Meta-Learning Augmented MPC for Disturbance-Aware Motion Planning and Control of Quadrotors

Dženan Lapandić[1], Fengze Xie[2], Christos K. Verginis[3], Soon-Jo Chung[2], Dimos V. Dimarogonas[1] and Bo Wahlberg[1]

***Abstract*— A major challenge in autonomous flights is unknown disturbances, which can jeopardize safety and cause collisions, especially in obstacle-rich environments. This paper presents a disturbance-aware motion planning and control framework for autonomous aerial flights. The framework is composed of two key components: a disturbance-aware motion planner and a tracking controller. The motion planner consists of a predictive control scheme and an online-adapted learned disturbance model. The tracking controller, developed using contraction control methods, ensures safety bounds on the quadrotor's behavior near obstacles with respect to the motion plan. The algorithm is tested in simulations with a quadrotor facing strong crosswind and ground-induced disturbances.**

I. Introduction

Enhancing the autonomy of unmanned aerial vehicles (UAVs) has made safe autonomous landing in harsh environments a key research challenge. This capability is relevant in various domains and applications, including air mobility, search and rescue, and drone delivery [1]–[3]. Developing robust quadrotor landing algorithms is challenging due to disturbances and safety-critical constraints near obstacles. Therefore, planning and control algorithms must account for these disturbances and their effects on UAV performance.

Rotor-based aircraft experience increased thrust near the ground due to reduced downwash, known as the ground effect, first documented for helicopters in [4]. Modeling ground effects is complex, and neglecting them can pose significant safety risks. Accurately identifying or learning these disturbance models can lead to smoother, safer UAV landings. Ground effect disturbances, a type of interaction-produced disturbance, can be modeled using neural networks that take the relative position of the UAV from the ground as input [5]. Incorporating such models into motion planning enables UAVs to find and execute optimal trajectories that account for ground effect disturbances.

In this paper, we focus on augmenting the nominal quadrotor dynamics with a learned disturbance model inside the MPC scheme for predicting future behaviour. The disturbance model is acquired and refined through meta-learning,

*This work was supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP), the Swedish Research Council and Knut and Alice Wallenberg Foundation (KAW).

[1]Dženan Lapandić, Dimos V. Dimarogonas and Bo Wahlberg are with Division of Decision and Control Systems, KTH Royal Institute of Technology, Stockholm, Sweden. `lapandic,dimos,bo@kth.se`

[2]Fengze Xie and Soon-Jo Chung are with Division of EAS, Caltech, Pasadena, CA, USA `fxxie,sjchung@caltech.edu`

[3]Christos K. Verginis is with Division of Signals and Systems, Department of Electrical Engineering, Uppsala University,Uppsala, Sweden. `christos.verginis@angstrom.uu.se`

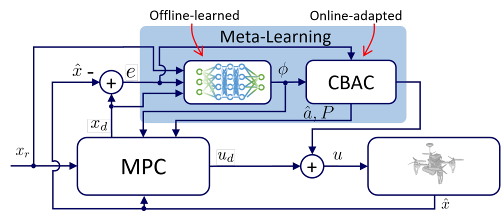


Fig. 1: Block diagram of the proposed disturbance-aware motion planning and control algorithm.

defined here as an online adaptation of the offline-trained disturbance model to the observed environmental conditions. Our meta-learning approach leverages deep neural networks, with the final layer dynamically adapted online through adaptive control mechanisms [6]. Since the application is safety-critical, collecting data to learn the representation must be done efficiently and safely. During each MPC iteration, the meta-learning algorithm updates the parameter estimates and covariance matrix, which are then applied in the next prediction step to refine the disturbance model. A low-level contraction-based controller complements the feedforward MPC control action, ensuring convergence to the planned trajectory. Contraction theory [7] provides performance bounds *a priori*, with respect to the desired trajectory. The planner can utilize this information to guarantee collision-free behaviour near obstacles. Thus, the proposed framework achieves disturbance-aware planning and control with theoretical safety guarantees.

The main contributions of this paper are

i) the augmentation of the model predictive control (MPC) with the learned model of disturbances within the proposed meta-learning framework,
ii) stability guarantees for the models with approximated state-dependent coefficients,
iii) theoretical considerations on chance-constrained upper bounds for safety.

A. Related Work

Recent research on quadrotor control and disturbance handling often uses optimization-based controllers like model predictive control (MPC) with adaptive methods for model mismatch [8]. In [9], Gaussian processes model aerodynamic effects and propagate corrected dynamics within MPC, outperforming linear compensation approaches [10]. Disturbances are categorized as matched and unmatched in [11], with online estimation and adaptation countering matched disturbances while ensuring unmatched ones remain bounded. Ground-effect disturbances [4], [12]–[14] have been effectively represented and cancelled using spectrally-normalized deep neural networks (DNNs) [5]. The characteristics of interaction-produced disturbances have been studied in [15], [16]. In [17], [18], authors use meta-learning to combine offline learning and online adaptation to cancel the wind disturbances represented by a learned deep neural network and a set of linear coefficients adapted online for the current wind conditions. Such a decomposition effectively represents the unknown wind dynamics. Furthermore, meta-learning approaches have been explored in [19]–[24].

Note that a similar study on using interaction-aware disturbances for motion planning has been conducted in [16]. However, the main differences to this work are that we now consider the nonlinear model of quadrotor dynamics instead of double-integrator dynamics and use the control inputs provided as the outcome of the optimal control problem as a feedforward input to the low-level controller.

The paper is organized as follows. The problem formulation is given in Sec. II, and the meta-learning augmented MPC is presented in Sec. III. Finally, Sec. IV presents the simulation results, and Sec. V concludes the paper.

B. Notation

For a matrix $A \in \mathbb{R}^{n\times n}$, $A \succ 0$ denotes that A is positive definite, $A \succeq 0$ that A is positive semi-definite, and $\lambda_{\min}(A)$ and $\lambda_{\max}(A)$ are minimal and maximal eigenvalues of A, respectively. Also, $2\text{sym}(A) = A + A^T$.

