Quadrotor Motion Planning in Stochastic Wind Fields

Marcus Greiff, Abraham Vinod, Saleh Nabi, and Stefano Di Cairano

***Abstract*— In this paper, we propose a motion planner for quadrotors in windy environments. We extend a well-known convex polynomial optimization (CPO) method to incorporate known stochastic input uncertainties. In particular, we focus on a quadrotor unmanned aerial vehicle (UAV), and propose a new objective for direct minimization of the squared $\mathcal{L}_2$-norm of the UAV thrust, $\|f\|^2_{\mathcal{L}_2}$. We show that the first two moments of $\|f\|^2_{\mathcal{L}_2}$ are convex in the optimization variables of the CPO problem, and can be minimized directly. Furthermore, we demonstrate that a constrained CPO approach can be used in this setting, contrary to the more popular unconstrained approaches. We provide examples demonstrating: (i) that inclusion of wind can yield significant improvements in the considered cost; (ii) that re-planning of complex paths at can be done at rates exceeding 100 Hz; and (iii) that the proposed method facilitates online re-planning leveraging wind in free-space defined as the union of convex sets.**

I. INTRODUCTION

With their high thrust-to-weight ratio, modern quadrotors are capable of extreme feats of agility, with applications ranging from indoor UAV racing [1] to outdoor resource monitoring [2]. In both scenarios, online re-planning of the motion plan is necessary at high rates, in which case the property of differential flatness [3] is often used in combination with simple parametrizations of the motion plan to formulate tractable optimization problems [4]. Recently, it has become popular to consider trajectories formed in Beziér curves [5], [6], due to the ease with which the trajectories can be confined to convex sets constituting free-space. Similarly, a related polynomial parametrization of the UAV motion is frequently used to define the planning problem [4], [7], [8], and to compute motion primitives [9]. We refer broadly to such polynomial-based approaches as convex polynomial optimization (CPO) planning methods.

While CPO approaches permit fast and flexible re-planning, they come with two significant drawbacks. First, CPO methods rely heavily on the differential flatness equations for trajectory tracking, which limit their flexibility. For instance, the approaches in [4], [7], [8] cannot easily incorporate estimated exogenous disturbances such as wind or drag when computing the motion plan. Second, the objective in the CPO relates to the total variation of the derivatives of the flat output trajectories. This can accommodate the kinetic and potential energy of the UAV, but signals such as the UAV thrust are generally not incorporated directly in the cost function of the CPO methods.

We address these drawbacks in existing methods using the observation that linear combinations of the flat output derivatives under the squared $\mathcal{L}_2$-norm can be included directly in the CPO objective function. Leveraging this insight, we demonstrate that signals such as the thrust in the non-linear UAV dynamics, f, can be minimized over the polynomial coefficients when known forces are acting on the UAV. Additionally, we show that the moments of this modified cost are convex in the original CPO optimization variables when the disturbance is modeled as stochastic. Consequently, we obtain a novel version of the CPO that incorporates input disturbances, with direct application to planning in wind vector fields. With the proposed modification to the CPO cost, the online re-planning [4], [7], [9] which are typically used for agile indoor flight can be employed for outdoor flight and leverage wind information in the path planning.

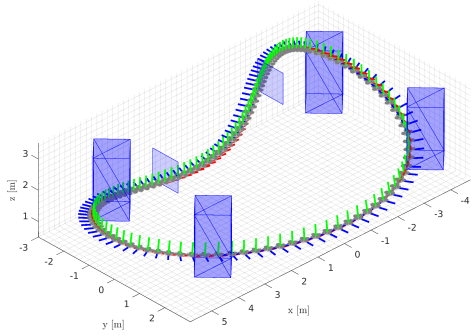

Fig. 1. A UAV motion plan generated with the proposed CPO method.

A. Contributions

The contributions of this paper are threefold:

- We show that a linear combination of polynomial derivatives under the squared $\mathcal{L}_2$-norm is convex and can be minimized together with the usual CPO cost.
- We use this to plan the motion of a UAV perturbed by a viscous friction term in the wind speed, minimizing $\|f\|^2_{\mathcal{L}_2}$ with the wind described as a polynomial in time.
- We show that, for stochastic wind (specifically Gaussian distributed wind speeds), the first two moments of $\|f\|^2_{\mathcal{L}_2}$ are convex in the CPO decision variables.

We demonstrate the approach to motion planning in several simulation examples (see Fig. 1), highlighting the impact of using wind information in the planner (even if it is uncertain).

B. Outline

We give the preliminaries in Sec. II, followed by a brief review of the CPO method in Sec III. A set of modeling assumptions related to the wind disturbance are introduced in Sec. IV, followed by the definition of a new objective function with deterministic wind speeds in Sec. IV-B, and stochastic wind speeds in Sec. IV-C. Supporting numerical results for UAV path planning are given in Sec. V.

Mitsubishi Electric Research Labs (MERL), 02139 Cambridge, MA, USA. Email: greiff@merl.com

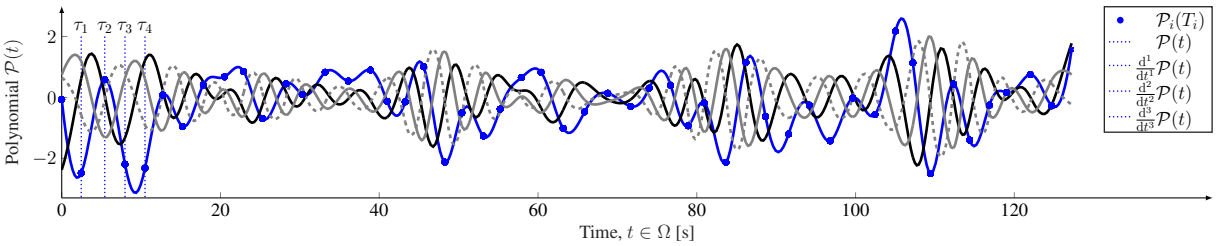

Fig. 2. Example of a convex polynomial optimization solution with $m = 50$ splines of degree $n = 6$, enforcing continuity in the first four derivatives.

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 $[\boldsymbol{e}_i]_i = 1$. Matrices are indicated in bold font as $\boldsymbol{X}$, and the element on row i and column j of $\boldsymbol{X}$ is written $[\boldsymbol{X}]_{i,j}$. Here, $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ indicates that $\boldsymbol{x}$ is Gaussian distributed with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$, and $\boldsymbol{x} \sim \mathcal{U}(I)$ indicates that $\boldsymbol{x}$ is uniformly distributed over an interval I. Additionally, we let $\|\boldsymbol{x}\|_2 = (\boldsymbol{x}^\top \boldsymbol{x})^{1/2}$, denote $\|\boldsymbol{x}\|_{\boldsymbol{A}} = (\boldsymbol{x}^\top \boldsymbol{A}\boldsymbol{x})^{1/2}$, and write the matrix 1-norm as $\|\boldsymbol{A}\|_1 = \max_j \sum_i |[\boldsymbol{A}]_{i,j}|$. The *reciprocal condition number*, defined as

$$\kappa(\boldsymbol{A}) \triangleq \frac{1}{\|\boldsymbol{A}\|_1 \|\boldsymbol{A}^{-1}\|_1} \in [0, 1]. \tag{1}$$

For a univariate random variable X with density function $p_X(x)$, we let $\mathbb{E}_{p_X(x)}[x] = \int_{-\infty}^{\infty} x p_X(x) \mathrm{d}x$ denote its mean (often written $\mathbb{E}[x]$ for brevity), and $\mathbb{V}[x] = \int_{-\infty}^{\infty} (x - \mathbb{E}[x])^2 p_X(x) \mathrm{d}x$ denote its variance. Here, we recollect a result for the computation of moments of quadratic forms.

