Adaptive Model Predictive Control for High-Accuracy Trajectory Tracking in Changing Conditions

Karime Pereida and Angela P. Schoellig

***Abstract*— Robots and automated systems are increasingly being introduced to unknown and dynamic environments where they are required to handle disturbances, unmodeled dynamics, and parametric uncertainties. Robust and adaptive control strategies are required to achieve high performance in these dynamic environments. In this paper, we propose a novel adaptive model predictive controller that combines model predictive control (MPC) with an underlying $\mathcal{L}_1$ adaptive controller to improve trajectory tracking of a system subject to unknown and changing disturbances. The $\mathcal{L}_1$ adaptive controller forces the system to behave in a predefined way, as specified by a reference model. A higher-level model predictive controller then uses this reference model to calculate the optimal reference input based on a cost function, while taking into account input and state constraints. We focus on the experimental validation of the proposed approach and demonstrate its effectiveness in experiments on a quadrotor. We show that the proposed approach has a lower trajectory tracking error compared to non-predictive, adaptive approaches and a predictive, non-adaptive approach, even when external wind disturbances are applied.**

I. INTRODUCTION

Robots and automated systems are increasingly being introduced to unknown and dynamic environments, which requires the design of sophisticated control methods that can achieve high overall performance in these situations. Unlike traditional controllers where small changes in the conditions may significantly deteriorate the controller performance and may cause instability (see [1], [2], and [3]), controllers deployed in changing environments must be robust to model uncertainties, unknown disturbances, and changing dynamics. Robotic applications in these increasingly challenging environments include autonomous driving, assistive robotics, and unmanned aerial vehicle (UAV) applications. For example, in airborne package delivery, packages to be delivered can have different properties (mass, center of gravity, and inertia) and the vehicles may encounter different weather conditions. As such, it is not feasible to design a controller for each environmental or system condition.

In this paper, we present a controller that achieves high-accuracy tracking performance and is robust to unknown disturbances and changing dynamics. We consider nonlinear systems and propose a novel architecture that combines $\mathcal{L}_1$ adaptive control and model predictive control (MPC), see Fig. 1. MPC is an attractive control scheme because

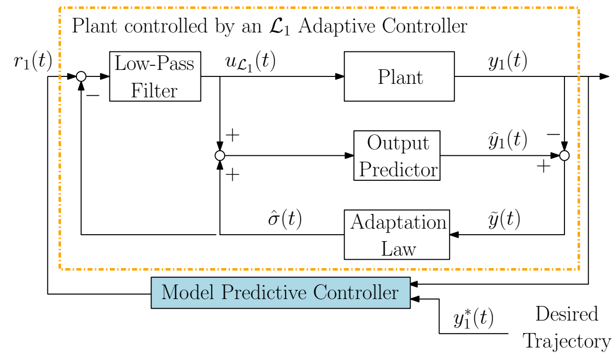

Fig. 1. The proposed adaptive model predictive control architecture. The underlying $\mathcal{L}_1$ adaptive controller forces the system to behave close to a specified linear reference model even in the presence of uncertainties and disturbances. Using the specified linear reference model, the model predictive controller calculates at each time step an input to the plant controlled by the $\mathcal{L}_1$ adaptive controller (which now behaves as the specified reference model) such that the trajectory tracking error is minimized.

constraints can easily be incorporated and because of its predictive capability. It is an optimal control scheme; however, its performance depends on the accuracy of the model used in the optimization. To overcome this, we propose an underlying $\mathcal{L}_1$ adaptive controller, which forces a potentially nonlinear system to behave in the same predefined way, as specified by a linear reference model, even in the presence of model uncertainties. As a result, a standard linear MPC is able to achieve high-accuracy trajectory tracking of the overall system. We validate the proposed approach through extensive experiments on a quadrotor. We show that the predictive capabilities of MPC and the robustness to disturbances of the $\mathcal{L}_1$ adaptive control enable the proposed approach to achieve better trajectory tracking performance compared to non-predictive and/or non-adaptive approaches, especially when external, unknown disturbances are present.

Model predictive control solves an optimization problem at each time step, based on a model of the system and the current state of the system, to find an optimal control sequence that minimizes the given objective function subject to input and state constraints. The first input value of that sequence is applied to the system. In the next time step, the optimization problem is solved again using the new state of the system. MPC has been widely applied in practice, where satisfaction of constraints is important [4]. The standard implementation of MPC using a nominal model of the system dynamics exhibits nominal robustness to small disturbances [5]. However, these robustness guarantees may be insufficient in practical situations where larger disturbances may be present.