II. Problem Formulation

We consider the quadrotor UAV model:

$$\dot{p} = v, \tag{1a}$$
$$m\dot{v} = R(\eta)f_T + f_{dist} - mg, \tag{1b}$$
$$\dot{\eta} = R_T(\eta)\omega, \tag{1c}$$
$$J\dot{\omega} = J\omega \times \omega + \tau_{dist} + \tau_u \tag{1d}$$

where $x = [p^T, v^T, \eta^T, \omega^T]^T \in \mathcal{X} = \mathbb{R}^{12}$, $p \in \mathbb{R}^3$ is the position in the inertial frame, $v \in \mathbb{R}^3$ is the linear velocity, $\eta = [\phi, \theta, \psi]^T \in \mathbb{T} = (-\frac{\pi}{2}, \frac{\pi}{2}) \times (-\frac{\pi}{2}, \frac{\pi}{2}) \times (-\pi, \pi)$ is the vector of Euler angles representing the attitude (roll, pitch, yaw angles), and $\omega = [\omega_\phi, \omega_\theta, \omega_\psi]^T$ is the angular velocity, expressed in the body frame; $f_T = [0, 0, f_u]^T$ is the controlled thrust, and τ_u is the controlled torque; $R : \mathbb{T} \to SO(3)$ is the rotation matrix from the body to the inertial frame, and $SO(3)$ is the special orthonormal group in 3D and $I_n \in \mathbb{R}^{n\times n}$ is the identity matrix; Furthermore, $R_T : \mathbb{T} \to \mathbb{R}^{3\times 3}$ is the mapping from the angular velocity to the time derivatives of the Euler angles. Wind disturbances, ground effects, unmodeled aerodynamic forces, and other external disturbances are captured by f_{dist}. Since the disturbance torque τ_{dist} is two orders of magnitude smaller than other feedback terms in a standard attitude controller loop [16], we focus on learning f_{dist} alone. Finally, m and J are the mass and positive definite inertia matrix of the UAV.

Because of the assumption on additive disturbances as in (1b), the quadrotor model can be rewritten in the following form

$$\dot{x} = f(x) + B(x)u + f_d \tag{2}$$

where $f_d = [0^T, f_{dist}^T, 0^T, 0^T]^T$ because we assume $\tau_{dist} \approx 0$. We consider the quadrotor tasked with landing on a moving platform with the reference state $x_r(t)$ being derived from the motion of the platform.

Problem 1: Given the state estimate of the moving landing platform $x_r(t) \in \mathcal{X}$, $t \geq 0$, derive a trajectory $x : [0, t_f] \to \mathcal{X}$ and a control input $u = [f_u, \tau_u^T]^T$, with control thrust f_u and torque τ_u such that the state of quadrotor $x(t_f)$ at t_f is in the goal region $\mathcal{X}_{\text{goal}}(x_r(t_f))$ determined by the state of the landing platform and state and control constraints $x \in \mathcal{X}$, $u \in \mathcal{U}$ are satisfied for all t.

III. Meta-learning Augmented MPC

The interaction-produced disturbances, mainly generated by the ground effect, can also be modelled with a neural network that, as input, takes the relative position from the ground surface [5]. In this work, we introduce a disturbance model that incorporates the position of the targeted ground surface, setting it apart from the approach in [17]

$$f_d(x, x_r, w) \approx \phi(x, x_r, \Theta)a(w) \tag{3}$$

where $a(w)$ is an unknown parameter that depends on $w \in \mathbb{R}^m$ which is an unknown hidden state representing the underlying environmental conditions and can also be time-varying, and ϕ is a neural network with parameters Θ. Note that ϕ has non-zero components only in the linear velocity term (1b), so our $\phi = [0, \phi_v, 0]$. The function ϕ constitutes a basis function in the meta-learning approach that is invariant to the specific environment conditions. The specific conditions are tackled online by dynamically adapting the parameter a in the disturbance model which, practically, can be seen as changing the final layer of the learned neural network model.

A. Neural Network Model of Disturbances

The first stage is an offline training of a neural network based on the synthetically generated dataset $\mathcal{D}_{\text{meta}} = \{D_1, ..., D_M\}$ consisting of M different environmental conditions subsets D_i with N_k samples

$$D_i = \left\{x_k^{(i)}, y_k^{(i)} = f_k(x_k^{(i)}, u_k^{(i)}) + \epsilon_k^{(i)}, x_{r,k}^{(i)}\right\}_{k=1}^{N_k} \tag{4}$$

where $\epsilon_k^{(i)}$ is the residual obtained by capturing the discrepancy between the discretized known model dynamics $f_k(x_k^{(i)}, u_k^{(i)})$ of (2) and the measured dynamical state. The Deep Neural Network (DNN) model is based on L fully connected layers with element-wise ReLU activation function $g(\cdot) = \max(\cdot, 0)$ and the DNN weights $\Theta = \{W^1, ..., W^{L+1}\}$

$$\phi(x, x_r, \Theta) = W^{L+1}g(W^L g(\cdots g(W^1[x^T, x_r^T]^T)\cdots)) \tag{5}$$

Thus, the meta-learning model of the disturbances is

$$\hat{f}_d(x(t), x_r(t)) = \phi(x(t), x_r(t), \Theta)\hat{a}(t) \tag{6}$$

where $\hat{a}(t)$ is the estimate of $a(w)$. Define the loss function

$$\mathcal{L}(\Theta, \{a_i\}_1^M) = \sum_{i=1}^{M}\sum_{k=1}^{N_k}\left\|\epsilon_k^{(i)} - \phi(x_k^{(i)}, x_{r,k}^{(i)}, \Theta)a_i\right\|_2^2. \tag{7}$$

Learning is performed using stochastic gradient descent (SGD) and spectral normalization. A parameter a_i is introduced for each dataset to capture the specific environmental conditions. Domain-invariant learning [17] ensures the neural network basis remains invariant to these conditions. This approach enables us to employ adaptive control to learn and adapt in real-time to the current conditions encoded with the parameter $\hat{a}$. It also enforces $\|\varphi(x, x_d, x_r)\| \leq \gamma \|x - x_d\|$ where $\varphi(x, x_d, x_r) = \phi(x, x_r) - \phi(x_d, x_r)$, and γ is specified Lipschitz constant for the spectrally normalized DNN [5].

