denote the Kronecker product, and for smooth curves $\boldsymbol{B} \in \mathcal{C}^k([0, T], \mathbb{R}^d)$ we write their squared $\mathcal{L}_2$-norms

$$\|\boldsymbol{B}\|^2_{\mathcal{L}_2([0,T])} = \int_0^T \|\boldsymbol{B}(t)\|_2^2 \mathrm{d}t. \quad (1)$$

A. Bézier Curves

To represent the UAV motion and analyze the effects of wind on its thrust, we consider Bézier curves [12].

Definition 1 (Bézier curve) *A Bézier curve of degree n denoted $\boldsymbol{B} : [0, 1] \mapsto \mathbb{R}^d$ is defined by $n + 1$ control points $\mathcal{P} = \{\bar{\boldsymbol{p}}_i \in \mathbb{R}^d : d > 1, i \in [0, ..., n]\}$ as the interpolation*

$$\boldsymbol{B}(\lambda; \mathcal{P}) = \sum_{i=0}^{n} \binom{n}{i} (1-\lambda)^{(n-i)} \lambda^i \bar{\boldsymbol{p}}_i \triangleq \sum_{i=0}^{n} \alpha_i^n(\lambda)\bar{\boldsymbol{p}}_i. \quad (2)$$

As it is sometimes useful to consider $\mathcal{P}$ as a set of points, and other times as a vector, let $\boldsymbol{p} = \text{vec}(\mathcal{P}) \triangleq (\bar{\boldsymbol{p}}_0^\top, ..., \bar{\boldsymbol{p}}_n^\top)^\top$. The Bézier curve is a convex interpolation in its control points, resulting in the following properties (see, e.g., [12]).

Property 1 *Any Bézier curve is contained in the convex hull of its control points $\boldsymbol{B}(\lambda; \mathcal{P}) \in \text{hull}(\mathcal{P})$ for all $\lambda \in [0, 1]$.*

Property 2 *The kth derivative of a Bézier curve is a Bézier curve of order $n - k$, $(\mathrm{d}^k/\mathrm{d}\lambda^k)\boldsymbol{B}(\lambda; \mathcal{P}) = \boldsymbol{B}(\lambda; \mathcal{P}^{(k)})$, and the control points $\mathcal{P}$ and $\mathcal{P}^{(k)}$ are related by a linear map.*

Property 3 *The squared $\mathcal{L}_2$-norm of $\boldsymbol{B}(\lambda; \mathcal{P})$ is strictly convex in the control points $\mathcal{P}$ over its domain $\lambda \in [0, 1]$.*

B. Time-scaled Bézier Curves and Trajectories

In the following, we relate motions along these curves to signals such as wind speeds, which are defined in time. To this end, we consider a time-scaled Bézier curve as follows.

Definition 2 (Time-scaled Bézier curve) *For a Bézier curve $\boldsymbol{B}(\lambda; \mathcal{P})$, its time scaled equivalent is defined by a coordinate transform $\lambda = tT^{-1}$ with $t \in [0, T]$ for $T > 0$.*

With this time-scaling, Property 1 holds and Properties 2–3 can easily be modified to derivatives/integration in t. In the following, we consider a set of m segments $\{\mathcal{B}_i : [0, T_i] \mapsto \mathbb{R}^d\}_{i=1}^m$, where $\mathcal{B}_i(t) = \boldsymbol{B}(tT_i^{-1}; \mathcal{P}_i)$ for some $T_i > 0, d > 0$, and construct a Bézier trajectory $\mathcal{B} : [0, T] \mapsto \mathbb{R}^d$, as

$$\mathcal{B}(t) = \mathcal{B}_i(t - \tau_i) \quad \forall t \in [\tau_i, \tau_{i+1}], \quad \tau_i = \sum_{j=1}^{i-1} T_j, \quad (3)$$

where $T = \sum_{i=1}^m T_i$ and $T_0 = 0$. To simplify notation, we let $\boldsymbol{p}_i = \text{vec}(\mathcal{P}_i) \in \mathbb{R}^{d(n+1)}$ be the control points associated with the ith curve, and define all of the control points associated with $\mathcal{B}$ in (3) as $\boldsymbol{p} = (\boldsymbol{p}_1^\top, \dots, \boldsymbol{p}_m^\top)^\top \in \mathbb{R}^{md(n+1)}$ and collect time intervals in $\boldsymbol{t} = (T_1, \dots, T_m) \in \mathbb{R}^m_+$.

C. UAV Dynamics and Properties

The UAV is configured in $(\boldsymbol{r}, \boldsymbol{R}) \in \mathbb{R}^3 \times \mathsf{SO}(3)$ with $\mathsf{SO}(3) = \{\boldsymbol{R} \in \mathbb{R}^{3\times 3} : \boldsymbol{R}^\top \boldsymbol{R} = \boldsymbol{I}, \det(\boldsymbol{R}) = 1\}$. Here, $\{G\}$ is a global frame with a basis $\{\boldsymbol{e}_i\}_{i=1}^3$, and $\{O\}$ denotes the body frame with a basis $\{\boldsymbol{e}_i^O\}_{i=1}^3$, where $[\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3] = \boldsymbol{R}^\top[\boldsymbol{e}_1^O, \boldsymbol{e}_2^O, \boldsymbol{e}_3^O]$. From first principles [13, Chapter 2.3.5],

$$\dot{\boldsymbol{r}} = \boldsymbol{v} \quad (4a)$$

$$m\dot{\boldsymbol{v}} = f\boldsymbol{R}\boldsymbol{e}_3 - mg\boldsymbol{e}_3 \quad (4b)$$

$$\dot{\boldsymbol{R}} = \boldsymbol{R}\boldsymbol{S}(\boldsymbol{\omega}) \quad (4c)$$

$$\boldsymbol{J}\dot{\boldsymbol{\omega}} = \boldsymbol{S}(\boldsymbol{J}\boldsymbol{\omega})\boldsymbol{\omega} + \boldsymbol{\tau}, \quad (4d)$$

where $\boldsymbol{r} \in \mathbb{R}^3$ [m] defines the position of the UAV in the global frame $\{G\}$; $\boldsymbol{v} \in \mathbb{R}^3$ [m/s] defines the velocity of the UAV in $\{G\}$; $\boldsymbol{R} \in \mathsf{SO}(3)$ defines the attitude of the UAV; $\boldsymbol{\omega} \in \mathbb{R}^3$ [rad/s] denotes the usual attitude rates defined in $\{O\}$; $f > 0$ [N] defines the thrust generated by the rotors; $\boldsymbol{\tau}$ [N·m] denotes the torques defined in $\{O\}$. The model is parameterized by an inertia matrix $\boldsymbol{J} \in \mathbb{S}^3_{++}$ [kg·m^2], a mass $m > 0$ [kg], and the gravitational acceleration $g > 0$ [m/s^2]. It is well known that the UAV dynamics are differentially flat [2], [9]. For the purpose of modeling Newtonian wind interactions, we note that (4) remains differentially flat if there is a known and sufficiently smooth, additive disturbance on (4b) by following the proofs in [2]. The maps from flat outputs to states and control signals can be found in [2], [9].