To handle model uncertainties, adaptive MPC schemes

The authors are with the Dynamic Systems Lab (www.dynsyslab.org) at the University of Toronto Institute for Aerospace Studies (UTIAS), Canada. Email: k.pereidaperez@mail.utoronto.ca, schoellig@utias.utoronto.ca

This research was supported in part by the Mexican National Council of Science and Technology (abbreviated CONACYT).

Sub version. Accepted at *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems.*

have been introduced, which update the model online based on measurement data. In [6], an adaptive MPC method for a class of constrained linear, time-invariant systems is proposed. This approach is based on a novel method to estimate the parameters of a model suitable for MPC. The algorithm is initially conservative due to large parameter uncertainties, but performance is improved over time as the parameter estimate's uncertainty is reduced and the estimate converges to the true value. However, the quality of the estimation depends on the excitation of the state. Similar adaptive approaches have been proposed for nonlinear systems; however, these are challenging to design and computationally expensive. An adaptive nonlinear MPC scheme for constrained nonlinear systems is presented in [7]. It uses min-max nonlinear MPC in combination with an adaptation mechanism to address the issue of robustness while the estimated value is evolving. Once again, the algorithm is initially conservative due to large parameter uncertainties, and the conservativeness reduces as the parameter estimate improves. However, this approach is still computationally expensive. Finally, learning-based MPC approaches have used neural networks (see [8] and [9]) or Gaussian processes (see [10] and [11]) to learn the dynamics of the system used in the controller. These approaches require a significant amount of data in order to build an accurate model and often do not adapt to changes in the environment in real time and at high update rates.

Further, $\mathcal{L}_1$ adaptive control is based on the model reference adaptive control (MRAC) architecture but includes a low-pass filter that decouples robustness from adaptation [12]. As a result, arbitrarily high adaptation gains can be chosen for fast adaptation. It has successfully been applied to fixed-wing vehicles [13], [14], quadrotors [15], [16], a NASA AirSTAR flight test vehicle [17], a tailless unstable aircraft [18], and hexacopter and octocopter vehicles [19]. The $\mathcal{L}_1$ adaptive controller forces a system, which may be nonlinear and subject to model uncertainties, to behave as a predefined linear reference model. Leveraging this characteristic, it was used in combination with iterative learning control (ILC), where ILC enabled the system to improve trajectory tracking over iterations [16]. ILC computes the input for the next iteration offline after the previous iteration is completed. In contrast to [16], in this work, we calculate the inputs online using MPC, which enables the system to achieve high-accuracy trajectory tracking on the first iteration.

The key contributions of this paper are:

- to demonstrate the effectiveness of a combined $\mathcal{L}_1$ adaptive control and a linear MPC approach to control nonlinear systems with reduced computational cost as compared to nonlinear MPC schemes;
- to remove the need for persistent excitation to achieve accurate adaptation as in existing adaptive MPC strategies [20];
- to validate the novel $\mathcal{L}_1$ adaptive control and MPC approach through extensive experimental results;
- to demonstrate the robustness of the proposed approach to external disturbances, (e.g. wind), as shown in experimental results with a quadrotor;
- to demonstrate the improved tracking performance on a quadrotor compared to two non-predictive and adaptive frameworks, and a predictive and non-adaptive scheme.

The remainder of this paper is organized as follows. The problem is defined in Section II. The details of the proposed approach are presented in Section III. Experimental results on a quadrotor are presented in Section IV. Conclusions are provided in Section V.

