2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Unified Meta-Representation and Feedback Calibration for General Disturbance Estimation

Zihan Yang[1], Jindou Jia[2], Meng Wang[2], Yuhang Liu[1], Kexin Guo[1,3†] and Xiang Yu[2,3]

Abstract— **Precise control in modern robotic applications is always an open issue due to unknown time-varying disturbances. Existing meta-learning-based approaches require a shared representation of environmental structures, which lack flexibility for realistic non-structural disturbances. Besides, representation error and the distribution shifts can lead to heavy degradation in prediction accuracy. This work presents a generalizable disturbance estimation framework that builds on meta-learning and feedback-calibrated online adaptation. By extracting features from a finite time window of past observations, a unified representation that effectively captures general non-structural disturbances can be learned without predefined structural assumptions. The online adaptation process is subsequently calibrated by a state-feedback mechanism to attenuate the learning residual originating from the representation and generalizability limitations. Theoretical analysis shows that simultaneous convergence of both the online learning error and the disturbance estimation error can be achieved. Through the unified meta-representation, our framework effectively estimates multiple rapidly changing disturbances, as demonstrated by quadrotor flight experiments. See the project page for video, supplementary material and code: `https://nonstructural-metalearn.github.io`.**

I. Introduction

Modern intricate robotic systems can be confronted with dynamic and complex environments that introduce unknown disturbances in the nominal case, including latent dynamics variation and other external disturbances. To retain high control precision, such disturbances must be handled properly. Several classical methods, such as Disturbance Observer (DO) [1], Incremental Nonlinear Dynamic Inversion (INDI) [2] and $\mathcal{L}_1$ adaptive control [3], [4], directly estimate the lumped disturbance, but are limited by the trade-off between disturbance estimation lag and noise amplification. The utilization of the feedforward model can greatly improve the estimation performance with a specific type of disturbance. Multi-model-based estimation of multiple, heterogeneous, and isomeric disturbances is studied in [5] with remarkable results. Recently, adaptive control with a meta-learning scheme has allowed robots to learn from prior experiences and rapidly adapt to new environments [6]–[8]. Such frameworks enable the rapid estimation of environmental disturbances with shared features, benefiting from the power of offline learning and online adaptation.

†Corresponding author (kxguo@buaa.edu.cn)
[1]School of Aeronautic Science and Engineering, Beihang University, Beijing 100191, China.
[2]School of Automation Science and Electric Engineering, Beihang University, Beijing 100191, China.
[3]Hangzhou Innovation Institute of Beihang University, Hangzhou, 310051, China.

Despite their promising results, several drawbacks are worth mentioning. Firstly, these methods are designed for disturbances with shared structural representation. The disturbances are assumed to be functions of the state representation and an environment configuration, e.g., the wind disturbances [7], [8]. Nevertheless, the premise of the existence of shared representation may not hold in a general unstructured environment, e.g., sudden external forces or coupled effects with unknown disturbances. Moreover, some well-conditioned environments are tricky to construct in real-world data collection for domain-invariant meta-learning. Secondly, the learning residuals are not explicitly considered in existing frameworks. The approximation capability of the meta-learning approach is guaranteed by collecting information in various cases [9], [10]. However, generalization degrades when the environment lies outside the support of the training task distribution. While meta-learned representations are trained on disturbances from a specific distribution, real-world disturbances may not match this distribution, leading to a performance drop.

Motivated by these challenges, we propose a feedback-calibrated meta-adaptive framework to aid generalization from both the representation and calibration perspectives. Initially, a unified meta-representation that captures general disturbance effects is constructed using a finite time window of past observations, without requiring any predefined structural assumptions. Domain randomization in simulation generates diverse disturbances using random Fourier series [11], [12], enabling strong generalization for both simulation and real-world tasks. Next, a feedback-calibrated mechanism is incorporated to correct prediction errors and attenuate residuals arising from representation limitations and generalizability loss, ensuring reliable online adaptation and disturbance estimation. To demonstrate the effectiveness of the proposed method, we incorporate the framework with a baseline controller and construct extensive empirical studies on the trajectory tracking control of a quadrotor under multiple rapid-changing disturbances. Such disturbances include aerodynamic drag, unknown external forces, unknown fixed and suspended payloads, as well as external wind disturbances. The results indicate that using a unified meta-representation with online feedback calibration can generalize to such non-structural disturbances, also with significant improvements over the state-of-the-art approaches. The main contributions of this work can be summarized as follows:

- A unified meta-representation for modelling general non-structural disturbances without predefined struc-

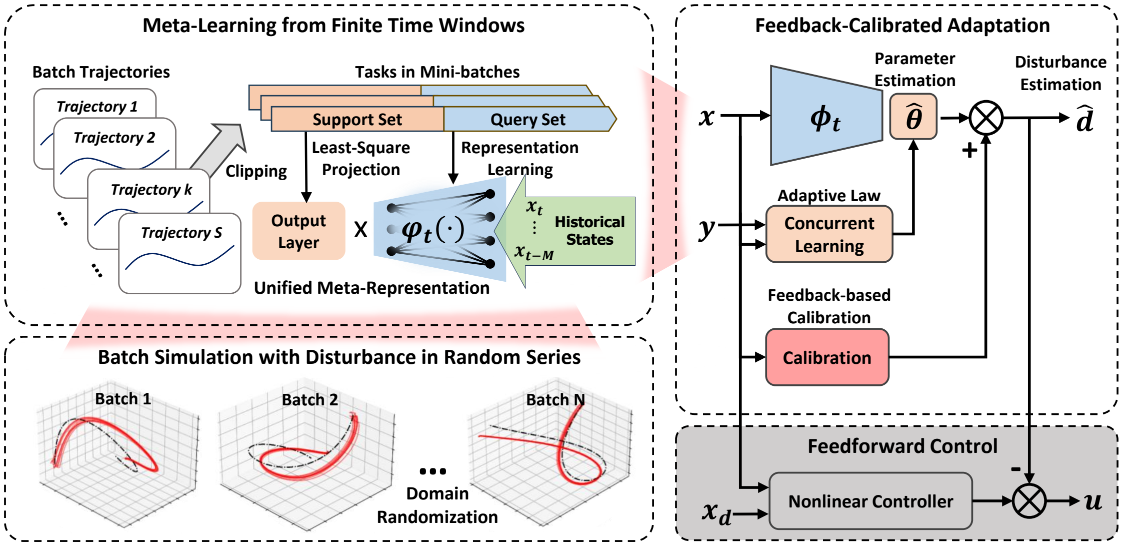

Fig. 1. Schematic of the proposed framework. The meta-representation is learned from a finite time window of past observations with domain-randomized disturbances. The online adaptation is calibrated with a feedback mechanism to attenuate the learning residual, which can be further integrated with a baseline controller for disturbance rejection.

tural assumptions.