III. Wind Modeling

At its most granular level, we model the wind as governed by the incompressible Naviér-Stokes equations, with kinematic pressure $\boldsymbol{\rho} \in \mathbb{R}^3$ [m^2/s^2] and wind velocity $\boldsymbol{u} \in \mathbb{R}^3$ [m/s], evolving by the partial differential equations (PDEs)

$$\nabla_{\boldsymbol{x}} \cdot \boldsymbol{u} = \boldsymbol{0} \quad (5a)$$

$$(\partial/\partial t)\boldsymbol{u} + \nabla_{\boldsymbol{x}}(\boldsymbol{u}\boldsymbol{u}^\top) - \nabla_{\boldsymbol{x}} \cdot \boldsymbol{R}_u = -\nabla_{\boldsymbol{x}}\boldsymbol{\rho} + \boldsymbol{S}_u, \quad (5b)$$

where $\boldsymbol{R}_u$ is a stress tensor and $\boldsymbol{S}_u$ is a momentum source (see, e.g., [14]). This PDE is considered over a compact set $\Omega \subset \mathbb{R}^3$ with boundary $\text{bnd}(\Omega) = \bigcup_i B_i$ partitioned into an inlet surface B_1, a ground surface B_2, an outlet surface B_3, and a boundary surface B_4. Specifically, we consider

$$\boldsymbol{u}(\boldsymbol{x}, t) = \boldsymbol{u}_0(\boldsymbol{x}), \quad \forall \boldsymbol{x} \in B_1 \quad \text{(inlet)} \quad (6a)$$

$$\boldsymbol{u}(\boldsymbol{x}, t) = \boldsymbol{0}, \quad \forall \boldsymbol{x} \in B_2 \quad \text{(ground)} \quad (6b)$$

$$\nabla_{\boldsymbol{x}}\boldsymbol{u}(\boldsymbol{x}, t) = \boldsymbol{0}, \quad \forall \boldsymbol{x} \in B_3 \quad \text{(zero grad)} \quad (6c)$$

$$\nabla_{\boldsymbol{x}}\boldsymbol{u}(\boldsymbol{x}, t) \cdot \boldsymbol{n}(\boldsymbol{x}) = \boldsymbol{0}, \quad \forall \boldsymbol{x} \in B_4 \quad \text{(slip)} \quad (6d)$$

where $\boldsymbol{n}(\boldsymbol{x})$ denotes the normal of the surface $\boldsymbol{x} \in \mathrm{bnd}(\Omega)$ pointing into the interior of Ω. The inlet flow is uniform

$$\boldsymbol{u}_0(\boldsymbol{x}) = V(\cos(\psi), \sin(\psi), 0) \;\; \forall \boldsymbol{x} \in B_1, \tag{7}$$

such that we can characterize the wind by means of an inlet velocity and a wind direction. The boundary conditions (BCs) are determined by $\boldsymbol{\theta} = (V, \psi, B_1, B_2, B_3, B_4)$. In the experiments, the wind is created by a fan in a small wind tunnel (see Fig. 1), and Ω is taken to be a box with $\mathrm{vol}(\Omega) \approx 47m^3$. In this setting, the inlet is a small face $B_1 = \{(x, y, z) \in \Omega | x = -1.2, -1.35 \leq y \leq -1.05, 0.42 \leq z \leq 0.7\}$; the bottom plane and buildings are ascribed the ground BC in (6b); the walls are ascribed the zero gradient BC in (6c); and the top is given a slip BC in (6d).

To solve (5), we employ the `OpenFoam` software [14] with the `SIMPLE` algorithm [15] and a finite element (FE) discretization of Ω to approximately solve (5) over a mesh by given a set of boundary conditions and general non-convex obstacle geometries, $\mathcal{G}$, represented as `.stl`-files. The stationary velocity field solution of (5) computed for a specific $\boldsymbol{\theta}_i$ is denoted $\boldsymbol{u}_i(\boldsymbol{x})$, stored as a large number of three-dimensional vectors associated with each element in the mesh. An example of a geometry $\mathcal{G}$ and solution computed for $(V, \psi) = (2, 0)$ is shown in Fig. 1.

A. Look-up Tables

As solving (5) takes a significant amount of time, this is done offline. Having done so, we get a set of N stationary solutions to (5) corresponding to the parameters $\{\boldsymbol{\theta}_i\}_{i=1}^N$. Note that we can also compute a set of transient solutions $\boldsymbol{\theta}_i$ with the same boundary conditions. Yet this paper only considers stationary solutions with the intention of finding the $\boldsymbol{u}_i(\boldsymbol{x})$ which most closely matches the real-time estimates of methods such as [1] and approximating the future wind flow with the corresponding stationary solution to (5).

IV. Wind Representation and Interaction

In keeping with the literature on UAV planning over Bézier curves [6], [7], we leverage Property 1 and consider the trajectory in (3) as being confined to sets in space over intervals in time. These sets are referred to as *free-space sets* $\mathcal{F} = \bigcup_i \mathcal{F}_i$, where $\mathcal{F}_i$ is a polyhedral set. The wind speed representation and regression are discussed in Sec. IV-A, before modeling the UAV-wind interaction in Sec. IV-B.