II. PROBLEM STATEMENT

The objective of this work is to achieve high-accuracy trajectory tracking in the presence of uncertain, and possibly changing conditions on the first iteration. Consider a system whose dynamics ('Plant' block in Fig. 1) are unknown, can change over time, and can be described by a single-input, single-output (SISO) system (this approach is later extended to multi-input, multi-output (MIMO) systems) identical to [12] for output feedback:

$$y_1(s) = A(s)(u_{\mathcal{L}_1}(s) + d_{\mathcal{L}_1}(s)), \tag{1}$$

where $y_1(s)$ is the Laplace transform of the output $y_1(t)$, $A(s)$ is a strictly-proper unknown transfer function that can be stabilized by a proportional-integral controller, $u_{\mathcal{L}_1}(s)$ is the Laplace transform of the input signal, and $d_{\mathcal{L}_1}(s)$ is the Laplace transform of the disturbance signal defined as $d_{\mathcal{L}_1}(t) \triangleq f(t, y_1(t))$, where $f : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ is an unknown map subject to the following assumption:

Assumption 1 (Lipschitz continuity)**.** *There exist constants $L > 0$ and $L_0 > 0$, such that the following inequalities hold uniformly for all t:*

$$|f(t, v) - f(t, w)| \leq L|v - w| \text{ , and} \tag{2}$$

$$|f(t, w)| \leq L|w| + L_0 \text{ .} \tag{3}$$

The system is required to accurately track a desired trajectory $y_1^*(t)$ defined over a finite time interval. Given $y_1^*(t)$, we aim to compute $u_{\mathcal{L}_1}$ such that the system (1) is able to achieve high-accuracy trajectory tracking at a low computational cost. To achieve this, we propose a control architecture as shown in Fig. 1, where a model predictive controller is used on top of an underlying $\mathcal{L}_1$ adaptive controller.

III. METHODOLOGY

We consider two main subsystems: the $\mathcal{L}_1$ adaptive controller (orange dashed box in Fig. 1) and the model predictive controller. The $\mathcal{L}_1$ adaptive controller is presented in Section III-A. Section III-B introduces the MPC. The combined $\mathcal{L}_1$ adaptive control and MPC framework is described in Section III-C.

A. $\mathcal{L}_1$ Adaptive Control

The $\mathcal{L}_1$ adaptive controller is used to make the system behave as a predefined, linear system even when unknown, changing disturbances act on the system. In this work, the standard $\mathcal{L}_1$ adaptive output feedback controller for SISO systems [12] is used. For convenience and completeness, we present a brief summary of the $\mathcal{L}_1$ adaptive control scheme.

Problem Formulation: The output feedback $\mathcal{L}_1$ adaptive controller computes a control input $u_{\mathcal{L}_1}(t)$ such that $y_1(t)$ tracks the bounded piecewise continuous reference input $r_1(t)$ according to the following first-order reference system:

$$M(s) = \frac{m}{s+m}\,, \quad m > 0\,. \tag{4}$$

Definitions and $\mathcal{L}_1$-Norm Condition: We can rewrite system (1) in terms of the reference system (4) as follows:

$$y_1(s) = M(s)(u_{\mathcal{L}_1}(s) + \sigma(s))\,, \tag{5}$$

where uncertainties in $A(s)$ and $d_{\mathcal{L}_1}(s)$ are combined into $\sigma(s)$:

$$\sigma(s) \triangleq \frac{(A(s) - M(s))u_{\mathcal{L}_1}(s) + A(s)d_{\mathcal{L}_1}(s)}{M(s)}\,. \tag{6}$$

The low-pass filter $C(s)$ (see Fig. 1) is assumed to be strictly proper with $C(0) = 1$, with:

$$H(s) \triangleq \frac{A(s)M(s)}{C(s)A(s) + (1 - C(s))M(s)} \quad \text{being stable,} \tag{7}$$

and the following $\mathcal{L}_1$-norm condition being satisfied:

$$\|G(s)\|_{\mathcal{L}_1} L < 1\,, \quad \text{where } G(s) \triangleq H(s)(1 - C(s))\,, \tag{8}$$

where L is the Lipschitz constant defined in Assumption 1.

The above assumption holds for cases when $A(s)$ can be stabilized by a proportional-integral controller, as shown in [12]. The equations that describe the implementation of the $\mathcal{L}_1$ output feedback architecture are presented below.