- A feedback-calibrated mechanism for the attenuation of the meta-adaptation residual, thereby improving the model accuracy and generalizability.
- Theoretical analysis for the simultaneous convergence of the online learning error and the disturbance estimation error.

II. Related Works

A. Offline Meta-Learning

Meta-learning aims at learning a representation from previous experiences that can adapt to new tasks quickly [13], [14]. In the context of model-based control, meta-learning has been applied to adaptive control [7], [8], [15] and model-based reinforcement learning (MBRL) [6], [16] with promising results. Meta-learners can be designed in various ways, including optimizing hyperparameters of the base-learner and/or learning a representation (usually a neural network) for the adaptation task of the base-learner. A domain-adversarial invariant approach is proposed to learn a shared spatial representation of different wind disturbances with constant wind speeds [8]. A control-oriented method is presented for end-to-end learning of the adaptation policy [7]. Model predictive control (MPC) with adaptation is constructed for fast adaptation to new conditions [17]. Learning representations with structures from bilinear models to deep neural networks (DNNs) has been studied [18]. An intriguing approach learns to adapt the control gains concerning various environments, which is augmented with a Kalman filter for parameter learning [19]. Despite the effectiveness of structural disturbances, these methods require a shared spatial representation among varied disturbances or tasks, which is unsuitable for non-structural disturbances, e.g., the joint effects of external gust disturbance and unknown payload. A framework that refines the tasks into previous trajectory segments and learns both the base-learner hyperparameter and the representation has been proposed [6], enabling MBRL on different failure modes and loading conditions of legged robots and manipulators. A hierarchical framework [15] that enables the capture of unmanageable environmental disturbances can alleviate such an issue. Our method also focuses on representation learning in a smooth and continuous manner, and demonstrates that a powerful unified representation can be yielded based on finite-time observations and domain randomization. To alleviate the difficulties of data collection for meta-learning in previous methods, a simulation-based domain randomization is utilized to collect various non-structural disturbances for representation learning.

B. Online Adaptation

In the online adaptation phase of meta-learning, the model parameter is updated with upcoming data. In the context of model-based control, online adaptation is usually achieved by adaptive laws [20], [21] in adaptive controllers and stochastic gradient descent [6], [13], [22] in MBRL or MPC approaches. The composite adaptive law consists of multiple sources of feedback to boost the convergence [7], [8], [15]. In real-world applications, the utilization of Kalman filters [8], [19] enables parameter learning from noisy measurements. However, none of the existing adaptation methods explicitly considered the learning residual that can originate from representation error and distribution shifts. We propose a feedback-calibrated mechanism that attenuates the learning residual that favors the final estimation performance.

III. Problem Formulation

We consider a control affine system, a general case for rigid-body robots:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})\boldsymbol{u} + \boldsymbol{d}(\boldsymbol{x}, \boldsymbol{h}(t)) \quad (1)$$

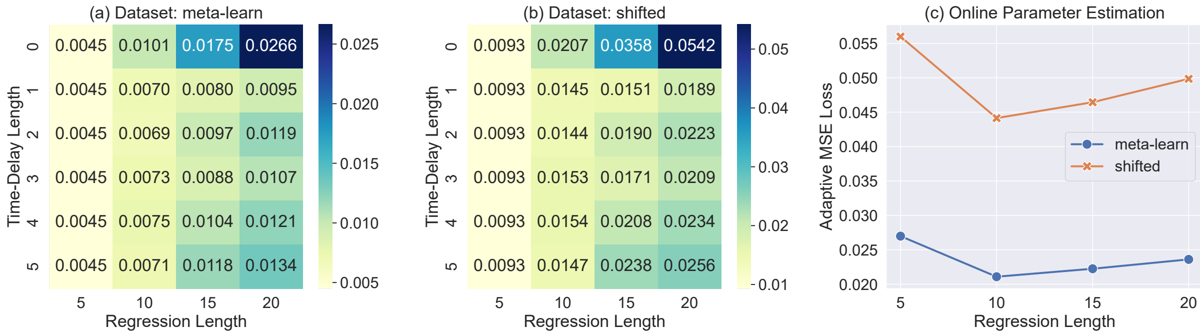

Fig. 2. The result of ablation study, including model performance (prediction loss in mean squared error) on *meta-learn* dataset (a), *shifted* dataset (b) and the effect of online parameter estimation (c).

where $\boldsymbol{x} \in \mathbb{R}^n$ and $\dot{\boldsymbol{x}} \in \mathbb{R}^n$ are the state and its derivative, respectively. $\boldsymbol{u} \in \mathbb{R}^m$ is the control input and $\boldsymbol{f}(\boldsymbol{x})$ and $\boldsymbol{g}(\boldsymbol{x})$ are continuously differentiable mappings. The disturbance $\boldsymbol{d}(\cdot) \in \mathbb{R}^n$ originates from state-related internal effects and external time-varying environmental impact $\boldsymbol{h}(t) \in \mathbb{R}^h$. Our key objective is to represent general $\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{h}(t))$ based on a unified meta-representation that covers the influence of $\boldsymbol{h}(t)$ and can adapt to new disturbances. As shown in [7], [8], [15], we formulate the disturbance as:

$$\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{h}(t)) = \boldsymbol{\Xi}\boldsymbol{\varphi}(\boldsymbol{x}, \boldsymbol{h}(t)) + \boldsymbol{\epsilon} \tag{2}$$

where $\boldsymbol{\Xi} \in \mathbb{R}^{n \times k}$ a general online-learned model parameter, $\boldsymbol{\varphi}(\cdot) : \mathbb{R}^n \times \mathbb{R}^h \to \mathbb{R}^k$ is the shared representation among different disturbance structures. $\boldsymbol{\epsilon} \in \mathbb{R}^n$ refers to the learning residual, indicating that the formulation is imperfect under the representation error and model generalizability loss. Previous approaches assume that $\boldsymbol{h}(t)$ is a constant or slow-varying environment configuration represented by $\boldsymbol{\Xi}$, such as wind speed or payload mass, then $\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{h}(t))$ is approximated with the formulation $\boldsymbol{\Xi}\boldsymbol{\varphi}(\boldsymbol{x})$. This limits the representation in a structured pattern characterized by $\boldsymbol{\varphi}(\boldsymbol{x})$.