A. Wind Regression

To facilitate real-time motion planning, we now discuss a convexity-preserving, regression-based strategy to incorporate the wind information from the CFD solutions into the motion planner. It is therefore interesting to consider the wind speeds along trajectories in Ω. Here, we do this by means of regression. Specifically, we assume that there exists a motion trajectory $\mathcal{B}(t)$ that passes through a connected sequence of m free-space sets characterized by the parameters $(\boldsymbol{p}, \boldsymbol{t})$. We then define a wind trajectory $\mathcal{W}(t)$ as a yet another Bézier trajectory of the same degree, with parameters $(\boldsymbol{w}, \boldsymbol{t})$. Finally, we define a loss function in terms of the total variation distance

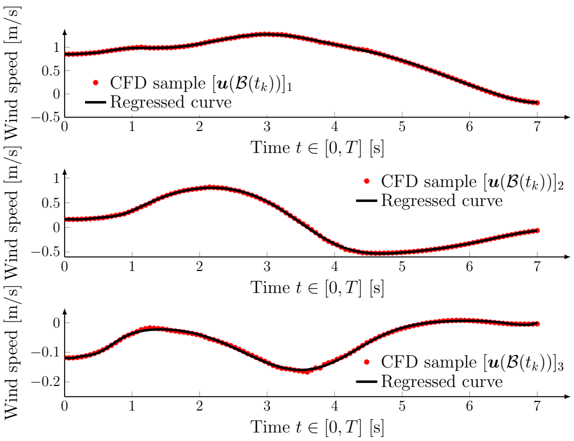

Fig. 2. The wind speeds (red) interpolated along a motion trajectory in space from CFD solution depicted in Fig. 1, and a regression of a Bézier curve to this data, computed by solving (10) for $\boldsymbol{w}$.

$$J_R(\boldsymbol{w}; \boldsymbol{p}, \boldsymbol{t}) = \|\mathcal{W}(t) - \boldsymbol{u}(\mathcal{B}(t))\|^2_{\mathcal{L}_2([0,T])}. \tag{8}$$

where $\boldsymbol{u}$ is a CFD solution to the wind speeds. This objective is not generally convex, but can be rewritten by discretizing time as $\{t_k = hk\}_{k=0}^K$ such that $t_0 = 0$ and $t_K = T$. Each point $\boldsymbol{u}(\mathcal{B}(t_k))$ along the trajectory can then be associated with a spline index j and a distance along that curve, as

$$\mathcal{D}_j = \{(\lambda_k, \boldsymbol{u}(\mathcal{B}(t_k)) \mid \lambda_k = (t_k - \tau_j)T_j^{-1} \in [0, 1]\}. \tag{9}$$

By Riemann-integration, the loss in (8) can be approximated

$$\begin{aligned} J_R(\boldsymbol{w}; \boldsymbol{p}, \boldsymbol{t}) &\approx \frac{1}{h} \sum_{j=1}^{m} \sum_{(\bar{\lambda}_k, \bar{\boldsymbol{u}}_k) \in \mathcal{D}_j} \|(\boldsymbol{\alpha}(\bar{\lambda}_k) \otimes \boldsymbol{I}_d)\boldsymbol{w}_j - \bar{\boldsymbol{u}}_k\|_2^2 \\ &\propto \|\boldsymbol{A}\boldsymbol{w} - \boldsymbol{D}\|_2^2, \end{aligned} \tag{10}$$

where $\boldsymbol{D}^\top = (\boldsymbol{d}_1^\top, ..., \boldsymbol{d}_m^\top)$, $\boldsymbol{A} = \mathrm{diag}(\boldsymbol{A}_1, \cdots, \boldsymbol{A}_m)$, and

$$\boldsymbol{A}_j = \begin{bmatrix} \vdots & & \vdots \\ \alpha_0^n(\bar{\lambda}_k) & \cdots & \alpha_n^n(\bar{\lambda}_k) \\ \vdots & & \vdots \end{bmatrix} \otimes \boldsymbol{I}_d, \;\; \boldsymbol{d}_j = \begin{bmatrix} \vdots \\ \bar{\boldsymbol{u}}_k \\ \vdots \end{bmatrix}, \tag{11}$$

for all $(\bar{\lambda}_k, \bar{\boldsymbol{u}}_k) \in \mathcal{D}_j$. This problem can be solved over each curve independently and the problem is well posed if and only if K is chosen such that $\min_j |\mathcal{D}_j| \geq n + 1$, implying that $\boldsymbol{A}^\top \boldsymbol{A}$ is of full rank. Here, we need to formulate it across all of the curves (10) to enforce equality constraints in the regression which ensure $\mathcal{W} \in \mathcal{C}^k([0, T], \mathbb{R}^3)$. This is required to expand $\mathcal{B}$ and $\mathcal{W}$ into the flat output space of the UAV. The formulation of such constraints will be discussed in Sec. V-B, and a solution of $\mathcal{W}(t)$ for a given $\mathcal{B}(t)$ computed with respect to the solution in Fig. 1 is depicted in Fig. 2.

B. Wind Interaction

When considering UAV-wind interaction, it is relatively common to model wind as a forcing term (viscous friction)

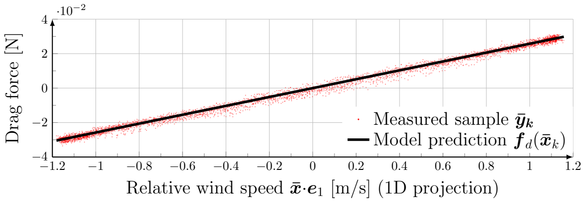

Fig. 3. Sampled data and the regressed linear model in the $\boldsymbol{e}_1$-direction.

experienced by an object moving through a fully enclosing fluid [16, Chap. 4.5]. This results in a quadratic force,

$$f_d(\delta) = \frac{1}{2}\rho c_d A \delta^2 \in \mathbb{R}, \tag{12}$$

where δ [m/s] is the velocity of the object relative to the fluid, ρ [kg/m^3] is the fluid mass density, A [m^2] is the reference area of the object, and c_d [$\cdot$] is the drag coefficient. In the following, we consider the translational dynamics of the UAV (4b), and model the forcing term as

$$m\dot{\boldsymbol{v}} \;= f\boldsymbol{R}\boldsymbol{e}_3 - mg\boldsymbol{e}_3 + \boldsymbol{f}_d(\boldsymbol{u}(\boldsymbol{r}) - \boldsymbol{v}), \tag{13}$$

where $\boldsymbol{u}(\boldsymbol{r})$ is the wind speed along the UAV motion in time. Instead of directly deriving the forcing term by measuring parameters in (12), we take a data-driven approach and define

$$\bar{\boldsymbol{y}} = m\dot{\boldsymbol{v}} - f\boldsymbol{R}\boldsymbol{e}_3 + mg\boldsymbol{e}_3 \in \mathbb{R}^3 \tag{14a}$$

$$\bar{\boldsymbol{x}} = \boldsymbol{u}(\boldsymbol{r}) - \boldsymbol{v} \in \mathbb{R}^3. \tag{14b}$$