B. Contraction-Based Adaptive Controller (CBAC)

In this section, we present a controller that bounds system behavior to a target trajectory using contraction theory. We first define the error dynamics, then solve a convex optimization problem for the contraction metric, and conclude with a stability theorem and proof. Let us consider the system in (2) and a given target trajectory (x_d, u_d)

$$\dot{x} = f(x) + B(x)u + \phi(x, x_r)a + d(x) \tag{8}$$

$$\dot{x}_d = f(x_d) + B(x_d)u_d(x_d) + \phi(x_d, x_r)\hat{a} \tag{9}$$

where $x, x_d : \mathbb{R}_{\geq 0} \to \mathbb{R}^n$, $u_d : \mathbb{R}^n \to \mathbb{R}^m$, $f_d(x, x_r) = \phi(x, x_r)a$ captures the interaction-produced and wind disturbances, $f_d(x_d, x_r) = \phi(x_d, x_r)\hat{a}$ are the disturbance estimate used to determine the target trajectory, $d(x)$ are the unmodelled remainder of the bounded disturbances with $\bar{d} = \sup_x \|d(x)\|$ and $f : \mathbb{R}^n \to \mathbb{R}^n$, $B : \mathbb{R}^n \to \mathbb{R}^{n \times m}$ are known smooth functions.

Let us define an approximated state-dependent coefficient (SDC) parametrization as $A(x, x_d)$, such that

$$\begin{aligned} f(x) + B(x)u_d - f(x_d) - B(x_d)u_d = \\ A(x, x_d)(x - x_d) + \varepsilon_A(x, x_d) \end{aligned} \tag{10}$$

where $\varepsilon_A(x, x_d)$ is a parametrization error that can be considered as disturbances $d'(x, x_d) = d(x) + \varepsilon_A(x, x_d)$. Then, by choosing the control law

$$u = u_d - K(x, x_d)(x - x_d) - B^\dagger(x)\varphi(x, x_d, x_r)\hat{a} \tag{11}$$

the dynamics in (8) can be equivalently written as

$$\begin{aligned} \dot{x} =& \dot{x}_d + (A(x, x_d) - B(x)K(x, x_d))(x - x_d) \\ &- B(x)B^\dagger(x)\varphi(x, x_d, x_r)\hat{a} + \varphi(x, x_d, x_r)a \\ &- \phi(x_d, x_r)\tilde{a} + d'(x) \end{aligned} \tag{12}$$

where $\tilde{a} = \hat{a} - a$ is the error between the estimate $\hat{a}$ and actual parameter a, $K(x, x_d) = R^{-1}(x, x_d)B^T(x)M(x, x_d)$ is a state-feedback control gain based on the contraction metric $M(x, x_d)$ [7], [25], [26] that will be explained later and a weight matrix $R(x, x_d) \succ 0$, where $R(x, x_d) \succ 0$ denotes that $R(x, x_d)$ is positive definite, $B^\dagger(x) = (B^T(x)B(x))^{-1}B^T(x)$ is the Moore-Penrose inverse of the matrix $B(x)$ which has linearly independent columns and $\varphi(x, x_d, x_r) = \phi(x, x_r) - \phi(x_d, x_r)$. Note that in such a formulation, the disturbances $f_d(x) = \phi(x, x_r)a$ are matched through two parts. First, the function $\varphi(x, x_d, x_r)$ corresponds to the cancellation of the disturbances based on the discrepancy from the target trajectory x_d, and the term $-B(x)B^\dagger(x)\varphi(x, x_d, x_r)\hat{a}$ which corresponds to the online adaptation part that acts through the control input and is fundamentally limited via matrix $B(x)$.

Problem 2: Let $\underline{\omega}, \bar{\omega} \in (0, \infty)$, $\omega_\chi = \bar{\omega}/\underline{\omega}$. Determine the contraction metric $M(x, x_d) = W^{-1}(x, x_d) \succ 0$, with $\lambda_{\min}(W) = \underline{\omega}$, $\lambda_{\max}(W) = \bar{\omega}$, by solving the convex optimization problem for a given value of $\alpha \in (0, \infty)$:

$$\min_{\nu > 0, \omega_\chi \in \mathbb{R}, \bar{W} \succ 0} \omega_\chi \tag{13}$$

subject to the convex constraints

$$-\dot{\bar{W}} + 2\text{sym}(A\bar{W}) - 2\nu BR^{-1}B^T \preceq -2\alpha\bar{W}, \tag{14}$$

$$\partial_{b_j(x)}\bar{W} + \partial_{b_j(x_d)}\bar{W} = 0, \quad j = 1, ..., m \tag{15}$$

$$I \preceq \bar{W} \preceq \omega_\chi I, \tag{16}$$

where $A = A(x, x_d)$ and $B = B(x) = [b_1(x), ..., b_m(x)]$ are the state-dependent coefficients defined in (10), $\bar{W} = \bar{W}(x) = \nu W(x)$, $\nu = 1/\underline{\omega}$, and $\partial_{b_j(x)}\bar{W} = \sum_{i=1}^n \frac{\partial W}{\partial x_i} b_{ij}(x)$ is the notation for directional derivative where $b_{ji}(x)$ is the ith element of the column vector b_j.

Remark 1: We minimize the eigenvalue ratio of W to optimize the transient response and error bounds, while ensuring stability with constraint (14) and relaxing the need for *a priori* knowledge of the closed-loop controller K (15). The structure of matrix M is a non-trivial problem that depends on the considered dynamical system. It can be computed using SOS programming [26] or, alternatively, pointwise for the state space of interest [25], with the results then fitted to a neural network for a representation valid across the entire state space [27]. In Problem 2, α is treated as given, simplifying the optimization. For a fixed convergence rate, this approach obtains an appropriate contraction metric and robust set estimates, as shown in Theorem 1. However, obtaining the optimal contraction rate α and metric M can be achieved through a line search on α [28].