Lemma 1 *If $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ and $\boldsymbol{A} = \boldsymbol{A}^\top$, the first moments of $\mathcal{Q}(\boldsymbol{x}) = (\boldsymbol{x} - \boldsymbol{\mu})^\top \boldsymbol{A}(\boldsymbol{x} - \boldsymbol{\mu}) + \boldsymbol{b}^\top (\boldsymbol{x} - \boldsymbol{\mu}) + c$ are*

$$\mathbb{E}[\mathcal{Q}(\boldsymbol{x})] = \mathrm{Tr}(\boldsymbol{A}\boldsymbol{\Sigma}) + c, \tag{2a}$$
$$\mathbb{V}[\mathcal{Q}(\boldsymbol{x})] = \mathrm{Tr}(\boldsymbol{A}\boldsymbol{\Sigma}\boldsymbol{A}\boldsymbol{\Sigma}) + \boldsymbol{b}^\top \boldsymbol{\Sigma} \boldsymbol{b}. \tag{2b}$$

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.

A. Polynomials and $\mathcal{L}_2$-norms

Let $\mathcal{P} : [0, T] \mapsto \mathbb{R}^d$ be a curve comprised of m segments $\{\mathcal{P}_i : \Omega_i \mapsto \mathbb{R}^d\}_{i=1}^m$, where $\Omega_i = [0, T_i], T_i > 0, d > 0$, and

$$\mathcal{P}(t) = \mathcal{P}_i(t - \tau_i), \quad \tau_i = \sum_{j=1}^{i-1} T_j, \quad T = \sum_{i=1}^{m} T_i, \tag{3}$$

with $T_0 = 0$ (see Fig. 2). Furthermore, let each segment be defined by sets of polynomial splines $\mathcal{P}_i^j : \Omega_i \mapsto \mathbb{R}$ such that

$$\mathcal{P}_i(t) = (\mathcal{P}_i^1(t), \ldots, \mathcal{P}_i^d(t)) \in \mathbb{R}^d. \tag{4}$$

As we concerned with polynomials, these are defined as tuples $(\mathcal{P}_i^j, \boldsymbol{p}_i^j)$, where $\boldsymbol{p}_i^j \in \mathbb{R}^{n+1}$ denotes a set of coefficients such that if $\boldsymbol{t} = (1, t, \ldots, t^n) \in \mathbb{R}^{n+1}$, then $\mathcal{P}_i^j(t) = \boldsymbol{p}_i^j \cdot \boldsymbol{t}$. Let $\|\mathcal{P}\|_{\mathcal{L}_2(\Omega)}$ denote the standard $\mathcal{L}_2$-norm,

$$\|\mathcal{P}\|_{\mathcal{L}_2(\Omega)} = \Big(\int_\Omega \mathcal{P}(t) \cdot \mathcal{P}(t) \mathrm{d}t \Big)^{1/2}. \tag{5}$$

When obvious, we omit the domain, permitting a shorthand

$$\|\mathcal{P}\|_{\mathcal{L}_2}^2 = \sum_{i=1}^{m} \|\mathcal{P}_i\|_{\mathcal{L}_2}^2 = \sum_{i=1}^{m} \sum_{j=1}^{d} \|\mathcal{P}_i^j\|_{\mathcal{L}_2}^2. \tag{6}$$

In the following, the sub-index $i \in [1, m]$ refers to a *spline index*, and the super-index $j \in [1, d]$ refers to a *dimension*.

B. UAV Dynamics and Differential Flatness

The UAV is configured on $(\boldsymbol{r}, \boldsymbol{R}) \in \mathbb{R}^3 \times \mathsf{SO(3)}$ with $\mathsf{SO(3)} = \{\boldsymbol{R} \in \mathbb{R}^{3\times 3} : \boldsymbol{R}^\top \boldsymbol{R} = \boldsymbol{I}, \det(\boldsymbol{R}) = 1\}$. Here, $\{\mathcal{G}\}$ is a global frame with basis $\{\boldsymbol{e}_i\}_{i=1}^3$, and $\{\mathcal{B}\}$ denotes the body frame of the UAV with a basis $\{\boldsymbol{e}_i^{\mathcal{B}}\}_{i=1}^3$, where $[\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3] = \boldsymbol{R}^\top [\boldsymbol{e}_1^{\mathcal{B}}, \boldsymbol{e}_2^{\mathcal{B}}, \boldsymbol{e}_3^{\mathcal{B}}]$, as depicted in Fig. 3. From the Euler-Lagrange equations [10, Chapter 2.3.5], we have

$$\dot{\boldsymbol{r}} = \boldsymbol{v} \tag{7a}$$
$$m\dot{\boldsymbol{v}} = f\boldsymbol{R}\boldsymbol{e}_3 - mg\boldsymbol{e}_3 \tag{7b}$$
$$\dot{\boldsymbol{R}} = \boldsymbol{R}\boldsymbol{S}(\boldsymbol{\omega}) \tag{7c}$$
$$\boldsymbol{J}\dot{\boldsymbol{\omega}} = \boldsymbol{S}(\boldsymbol{J}\boldsymbol{\omega})\boldsymbol{\omega} + \boldsymbol{\tau}, \tag{7d}$$

where $\boldsymbol{r} \in \mathbb{R}^3$ [m] defines the position of the UAV in the global frame $\{\mathcal{G}\}$; $\boldsymbol{v} \in \mathbb{R}^3$ [m/s] defines the velocity of the UAV in $\{\mathcal{G}\}$; $\boldsymbol{R} \in \mathsf{SO(3)}$ defines the attitude of the UAV; $\boldsymbol{\omega} \in \mathbb{R}^3$ [rad/s] denotes the usual attitude rates defined in $\{\mathcal{B}\}$; $f > 0$ [N] defines the thrust generated by the rotors; $\boldsymbol{\tau}$ [N·m] denotes the torques defined in $\{\mathcal{B}\}$. The model is parameterized by a positive definite symmetric inertia $\boldsymbol{0} \prec \boldsymbol{J} \in \mathbb{R}^{3\times 3}$ [kg·m^2], a positive mass $m > 0$ [kg], and a constant positive gravitation acceleration $g > 0$ [m/s^2]. An interesting property of the UAV dynamics is that the system is differentially flat [7], [11].