Output Predictor: The output predictor is used to estimate the system output based on the reference model (5):

$$\dot{\hat{y}}_1(t) = -m\hat{y}_1(t) + m(u_{\mathcal{L}_1}(t) + \hat{\sigma}(t))\,, \quad \hat{y}_1(0) = 0\,,$$

where $\hat{\sigma}(t)$ is the estimate of $\sigma(t)$. In the Laplace domain,

$$\hat{y}_1(s) = M(s)(u_{\mathcal{L}_1}(s) + \hat{\sigma}(s))\,. \tag{9}$$

Adaptation Law: The following adaptation law is used to update the estimate $\hat{\sigma}(t)$:

$$\dot{\hat{\sigma}}(t) = \Gamma\,\mathrm{Proj}(\hat{\sigma}(t), -\tilde{y}(t))\,, \quad \hat{\sigma}(0) = 0\,, \tag{10}$$

where $\tilde{y}(t) \triangleq \hat{y}_1(t) - y_1(t)$, $\Gamma \in \mathbb{R}^+$ is the adaptation rate subject to the lower bound specified in [12]. To ensure fast adaptation, the adaptation rate is chosen to be very large. The projection operator Proj$(\cdot)$, as defined in [12], guarantees that the estimated value of $\sigma(t)$ remains within a specified convex set.

Control Law: The control input is calculated by passing the difference between the desired trajectory signal $r_1(t)$ and the adaptive estimate $\hat{\sigma}(t)$ through a low-pass filter:

$$u_{\mathcal{L}_1}(s) = C(s)(r_1(s) - \hat{\sigma}(s))\,. \tag{11}$$

This means that the $\mathcal{L}_1$ adaptive controller compensates only for low-frequency uncertainty components of $A(s)$ and $d_{\mathcal{L}_1}(s)$, i.e., the frequencies that the system is able to counteract. The low-pass filter attenuates high-frequency components.

Performance Bounds: We choose the $\mathcal{L}_1$ adaptive controller as it provides performance bounds on the estimation and output errors. The closed-loop reference system for the $\mathcal{L}_1$ adaptive controller is described as follows:

$$y_{1,ref}(s) = M(s)(u_{ref}(s) + \sigma_{ref}(s)) \tag{12}$$

$$u_{ref}(s) = C(s)(r_1(s) - \sigma_{ref}(s))\,, \tag{13}$$

where

$$\sigma_{ref}(s) = \frac{(A(s) - M(s))u_{ref}(s) + A(s)d_{ref}(s)}{M(s)}$$

and $d_{ref}(s)$ is the Laplace transform of $d_{ref}(t) \triangleq f(t, y_{1,ref}(t))$. If $C(s)$ and $M(s)$ satisfy the $\mathcal{L}_1$-norm condition (8), then the closed-loop reference system described in (12) and (13) is bounded-input, bounded-output (BIBO) stable.

In [12], it is shown that when system (1) is subject to the $\mathcal{L}_1$ output feedback adaptive controller described in (9), (10), and (11), and $C(s)$ and $M(s)$ satisfy the $\mathcal{L}_1$-norm condition in (8), the following bounds hold:

$$\|\tilde{y}\|_{\mathcal{L}_\infty} < \gamma_0\,, \tag{14}$$

$$\|y_{1,ref} - y_1\|_{\mathcal{L}_\infty} < \gamma_1\,, \tag{15}$$

where $\tilde{y}(t) \triangleq \hat{y}_1(t) - y_1(t)$, γ_0 is defined in equation (4.31) in [12] and

$$\gamma_1 \triangleq \frac{\left\|\frac{C(s)H(s)}{M(s)}\right\|_{\mathcal{L}_1}}{1 - \|G(s)\|_{\mathcal{L}_1} L}\gamma_0\,.$$

It is important to note that $\gamma_0 \propto \sqrt{\frac{1}{\Gamma}}$, which means that for high adaptation gains, the estimation error is bounded and can be made arbitrarily small. Furthermore, the actual system output approaches the behavior of the reference system (12).

When the uncertainties are within the bandwidth of the low-pass filter, the controller is able to cancel the uncertainties exactly. In this ideal scenario, the system response is the following:

$$y_{1,id}(s) = M(s)r_1(s)\,. \tag{16}$$

Even when the ideal case is not achieved, the transient and steady-state performance of the controller have the performance guarantees provided in (14) and (15). For the proposed $\mathcal{L}_1$-MPC framework, we require that the system controlled by the $\mathcal{L}_1$ adaptive controller behaves as close as possible to a known, linear model. As shown in (14), in order to achieve this, we must choose Γ to be very large.