Theorem 1: Suppose there exists the contraction metric $M(x, x_d) \succ 0$ and $M(x, x_d) = W^{-1}(x, x_d)$ obtained by solving Problem 2 for a given value of $\alpha \in (0, \infty)$ and that $\sup \|d'(x, x_d, u_d)\| \leq \bar{d}$. Suppose further that the system is controlled by the following adaptive control law:

$$u = u_d - Ke - B^\dagger\varphi\hat{a} \tag{17}$$

$$\dot{\hat{a}} = -\sigma\hat{a} + P\phi^T R^{-1}(y - \phi\hat{a}) + P(BB^\dagger\varphi)^T Me \tag{18}$$

$$\dot{P} = -2\sigma P + Q - P\phi^T R^{-1}\phi P \tag{19}$$

where $e = x - x_d$, $K = R(x, x_d)^{-1}B(x)^T M(x, x_d)$, $\varphi = \varphi(x, x_d, x_r)$, $\phi = \phi(x, x_r)$, y is the measured discrepancy between the observed error dynamics and the known dynamics, P is the covariance matrix, $Q \succ 0$ is a weight matrix and $\sigma \in \mathbb{R}_{\geq 0}$. If there exists $\bar{\alpha} > 0$ such that

$$-\begin{bmatrix} 2\alpha M & 2M\phi \\ 0 & P^{-1}QP^{-1} + \phi^T R^{-1}\phi \end{bmatrix} \preceq -2\bar{\alpha}\begin{bmatrix} M & 0 \\ 0 & P^{-1} \end{bmatrix} \tag{20}$$

holds, then the error dynamics e are bounded with

$$\|e\| \leq \lambda_{\mathcal{M}} e^{-\bar{\alpha}t}(\|e(0)\| + \|\tilde{a}(0)\|) + \left(1 - e^{-\bar{\alpha}t}\right)\frac{D}{\bar{\alpha}\lambda_{\min}(\mathcal{M})},$$

where

$$D = \sup_t \left\| \begin{bmatrix} M(I-BB^\dagger)\varphi a + Md \\ \phi^T R^{-1}\varepsilon - \sigma P^{-1} a - P^{-1}\dot{a} \end{bmatrix} \right\| \tag{21}$$

and $\varepsilon = y - \phi a$, $\mathcal{M} = \begin{bmatrix} M & 0 \\ 0 & P^{-1} \end{bmatrix}$, $\lambda_{\mathcal{M}} = \sqrt{\frac{\lambda_{\max}(\mathcal{M})}{\lambda_{\min}(\mathcal{M})}}$.

Proof: By satisfying conditions (14) for the matrix $\bar{W}(x, x_d)$, its scaled inverse $M(x, x_d)$ satisfies $\dot{M} + 2\mathrm{sym}(A_K M) \leq -2\alpha M$ (P1) [26], where $A_K = A - BK$. Rewrite the error dynamics $e = x - x_d$ using (12) as

$$\begin{aligned}\dot{e} &= (A-BK)e - BB^\dagger\varphi\hat{a} + \varphi a - \phi\hat{a} + d' \\ &= (A-BK)e - BB^\dagger\varphi\tilde{a} + (I - BB^\dagger)\varphi a - \phi\hat{a} + d'\end{aligned}$$

where $\tilde{a} = \hat{a} - a$. Furthermore, $\dot{\tilde{a}} = \dot{\hat{a}} - \dot{a}$ and using (18)

$$\dot{\tilde{a}} = -\sigma\tilde{a} - \sigma a + P\phi^T R^{-1}(\varepsilon - \phi\tilde{a}) - P(BB^\dagger\varphi)^T M e - \dot{a}$$

where $\varepsilon = y - \phi a$. Similar to [29, Theorem 2], we prove the formal stability and robustness guarantees using the contraction analysis. For a Lyapunov function $V = e^T M e + \tilde{a}^T P^{-1}\tilde{a}$, and using $\frac{\mathrm{d}}{\mathrm{d}t}P^{-1} = -P^{-1}\dot{P}P^{-1} \overset{(19)}{=} 2\sigma P^{-1} - P^{-1}QP^{-1} + \phi^T R^{-1}\phi$ we obtain

$$\begin{aligned}\dot{V} = &\begin{bmatrix} e \\ \tilde{a}\end{bmatrix}^T \begin{bmatrix} \dot{M} + 2\mathrm{sym}(MA_K) & 2MBB^\dagger\varphi - 2M\phi \\ -2(BB^\dagger\varphi)^T M & -P^{-1}QP^{-1} - \phi^T R^{-1}\phi \end{bmatrix} \begin{bmatrix} e \\ \tilde{a}\end{bmatrix} \\ &+ 2\begin{bmatrix} e \\ \tilde{a}\end{bmatrix}^T \begin{bmatrix} M(I-BB^\dagger)\varphi a + Md' \\ \phi^T R^{-1}\varepsilon - \sigma P^{-1}a - P^{-1}\dot{a}\end{bmatrix} \\ &\overset{(P1)}{\leq} -\begin{bmatrix} e \\ \tilde{a}\end{bmatrix}^T \begin{bmatrix} 2\alpha M & 2M\phi \\ 0 & P^{-1}QP^{-1} + \phi^T R^{-1}\phi \end{bmatrix} \begin{bmatrix} e \\ \tilde{a}\end{bmatrix} \\ &+ 2\begin{bmatrix} e \\ \tilde{a}\end{bmatrix}^T \begin{bmatrix} M(I-BB^\dagger)\varphi a + Md' \\ \phi^T R^{-1}\varepsilon - \sigma P^{-1}a - P^{-1}\dot{a}\end{bmatrix}\end{aligned}$$