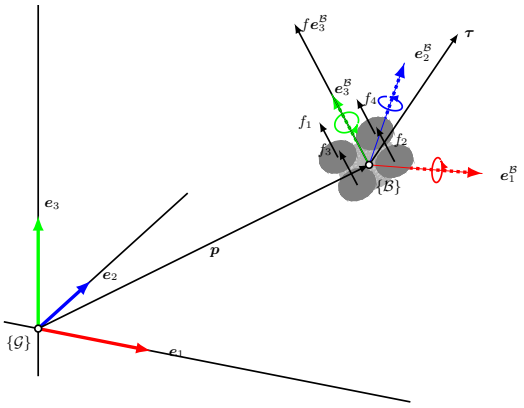

Fig. 3. The UAV geometry and frames (same color coding as in Fig 1).

Definition 1 (Differential Flatness [3]) *A system $\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u})$ with $\boldsymbol{x} \in \mathbb{R}^n$, $\boldsymbol{u} \in \mathbb{R}^m$, and where $\boldsymbol{f}$ is a smooth vector field, is differentially flat if there exist outputs $\boldsymbol{\gamma} \in \mathbb{R}^m$,*

$$\boldsymbol{\gamma} = \boldsymbol{h}(\boldsymbol{x}, \boldsymbol{u}, \tfrac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{u}, \ldots, \tfrac{\mathrm{d}^r}{\mathrm{d}t^r}\boldsymbol{u}), \tag{8}$$

and associated functions $\boldsymbol{x} = \boldsymbol{\phi}(\boldsymbol{\gamma}, \tfrac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\gamma}, \ldots, \tfrac{\mathrm{d}^q}{\mathrm{d}t^q}\boldsymbol{\gamma})$ and $\boldsymbol{u} = \boldsymbol{\beta}(\boldsymbol{\gamma}, \tfrac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\gamma}, \ldots, \tfrac{\mathrm{d}^q}{\mathrm{d}t^q}\boldsymbol{\gamma})$ where $\{\boldsymbol{h}, \boldsymbol{\phi}, \boldsymbol{\beta}\}$ are smooth.

The system (7) is differentially flat in the position, $\boldsymbol{r}$, and a one-dimensional parametrization of the rotation, $\boldsymbol{R}$ (often the so-called yaw angle). For a detailed derivation of these maps, refer to [12], [13]. The implications of the UAV being differentially flat are that feasible motion plans can be defined in smooth functions of time, such as polynomials [4]. In the following, we focus on the positional subspace of the flat outputs, and plan motions in the form $\mathcal{P} \in \mathcal{C}^4(\Omega, \mathbb{R}^3)$ using the polynomial representation in Sec II-A.

III. Convex Polynomial Optimization

It is well known that the integral $\|\mathcal{P}_i^j\|_{\mathcal{L}_2}^2$ can be written as a quadratic form in the polynomial coefficients (see Appendix B).[1] This is leveraged in the CPO methods, which define a cost in the $\mathcal{L}_2$-norms of the polynomial derivatives

$$J_i^j(\boldsymbol{p}_i^j) = \sum_{k=0}^{n} c_k \|(\mathrm{d}^k/\mathrm{d}t^k)\mathcal{P}_i^j(t)\|_{\mathcal{L}_2}^2 = \|\boldsymbol{p}_i^j\|_{\boldsymbol{Q}_i^j(T_i)}^2, \tag{9}$$

where $\boldsymbol{c} = (c_0, \ldots, c_n) \in \mathbb{R}_{>0}^{n+1}$ is a vector that defines a relative weighting of variations in the derivatives of $\mathcal{P}_i^j(t)$. This cost can be minimized efficiently, and it is simple to impose useful constraints on the polynomials. For instance,

$$\boldsymbol{A}_i^j(t)\boldsymbol{p}_i^j = \boldsymbol{b}_i^j(t) = \begin{bmatrix} \frac{\mathrm{d}^0}{\mathrm{d}t^0}\mathcal{P}_i^j(t) \\ \vdots \\ \frac{\mathrm{d}^n}{\mathrm{d}t^n}\mathcal{P}_i^j(t) \end{bmatrix} \in \mathbb{R}^{n+1}, \tag{10}$$

can be implemented to encode end-point constraints, which are typically defined in terms of $\{\boldsymbol{A}_i^j(0), \boldsymbol{b}_i^j(0)\}_{i,j}$ and $\{\boldsymbol{A}_i^j(T_i), \boldsymbol{b}_i^j(T_i)\}_{i,j}$ (see Appendix B). Similarly, continuity across end-points can be expressed with equality constraints

$$\boldsymbol{A}_i^j(T_i)\boldsymbol{p}_i^j - \boldsymbol{A}_{i+1}^j(0)\boldsymbol{p}_{i+1}^j = \boldsymbol{0}, \tag{11}$$

for every $j \in [1, d], i \in [1, m-1]$. Of course, only a subset of these constraints need be applied, and care must to be taken in ensuring that the resulting set of constraints are feasible. By the property in (6), we can write the CPO cost compactly

$$J(\boldsymbol{p}) = \sum_{k=0}^{n} c_k \|\tfrac{\mathrm{d}^k}{\mathrm{d}t^k}\mathcal{P}(t)\|_{\mathcal{L}_2}^2 = \sum_{i=1}^{n}\sum_{j=1}^{m} J_i^j(\boldsymbol{p}_i^j) = \|\boldsymbol{p}\|_{\boldsymbol{Q}}^2, \tag{12}$$

and define the associated quadratic program

$$\min_{\boldsymbol{p}} J(\boldsymbol{p}) \qquad \text{subject to} \qquad \boldsymbol{A}\boldsymbol{p} = \boldsymbol{b}. \tag{13}$$

[1] This is seen by computing the Hessian of $J_i^j(\boldsymbol{p}_i^j)$ in (9) with respect to the polynomial coefficients $\boldsymbol{p}_i^j$, yielding $2\boldsymbol{Q}_i^j(T_i) \succ \boldsymbol{0}$ if $c_0 \neq 0$. The exact way to compute this matrix is given in Algorihm 1 in the appendix.

IV. Planning with Wind

To introduce wind in the path planning, we make a set of assumptions on the UAV–wind interaction, here modeled exogenous forces acting on the translational dynamics (7b). To this end, we consider the drag equation, expressing the force experienced by an object moving through a fully enclosing fluid [14, Chap. 4.5]. The force on the object is

$$f_d(u) = \frac{1}{2}\rho c_d A u^2 \in \mathbb{R}, \tag{14}$$

where u [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 a drag coefficient. Taking a first order Taylor expansion at a nominal $u_\circ$ yields $f_d(u) = \frac{1}{2}\rho c_d A u_\circ^2 + \rho c_d A u_\circ(u - u_\circ) + o(\|u - u_\circ\|^2)$. As such, we model the force on the UAV as a piece wise linear function in the relative velocities of the wind and the UAV.