These signals are sampled over multiple experiment trajectories, and stored as K input/output pairs $\mathcal{D} = \{(\bar{\boldsymbol{x}}_k, \bar{\boldsymbol{y}}_k)\}_{k=1}^K$. An approximate model of the forcing term is defined as

$$\boldsymbol{f}_d(\bar{\boldsymbol{x}};\boldsymbol{\alpha}) = \sum_{l=1}^{L} \boldsymbol{\phi}_l(\bar{\boldsymbol{x}})\boldsymbol{\alpha}_l, \quad \boldsymbol{\alpha} = (\boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_L) \in \mathbb{R}^{3L}, \tag{15}$$

with L differentiable kernels $\phi_l : \mathbb{R}^3 \mapsto \mathbb{R}^{3\times 3}$, before solving

$$\mathrm{argmin}_{\boldsymbol{\alpha}} \|\bar{\boldsymbol{y}}_k - \boldsymbol{f}_d(\bar{\boldsymbol{x}}_k;\boldsymbol{\alpha})\|_2^2. \tag{16}$$

The forcing term (13) only depends on the relative wind speed, and it is therefore easiest computed in the absence of wind. After forming $\mathcal{D}$ from several trajectories with $\boldsymbol{u}(\boldsymbol{r}) \equiv \boldsymbol{0}$ using a Crazyflie 2.1 [11], we obtain a good fit with a linear model $\boldsymbol{\phi}_1(\bar{\boldsymbol{x}}) = \mathrm{diag}(\bar{\boldsymbol{x}})$, $\boldsymbol{\alpha}_1 = 10^{-2}(3.3, 2.7, 2.2)$ and $L = 1$ (see Fig. 3). Additional terms result in models that are rejected by standard Bayesian information criteria [17].

The simple Newtonian UAV-wind interaction in (13) is approximate, especially for lightweight UAVs such as the Crazyflie, which are notoriously sensitive to turbulence effects [10]. Furthermore, the lift and drag forces of the rotors are non-trivially affected by Bernoulli effects on the rotor blades [18]. Some of these phenomena may be captured by regressing a wind model using real-time wind measurements. However, this is beyond the scope of the present paper.

V. Planning with Wind over Bézier Curves

A. Objective function

Equipped with the CFD wind information, we next formulate a planing objective including the snap of the trajectory, J_S, the thrust consumption due to wind, J_W, and the total time of the maneuver. The objective function is defined as

$$J(\boldsymbol{p},\boldsymbol{t};\boldsymbol{w}) = k_S J_S(\boldsymbol{p},\boldsymbol{t}) + k_W J_W(\boldsymbol{p},\boldsymbol{t};\boldsymbol{w}) + k_T T, \tag{17}$$

for some weights $k_S, k_W, k_T > 0$. Notably, when fixing $\boldsymbol{w}$, (17) can be shown to be bi-convex in $(\boldsymbol{p},\boldsymbol{t})$ if $T_j > 0$, $\forall j$, and its gradient and hessian in $(\boldsymbol{p},\boldsymbol{t})$ are are known in closed form. By setting $k_W = 0$ we can get a minimum snap objective, and by adjusting the ratio k_W/k_S we can get solutions that leverage the wind to a varying extent. The components of the cost function is defined in the following.

1) Minimum Snap: For path planning in parametric curves, it is common to define a cost that penalizes the total variation of higher-order derivatives of the motion plan [3]. In the context of time-scaled Bézier curves, this objective can be expressed as

$$J_S(\boldsymbol{p},\boldsymbol{t}) = \sum_{k=0}^{n} c_k \|(\mathrm{d}^k/\mathrm{d}t^k)\mathcal{B}(t)\|^2_{\mathcal{L}_2([0,T])} = \sum_{j=1}^{m} \boldsymbol{p}_j^\top \boldsymbol{Q}_S(T_j)\boldsymbol{p}_j \tag{18}$$

with weights $c_k > 0$ for all $k = 0, ..., n$. This is notably a strictly bi-convex function in $\boldsymbol{p}$ and $\boldsymbol{t}$, on the domain $(\boldsymbol{p},\boldsymbol{t}) \in \mathbb{R}^{dm(n+1)} \times \mathbb{R}^m_{>0}$. While this minimum-snap-like cost function captures some signals of interest, such as translational kinetic energy, it is agnostic to any external disturbances acting on the UAV. As such, we express the squared $\mathcal{L}_2$-norm of the UAV thrust as an objective.

2) Minimum Thrust: To express a cost in the thrust of the UAV given (13), we start by noting that

$$\|f\|_2^2 = \|m\dot{\boldsymbol{v}} + mg\boldsymbol{e}_3 - \boldsymbol{K}(\boldsymbol{u}(\boldsymbol{r}) - \boldsymbol{v})\|_2^2, \tag{19}$$

for a matrix $\boldsymbol{K} = (\partial/\partial\bar{\boldsymbol{x}})\boldsymbol{f}_d(\bar{\boldsymbol{x}}) = \mathrm{diag}(\boldsymbol{\alpha}_1)$, given that (15) is linear in $\bar{\boldsymbol{x}}$. Consequently, the $\mathcal{L}_2$-norm of the thrust can be written in terms of the trajectories $\mathcal{B}$ and $\mathcal{W}$ as

$$\begin{aligned} J_W(\boldsymbol{p},\boldsymbol{t};\boldsymbol{w}) &= \|f\|^2_{\mathcal{L}_2([0,T])} \\ &= \|m\tfrac{\mathrm{d}^2}{\mathrm{d}t^2}\mathcal{B}(t) + \boldsymbol{K}\tfrac{\mathrm{d}}{\mathrm{d}t}\mathcal{B}(t) + mg\boldsymbol{e}_3 - \boldsymbol{K}\mathcal{W}(t)\|^2_{\mathcal{L}_2([0,T])} \end{aligned} \tag{20}$$

Here, we can utilize the fact that a sum of two Bézier curves of degree n is another Bézier curve of degree n, with an additive relationship between the control points. As such, we can define a new curve $\mathcal{V}$ with points

$$\boldsymbol{v} = \boldsymbol{1}_{(n+1)m} \otimes mg\boldsymbol{e}_3 - (\boldsymbol{I}_{(n+1)m} \otimes \boldsymbol{K})\boldsymbol{w}, \tag{21}$$

whereby (20) can be written as a quadratic in $\boldsymbol{p}$,

$$\begin{aligned} J_W(\boldsymbol{p},\boldsymbol{t};\boldsymbol{w}) &= \|m\tfrac{\mathrm{d}^2}{\mathrm{d}t^2}\mathcal{B} + \boldsymbol{K}\tfrac{\mathrm{d}}{\mathrm{d}t}\mathcal{B} + \mathcal{V}\|^2_{\mathcal{L}([0,T])} \\ &= \sum_{j=1}^{m} (\boldsymbol{p}_j + \boldsymbol{v}_j)^\top \boldsymbol{Q}_W(T_j)(\boldsymbol{p}_j + \boldsymbol{v}_j). \end{aligned} \tag{22}$$

Similar to (18), for fixed $\boldsymbol{w}$, (22) is bi-convex in $(\boldsymbol{p},\boldsymbol{t})$.