Unlike previous methods [7], [8], [15], which extract a global, state-dependent structure from the entire disturbance space, we propose a unified representation for unstructured disturbances that encodes a sequence of historical states to implicitly address the influence of $\boldsymbol{h}(t)$ in $\boldsymbol{\varphi}(\cdot)$. Intuitively, the presence of disturbances is reflected in the sequential state variations within a finite-time window, preserving temporal variability and enabling flexible adaptation to general non-structural disturbances. Additionally, a feedback calibration mechanism is introduced to handle the learning residual, enhancing model accuracy and generalizability.

IV. Methods

A. Meta-learning for Non-Structural Disturbances

For notational convenience, we reformulate $\boldsymbol{\Xi}\boldsymbol{\varphi}(\cdot)$ as $\boldsymbol{\phi}(\cdot)\boldsymbol{\xi}$, where $\boldsymbol{\xi} = [\Xi_1, \cdots, \Xi_k]^\top \in \mathbb{R}^{nk}$ is formed by stacking the rows of $\boldsymbol{\Xi}$, and $\boldsymbol{\phi}(\cdot) = \text{diag}\{\boldsymbol{\varphi}(\cdot)^\top, \cdots, \boldsymbol{\varphi}(\cdot)^\top\} \in \mathbb{R}^{n \times nk}$ denotes the block-diagonal matrix with $\boldsymbol{\varphi}(\cdot)^\top$ repeated n times along the diagonal.

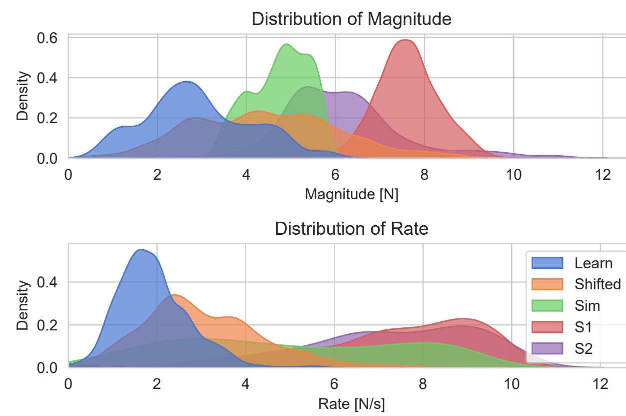

Fig. 3. Distribution differences of disturbances in the dataset of *meta-learn* (Learn), *shifted* (Shifted), simulations (Sim) and real-world scenarios (S1 and S2).

1) Unified Meta-Representation: Given a disturbance model in the form of (2), we aim to learn a meta-representation for $\boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{h}(t))$ without the knowledge of $\boldsymbol{h}(t)$. Inspired by time-delay embedding [23], a meta-learned representation that encodes a historical state segment into a latent space for disturbance prediction is introduced. Existing approaches have shown that the time-delay embedding can be utilized for disturbance and dynamics modeling [24], [25]. The disturbance model is reformulated as:

$$\boldsymbol{d}(\boldsymbol{x}, \boldsymbol{h}(t)) = \boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{h}(t))\boldsymbol{\xi} + \boldsymbol{\epsilon} = \boldsymbol{\phi}_t(\boldsymbol{z})\boldsymbol{\theta} + \boldsymbol{\gamma} \tag{3}$$

where $\boldsymbol{z}_k = [\boldsymbol{x}_k, \boldsymbol{x}_{k-1}, \cdots, \boldsymbol{x}_{k-M}]$ and M is the embedding dimension. $\boldsymbol{\theta} \in \mathbb{R}^{nk}$, $\boldsymbol{\phi}_t(\cdot) : \mathbb{R}^{nM} \to \mathbb{R}^{n \times nk}$ and $\boldsymbol{\gamma} \in \mathbb{R}^n$ are the model parameter, the meta-learned representation and learning residual under time-delay embedding, respectively.

2) Learning from Finite Time Window: Generally, the meta-learning problem is achieved by bi-level optimization. The inner base-learning problem learns the model parameter for adaptation based on the task-specified support set $\mathcal{D}_{support}$ and a given representation. The outer meta-learning problem learns an optimal representation for the adaptation to new tasks (query set) $\mathcal{D}_{query}$. The meta-learning algorithm is designed for future prediction based on past data-based adaptation. In particular, a task $\mathcal{D}$ is defined as multiple trajectory segments $\mathcal{D} = \{\mathcal{D}^1, \cdots, \mathcal{D}^{N_m}\}$, where $\mathcal{D}^i =$

$\{\boldsymbol{x}^i_{1:N+M+H}, \bar{\boldsymbol{d}}^i_{1:N+M+H}\}$ is the state-disturbance sequence. $\bar{\boldsymbol{d}}$ is the real disturbance that can be obtained from simulation or offline-filtered real-world data. $\mathcal{D}^i_{support}$ and $\mathcal{D}^i_{query}$ are divided from $\mathcal{D}^i$ and refer to the past N and future H state-disturbance sequences, respectively. The base-learner is a regularized least-squares method that learns the model parameter $\boldsymbol{\theta}^*$ with a given $\boldsymbol{\phi}_t$ from past N state-disturbance sequence. We note that $\boldsymbol{\phi_t}(\cdot)$ is parameterized by $\boldsymbol{\eta}$. For the meta-learner, the objective is to learn an optimal $\boldsymbol{\eta}$ with the base-learned $\boldsymbol{\theta}^*$ for the prediction of future H disturbances:

$$\begin{aligned} &\min_{\boldsymbol{\eta}} \frac{1}{H}\sum_{k=1}^{H}\frac{1}{2}\|\boldsymbol{\phi}_t(\boldsymbol{z}_k)\boldsymbol{\theta}^* - \bar{\boldsymbol{d}}_{kU}\|_1^2 + \lambda_1\|\boldsymbol{\phi}_t\|_1, \\ &\text{s.t. } \boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \frac{1}{N}\sum_{k=1}^{N}\frac{1}{2}\|\boldsymbol{\phi}_t(\boldsymbol{z}_k)\boldsymbol{\theta} - \bar{\boldsymbol{d}}_{kL}\|_2^2 + \lambda_2\|\boldsymbol{\theta}\|_2^2. \end{aligned} \tag{4}$$

where $\bar{\boldsymbol{d}}_{kU} \in D^i_{query}$ and $\bar{\boldsymbol{d}}_{kL} \in D^i_{support}$. λ_1, λ_2 are the L1 and L2 regularization parameters that encourage simpler representation. The inner problem is with the closed-form solution:

$$\boldsymbol{\theta}^* = (\boldsymbol{\Phi}^\top\boldsymbol{\Phi} + \lambda_2\boldsymbol{I})^{-1}\boldsymbol{\Phi}^\top\bar{\boldsymbol{\Delta}} \tag{5}$$

where $\boldsymbol{\Phi} = [\boldsymbol{\phi}_t(\boldsymbol{z}_1), \cdots, \boldsymbol{\phi}_t(\boldsymbol{z}_N)]^\top$ and $\bar{\boldsymbol{\Delta}} = [\bar{\boldsymbol{d}}_1, \cdots, \bar{\boldsymbol{d}}_N]^\top$ are the concatenated vectors for regression. The original bi-level optimization problem (4) is therefore a single layer one.