Assumption 1 *Over the interval $\sum_{j=0}^{i-1} T_j \leq t < \sum_{j=0}^{i} T_j$, the UAV–wind force interaction can be modeled approximately as a linear term in the relative velocity of the UAV, $\boldsymbol{f}_w(t) = \boldsymbol{l}_i - \boldsymbol{K}_i(\boldsymbol{v}(t) - \boldsymbol{v}_{\boldsymbol{w}}(t))$, where $\boldsymbol{v}_{\boldsymbol{w}} \in \mathbb{R}^3$ is the wind speed $\boldsymbol{l}_i = (l_i^1, l_i^2, l_i^3) \in \mathbb{R}^3$, and $\boldsymbol{K}_i \in \mathbb{R}^{3\times 3}$.*

The parameters $\{(\boldsymbol{l}_i, \boldsymbol{K}_i)\}_{i=1}^m$ can be found by a direct linearization of the drag equation (14), or learned from data.

Assumption 2 *The UAV-wind interaction is isotropic, in the sense that $\boldsymbol{K} = \mathrm{diag}(k_i^1, k_i^2, k_i^3)$ with $k_i^j > 0\ \forall i \in [1, m]$.*

Such a UAV–wind interaction is compatible with the CPO methods, and we can proceed by modeling the wind speed $\boldsymbol{v}_{\boldsymbol{w}}$ as a set of polynomials $\{(\mathcal{W}_i^j, \boldsymbol{w}_i^j)\}_{i=1,j=1}^{i=m,j=3}$ defined over the same domains $\{\Omega_i\}_{i=1}^m$ as $\{(\mathcal{P}_i^j, \boldsymbol{p}_i^j)\}_{i=1,j=1}^{i=m,j=3}$. We assume following of the polynomial coefficients defining $\mathcal{W}$.

Assumption 3 *Assume one of the following wind models:*

- **3A** ***(Deterministic)*** *The coefficients $\boldsymbol{w}_i^j$ of each $\mathcal{W}_i^j : \Omega_i \mapsto \mathbb{R}$ are known and deterministic $\forall i \in [1, m], j \in [1, d]$.*
- **3B** ***(Stochastic)*** *The coefficients $\boldsymbol{w}_i^j$ of each polynomial $\mathcal{W}_i^j : \Omega_i \mapsto \mathbb{R}$ are stochastic and Gaussian distributed with $\boldsymbol{w}_i^j \sim \mathcal{N}(\bar{\boldsymbol{\mu}}_i^j, \bar{\boldsymbol{\Sigma}}_i^j)$ for all $i \in [1, m], j \in [1, d]$.*

The resulting wind model is interesting, as it is directly compatible with a high-dimensional Gaussian process model over the coefficients of $\mathcal{W}$. We would likely not leverage the full flexibility of such an approach, but rather a linear or quadratic approximation of the wind speed in time. Next we consider the implications of **Assumptions 1–3** in CPO.

A. Implications of the Wind Modeling

Under **Assumption 1**, the translational dynamics of the UAV perturbed by forces caused by the wind takes the form

$$m\dot{\boldsymbol{v}}(t) = f(t)\boldsymbol{R}(t)\boldsymbol{e}_3 - mg\boldsymbol{e}_3 + \boldsymbol{f}_w(t), \tag{15}$$

and as $\boldsymbol{R}$ vanishes under the norm (and dropping the indices),

$$\|f\|_2^2 = \|m\dot{\boldsymbol{v}} + mg\boldsymbol{e}_3 - \boldsymbol{f}_w\|_2^2 = \|m\dot{\boldsymbol{v}} + mg\boldsymbol{e}_3 - \boldsymbol{l} + \boldsymbol{K}(\boldsymbol{v} - \boldsymbol{v}_{\boldsymbol{w}})\|_2^2$$

If we consider the motion of the UAV as parameterized by the polynomials $\mathcal{P}(t)$, we can express a polynomial $\mathcal{U}(t) \in \mathbb{R}^3$ with splines $(\mathcal{U}_i^j, \boldsymbol{u}_i^j)$, such that $\|f\|_{\mathcal{L}_2}^2 = \|\mathcal{U}\|_{\mathcal{L}_2}^2$, where

$$\mathcal{U}_i(t) = m\tfrac{\mathrm{d}^2}{\mathrm{d}t^2}\mathcal{P}_i(t) + mg\boldsymbol{e}_3 - \boldsymbol{l}_i + \boldsymbol{K}_i(\tfrac{\mathrm{d}}{\mathrm{d}t}\mathcal{P}_i(t) - \mathcal{W}_i(t)). \tag{16}$$

As such, we can define a cost in the polynomial coefficients

$$C(\boldsymbol{p}) = \|f\|_{\mathcal{L}_2}^2 = \sum_{i=1}^{m} \|\mathcal{U}_i\|_{\mathcal{L}_2}^2 = \sum_{i=1}^{m}\sum_{j=1}^{3} \|\mathcal{U}_i^j\|_{\mathcal{L}_2}^2. \tag{17}$$

where the last equality holds given **Assumption 2** (while realistic and simplifying implementations, it is not necessary).

B. Planning with a Deterministic Wind Model

As $\mathcal{U}_j^i(t)$ is a linear combination of the derivatives of $\mathcal{P}_i^j(t)$ and $\mathcal{W}_i^j(t)$, there exists a map relating their coefficients,

$$\boldsymbol{u}_i^j = \boldsymbol{M}_i^j \boldsymbol{p}_i^j + \boldsymbol{m}_i^j, \tag{18}$$

where, in the context of the differential relationship in (16),

$$\boldsymbol{m}_i^j = -k_i^j \boldsymbol{w}_i^j + mg\boldsymbol{e}_1 - \boldsymbol{l}_i^j, \tag{19a}$$

$$[\boldsymbol{M}_i^j]_{r,c} = \begin{cases} k_i^j r, & \text{if} \quad r+1 = c \\ mr(r+1), & \text{if} \quad r+2 = c \\ 0 & \text{otherwise} \end{cases} \tag{19b}$$

which can be formed under **Assumption 3A**. To proceed, let $\bar{\boldsymbol{Q}}_i^j(T_i)$ denote the matrix in (9), evaluated with $c_0 = 1$ and $c_i = 0$ for all $i \neq 0$. Dropping the argument and utilizing (9),

$$\begin{aligned} \|\mathcal{U}_i^j\|_{\mathcal{L}_2}^2 &= \|\boldsymbol{u}_i^j\|_{\bar{\boldsymbol{Q}}_i^j}^2 \\ &= \|\boldsymbol{M}_i^j \boldsymbol{p}_i^j + \boldsymbol{m}_i^j\|_{\bar{\boldsymbol{Q}}_i^j}^2 \\ &= \|\boldsymbol{p}_i^j\|_{(\boldsymbol{M}_i^j)^\top \bar{\boldsymbol{Q}}_i^j \boldsymbol{M}_i^j}^2 + (\boldsymbol{m}_i^j)^\top \bar{\boldsymbol{Q}}_i^j \boldsymbol{M}_i^j \boldsymbol{p}_i^j + \|\boldsymbol{m}_i^j\|_{\bar{\boldsymbol{Q}}_i^j}^2 \end{aligned} \tag{20}$$

which is notably convex in $\boldsymbol{p}_i^j$, as $(\boldsymbol{M}_i^j)^\top \bar{\boldsymbol{Q}}_i^j(T_i)\boldsymbol{M}_i^j \succeq \boldsymbol{0}$. This is useful, as the cost $C(\boldsymbol{p})$ directly relates to signals in (7), but here we can make a more general statement.