As $P^{-1}QP^{-1}$, M and P^{-1} are all positive definite and uniformly bounded and $\phi^T R^{-1}\phi$ is positive semidefinite, there exists some $\bar{\alpha} > 0$ such that (20) holds for all t [17]. Then, $\dot{V} \leq -2\bar{\alpha}V + 2\sqrt{\frac{V}{\lambda_{\min}(\mathcal{M})}}D$, where D as in (21). Using the Comparison lemma [30], and $\left\|\begin{bmatrix} e \\ \tilde{a}\end{bmatrix}\right\| \leq \sqrt{\frac{V}{\lambda_{\min}(\mathcal{M})}}$, we obtain $\left\|\begin{bmatrix} e \\ \tilde{a}\end{bmatrix}\right\| \leq \lambda_{\mathcal{M}} e^{-\bar{\alpha}t}\left\|\begin{bmatrix} e(0) \\ \tilde{a}(0)\end{bmatrix}\right\| + (1 - e^{-\bar{\alpha}t})\frac{D}{\bar{\alpha}\lambda_{\min}(\mathcal{M})}$. The final result follows by $\|e\| \leq \left\|\begin{bmatrix} e \\ \tilde{a}\end{bmatrix}\right\| \leq \|e\| + \|\tilde{a}\|$. ∎

Remark 2: A similar problem formulation to Theorem 1 can be found in [29, Theorem 2]. Our work differs by not assuming the matched uncertainty condition $\varphi(x, x_d, x_r)a \in \mathrm{span}(B(x))$, leading to a less conservative stability theorem valid for a broader class of systems. Unmatched disturbances are addressed by generating the target trajectory x_d, accounting for learned disturbances in the optimization problem (Problem 3 below). The covariance matrix P for the adaptation variable $\hat{a}$ is updated analogously as in Kalman-Bucy filter [31]. In practice, we implemented the discretized version of CBAC at a frequency of 100 Hz. At this rate, the discretization error is minimal and should not significantly affect performance or stability guarantees. The control law remains unchanged, while the adaptation law follows the two-step Kalman filter approach [17, Section S4]. This result enables us to further quantify the upper bound (21) as done in Corollary 1.

C. Chance-constrained Upper Bound

Based on the covariance matrix P, and a user-specified small probability of failure $\delta > 0$, $\delta \in (0, 1)$, we can determine the uncertainty sets as

$$\mathcal{S}_P(\hat{a}, P, \delta) := \{a : \|\hat{a} - a\|_P^2 \leq \chi_k^2(1-\delta)\} \tag{22}$$

where $\chi_k^2(p)$ is the Inverse Cumulative Distribution Function (ICDF) of the chi-square distribution with k degrees of freedom, evaluated at the probability values in p.

Corollary 1: Assume that the unknown parameter a, estimated through the adaptation law (18), varies slowly such that $\dot{a} \approx 0$, and that the estimation $\hat{a}$ has reached a steady state. Then D in (21) can be upper bounded on a set $x \in \mathcal{X}$ with

$$D \leq \bar{D} := \frac{\bar{d}}{\omega_\chi} + \bar{\phi}\bar{\varepsilon}\lambda_{\max}(R) + \left(\frac{\bar{b}\bar{\phi}}{\omega_\chi} + \lambda_{\min}(P)\sigma\right)\sup_t \|a\|,$$

where $\bar{b} = \lambda_{\max}(I - BB^\dagger)$, $\bar{\varphi} = \sup_{x\in\mathcal{X}}\|\varphi\|$, $\bar{\phi} = \sup_{x\in\mathcal{X}}\|\phi\|$ and $\bar{\varepsilon} = \sup\|y - f_d\|$ is the upper bound on the measurement noise. Furthermore, a chance-constrained bound can be derived by using $\sup_t\|a\| \leq \|\hat{a}\| + \sqrt{\frac{\chi_k^2(1-\delta)}{\lambda_{\min}(P)}}$.

Remark 3: The assumption that $\dot{a}$ changes slowly (practically zero) is needed to establish an upper bound that can be calculated, as estimating this parameter beforehand is difficult. This assumption holds in practice since wind conditions remain relatively constant during algorithm execution. Due to the non-constant terms in the upper bound that depend on the time-varying matrix P, the corollary is valid only when the estimation has converged. Numerically, $\bar{D}$ can be computed at each MPC (Section III-D) time step. Finally, the initial adaptation error $\|\tilde{a}(0)\|$ in Theorem 1, can be written as $\|\tilde{a}(0)\| \leq \sqrt{\frac{\chi_k^2(1-\delta)}{\lambda_{\min}(P_{\mathcal{D}_{\text{meta}}})}}$ assuming $\hat{a}(0)$ comes from the neural network training step (Section III-A).

Corollary 2: Let $\bar{D}$ be defined as in Corollary 1. Assume that the current wind conditions a have been previously recorded in the dataset $\mathcal{D}_{\text{meta}}$, and that the initial value for the estimated parameter $\hat{a}(0)$ is determined as described in Section III-A. Then, $\|\tilde{a}(0)\| \leq \sqrt{\frac{\chi_k^2(1-\delta)}{\lambda_{\min}(P_{\mathcal{D}_{\text{meta}}})}}$ and the error dynamics can be upper bounded with

$$\begin{aligned}\bar{e}(t, \hat{a}, P, \delta) = &e^{-\bar{\alpha}t}\left(\lambda_{\mathcal{M}}\|e(0)\| + \lambda_{\mathcal{M}}\sqrt{\frac{\chi_k^2(1-\delta)}{\lambda_{\min}(P_{\mathcal{D}_{\text{meta}}})}}\right) \\ &+ \left(1 - e^{-\bar{\alpha}t}\right)\frac{\bar{D}}{\bar{\alpha}\lambda_{\min}(\mathcal{M})},\end{aligned}$$

Remark 4: Corrolary 2 provides a way to compute the error bound, valid for the current conditions, that can be used as an ingredient to obtain safe target trajectories through MPC. This bound certifies safe behaviour near obstacles.