Algorithm 1 Meta-Learning from Segments

Input: Base-learner regression size N, time-delay-embedding size M, mini-batch size N; dataset $\mathcal{D}$, objective function J in (4).
Result: Representation $\boldsymbol{\phi}_t$.
Initialize: Representation parameters $\boldsymbol{\eta}$; slice $\mathcal{D}$ into N_m segments $\mathcal{D} = \{\mathcal{D}^1, \cdots, \mathcal{D}^{N_m}\}$ with length H each, $\mathcal{D}^i = \{\boldsymbol{x}^i_k, \bar{\boldsymbol{d}}^i_k\}$, $i = 1, \cdots, N_m$.

```
1: repeat
2:   for {x^i_{1:H}, d̄^i_{1:H}} in {D_{i=1,...,N_m}} do
3:     Compute Gradient ∇_η J based on (4);
4:     Compute step Δη using Adam or other methods;
5:     Update η with η ← η + Δη;
6:   end for
7: until convergence
```

The mini-batching technique is applied for rolling out all trajectory segments with gradient descent on the parameter of $\boldsymbol{\phi}_t$ based on the bi-level optimization (4). Since it is defined for a single trajectory segment $\mathcal{D}^i$, mini-batch learning can be applied to the complete dataset $\mathcal{D}$. The representation learning algorithm (Algorithm 1) is lightweight and computationally efficient, completing training within a few hours on a laptop equipped with an NVIDIA RTX 4060 GPU, depending on the value of N and the size of the neural network. The real-time computation of the least-squares method with large parameter dimensions can be heavy for limited-sourced robots. Therefore, the $\boldsymbol{\theta}^*$ is estimated with adaptive laws in the online adaptation phase. More details on the meta-learning can be found in the Supplementary Material I-C.

3) Ablation Study on Model Parameters: Two datasets are constructed for the ablation study. Besides *meta-learn* set for training, *shifted* set is loaded with disturbances containing larger magnitudes and faster rates compared with that of *meta-learn*. As shown in Figure.2, the time-delay embedding with $M = 3$ outperforms the one with $M = 1$ in both datasets and further improves as the regression length N increases. While a smaller N leads to better model accuracy, the online performance is degraded if online parameter estimation (8) is applied instead of directly solving the least-squares problem (5). A trade-off between the convergent speed and the model performance can be obtained with $N = 10$. More details on the meta-learned models can be found in the Supplementary Material I-A

4) Domain Randomization for Meta-learning: To generalize to arbitrary non-structural disturbances, $\mathcal{D}$ is constructed by simulations. Domain randomization [11], [12] is applied in a batch simulator to construct non-structural disturbances in random series with different magnitudes and rates under various closed-loop conditions. Despite the abundance of simulated cases, the model performance can be damaged by the representation error and distribution shifts. In the next section, we introduce a feedback-calibrated mechanism that tackles such issues while doing online parameter estimation. See the Supplementary Material I-B for more information.

B. Feedback-Calibrated Online Adaptation

In attempt to tamp the learning residual $\boldsymbol{\gamma}$, our framework is furthermore augmented with the feedback-calibration phase by online learning the model parameter $\boldsymbol{\theta}$. To construct the online adaptation algorithm, we start with a feedback-based correction that can be seen in the design of luenberger observers [26] and disturbance observers [1].

$$\dot{\hat{\boldsymbol{d}}} = \dot{\boldsymbol{d}}_{model} + \boldsymbol{L}(\boldsymbol{d} - \hat{\boldsymbol{d}}) \tag{6}$$

where $\boldsymbol{L}$ is the positive-definite feedback gain, $\hat{\boldsymbol{d}}$ is the estimated disturbance, $\dot{\boldsymbol{d}}_{model}$ refers to the derivative of the disturbance model $\boldsymbol{\phi}_t(\boldsymbol{z})\boldsymbol{\theta}$ and $\boldsymbol{d}$ is the real disturbance. With $\boldsymbol{d} = \dot{\boldsymbol{x}} - \boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{g}(\boldsymbol{x})\boldsymbol{u}$, the feedback-calibrated mechanism is designed as:

$$\begin{aligned} \dot{\boldsymbol{\xi}} &= -\boldsymbol{L}(\boldsymbol{\xi} + \boldsymbol{d}_{model} + \boldsymbol{L}\boldsymbol{x} + \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})\boldsymbol{u}) \\ \hat{\boldsymbol{d}} &= \boldsymbol{d}_{model} + \boldsymbol{L}\boldsymbol{x} + \boldsymbol{\xi} \end{aligned} \tag{7}$$

where $\boldsymbol{\xi} \in \mathbb{R}^n$ is an auxiliary variable to avoid the usage of state derivative. Such calibration brings us the following benefit with the assumption on the learning residual.

Assumption 1: In the disturbance model $\boldsymbol{d} = \boldsymbol{\phi}_t(\boldsymbol{z})\boldsymbol{\theta}^* + \boldsymbol{\gamma}$, the learning residual $\boldsymbol{\gamma}$ is bounded smooth, i.e., $\|\boldsymbol{\gamma}\| \leq \bar{\gamma}$ and $\|\dot{\boldsymbol{\gamma}}\| \leq \bar{d}_\gamma$, where $\boldsymbol{\theta}^*$ is the optimal model parameter, $\bar{\gamma} > 0$ and $\bar{\boldsymbol{d}}_{\boldsymbol{\gamma}} > 0$ are positive constants.

Theorem 1: Under Assumption 1, with $\boldsymbol{\theta} = \boldsymbol{\theta}^*$, the disturbance estimation error $\tilde{\boldsymbol{d}} = \hat{\boldsymbol{d}} - \boldsymbol{d}$ exponentially converges to a bounded set regularized by $\boldsymbol{L}$ and $\bar{d}_\gamma$.