Remark 1 *A linear combination of derivatives of $\mathcal{P}$ under the squared $\mathcal{L}_2$-norm is convex in the coefficients $\boldsymbol{p}$.*

This insight permits a modification of the CPO, as follows

$$\min_{\boldsymbol{p}} J(\boldsymbol{p}) + \alpha C(\boldsymbol{p}) \qquad \text{subject to} \qquad \boldsymbol{A}\boldsymbol{p} = \boldsymbol{b}. \tag{21}$$

for some relative weighting factor $\alpha > 0$. This equality-constrained quadratic program can be solved efficiently by inverting the associated the KKT conditions [15].[2]

C. Planning with a Stochastic Wind Model

While (21) can be solved under **Assumption 3A**, this assumption can be challenged. The exogenous forces, $\boldsymbol{f}_w$, caused by the UAV-wind interaction in (15) will never be known with certainty. Even if the model can be considered accurate, the wind speed will have to be estimated or assumed. In both cases, it is interesting to see how stochastic wind speeds affect the cost used in the CPO planning. To study this, we instead consider **Assumption 3B** and take the wind speed to be Gaussian distributed in its polynomial coefficients, that is, $\boldsymbol{w}_i^j \sim \mathcal{N}(\bar{\boldsymbol{\mu}}_i^j, \bar{\boldsymbol{\Sigma}}_i^j)$. The reason for considering such a wind model is that a likely candidate for wind estimation is a spatio-temporal Gaussian process over the polynomial coefficients $\boldsymbol{w}_i^j$. Here, we note that uncertainty of this kind only enters the constant term $\boldsymbol{m}_i^j \sim \mathcal{N}(\boldsymbol{\mu}_i^j, \boldsymbol{\Sigma}_i^j)$ in the map (18), where then

$$\boldsymbol{\mu}_i^j = -k^j \bar{\boldsymbol{\mu}}_i^j + mg\boldsymbol{e}_1 - \boldsymbol{l}_i^j, \qquad \boldsymbol{\Sigma}_i^j = (k_i^j)^2 \bar{\boldsymbol{\Sigma}}_i^j. \tag{22}$$

By completion of squares and leveraging Lemma 1, the first two moments of $\|\mathcal{U}_i^j\|_{\mathcal{L}_2}^2$ can be expressed analytically as

$$\begin{aligned} \mathbb{E}_{p(\boldsymbol{w}_i^j)}[\|\mathcal{U}_i^j\|_{\mathcal{L}_2}^2] &= \mathrm{Tr}(\bar{\boldsymbol{Q}}\boldsymbol{\Sigma}_i^j) + 2(\boldsymbol{\mu}_i^j + \boldsymbol{M}_i^j \boldsymbol{p}_i^j)^\top \bar{\boldsymbol{Q}}_i^j \boldsymbol{\mu}_i^j \\ &\quad - \boldsymbol{\mu}_i^j \bar{\boldsymbol{Q}}_i^j \boldsymbol{\mu}_i^j + (\boldsymbol{M}_i^j \boldsymbol{p}_i^j)^\top \bar{\boldsymbol{Q}}_i^j \boldsymbol{M}_i^j \boldsymbol{p}_i^j, \end{aligned} \tag{23a}$$

$$\begin{aligned} \mathbb{V}_{p(\boldsymbol{w}_i^j)}[\|\mathcal{U}_i^j\|_{\mathcal{L}_2}^2] &= \mathrm{Tr}(\bar{\boldsymbol{Q}}_i^j \boldsymbol{\Sigma}_i^j \bar{\boldsymbol{Q}}_i^j \boldsymbol{\Sigma}_i^j) \\ &\quad + 4(\boldsymbol{\mu}_i^j + \boldsymbol{M}_i^j \boldsymbol{p}_i^j)^\top \bar{\boldsymbol{Q}}_i^j \boldsymbol{\Sigma}_i^j \bar{\boldsymbol{Q}}_i^j (\boldsymbol{\mu}_i^j + \boldsymbol{M}_i^j \boldsymbol{p}_i^j), \end{aligned} \tag{23b}$$

Notably, both of these expressions are convex in the polynomial parameters: (23a) for the same reason the (20) is convex, and (23b) due to $\bar{\boldsymbol{Q}}_i^j \boldsymbol{\Sigma}_i^j \bar{\boldsymbol{Q}}_i^j \succeq 0$, even if $\boldsymbol{\Sigma}_i^j$ is degenerate. Consequently, in addition to directly minimizing the expected UAV thrust subject to wind, we may directly minimize the variance of the cost over the polynomial coefficients,

$$\min_{\boldsymbol{p}} J(\boldsymbol{p}) + \alpha \mathbb{E}_{p(\boldsymbol{w})}[C(\boldsymbol{p})] + \beta \mathbb{V}_{p(\boldsymbol{w})}[C(\boldsymbol{p})]. \tag{24}$$

It is important to note that the random variable $\|\mathcal{U}_i^j\|_{\mathcal{L}_2}^2$ is not Gaussian and generally has a non-zero kurtosis, but if we have a large number of splines and many different realizations of $\boldsymbol{w}_i^j$, the cost distribution will approach a Gaussian by the central limit theorem. While the expressions in (23) are exact, they do not comprise sufficient statistics to represent the cost distribution, but still prove useful in reasoning about and minimizing uncertainty in the cost.

V. Numerical Results

In this section, we provide several numerical results to illustrate the effects of modifying the CPO cost. First, we verify the theoretical developments in a simplified planning example, highlighting that the wind can only be leveraged if the trajectory is sufficiently unconstrained. We then show the impact of increasing α in the modified cost function, and finally demonstrate the approach in a re-planning scenario.

A. Consequences of Incorporating Wind in the Planning

To illustrate the importance of incorporating the wind information in the motion plan, even if it is only known approximately, we consider a simplified example with:

(i) The CPO in (13) with no modeled wind ($\alpha = \beta = 0$);
(ii) The CPO in (24) with stochastic wind ($\alpha = \beta = 1$).

The two problems are solved with $(m, d, n) = (10, 3, 6)$, ensuring that $\mathcal{P} \in \mathcal{C}^4([0, T], \mathbb{R}^3)$ by enforcing (11), and applying end-point constraints at $\mathcal{P}(0) \sim \mathcal{N}(\boldsymbol{0}, 4\boldsymbol{I})$ and $\mathcal{P}(T) \sim \mathcal{N}(\boldsymbol{0}, 4\boldsymbol{I})$ using (10). To study the impact of derivative constraints on the resulting thrust, we let $\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{P}^5(T_5) \sim$

[2]To facilitate implementations, we give a convenient decomposition of the cost as $C(\boldsymbol{p}) = \boldsymbol{p}^\top \boldsymbol{H}\boldsymbol{p} + \boldsymbol{f}^\top \boldsymbol{p} + c$ by Algorithm 3 in Appendix B.