D. Optimal Control Problem as MPC

We formulate the optimal control problem with respect to the tracking objective x_r to determine the target trajectory (x_d, u_d) which will serve as an input to the CBAC.

Problem 3 (ML-MPC): Let the desired states of the system at time t be $x_d(t)$. Given the reference trajectory $x_r(\cdot|t)$, and the estimated error bound $\bar{e}(t, \hat{a}, P, \delta)$ the meta-learning augmented MPC is

$$\min_{u(\cdot|t)} J(\hat{x}_d(\cdot|t), u_d(\cdot|t), x_r(\cdot|t)) \tag{23a}$$

subject to

$$\hat{x}_d(k+1|t) = f_k(\hat{x}_d(k|t), u_d(k|t)), x_r(k|t)), \tag{23b}$$
$$\hat{x}_d(k|t) \in \mathcal{X}, \tag{23c}$$
$$u_d(k|t) \in \mathcal{U}, \tag{23d}$$
$$\hat{x}_d(k|t) \in \mathcal{X}_{\text{safe}}(\bar{e}(t_k, \hat{a}(t), P(t), \delta)), \tag{23e}$$
$$\hat{x}_d(N|t) \in \mathcal{X}_{\text{goal}}(x_r(N|t), \bar{e}(t_N, \hat{a}(t), P(t), \delta)), \tag{23f}$$

for $k = 0, 1, ..., N$, for all $i \in \mathcal{N}$, and $t_k = t + k\Delta t$, where $f_k(x_d, u_d, x_r)$ are discretized dynamics of $\dot{x}_d = f(x_d) + B(x_d)u_d + f_d(x_d, x_r)$.

Set $\mathcal{X}$ denotes the set of system dynamics state constraints, $\mathcal{U}$ the input constraints, $\mathcal{X}_{\text{safe}}(\bar{e}) := \{x \in \mathcal{X} : \{x\} \cap (\mathcal{O} \oplus \mathcal{B}(\bar{e})) = \emptyset\}$ is a safety set based on the obstacles occupying a closed set $\mathcal{O}$ and $\mathcal{B}(\bar{e})$ is a closed ball of radius $\bar{e}$ centered at the origin in $\mathbb{R}^3$; $\mathcal{X}_{\text{goal}}(x_r, \bar{e}) := \{x_d \in \mathcal{X}_{\text{safe}}(\bar{e}) : \|x_d - x_r\| \leq \varepsilon_l\}$ is the terminal set around the final point of x_r and $\varepsilon_l > 0$ is its radius. We define the cost function as

$$J(\hat{x}_d(\cdot|t), u_d(\cdot|t), x_r(\cdot|t)) = \|\hat{x}_d(N|t) - x_r(N|t)\|^2_{Q_m} + \sum_{k=0}^{N-1} \|\hat{x}_d(k|t) - x_r(k|t)\|^2_{Q_m} + \|u_d(k|t)\|^2_{R_m}$$

where the error between the target trajectory x_d and the tracking objective x_r as well as the control effort are penalized throughout the prediction horizon of length N and matrices $Q_m, R_m \succ 0$. The MPC formulation in Problem 3 yields a dynamically feasible and optimal target trajectory (x_d, u_d) that accounts for the disturbance model.

IV. Results

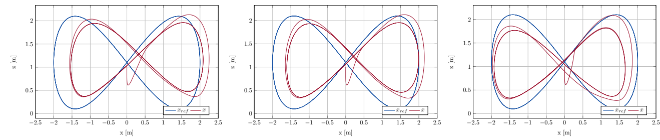
To evaluate the proposed algorithm, we use synthetic wind disturbance data and the ground-effect model presented in [14]. Side disturbances are computed as forces acting on the propellers under constant wind. The network is a four-layer fully connected DNN with ReLU activations, which proved to be effective on similar tasks [16], [17]. Given the temporal separability between the position and attitude dynamics of the quadrotor model, we design the position controller as CBAC and use a geometric controller valid on complete $SO(3)$ [32], [33] for the attitude. The CBAC for the position dynamics of the quadrotor, cast to the form of (2), is derived by considering the reduced state $x = [p^T, v^T, \eta^T]^T \in \mathcal{X} \subseteq \mathbb{R}^6 \times \mathbb{T}$, and input $u = [f_u, \eta_u^T]^T \in \mathcal{U} \subset \mathbb{R} \times \mathbb{T}$ and

$$f(x) = \begin{bmatrix} v \\ -ge_3 \\ 0 \end{bmatrix} B(x) = \begin{bmatrix} 0 & 0 \\ \frac{1}{m}R(\eta)e_3 & 0 \\ 0 & I_3 \end{bmatrix} f_d = \begin{bmatrix} 0 \\ \phi a \\ 0 \end{bmatrix}$$

where $e_3 = [0, 0, 1]^T$ and I_3 is the 3×3 identity matrix. One parametrization of the SDC matrix $A(x, x_d, u_d)$ can be obtained by considering Taylor's expansion of $r_3(\eta) = R(\eta)e_3 = r_3(\eta_d) + J(\eta_d)\tilde{\eta} + \frac{1}{2}\tilde{\eta}^T H(\eta_d)\tilde{\eta} + o(\|\Delta\eta\|^2)$ where $J(\eta_d) = \frac{\partial r_3}{\partial \eta}(\eta_d)$ is Jacobian matrix, and $H(\eta_d) = \frac{\partial J}{\partial \eta}(\eta_d)$ is Hessian tensor, $\tilde{\eta} = \eta - \eta_d$, and $o(\|\tilde{\eta}\|^2)$ is the little-o notation. Thus, for (10) to hold, we choose

$$A(x, x_d, u_d) = \begin{bmatrix} 0 & I_3 & 0 \\ 0 & 0 & \frac{f_u}{m}(J(\eta_d) + \frac{1}{2}\tilde{\eta}^T H(\eta_d)) \\ 0 & 0 & 0 \end{bmatrix} \tag{24}$$

and $\varepsilon_A(x, x_d, u_d) = [0^T, \frac{f_u}{m} o(\|\tilde{\eta}\|^2)^T, 0^T]^T$. The matrix $M(x, x_d, u_d)$ is obtained by solving the convex optimization problem in Problem 2 for a grid of points of the considered state space. We find optimal α using line search and approximate it with the neural network as described in Remark 1.