Proof. refers to the Supplementary Material II-A. The learning residual $\boldsymbol{\gamma}$ is therefore tamped by the feedback-based

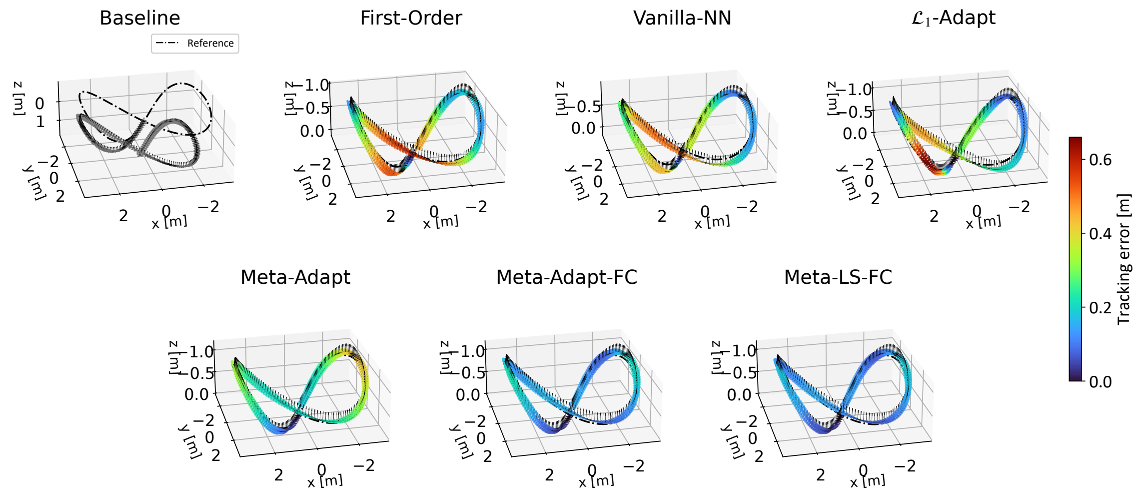

Fig. 4. Trajectory tracking results of the simulated cases.

correction mechanism, reducing to $\bar{d}_\gamma$ instead of $\bar{\gamma}$. The knowledge of $\dot{\boldsymbol{x}}$ with integration in (7) comes with the accessibility of real-time $\boldsymbol{d}$ feedback with $\boldsymbol{x}$.

As illustrated in adaptive control approaches [7], [8], the estimation model parameter $\boldsymbol{\theta}$ is achieved using concurrent learning adaptive laws [27], [28]:

$$\dot{\hat{\boldsymbol{\theta}}} = -\boldsymbol{P}\sum_{i=1}^{N_c}\boldsymbol{\phi}_t(\boldsymbol{z}_i)^\top(\boldsymbol{d}_i - \boldsymbol{\phi}_t(\boldsymbol{z})\hat{\boldsymbol{\theta}}) + \boldsymbol{\Gamma}\boldsymbol{\phi}_t(\boldsymbol{z})^\top(\boldsymbol{x}_d - \boldsymbol{x}) \quad (8)$$

where $\boldsymbol{P}$, $\boldsymbol{\Gamma}$ are the positive-definite gains, N_c is the number of concurrent learning samples, $\boldsymbol{d}_i$ is the disturbance measurement for $\boldsymbol{\theta}$-estimation. In real-world applications, Kalman filter-based estimation [8] or other direct filtering methods can be used to address the noise of $\boldsymbol{d}_i$. An assumption can be made for the online adaptation.

Assumption 2: The optimal model parameter $\boldsymbol{\theta}^*$ is slow time-varying, i.e., $\|\dot{\boldsymbol{\theta}}^*\| \le \bar{d}_\theta$. The representation is bounded in its magnitude and derivative, i.e., $\|\boldsymbol{\phi}_t(\boldsymbol{z})\| \le \bar{\phi}$, $\|\dot{\boldsymbol{\phi}}_t(\boldsymbol{z})\| \le \bar{d}_\phi$, $\bar{d}_\theta > 0$ and $\bar{d}_\theta > 0$ are positive constants.

With saturation functions, $\|\boldsymbol{\phi}_t(\boldsymbol{z})\| \le \bar{\phi}$ can be easily satisfied. Theorem 2 holds for online parameter learning and disturbance estimation without feedforward control i.e. $\boldsymbol{\Gamma} = \boldsymbol{0}$.

Theorem 2: Under Assumption 1 and Assumption 2, both the disturbance estimation error $\tilde{\boldsymbol{d}} = \hat{\boldsymbol{d}} - \boldsymbol{d}$ and the parameter estimation error $\tilde{\boldsymbol{\theta}} = \hat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*$ exponentially converges to bounded sets.

Proof. refers to the Supplementary Material II-B. With a general feedback control law with $\boldsymbol{g}(\boldsymbol{x})$ in full-rank: $\boldsymbol{u} = \boldsymbol{g}(\boldsymbol{x})^{-1}(-\boldsymbol{f}(\boldsymbol{x}) + \dot{\boldsymbol{x}}_d + \boldsymbol{K}(\boldsymbol{x} - \boldsymbol{x}_d) - \hat{\boldsymbol{d}})$ and positive-definite gains of $\boldsymbol{\Gamma}$ and $\boldsymbol{K}$, the closed-loop system also ends up with asymptotic stability. The full-rank assumption on $\boldsymbol{g}(\boldsymbol{x})$ indicates that the system is fully-actuated, but for underactuated systems, the feedback control can be achieved using cascade controllers. Here we skip the proof since it is similar to the proof of Theorem 2.

V. Empirical Study

A. Simulated Experiments

In this part, the proposed framework is validated in a simulated quadrotor under mass uncertainty and aerodynamic drag. From Figure.3, the disturbances covered for learning is insufficient to represent the testing disturbances, yet the model is expected to generalize well due to the unified representation and feedback-calibrated mechanism. The quadrotor dynamics, controller configuration, disturbances, model parameters, and online adaptation settings can be found in the Supplementary Material III.

The abbreviations and explanations of the compared methods are as follows. First-order disturbance observer (**First-Order**) [1] can be seen as a special case of the feedback-calibration mechanism, where $\dot{\boldsymbol{d}}_{model} = \boldsymbol{0}$. Neural-network augmented disturbance observer (**Vanilla-NN**) is embedded with learned the aerodynamic effects $\boldsymbol{RDR}^\top\boldsymbol{v}$ that favors the disturbance estimation. $\mathcal{L}_1$-Adaptive Control (**$\mathcal{L}_1$-Adapt**) [4] estimates the disturbance via velocity feedback and a low-pass filter. **Meta-Adapt** refers to the proposed meta-learned model with a classic composite adaptive law as in [8]. **Meta-Adapt-FC** refers to the proposed meta-learned model with the feedback-calibrated online adaptation. **Meta-LS-FC** has the model parameter that is directly optimized by the base-learner (5) while enabling feedback calibration.