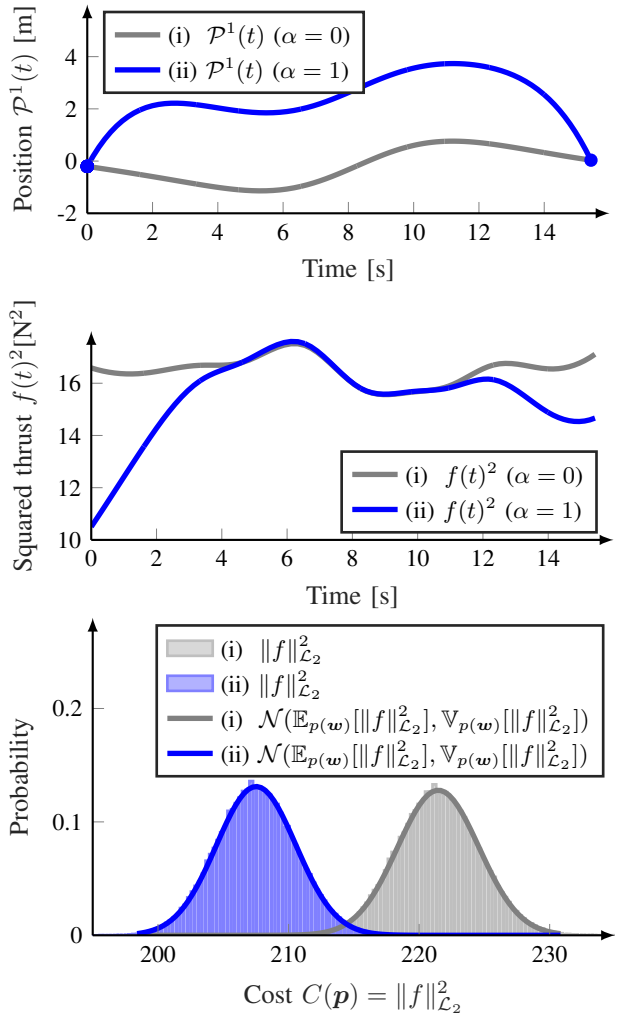

Fig. 4. *Top*: First dimension, $\mathcal{P}^1(t)$, computed using a CPO with (i) no modeled wind (gray) and (ii) stochastic modeled wind (blue). *Center*: The signal $\|f\|^2$ resulting from (i)–(ii). *Bottom*: The histogram of $C(\boldsymbol{p})$ for (i)–(ii), with Gaussians $\mathcal{N}(\mathbb{E}_{p(\boldsymbol{w})}[\|f\|^2_{\mathcal{L}_2}], \mathbb{V}_{p(\boldsymbol{w})}[\|f\|^2_{\mathcal{L}_2}])$ (thick).

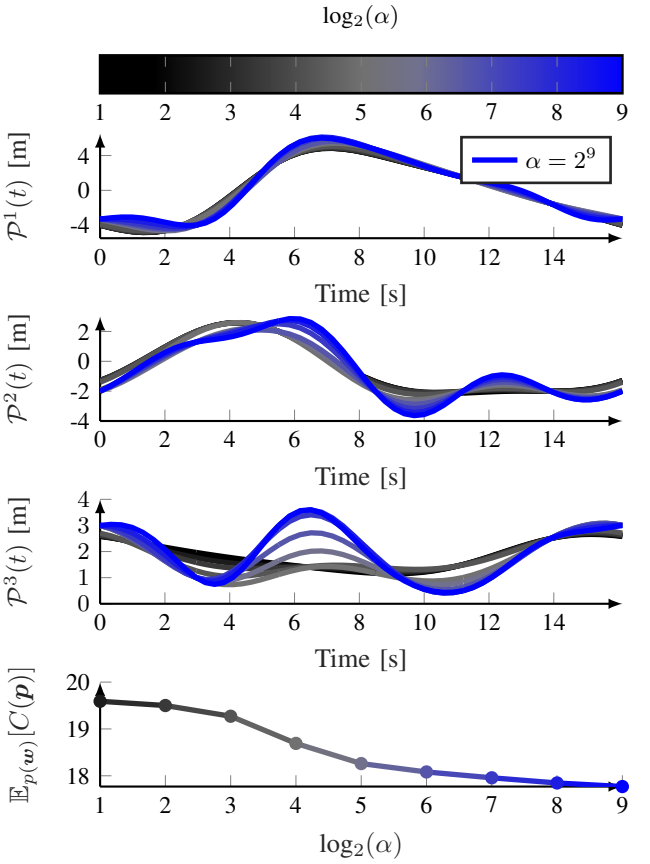

Fig. 5. *Top three plots:* The polynomial trajectory $\mathcal{P} \in \mathcal{C}^4(\Omega, \mathbb{R}^3)$ computed using a CPO for increasingly large α (from black to blue). *Bottom:* Expected thrust cost $\mathbb{E}_{p(\boldsymbol{w})}[C(\boldsymbol{p})]$ as a function of α used in defining the cost in (24).

$\mathcal{N}(\mathbf{0}, 4\boldsymbol{I})$ and $\frac{\mathrm{d}^2}{\mathrm{d}t^2}\mathcal{P}^5(T_5) \sim \mathcal{N}(\mathbf{0}, 4\boldsymbol{I})$. The domains are sampled as $T_i \sim \mathcal{U}([1,2])$, the remaining parameters are given in Appendix A. For simplicity, the means $\bar{\boldsymbol{\mu}}_i^j$ are found by solving a CPO similar to Figure 2, and $\bar{\boldsymbol{\Sigma}}_i^j = \gamma_1 \boldsymbol{e}_1 \boldsymbol{e}_1^\top + \gamma_2 \boldsymbol{e}_2 \boldsymbol{e}_2^\top + \gamma_3 \boldsymbol{e}_3 \boldsymbol{e}_3^\top + 0.01\boldsymbol{I}$, where $\gamma_i \sim \mathcal{U}([0.1, 0.2])$. In the Monte–Carlo runs, only the realization of the wind differs.

The solution associated with one realization of the wind model is depicted in Figure 4, along with the distribution of $C(\boldsymbol{p})$ from 10^6 Monte–Carlo runs, also plotting the associated first two moments of the cost computed analytically by (23). This example demonstrates that modeling of the wind and a mixed minimization of the original CPO cost and the thrust cost affects the motion plan significantly. Furthermore, minimizing over the thrust cost only has a significant impact if there is some freedom in the velocities and accelerations of the trajectory. Around $t \approx 8$, the forces of the two motion plans are near-identical due to the constrains imposed on $\frac{\mathrm{d}^k}{\mathrm{d}t^k}\mathcal{P}^5(T_5)$. Despite this, we note that the CPO in (ii) yields a statistically significant improvement over the CPO in (i) when considering $\|f\|^2_{\mathcal{L}_2}$. Finally we note that the impact of the wind can easily be studied through (23) without resorting to Monte–Carlo runs, as the analytical moment computations are consistent with the empirical statistics. As such, the formulas provided in (23) can be used both for diagnostics purposes and for trajectory generation.