B. Constraints

Next, we introduce associated constraints. Here we consider safety constraints (SCs), continuity constraints (CCs) across splines, and generalized waypoint constraints (WPs).

1) Safety Constraints: To ensure safety, we confine the segments of the trajectory in (3) to free space, leveraging Property 1. Specifically, for a connected route $\mathcal{R} = \{r_1, ..., r_m\}$ passing through a convex set $\mathcal{F}_{r_j} = \{\boldsymbol{x} \in \mathbb{R}^3 | \boldsymbol{A}_{r_j}\boldsymbol{x} \le \boldsymbol{b}_{r_j}\}$, we constrain the jth segment of the motion

$$(\boldsymbol{A}_{r_j} \otimes \boldsymbol{I}_{n+1})\boldsymbol{p}_j \le \boldsymbol{b}_{r_j} \otimes \mathbf{1}_{n+1}. \tag{23}$$

In the context of the curve $\mathcal{B}(t)$ in (3), enforcing (23) across all $j \in [1, m]$ implies that

$$\mathcal{B}(t) \in \bigcup_{j=1}^{m} \mathcal{F}_{r_j}, \qquad \forall t \in [0, T], \tag{24}$$

without invoking the times $\boldsymbol{t}$ in the inequality constraints.

2) Continuity Constraints: To guarantee that the planned motion is dynamically feasible, we need to ensure that the trajectory is sufficiently smooth across the end-points of the curve. Given Beziér trajectories $\mathcal{B}(t)$ of degree n,

$$(\mathrm{d}^k/\mathrm{d}t^k)\mathcal{B}_j(T_j) = (\mathrm{d}^k/\mathrm{d}t^k)\mathcal{B}_{j+1}(0) \tag{25}$$

as an equality constraint

$$\begin{bmatrix}\boldsymbol{M}_n^k(T_j) & -\boldsymbol{M}_n^k(T_{j+1})\end{bmatrix}\begin{bmatrix}\boldsymbol{p}_j \\ \boldsymbol{p}_{j+1}\end{bmatrix} = \mathbf{0}. \tag{26}$$

Enforcing the constraint (26) across the $j \in [1, m-1]$ connecting points for every derivative up until and including k ensures that the planned trajectory $\mathcal{B} \in \mathcal{C}^k([0, T], \mathbb{R}^3)$.

3) Generalized Waypoint Constraints: Finally, the end-points of the curve $\mathcal{B}(t)$ in (3) needs to be constrained. Here, we define feasible to a initial and terminal convex sets

$$\mathcal{B}(0) \in \mathcal{X}_\circ = \{\boldsymbol{x} \in \mathbb{R}^d | \boldsymbol{A}_\circ^I\boldsymbol{x} \le \boldsymbol{b}_\circ^I, \boldsymbol{A}_\circ^E\boldsymbol{x} = \boldsymbol{b}_\circ^E\} \subseteq \mathcal{F}_{r_1}$$
$$\mathcal{B}(T) \in \mathcal{X}_T = \{\boldsymbol{x} \in \mathbb{R}^d | \boldsymbol{A}_T^I\boldsymbol{x} \le \boldsymbol{b}_T^I, \boldsymbol{A}_T^E\boldsymbol{x} = \boldsymbol{b}_T^E\} \subseteq \mathcal{F}_{r_m}$$

which can be formulated in terms of $\boldsymbol{p}$, as

$$\begin{bmatrix}\boldsymbol{A}_\circ^I & \mathbf{0}_d \\ \mathbf{0}_d & \boldsymbol{A}_T^I\end{bmatrix}\boldsymbol{N}(n,d)\begin{bmatrix}\boldsymbol{p}_1 \\ \boldsymbol{p}_m\end{bmatrix} \le \begin{bmatrix}\boldsymbol{b}_\circ^I \\ \boldsymbol{b}_T^I\end{bmatrix} \tag{27a}$$

$$\begin{bmatrix}\boldsymbol{A}_\circ^E & \mathbf{0}_d \\ \mathbf{0}_d & \boldsymbol{A}_T^E\end{bmatrix}\boldsymbol{N}(n,d)\begin{bmatrix}\boldsymbol{p}_1 \\ \boldsymbol{p}_m\end{bmatrix} = \begin{bmatrix}\boldsymbol{b}_\circ^E \\ \boldsymbol{b}_T^E\end{bmatrix} \tag{27b}$$

where

$$\boldsymbol{N}(n,d) = \begin{bmatrix}\boldsymbol{I}_d & \mathbf{0}_{d\times 2dn} & \mathbf{0}_d \\ \mathbf{0}_d & \mathbf{0}_{d\times 2dn} & \boldsymbol{I}_d\end{bmatrix} \tag{28}$$

As such, we can constrain the initial position of the UAV to a point, a line, a plane or a region in space, and we can also to apply such constraints at any time $t \in [0, T]$ by modifying the map corresponding to $\boldsymbol{N}(n, d)$.

Remark 1 *For UAVs, it is sometimes desirable to start from a stable hovering configuration. Here, it is worth noting that* $\boldsymbol{v}(0) = \dot{\boldsymbol{v}}(0) = \boldsymbol{\omega}(0) = \dot{\boldsymbol{\omega}}(0) = \boldsymbol{\tau}(0) = \mathbf{0}$ *is ensured by*

$$\begin{bmatrix}\boldsymbol{I}_d & \mathbf{0}\end{bmatrix}\boldsymbol{M}_n^k(T_1)\boldsymbol{p}_1 = \mathbf{0} \quad \forall k > 0, \tag{29}$$

when picking a sufficiently large $k \le n$.