TABLE I
RMSE of Disturbance Estimation (m/s^2) and Trajectory Tracking Control (m).

Method	Estimation Loss	Control Loss
First-Order	0.738	0.209
Vanilla-NN	0.493	0.199
$\mathcal{L}_1$-Adapt	0.799	0.167
Meta-Adapt	0.245	0.158
Meta-Adapt-FC	**0.159**	**0.083**
Meta-LS-FC	**0.151**	**0.074**

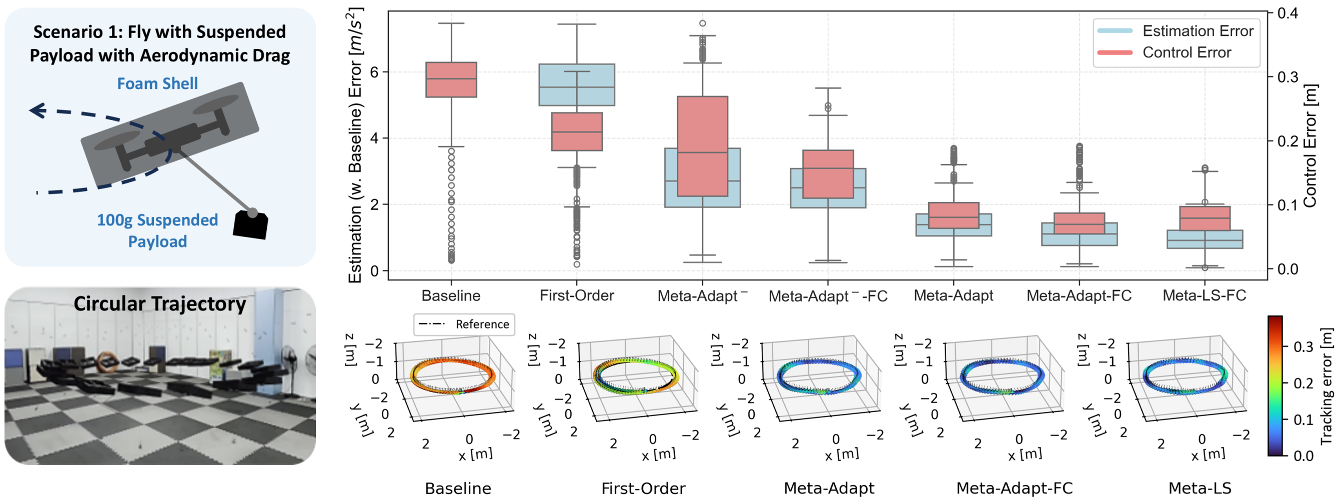

Fig. 5. Scenario.1, the quadrotor maneuvers in a circular trajectory with a suspended payload and aerodynamic drag. Boxplots of both estimation error and tracking error are provided. 3D trajectories are colored by the tracking error.

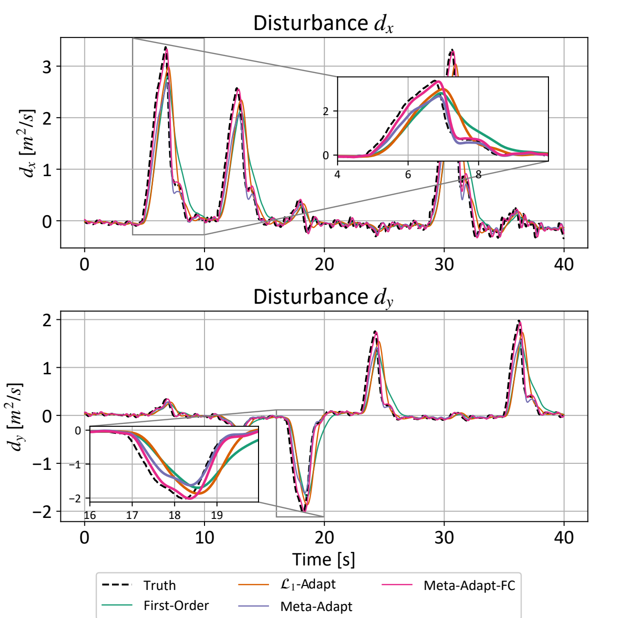

Fig. 6. The results of real-world push-rod force estimation in simulations.

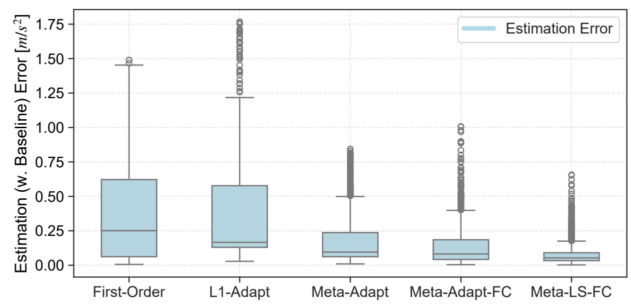

Fig. 7. The boxplots of real-world push-rod force estimation error.

For control tasks, the estimators are incorporated into the translational loop of a DFBC [29], [30] baseline controller and servers as a feedforward compensation on the desired acceleration. For estimation tasks, the estimator works with the baseline without feedforward compensation. The results are evaluated by root-mean-square error (RMSE) in Table.I. The baseline controller fails under the disturbances with the tracking RMSE of 0.705. **Vanilla-NN** outperforms the **First-Order** and **$\mathcal{L}_1$-Adapt** in estimation since the aerodynamic drag model is captured accurately. As shown in Figure.4, our adaptive approaches significantly boost the estimation and control performance. With the least-squares method **Meta-LS-FC**, the online adaptation is achieved without the convergent process, which ends with the highest estimation and tracking control performance. The feedback-calibration mechanism provides model correction, leading to an enhancement of 47.5% in the control task and 35.1% in estimation. Disturbance estimation plots can be found in the Supplementary Material III-D.

B. Real-world Experiments

In this section, the proposed framework is validated on a quadrotor platform with three challenging tasks involving non-structural disturbances. Our approach generalizes well to different scenarios using a single meta-representation under the distribution shift of real-world disturbances, as shown in Figure.3. The quadrotor uses a motion capture system and an onboard IMU for state and acceleration measurements, respectively. For the details of the platform, controller, model parameters, and the disturbance estimation plots, see the Supplementary Material IV.