B. A Realistic Cyclic Planning Example

Next, we provide a realistic planning example where the UAV is to fly through a sequence of convex polyhedral sets in space, effectively acting as mixed equality and inequality constraints on the polynomial coefficients. The definition of these constraints is omitted, but they are given in the form

$$\boldsymbol{C}_i^j \mathcal{P}_i(0) \leq \boldsymbol{c}_i^j, \quad \boldsymbol{D}_i^j \mathcal{P}_i(0) \leq \boldsymbol{d}_i^j, \quad \forall i = 1, \ldots, m. \tag{25}$$

Planning is done in the three-dimensional positional subspace of the flat outputs of the UAV. Continuity is enforced in the first four derivatives, and also between the endpoint of the terminal spline and the first spline resulting in a cyclic trajectory. In total, $m = 6$ splines of dimension $d = 3$ and degree $n = 6$ are used in the optimization, resulting in a problem with 126 decision variables and 106 constraints, of which 32 are inequality constraints. The resulting path plan with the CPO including wind is depicted Fig. 1 (with $\alpha = 2^4$), and the CPO solutions with an increasing weighting

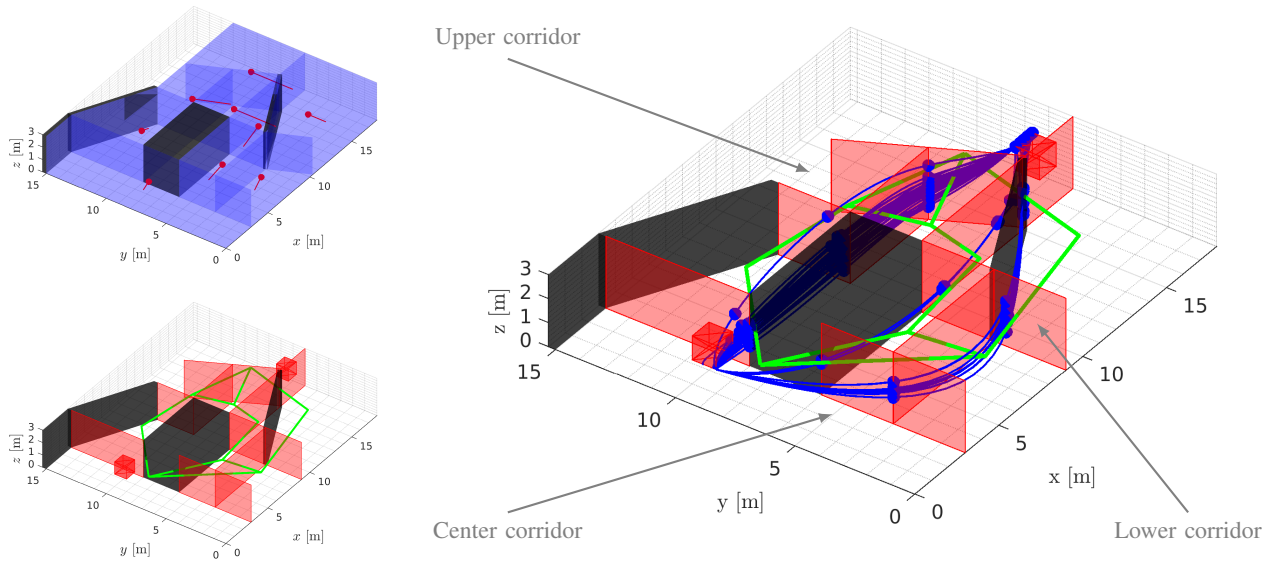

Fig. 6. Illustration of re-planning with wind using the proposed modified CPO. *Top left:* Three obstacles (black) and 9 regions constituting $\mathcal{C}$ defining free-space (blue), and each associated with a specific wind speed, here depicted in the mean (red) about the centroids. *Bottom left:* Depiction of the non-empty lower-dimensional sets $\mathcal{K}_{ab}$ (red) with the route between centroids of the sets $\mathcal{C}$ illustrating non-empty intersections as lines between these centroids (green). The figure also illustrates the start and terminal regions of the trajectory as two red boxes, each entirely contained in one of the sets $\mathcal{C}_l$. *Right:* Best solutions evaluated by the CPOs without accounting for wind (gray) and with wind information (blue) when re-planning over 10 [s].

of the thrust cost is depicted in Fig. 5. The computational times[3] for solving this problem is less than 10 [ms]. These computational times are exceptionally small, permitting re-planning of the entire trajectory in Fig. 1 at rates exceeding 100 [Hz]. Furthermore, the trajectory becomes more volatile as α increases, but the expected cost in $\|f\|^2_{\mathcal{L}_2}$ decreases with α as the UAV increasingly leverages wind information.

C. Re-planning with Wind in Free-Space

There are myriad ways in which the proposed modified cost function can be used in practice. One such example is to grow a lower-dimensional rapidly exploring random tree (RRT) in free-space [16], and apply the CPO as a final step to generate a smooth and dynamically feasible trajectory, as discussed in [4]. The proposed modification to the CPO is entirely compatible with this idea, but here we instead demonstrate how it can be leveraged in graph-based planning, leveraging the speed of the optimization to perform re-planning in free-space defined by a set of convex polyhedral sets analogous to [5]. To start, we consider a convex polyhedral set defined similar to (25). Let

$$\mathcal{C}_l = \{\boldsymbol{x} \in \mathbb{R}^3 | \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b}, \boldsymbol{A}_{\text{eq}}\boldsymbol{x} = \boldsymbol{b}_{\text{eq}}\}, \tag{26}$$

denote free-space, and consider a set of L regions $\mathcal{C} = \{\mathcal{C}_l\}_{l=1}^{L}$. Assume the existence of a function that takes two such regions, and represents their intersection $\mathcal{K}_{ab} = \mathcal{C}_a \cap \mathcal{C}_b$ in the same form. In graph-based planning, it is common to construct a distance matrix, defining the cost-to-go from one node to the next. Here, we instead consider a matrix of polyhedral sets $\mathcal{K}$, in which $[\mathcal{K}]_{a,b} = \mathcal{K}_{ab}$, and evaluate the cost of going from $\mathcal{C}_a$ to $\mathcal{C}_b$ by solving a CPO that enforces constraints in the form of (25). For instance, if we would like to steer a UAV from $\mathcal{C}_a$ to $\mathcal{C}_d$, then a possible solution is found by solving a CPO constrained by $\mathcal{K}_{ab}, \mathcal{K}_{bc}, \mathcal{K}_{cd}$ if these are all non-empty. The number of routes to be checked depends on L and the connectivity of $\mathcal{K}$.

To demonstrate this, consider the environment in Fig. 6 where three-dimensional space is decomposed into a set of $L = 11$ convex polyhedral sets (blue) with a resulting sparse $\mathcal{K}$ (red). In this example, there are only five routes to check (green), which given the computational speeds reported in the previous example can be done at approximately 20 [Hz]. Each region in space is associated with a wind speed, with the mean wind velocity illustrated in Fig. 6. The wind speed is here taken to be time-invariant with only the constant term of each wind polynomial $[\boldsymbol{w}_i^j]_1 \neq 0$. The wind is sampled from a Gaussian distribution at 10 [Hz] to simulate measurements on the wind speed taken by a sensor.