C. Complete Planning Algorithm

Given the cost in (17), and the constraints in (23), (26), and (27), for a specified route $\mathcal{R}$, we can formulate and solve

$$\begin{aligned} \text{Minimize } J(\boldsymbol{p}, \boldsymbol{t}) \quad & (30a)\\ \boldsymbol{A}\boldsymbol{p} \le \boldsymbol{b} \quad & (30b)\\ \boldsymbol{C}(\boldsymbol{t})\boldsymbol{p} = \mathbf{0} \quad & (30c)\\ \boldsymbol{t} \ge \mathbf{0}. \quad & (30d)\end{aligned}$$

As the objective function is bi-convex in $\boldsymbol{p}$ and $\boldsymbol{t}$ given $\boldsymbol{w}$, we propose a block coordinate descent (BCD) approach in which we start from a feasible point $(\boldsymbol{p}^{(0)}, \boldsymbol{t}^{(0)})$ and solve (30) with fixed $\boldsymbol{t} = \boldsymbol{t}^{(0)}$ as a quadratic program (QP). We then fix $\boldsymbol{p} = \boldsymbol{p}^{(1)}$ and minimize the objective (30a) using a Newton method (NM), leveraging the fact that we can express the gradient and Hessian of the cost functions analytically.

Algorithm 1 Wind planning algorithm (SQP version).

```
Receive: Geometry $\Omega$, $r$, $\mathcal{G}$, and BCs $\{\boldsymbol{\theta}_l\}_{l=1}^L$
// Compute CFD solutions from geometry
Solve (5) given $\{\boldsymbol{\theta}_l\}_{l=1}^L$ for a look-up table $\{\boldsymbol{u}_l(\boldsymbol{x})\}_{l=1}^L$.
// Compute free space sets
Compute $\mathcal{F}$ given $\{\boldsymbol{u}_l(\boldsymbol{x})\}_{l=1}^L$
// Compute UAV-wind interaction model
Identify $\boldsymbol{f}_d$ by solving (16) given $\{\boldsymbol{u}_l(\boldsymbol{x})\}_{l=1}^L$.
// Online re-planning (when queried)
Receive: Route $\mathcal{R}$, and initial/terminal sets $\mathcal{X}_0, \mathcal{X}_T$
Select specific CFD solution $\boldsymbol{u}_l(\boldsymbol{x})$
Define $\boldsymbol{X}^{(0)} = (\boldsymbol{p}^{(0)}, \boldsymbol{t}^0)$ as in (17) given $\mathcal{R}$
// Solve optimization problem
while $\|\boldsymbol{X}^{(i)} - \boldsymbol{X}^{(i-1)}\|_\infty > \epsilon$ do
    Compute $\boldsymbol{w}^{(i)}$ by minimizing $J_R$ in (10) given $\boldsymbol{X}^{(i)}$
    Solve (30) for $\boldsymbol{p}^{(i+1)}$ given $\boldsymbol{t}^{(i)}$ and $\boldsymbol{w}^{(i)}$
    Update $\boldsymbol{t}^{(i+1)}$ by a NM given $\boldsymbol{p}^{(i+1)}$ and $\boldsymbol{w}^{(i)}$
end while
```

VI. Numerical Examples

To illustrate the benefits of complementing the minimum-snap motion planning with a cost expressed in the form of (20), we give an example where a trajectory is to be planned in the positional three-dimensional ($d = 3$) subspace of the flat output space of the UAV. This is done with the geometry and CFD solution depicted in Fig. 1. For illustration purposes and to get a fair comparison, we fix the time intervals such that the motions are defined over an equal total time interval, with $T = 7.5$s and $T_j = T/m$. Furthermore, we chose the relative weighting of derivatives in (18) as $c_4 = 1$ with $c_k = 10^{-3}$ for all $k \neq 4$, producing a slightly regularized minimum snap trajectory. The free space sets are computed using a world geometry that is inflated along its normal directions by $r = 0.1$m. The degree of each Bézier segment is set to $n = 7$, and each curve is confined to a free-space set using the safety constraints in (23). Continuity constraints are enforced up to the fourth