1) Estimation of Push-Rod Forces: The proposed method is first validated on a quadrotor platform with a push-rod force estimation task. The sensor measurements of sudden force injections are collected, followed by force estimation in simulations. In this section, we provide the details of the external force estimation. The quadrotor undergoes a series of external forces provided manually via a push-rod, as shown in Figure.9. The IMU measurement of acceleration is collected from the quadrotor, which is then processed by an offline Gaussian Filter. In the simulated environment, the

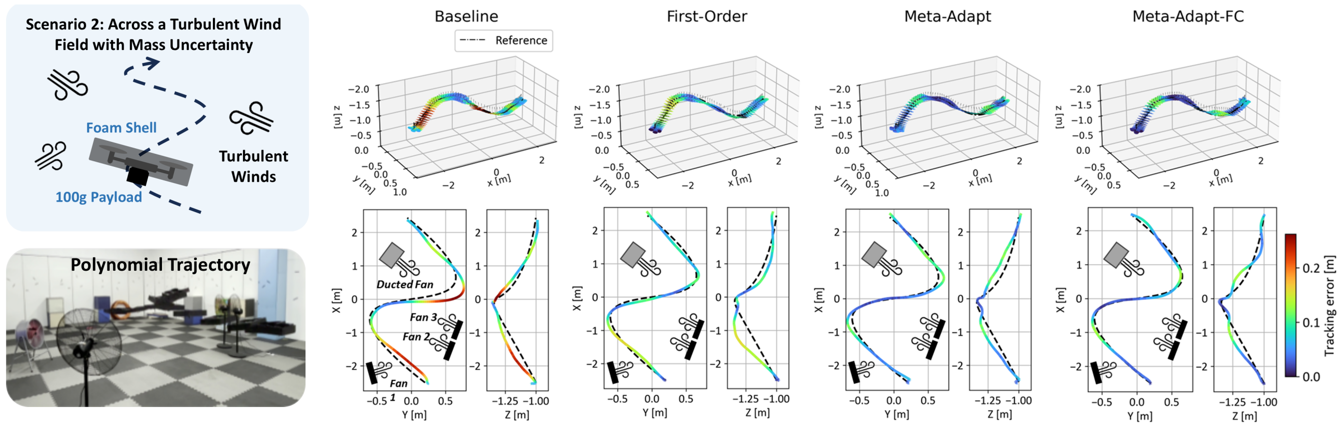

Fig. 8. Scenario.2, the quadrotor is commanded to fly across a turbulent wind field with mass uncertainty. Trajectories are plotted with control error in 3D view and top view with the location of wind sources marked.

Fig. 9. The quadrotor is disturbed by a series of external forces via a push-rod.

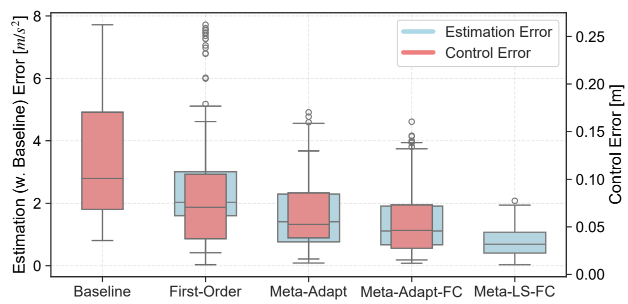

Fig. 10. Boxplots of estimation and control error in Scenario.2.

quadrotor is commanded to hover at a fixed position and the force is injected into the quadrotor in the earth-fixed frame. The results of the estimation and the boxplots of the force estimation are provided in Figure.6 and Figure.7, respectively. Our approach shows the least lag in disturbance estimation, even when the disturbances are completely state-independent and non-structural.

2) Scenario.1: Fly with Suspended Payload and Aerodynamic Drag: As shown in Figure.5, the quadrotor carries a suspended payload of $0.1\,\mathrm{kg}$ (about 12% of its mass) and is wrapped with a foam shell to increase aerodynamic drag. We compare two meta-learned models with different regression lengths: one trained with $N = 10$ and another with $N = 20$ (denoted by the superscript $^-$). As shown in the boxplots in Figure 5, the unified meta-representation enables effective estimation of coupled disturbances, even though such conditions were not present during training. While the quality of the representation (the regression length) is crucial for the online adaptation, the feedback calibration mechanism further improves overall performance and robustness. The least-squares method leads to the highest estimation performance but triggers an oscillation in tracking control. This can be explained as the low-level attitude controller fails to respond to the rapid change of desired acceleration due to the actuator delay.

3) Scenario.2: Across a Turbulent Wind Field with Mass Uncertainty: This scenario tests the method with a quadrotor flying through a turbulent wind field while carrying a $0.1\,\mathrm{kg}$ payload and foam shell. As shown in Figure 8, the wind field is created by multiple fans with varying speeds and directions. The quadrotor tracks a polynomial trajectory across the wind field, and control errors are shown with color-coded lines. As shown in Figure.10, our method managed to estimate the complex gust disturbances with mass uncertainty, which was never seen during training. The proposed feedback-calibration portion contributes to the model performance with a significant reduction in both the estimation error and control error. The model adaptation performance is further improved with the least-squares method in estimation, the corresponding control task is not included due to potential oscillation in the strong winds. Ablation study on the adaptation gain $\boldsymbol{P}$ can be found in Supplementary Material IV.

VI. Conclusion

We proposed a feedback-calibrated meta-adaptive framework that enables general non-structural disturbance prediction. By combining the meta-learned model with a feedback-calibrated online adaptation mechanism, the proposed approach effectively captures general non-structural disturbances. Theoretical analysis guarantees convergence of both model residuals and parameter estimation errors, while extensive simulations and real-world experiments on quadrotor platforms validate the superiority of the method in trajectory tracking and disturbance rejection. The proposed framework offers a generalizable solution for control and estimation of robotic systems in general disturbed environments, bridging adaptive control and estimation from structured assumptions to more realistic generalizations.

VII. Limitations and Open Problems

The accessibility of acceleration. When acceleration measurements are unavailable, online adaptation can still proceed using the control error or state estimation error by augmenting the proposed framework into a state observer, where the latter one makes the whole framework into a dual estimation one.

Trade-off between regression length and online adaptation loss. The proposed method is capable of achieving a good trade-off between model performance and online adaptation loss. The model performance is improved with a lower regression length N, but the online adaptation loss is increased as the parameter estimator fails to track the optimal parameter in time. As in [7], an end-to-end learning approach that learns the representation under the dynamical effect of parameter estimation can be worth studying.