(a) Nominal MPC, RMSE: 0.29720 (b) MPC+ML-CBAC, RMSE: 0.22470 (c) Our algorithm, RMSE: 0.17714

Fig. 2: Performance comparison of (a) nominal MPC, (b) nominal MPC with a feedback controller consisting of a neural network and an adaptive controller similar to [17] and (c) our algorithm. For fairness, the NNs in cases (b) MPC without and (c) with disturbance knowledge are identical and trained on the same dataset. Disturbances include the ground effect (ground at $z = 0$) and constant crosswind of 12 m/s.

1) Trajectory Tracking in a Figure-8 Pattern. The framework operates in real-time, achieving 100 Hz for CBAC and 5 Hz for MPC solved with IPOPT in CasADi, with the inference taking at most approximately 10 ms on a personal computer running Ubuntu 22.04 with an Intel i7 processor and an NVIDIA RTX A500 CUDA 12-enabled graphics card. Performance is evaluated with Root Mean Square Error (RMSE) relative to the reference trajectory, which is a rotated Figure 8 in the x-z plane. When the MPC is unaware of disturbances, discrepancies persist over multiple periods due to the inability of MPC to account for them. Adding a feedback controller does not improve the performance as it is limited by how well it can track the desired trajectory x_d generated by the MPC. Performance improves when the MPC incorporates knowledge of the disturbance model.

2) Autonomous Soft Landing. We focus on a specific problem of autonomous UAV landing [34]–[36]. In this setup, we assess whether the algorithm achieves a smooth landing, which means approaching the ground $z = 0$ as smoothly as possible. Figure 3 demonstrates the ability of the algorithm to compensate for the ground effect and land smoothly. It is worth noting that the algorithm in both examples uses the same NN trained on data from the first example.

V. Conclusion

This paper presented a meta-learning augmented MPC algorithm for disturbance-aware motion planning and control, guaranteed to improve performance with respect to desired state-input trajectories. The results highlight the

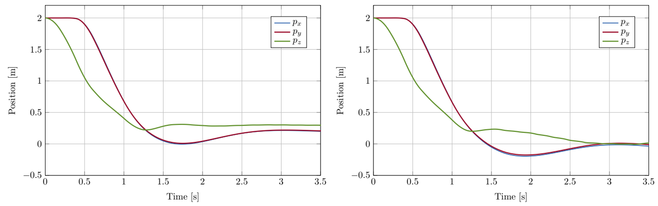


Fig. 3: (left) Nominal MPC fails to compensate, leaving a 25 cm error; (right) our algorithm compensates and lands.

importance of disturbance-aware planning for more accurate and reliable behavior. By incorporating disturbances into the planning loop, the algorithm improves robustness and trajectory tracking. Future work will focus on testing the control scheme in real-world experiments, extending it to explore other interaction-produced disturbances in unknown environments, and applying it to multi-agent model sharing.

References

[1] Aleksandar Bauranov and Jasenka Rakas. Designing airspace for urban air mobility: A review of concepts and approaches. *Progress in Aerospace Sciences*, 125:100726, 2021.

[2] Mohammad Sadra Rajabi, Pedram Beigi, and Sina Aghakhani. *Drone Delivery Systems and Energy Management: A Review and Future Trends*, pages 1–19. Springer International Publishing, Cham, 2021.

[3] Mingyang Lyu, Yibo Zhao, Chao Huang, and Hailong Huang. Unmanned aerial vehicles for search and rescue: A survey. *Remote Sensing*, 15(13), 2023.

[4] IC Cheeseman and WE Bennett. The Effect of the Ground on a Helicopter Rotor in Forward Flight. *Reports and Memoranda No. 3021, September, 1995*, 1955.

[5] Guanya Shi, Xichen Shi, Michael O'Connell, Rose Yu, Kamyar Azizzadenesheli, Animashree Anandkumar, Yisong Yue, and Soon-Jo Chung. Neural Lander: Stable Drone Landing Control Using Learned Dynamics. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 9784–9790. IEEE, 2019.

[6] Karl J Åström and Björn Wittenmark. *Adaptive Control*. Courier Corporation, 2013.

[7] Winfried Lohmiller and Jean-Jacques E Slotine. On Contraction Analysis for Non-Linear Systems. *Automatica*, 34(6):683–696, 1998.

[8] Drew Hanover, Philipp Foehn, Sihao Sun, Elia Kaufmann, and Davide Scaramuzza. Performance, Precision, and Payloads: Adaptive Nonlinear MPC for Quadrotors. *IEEE Robotics and Automation Letters*, 7(2):690–697, 2021.

[9] Guillem Torrente, Elia Kaufmann, Philipp Föhn, and Davide Scaramuzza. Data-Driven MPC for Quadrotors. *IEEE Robotics and Automation Letters*, 6(2):3769–3776, 2021.

[10] Matthias Faessler, Antonio Franchi, and Davide Scaramuzza. Differential Flatness of Quadrotor Dynamics Subject to Rotor Drag for Accurate Tracking of High-Speed Trajectories. *IEEE Robotics and Automation Letters*, 3(2):620–626, 2017.

[11] Rohan Sinha, James Harrison, Spencer M Richards, and Marco Pavone. Adaptive Robust Model Predictive Control With Matched and Unmatched Uncertainty. In *2022 American Control Conference (ACC)*, pages 906–913. IEEE, 2022.

[12] Vladimir Khromov and Omri Rand. Ground Effect Modeling for Rotary-Wing Simulation. In *26th International Congress of the Aeronautical Sciences*, pages 1–10. International Council of the Aeronautical Sciences Germany, 2008.