Multi-Input, Multi-Output Implementation The SISO architecture described above can be extended to a MIMO implementation. We assume that the states are decoupled, which can be achieved after applying an appropriate feedback linearization. Then, for n different outputs, the low-pass filter $C(s)$ and the first-order reference system (4) are implemented as $(n \times n)$ diagonal transfer function matrices:

$$\begin{aligned} C(s) &= \operatorname{diag}(C_1(s), \ldots, C_n(s))\,, \\ M(s) &= \operatorname{diag}(M_1(s), \ldots, M_n(s))\,, \end{aligned} \tag{17}$$

where $C_i(s) = \frac{\omega_i}{s+\omega_i}$, $\omega_i > 0$, for example, $M_i(s) = \frac{m_i}{s+m_i}$, $m_i > 0$, and $i = 1, \ldots, n$.

B. Model Predictive Control

MPC finds an optimal control input sequence that steers the system towards a desired trajectory, taking into account input and state constraints. The algorithm updates the control signal at every sampling time based on a known system model and desired future outputs. The key dynamics of the system to be controlled can be described by the following linear, discrete-time model:

$$y_1(k+1) = Ay_1(k) + Br_1(k)\,,\ y_1(0) = y_{1,0}\,, \tag{18}$$

where $y_1(k) \in \mathbb{R}^n$ is the output at time step k, and $r_1(k) \in \mathbb{R}^n$ is the control input. The desired output trajectory $y_1^*(k)$ is assumed to be feasible based on the model (18), where $(r_1^*(k), y_1^*(k))$ satisfy (18). At every time step $\bar{k}$, MPC solves the following open-loop optimization problem:

$$\min_{r_1(\bar{k}),\ldots,r_1(\bar{k}+N_h)} J(y_1, r_1) \tag{19}$$

subject to

$$\begin{aligned} &y_1(k+1) = Ay_1(k) + Br_1(k)\,,\ \forall k = \bar{k}, \ldots, \bar{k}+N_h\,, \\ &y_1(\bar{k}) = y_{1,\bar{k}} \end{aligned}$$

where $J(y_1, r_1)$ is defined as:

$$\begin{aligned} &J(y_1, r_1) = \\ &\textstyle\sum_{k=\bar{k}+1}^{k+N_h+1} (y_1^*(k) - y_1(k))^T Q (y_1^*(k) - y_1(k)) + \\ &\textstyle\sum_{k=\bar{k}}^{k+N_h} r_1^T(k) R r_1(k) + \delta r_1^T(k) S \delta r_1(k)\,, \end{aligned} \tag{20}$$

where $\delta r_1(k) = r_1(k) - r_1(k-1)$, $Q\,, R\,, S \in \mathbb{R}^{n \times n}$, and N_h is the length of the prediction horizon.

We apply the computed optimal $r_1(\bar{k})$ to the system and re-solve the optimization problem at $k = \bar{k}+1$ using the obtained measurement $y_1(\bar{k}+1)$.

C. Adaptive Model Predictive Control

The proposed architecture uses MPC with an underlying $\mathcal{L}_1$ adaptive controller (see Fig. 1), which guarantees that the controlled system behaves close to a reference model. The MPC updates the input signal to the underlying $\mathcal{L}_1$ adaptive controller $r_1(k)$ at every sampling time based on a known system model and desired future outputs. We assume that the $\mathcal{L}_1$ adaptive controller makes the system behave close to (16), which is achievable if the low-pass filter $C(s)$ in (17) is chosen appropriately. In the MIMO case, the ideal system response is described by (16) where $M(s)$ is a diagonal transfer function matrix defined as in (17). The ideal system response can then be discretized and written as:

$$y_1(k+1) = A_D y_1(k) + B_D(r_1(k) - y_1(k))\,, \tag{21}$$

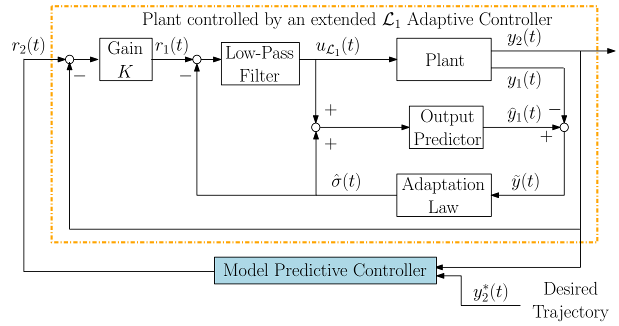

Fig. 2. The proposed adaptive model predictive control architecture with an extended $\mathcal{L}_1$ adaptive controller. The underlying $\mathcal{L}_1$ adaptive controller has been augmented with an outer feedback loop with a proportional gain.