The effect of the low-level controller. In real-world applications, the model performance is improved with the least-squares method but the tracking control performance is degraded. The low-level controller can fail to respond to the rapid change of feedforward compensation. Feedforward control that favors the low-level controller can be a potential future work.

Extension to other robotic systems. Further extension can be made to more diverse robotic platforms and tasks, including legged locomotion and manipulation under contact-rich environments. Meta-learning of joint-level disturbances can be a promising direction for future work.

References

[1] D.-X. Gao, "Disturbance attenuation and rejection for systems with nonlinearity via successive approximation approach," in *Proceedings of the 30th Chinese Control Conference*, 2011, pp. 250–255.

[2] F. Nan, S. Sun, P. Foehn, and D. Scaramuzza, "Nonlinear MPC for quadrotor fault-tolerant control," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 5047–5054, 2022.

[3] N. Hovakimyan and C. Cao, *L1 Adaptive Control Theory*. Society for Industrial and Applied Mathematics, 2010.

[4] K. Huang, R. Rana, A. Spitzer, G. Shi, and B. Boots, "DATT: Deep adaptive trajectory tracking for quadrotor control," in *7th Annual Conference on Robot Learning*, 2023.

[5] L. Guo and S. Cao, "Anti-disturbance control theory for systems with multiple disturbances: A survey," *ISA Transactions*, vol. 53, no. 4, pp. 846–849, 2014, disturbance Estimation and Mitigation.

[6] I. Clavera, A. Nagabandi, S. Liu, R. S. Fearing, P. Abbeel, S. Levine, and C. Finn, "Learning to adapt in dynamic, real-world environments through meta-reinforcement learning," in *International Conference on Learning Representations*, 2019.

[7] S. M. Richards, N. Azizan, J.-J. E. Slotine, and M. Pavone, "Adaptive-control-oriented meta-learning for nonlinear systems," in *Robotics: Science and Systems*, 2021.

[8] M. O'Connell, G. Shi, X. Shi, K. Azizzadenesheli, A. Anandkumar, Y. Yue, and S.-J. Chung, "Neural-fly enables rapid learning for agile flight in strong winds," *Science Robotics*, vol. 7, no. 66, p. eabm6597, 2022.

[9] A. Fallah, A. Mokhtari, and A. E. Ozdaglar, "Generalization of model-agnostic meta-learning algorithms: Recurring and unseen tasks," in *Advances in Neural Information Processing Systems*, 2021.

[10] A. G. Khoee, Y. Yu, and R. Feldt, "Domain generalization through meta-learning: A survey," *Artificial Intelligence Review*, vol. 57, no. 10, p. 285, 2024.

[11] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 23–30.

[12] X. B. Peng, M. Andrychowicz, W. Zaremba, and P. Abbeel, "Sim-to-real transfer of robotic control with dynamics randomization," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 3803–3810.

[13] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proceedings of the 34th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, D. Precup and Y. W. Teh, Eds., vol. 70. PMLR, 2017, pp. 1126–1135.

[14] T. Hospedales, A. Antoniou, P. Micaelli, and A. Storkey, "Meta-learning in neural networks: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 9, pp. 5149–5169, 2022.

[15] F. Xie, G. Shi, M. O'Connell, Y. Yue, and S.-J. Chung, "Hierarchical meta-learning-based adaptive controller," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 18 309–10 315.

[16] S. Belkhale, R. Li, G. Kahn, R. McAllister, R. Calandra, and S. Levine, "Model-based meta-reinforcement learning for flight with suspended payloads," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1471–1478, Apr. 2021.

[17] C. D. McKinnon and A. P. Schoellig, "Meta learning with paired forward and inverse models for efficient receding horizon control," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 3240–3247, 2021.

[18] G. Shi, K. Azizzadenesheli, M. O'Connell, S.-J. Chung, and Y. Yue, "Meta-adaptive nonlinear control: Theory and algorithms," in *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, Eds., vol. 34. Curran Associates, Inc., 2021, pp. 10 013–10 025.

[19] H. Sanghvi, S. Folk, and C. J. Taylor, "OCCAM: Online continuous controller adaptation with meta-learned models," in *8th Annual Conference on Robot Learning*, 2024.

[20] J.-J. E. Slotine and W. Li, "Composite adaptive control of robot manipulators," *Automatica*, vol. 25, no. 4, pp. 509–519, 1989.

[21] A. M. Annaswamy and A. L. Fradkov, "A historical perspective of adaptive control and learning," *Annual Reviews in Control*, vol. 52, pp. 18–41, 2021.

[22] A. Saviolo, J. Frey, A. Rathod, M. Diehl, and G. Loianno, "Active learning of discrete-time dynamics for uncertainty-aware model predictive control," *IEEE Transactions on Robotics*, vol. 40, pp. 1273–1291, 2024.

[23] F. Takens, "Detecting strange attractors in turbulence," in *Dynamical Systems and Turbulence, Warwick 1980*, D. Rand and L.-S. Young, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 1981, pp. 366–381.

[24] T. Z. Jiahao, K. Y. Chee, and M. A. Hsieh, "Online dynamics learning for predictive control with an application to aerial robots," *Conference on Robot Learning (CoRL)*, 2022.

[25] F. Piqué, H. T. Kalidindi, L. Fruzzetti, C. Laschi, A. Menciassi, and E. Falotico, "Controlling soft robotic arms using continual learning," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 5469–5476, 2022.

[26] D. Luenberger, "An introduction to observers," *IEEE Transactions on Automatic Control*, vol. 16, no. 6, pp. 596–602, 1971.

[27] R. Kamalapurkar, B. Reish, G. Chowdhary, and W. E. Dixon, "Concurrent learning for parameter estimation using dynamic state-derivative estimators," *IEEE Transactions on Automatic Control*, vol. 62, no. 7, pp. 3594–3601, 2017.

[28] G. Chowdhary and E. Johnson, "Concurrent learning for convergence in adaptive control without persistency of excitation," in *49th IEEE Conference on Decision and Control (CDC)*, 2010, pp. 3674–3679.

[29] D. Mellinger and V. Kumar, "Minimum snap trajectory generation and control for quadrotors," in *2011 IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 2520–2525.

[30] B. Morrell, M. Rigter, G. Merewether, R. Reid, R. Thakker, T. Tzanetos, V. Rajur, and G. Chamitoff, "Differential flatness transformations for aggressive quadrotor flight," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 5204–5210.