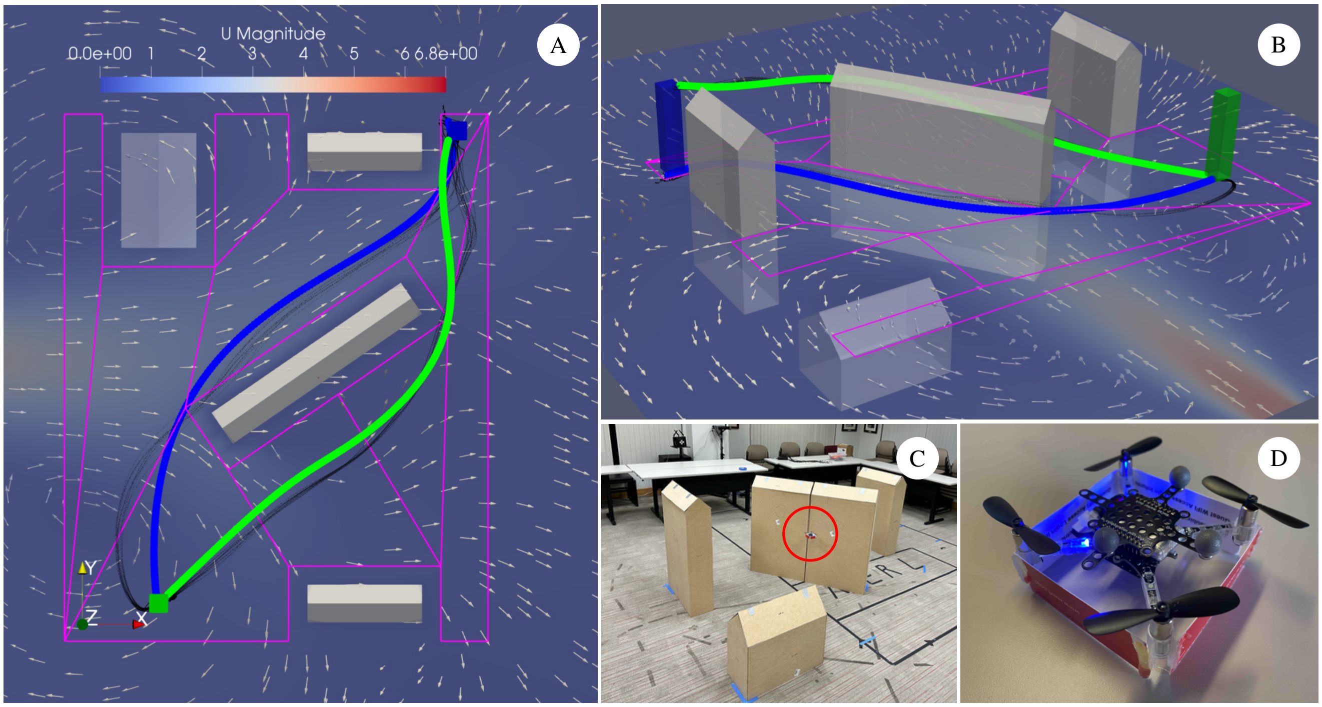

Fig. 4. Two views of the obstacle geometry (gray), the initial set $\mathcal{X}_0$ (blue), and the terminal set $\mathcal{X}_T$ (green), with a section of the CFD solution and projection of the free-space sets (magenta) at $z = 0.5$m. Optimal solutions with a minimum-snap cost (blue) and minimum-thrust cost (green) are shown, respectively. Faint black trajectories in (A) and (B) are plotted based on flight logs recorded in the lab with real wind (C) using the Crazyflie (D).

derivative as per (26). Generalized end-point constraints are enforced as in (27) and (29), with initial and terminal sets

$$\mathcal{X}_0 = \{\boldsymbol{x} \in \mathbb{R}^3 | \boldsymbol{x}_0 - \boldsymbol{\delta} \leq \boldsymbol{x} \leq \boldsymbol{x}_0 + \boldsymbol{\delta}\}, \tag{31a}$$

$$\mathcal{X}_T = \{\boldsymbol{x} \in \mathbb{R}^3 | \boldsymbol{x}_T - \boldsymbol{\delta} \leq \boldsymbol{x} \leq \boldsymbol{x}_T + \boldsymbol{\delta}\}, \tag{31b}$$

$$\boldsymbol{x}_0 = (2.00, -0.65, 0.75)^\top, \tag{31c}$$

$$\boldsymbol{x}_T = (0.25, -2.50, 0.75)^\top, \tag{31d}$$

$$\boldsymbol{\delta} = (0.05, 0.05, 0.25)^\top. \tag{31e}$$

In one planner, the objective in (17) is (i) defined with $k_W k_S^{-1} = 10^{-4}$, amounting to a minimum snap-like solution. In the other planner, (ii) the thrust cost is set to dominate the total objective, with $k_W k_S^{-1} = 10^4$. When considering motions for the possible routes in $\mathcal{R}$, we obtain two very different solutions shown in Fig. 4, with optimal costs:

$$\text{(i)} \quad J_S(\boldsymbol{p}^\star, \boldsymbol{t}) = 4.310, \quad \text{and} \quad J_W(\boldsymbol{p}^\star, \boldsymbol{t}) = 0.75 \tag{32a}$$

$$\text{(ii)} \quad J_S(\boldsymbol{p}^\star, \boldsymbol{t}) = 39.33, \quad \text{and} \quad J_W(\boldsymbol{p}^\star, \boldsymbol{t}) = 0.73 \tag{32b}$$

Including knowledge of the wind results in a small reduction of UAV thrust at the expense of making the trajectory more volatile (increasing the snap cost). When studying Fig. 4, it is clear that the planner achieves this by selecting the corridor with lesser wind, which is antagonistic in this example. A slight detour is taken in case (ii) in order to minimize wind exposure. The thrust cost difference is quite small in the case of the Crazyflie, due to the small impact of Newtonian drag and the low operating speeds. However, for larger UAVs operating at higher speeds with a greater influence of Newtonian drag, the performance gains will be much greater (as illustrated in Section VI-A).

When inflating the trajectories point-wise with spheres of radius 0.1m, the obstacle inflation and confinement of the trajectory to the free-space sets yield collision-free trajectories with the control points residing on the boundary of the safety sets. The resulting motions in (i) and (ii) are not conservative given the constraints imposed in Section V-B.

A. *Performance Improvement as a Function of Drag*

The considered objective only takes the Newtonian effects of wind into account. As such, it is relevant to explore how varying levels of drag affect the planner performance when introducing a minimum-thrust cost. To study this, we consider the same initial and terminal sets as in (31), and study two cases. In the first, we confine the trajectory to the corridor with the more wind, and compute the difference between thrust cost of solutions with (i) dominating minimum snap costs ($k_W k_S^{-1} = 10^{-4}$) and (ii) dominating minimum thrust costs ($k_W k_S^{-1} = 10^4$), respectively. We do this for a large number of drag models $\boldsymbol{K} = \beta \boldsymbol{K}^{\text{nominal}}$ for $\beta \in [0, 10]$, where $\boldsymbol{K}^{\text{nominal}}$ is the model regressed in Section IV-B, and compute the ratio

$$R(\beta) = \frac{J_W(\boldsymbol{p}^\star, \boldsymbol{t})|_{k_W k_S^{-1} = 10^{-4}}}{J_W(\boldsymbol{p}^\star, \boldsymbol{t})|_{k_W k_S^{-1} = 10^{4}}}. \tag{33}$$

This is a measure of thrust usage of (i) vs (ii) when varying the influence of drag. As expected, we note a clear performance improvement with increasing drag and permitting the planner to select either route (see Fig. 5, black) and this performance increases with the magnitude of the drag coefficient. If confining the trajectory to the same exposed path in both (i) and (ii) we see a lesser performance improvement, but even here, we note an improvement in the used thrust of the UAV of upwards of 15% just by accounting for how the wind varies thought the more exposed corridor.

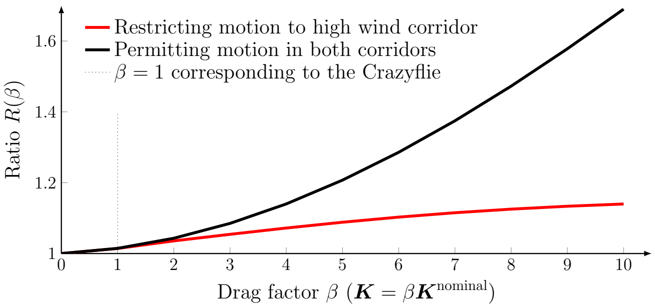

Fig. 5. Gap in thrust cost with (i) a planner that ignores wind and (ii) a planner that leverages wind when increasing the gains of the drag model.