[13] Xiang He, Marc Calaf, and Kam K Leang. Modeling and Adaptive Nonlinear Disturbance Observer for Closed-Loop Control of in-Ground-Effects on Multi-Rotor UAVs. In *Dynamic Systems and Control Conference*, volume 58295, page V003T39A004. American Society of Mechanical Engineers, 2017.

[14] Li Danjun, Zhou Yan, Shi Zongying, and Lu Geng. Autonomous Landing of Quadrotor Based on Ground Effect Modelling. In *2015 34th Chinese control conference (CCC)*, pages 5647–5652. IEEE, 2015.

[15] Guanya Shi, Wolfgang Hönig, Yisong Yue, and Soon-Jo Chung. Neural-Swarm: Decentralized Close-Proximity Multirotor Control Using Learned Interactions. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3241–3247. IEEE, 2020.

[16] Guanya Shi, Wolfgang Hönig, Xichen Shi, Yisong Yue, and Soon-Jo Chung. Neural-Swarm2: Planning and Control of Heterogeneous Multirotor Swarms Using Learned Interactions. *IEEE Transactions on Robotics*, 38(2):1063–1079, 2021.

[17] Michael O'Connell, Guanya Shi, Xichen Shi, Kamyar Azizzadenesheli, Anima Anandkumar, Yisong Yue, and Soon-Jo Chung. Neural-Fly Enables Rapid Learning for Agile Flight in Strong Winds. *Science Robotics*, 7(66), 2022.

[18] Fengze Xie, Guanya Shi, Michael O'Connell, Yisong Yue, and Soon-Jo Chung. Hierarchical meta-learning-based adaptive controller. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 18309–10315, 2024.

[19] Spencer M Richards, Navid Azizan, Jean-Jacques Slotine, and Marco Pavone. Adaptive-Control-Oriented Meta-Learning for Nonlinear Systems. *arXiv preprint arXiv:2103.04490*, 2021.

[20] James Harrison, Apoorva Sharma, and Marco Pavone. Meta-Learning Priors for Efficient Online Bayesian Regression. In *Algorithmic Foundations of Robotics XIII: Proceedings of the 13th Workshop on the Algorithmic Foundations of Robotics 13*, pages 318–337. Springer, 2020.

[21] Guanya Shi, Kamyar Azizzadenesheli, Michael O'Connell, Soon-Jo Chung, and Yisong Yue. Meta-Adaptive Nonlinear Control: Theory and Algorithms. *Advances in Neural Information Processing Systems*, 34:10013–10025, 2021.

[22] Christopher D McKinnon and Angela P Schoellig. Meta Learning With Paired Forward and Inverse Models for Efficient Receding Horizon Control. *IEEE Robotics and Automation Letters*, 6(2):3240–3247, 2021.

[23] Christos K Verginis, Zhe Xu, and Ufuk Topcu. Non-parametric neuro-adaptive control. In *2023 European control conference (ECC)*, pages 1–6. IEEE, 2023.

[24] Elena Sorina Lupu, Fengze Xie, James A. Preiss, Jedidiah Alindogan, Matthew Anderson, and Soon-Jo Chung. Magicvfm -meta-learning adaptation for ground interaction control with visual foundation models. *IEEE Transactions on Robotics*, pages 1–20, 2024.

[25] Hiroyasu Tsukamoto and Soon-Jo Chung. Robust Controller Design for Stochastic Nonlinear Systems via Convex Optimization. *IEEE Transactions on Automatic Control*, 66(10):4731–4746, 2020.

[26] Ian R Manchester and Jean-Jacques E Slotine. Control Contraction Metrics: Convex and Intrinsic Criteria for Nonlinear Feedback Design. *IEEE Transactions on Automatic Control*, 62(6):3046–3053, 2017.

[27] Hiroyasu Tsukamoto and Soon-Jo Chung. Neural Contraction Metrics for Robust Estimation and Control: A Convex Optimization Approach. *IEEE Control Systems Letters*, 5(1):211–216, 2020.

[28] Sumeet Singh, Benoit Landry, Anirudha Majumdar, Jean-Jacques Slotine, and Marco Pavone. Robust Feedback Motion Planning via Contraction Theory. *The International Journal of Robotics Research*, 42(9):655–688, 2023.

[29] Hiroyasu Tsukamoto, Soon-Jo Chung, and Jean-Jacques Slotine. Learning-Based Adaptive Control Using Contraction Theory. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 2533–2538. IEEE, 2021.

[30] Hassan K Khalil. *Nonlinear Systems; 3rd Ed.* Prentice-Hall, 2002.

[31] R. E. Kalman and R. S. Bucy. New Results in Linear Filtering and Prediction Theory. *Journal of Basic Engineering*, 83(1):95–108, 03 1961.

[32] Taeyoung Lee, Melvin Leok, and N Harris McClamroch. Geometric Tracking Control of a Quadrotor UAV on SE(3). In *49th IEEE conference on decision and control (CDC)*, pages 5420–5425. IEEE, 2010.

[33] Saptarshi Bandyopadhyay, Soon-Jo Chung, and Fred Y Hadaegh. Nonlinear Attitude Control of Spacecraft With a Large Captured Object. *Journal of Guidance, Control, and Dynamics*, 39(4):754–769, 2016.

[34] Linnea Persson and Bo Wahlberg. *Model Predictive Control for Autonomous Ship Landing in a Search and Rescue Scenario*. AIAA, 2019.

[35] Dženan Lapandić, Linnea Persson, Dimos V. Dimarogonas, and Bo Wahlberg. Aperiodic Communication for MPC in Autonomous Cooperative Landing. *IFAC-PapersOnLine*, 54(6):113–118, 2021. 7th IFAC Conference on Nonlinear Model Predictive Control NMPC 2021.

[36] Linnea Persson, Anders Hansson, and Bo Wahlberg. An optimization algorithm based on forward recursion with applications to variable horizon mpc. *European Journal of Control*, 75:100900, 2024.