When solving the CPO without wind, the same solution is found on all time steps. In contrast, the CPO incorporating wind information adapts the path with respect to the measured wind and its effect on the rotor thrusts. In some realizations (36%), the middle and lower paths are optimal, in which case the CPO including wind correctly selects these paths. In all realizations, the expected thrust cost, $C(\boldsymbol{p})$, is lower when including the wind. This demonstrates the utility of considering the effect of the wind of the UAV in planning.

Interestingly, the planner selects a point with high elevation in the initial set (box at large x) and lower elevation in the terminal set (box at small x). This is due to the gravitational field which is modeled in the thrust cost. Knowledge of this can be used to minimize the thrust used during the maneuver. This behavior is not seen in conventional CPOs, as the snap cost in (12) is agnostic to gravitational effects.

[3]Using Matlab's `quadprog` on an 11th Gen Intel i7 @ 2.80GHz.

VI. Conclusions

In this paper, we propose an additional term in the cost function of the conventional CPO planners which permits a direct minimization of the squared $\mathcal{L}_2$-norm of the UAV thrust, while incorporating deterministic or stochastic information on wind speeds. Examples were given with a constrained CPO approach, but one can formulate similar unconstrained versions with the same objective following [4]. This permits a flexible and fast path planning suitable for online re-planning, which is a necessity when considering wind information. While the ideas were developed specifically for UAVs subject to wind disturbances, we emphasize that Remark 1 holds for any signal that can be expressed as a linear combination of the derivatives of the flat outputs, and that such costs be expressed for any differentially flat system that currently employs CPO motion planning [8].

Future work will demonstrate the approach in practice using the Crazyflie 2.1 platform, performing re-planning of the UAV motions with partial estimates of the wind field.

References

[1] S. Sun, A. Romero, P. Foehn, E. Kaufmann, and D. Scaramuzza, "A comparative study of nonlinear MPC and differential-flatness-based control for quadrotor agile flight," *IEEE Trans. Robotics*, 2022.

[2] 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.

[3] R. M. Murray, M. Rathinam, and W. Sluis, "Differential flatness of mechanical control systems: A catalog of prototype systems," in *ASME Int. Mech. Eng. Congress and Exposition*, 1995.

[4] 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.

[5] 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.

[6] T. Marcucci, M. Petersen, D. von Wrangel, and R. Tedrake, "Motion planning around obstacles with convex optimization," arXiv, 2022. [Online]. Available: https://arxiv.org/abs/2205.04422

[7] 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.

[8] M. Greiff, "A time-warping transformation for time-optimal movement in differentially flat systems," in *American Control Conference (ACC)*. IEEE, 2018, pp. 6723–6730.

[9] M. W. Mueller, M. Hehn, and R. D'Andrea, "A computationally efficient motion primitive for quadrocopter trajectory generation," *IEEE Trans. Robotics*, vol. 31, no. 6, pp. 1294–1310, 2015.

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

[11] M. Greiff, "Modelling and control of the crazyflie quadrotor for aggressive and autonomous flight by optical flow driven state estimation," Master's thesis, 2017.

[12] D. Mellinger, "Trajectory generation and control for quadrotors," Ph.D. dissertation, University of Pennsylvania, 2012.

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

[14] M. Rieutord, *Fluid dynamics: an introduction*. Springer, 2014.

[15] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[16] S. M. LaValle *et al.*, "Rapidly-exploring random trees: A new tool for path planning," 1998.

Appendix

A. Nominal Simulation Parameters

In the simulations, we let $m = 0.1$, $g = 9.81$, $k_i^1 = k_i^2 = k_i^3 = 0.2$, and $l_i^j = 0 \; \forall \; i = 1, \dots, m, j = 1, \dots, 3$.

B. Matrix Expressions and Algorithms

The three key algorithms required to formulate the snap costs, the constraints, and the wind costs for individual spline segments are provided in Algorithms 1–3.

Algorithm 1 `get_Q`$(\cdot)$, factorization of $\mathcal{L}_2^2$-cost over $[0, T]$.

Receive: $\boldsymbol{c}, n$
Set $\boldsymbol{Q} = \boldsymbol{0} \in \mathbb{R}^{n+1 \times n+1}$
for $k = 0, \dots, n$ **do**
  Set $\boldsymbol{Q}_k = \boldsymbol{0} \in \mathbb{R}^{n+1 \times n+1}$
  for $i = 0, \dots, n$ **do**
    for $j = 0, \dots, n$ **do**
      if $i \geq k$ **and** $j \geq k$ **then**
        $[\boldsymbol{Q}_k]_{i+1,j+1} = \frac{T^{i+j-2k+1}}{(i+j-2k+1)}$
        for $l = 0, \dots, k-1$ **do**
          $[\boldsymbol{Q}_k]_{i+1,j+1} = [\boldsymbol{Q}_k]_{i+1,j+1}(i-l)(j-l)$
        end for
      end if
    end for
  end for
  $\boldsymbol{Q} = \boldsymbol{Q} + c_k \boldsymbol{Q}_k$
end for
Output $\boldsymbol{Q}$

Algorithm 2 `get_A`$(\cdot)$, constraints on polynomial at time t.

Receive: t, n
Set $\boldsymbol{A} = \boldsymbol{0} \in \mathbb{R}^{n+1 \times n+1}$
for $i = 0, \dots, n$ **do**
  for $j = 0, \dots, n$ **do**
    if $j \geq i - 1$ **then**
      $[\boldsymbol{A}]_{i+1,j+1} = t^{j-i}$
      for $l = 0, \dots, i-1$ **do**
        $[\boldsymbol{A}]_{i+1,j+1} = [\boldsymbol{A}]_{i+1,j+1}(j-l)$
      end for
    end if
  end for
end for
Output $\boldsymbol{A}$

Algorithm 3 `get_QP`$(\cdot)$, QP-factorization of cost with wind.

Receive: $n, \boldsymbol{w}, k, m, g, l$
Set $\boldsymbol{M} = \boldsymbol{0} \in \mathbb{R}^{n+1 \times n+1}$
Set $\boldsymbol{m} = -k\boldsymbol{w} \in \mathbb{R}^{n+1 \times 1}$
Set $[\boldsymbol{m}]_1 = [\boldsymbol{m}]_1 + mg - l$
for $i = 1, \dots n$ **do**
  $[\boldsymbol{M}]_{i,i+1} = ik$
  if $i + 2 < n + 1$ **then**
    $[\boldsymbol{M}]_{i,i+2} = mi(i+1)$
  end if
end for
Compute $\bar{\boldsymbol{Q}} = \texttt{get\_Q}(n, T, \boldsymbol{e}_1)$
Set $\boldsymbol{H} = \boldsymbol{M}^\top \bar{\boldsymbol{Q}} \boldsymbol{M}$, $\boldsymbol{f} = 2\boldsymbol{M}^\top \bar{\boldsymbol{Q}} \boldsymbol{m}$, $c = \boldsymbol{m}^\top \bar{\boldsymbol{Q}} \boldsymbol{m}$
Output $\boldsymbol{H}, \boldsymbol{f}, \boldsymbol{c}$