B. Experiments with Real Wind

To verify the implementation and further demonstrate planning approach, a Crazyflie UAV [11] was flown along the trajectories computed in Fig. 4 using an Optitrack motion capture system. The UAV was controlled using the nonlinear geometric tracking controller in [19], and the thrust-cost in (20) was computed directly from the total thrust used to compute the motor commands. Note that this is not the true thrust of the UAV, which we have no way of measuring, but it is a good proxy. The experiment was repeated $N = 10$ times per trajectory with the positional response shown in Fig. 4 (A and B), resulting in the mean thrust usage

$$\text{(i)} \quad \mathbb{E}[\|f\|^2_{\mathcal{L}_2([0,T])}] = 0.90 \quad (k_W k_S^{-1} = 10^{-4}) \tag{34a}$$

$$\text{(ii)} \quad \mathbb{E}[\|f\|^2_{\mathcal{L}_2([0,T])}] = 0.88 \quad (k_W k_S^{-1} = 10^{4}). \tag{34b}$$

The corresponding theoretical values in (32) are an under approximation of the experimental results in (34), but we note that the relative difference in thrust usage is strikingly similar. However, we emphasize that there are many sources of uncertainty, warranting a much larger number of sampled trajectories due to the large number of factors affecting the UAV thrust. Such factors include the absence of feed-forward terms in the controller, the turbulent nature of the wind generated by the fan, and Bernoulli effects that seem to dominate the UAV-wind interaction with the Crazyflie in our experiment setup. Such small difference in thrust cost is expected when operating the Crazyflie at low speeds, as indicated by (32) and Fig. 5, but larger UAVs are likely to see a greater benefit of incorporating the wind information.

VII. Conclusions

In this paper, we demonstrate that classical minimum-snap-like motion planning over Bézier curves can be complemented by additional cost functions that describe the total thrust used by the UAV actuators during a maneuver. We show that if formulated appropriately, the thrust cost is bi-convex in the control points of the Bézier curve and the time-intervals under a linear Newtonian UAV-wind interaction. As such, any Bézier-based motion planner that operates with an objective function expressed in the weighted sum of $\mathcal{L}_2$-norms of high-order motion derivatives can be complemented by the proposed cost, and may leverage wind information.

Using the developed theory, we describe how information from high-fidelity CFD simulators can be leveraged to better plan the motions of the UAVs while retaining computational tractability. The key being a temporal regression of the disturbance in the same function class that motion of the UAV is represented by. To this end, the UAV-wind interaction was studied experimentally, and the expected theoretical reduction in thrust was subsequently seen in the experiments. However, it should be noted that the impact of including thrust in the planner will be significantly greater for larger UAVs with larger drag coefficients (as shown in Fig. 5).

References

[1] S. Nabi, N. Nishio, P. Grover, R. Matai, Y. Kajiyama, N. Kotake, S. Kameyama, W. Yoshiki, and M. Iida, "Improving LiDAR performance on complex terrain using CFD-based correction and direct-adjoint-loop optimization," in *Jour. of Physics*, vol. 1452, no. 1, 2020.

[2] D. Mellinger, N. Michael, and V. Kumar, "Trajectory generation and control for precise aggressive maneuvers with quadrotors," *Int. Robotics Research*, vol. 31, no. 5, pp. 664–674, 2012.

[3] C. Richter, A. Bry, and N. Roy, "Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments," in *Robotics research*. Springer, 2016, pp. 649–666.

[4] A. Loquercio, E. Kaufmann, R. Ranftl, M. Müller, V. Koltun, and D. Scaramuzza, "Learning high-speed flight in the wild," *Science Robotics*, vol. 6, no. 59, 2021.

[5] M. Greiff, A. Vinod, S. Nabi, and S. Di Cairano, "Quadrotor motion planning in stochastic wind fields," in *American Cont. Conf.*, 2023, pp. 4619–4625.

[6] J. Park and H. J. Kim, "Fast trajectory planning for multiple quadrotors using relative safe flight corridor," in *Int. Conf. on Intelligent Robots and Systems (IROS)*. IEEE/RSJ, 2019, pp. 596–603.

[7] J. Tordesillas, B. T. Lopez, M. Everett, and J. P. How, "FASTER: Fast and safe trajectory planner for navigation in unknown environments," *IEEE Trans. Robotics*, vol. 38, no. 2, pp. 922–938, 2021.

[8] T. Marcucci, P. Nobel, R. Tedrake, and S. Boyd, "Fast path planning through large collections of safe boxes," *arXiv preprint arXiv:2305.01072*, 2023.

[9] M. Greiff, "Nonlinear control of unmanned aerial vehicles: Systems with an attitude," Ph.D. dissertation, 2021.

[10] D. A. Olejnik, S. Wang, J. Dupeyroux, S. Stroobants, M. Karasek, C. De Wagter, and G. de Croon, "An experimental study of wind resistance and power consumption in MAVs with a low-speed multi-fan wind system," in *Int. Conf. on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 2989–2994.

[11] Bitcraze, "Crazyflie 2.0," 2021, last accessed: 24-2-2023. [Online]. Available: www.bitcraze.io/products/old-products/crazyflie-2-0

[12] C. De Boor, *A practical guide to splines: Revised Version*. Springer-verlag New York, 2001, vol. 27, ISBN: 0-387-95366-3.

[13] T. Lee, "Computational geometric mechanics and control of rigid bodies," Ph.D. dissertation, University of Michigan, 2008.

[14] H. G. Weller, G. Tabor, H. Jasak, and C. Fureby, "A tensorial approach to computational continuum mechanics using object-oriented techniques," *Computers in physics*, vol. 12, no. 6, pp. 620–631, 1998.

[15] J. H. Ferziger, M. Perić, and R. L. Street, *Computational methods for fluid dynamics*. Springer, 2002, vol. 3, ISBN: 3-540-42074-6.

[16] M. Rieutord, *Fluid dynamics: an introduction*. Springer, 2014, vol. 1, ISBN: 978-3-319-09350-5.

[17] D. Anderson and K. Burnham, "Model selection and multi-model inference," *NY: Springer-Verlag*, vol. 63, no. 2020, p. 10, 2004.

[18] M. Bangura, M. Melega, R. Naldi, and R. Mahony, "Aerodynamics of rotor blades for quadrotors," *arXiv preprint arXiv:1601.00733*, 2016.

[19] T. Lee, M. Leok, and N. H. McClamroch, "Geometric tracking control of a quadrotor UAV on SE(3)," in *Conf. on decision and control (CDC)*. IEEE, 2010, pp. 5420–5425.