where A_D and B_D are the discretized state-space matrices related to $M(s)$. We can then reformulate the open-loop optimization problem as:

$$\begin{aligned} &\min\nolimits_{r_1(\bar{k}),\ldots,r_1(\bar{k}+N_h)} \\ &\Big[\textstyle\sum_{k=\bar{k}+1}^{\bar{k}+N_h+1} (y_1^*(k) - y_1(k))^T Q (y_1^*(k) - y_1(k)) + \\ &\textstyle\sum_{k=\bar{k}}^{\bar{k}+N_h} r_1^T(k) R r_1(k) + \delta r_1^T(k) S \delta r_1(k)\Big] \end{aligned} \tag{22}$$

subject to

$$\begin{aligned} y_1(k+1) &= A_D y_1(k) + B_D(r_1(k) - y_1(k))\,, \\ y_1(\bar{k}) &= y_{1,\bar{k}}\,, \\ |\ddot{r}_1(k)| &\leq r_{max}\,,\ \forall k = \bar{k}, \ldots, \bar{k}+N_h\,, \end{aligned}$$

where the second constraint may be added to ensure a smooth input and r_{max} is designed based on the system's physical constraints.

IV. EXPERIMENTAL RESULTS

This section shows experimental results of the proposed framework combining $\mathcal{L}_1$ adaptive control and MPC (MPC-$\mathcal{L}_1$) applied to a quadrotor flying three-dimensional trajectories and experiencing dynamic disturbances. We assess two main aspects of the proposed framework: *(i)* predictive performance and *(ii)* robustness to external disturbances, as compared to non-predictive and/or non-adaptive frameworks.

A. Quadrotor Setup

The desired trajectory is described in the x, y and z directions. We implement the MIMO version of the $\mathcal{L}_1$ adaptive controller described in Section III-A, where we assume that the x, y and z directions are decoupled. We use a slightly modified $\mathcal{L}_1$ adaptive controller structure (see Fig. 2) as previously proposed in [16], [21]. The difference is that this extended $\mathcal{L}_1$ adaptive controller includes an additional outer feedback loop with a proportional gain. For the quadrotor example, we define $r_1(s)$ and $r_2(s)$ as the Laplace transforms of the desired translational velocity $r_1(t) \in \mathbb{R}^3$ and the desired bounded position reference input $r_2(t) \in \mathbb{R}^3$, respectively. We also define $y_1(s)$ and $y_2(s)$ as the Laplace transforms of the vehichle's translational velocity $y_1(t) \in \mathbb{R}^3$ and position $y_2(t) \in \mathbb{R}^3$.

We implement an outer-loop proportional controller as described in [16], [21], where

$$r_1(s) = K(r_2(s) - y_2(s))\,,$$

where $K = \mathrm{diag}(K_1, K_2, K_3)$ and $K_i > 0$ for all $i = 1, 2, 3$. This extension to the $\mathcal{L}_1$ adaptive controller results in the following ideal linear reference system:

$$y_{2,id}(s) = \begin{bmatrix} D_1(s) & 0 & 0 \\ 0 & D_2(s) & 0 \\ 0 & 0 & D_3(s) \end{bmatrix} r_2(s)\,, \tag{23}$$

where

$$D_i(s) = \frac{K_i m_i}{s^2 + m_i s + K_i m_i} \quad \text{for } i = 1, 2, 3\,. \tag{24}$$

The system in (23) is used to obtain A_D and B_D used for the equality constraints in the MPC implementation (22). We use the IBM CPLEX solver to compute the desired positions $r_2(k) = (r_{2,x}(k), r_{2,y}(k), r_{2,z}(k))$. The interface to the real quadrotor requires commanded roll ϕ_{des}, pitch θ_{des}, vertical velocity $\dot{z}_{des}$, and yaw rate $\dot{\psi}_{des}$. Therefore, the signal $u_{\mathcal{L}_1}(k)$ (see Fig. 2) is transformed through the following equations:

$$\begin{aligned} \phi_{des} &= -\arcsin(-u_{\mathcal{L}_1,x}\sin(\psi) + u_{\mathcal{L}_1,y}\cos(\psi)) \\ \theta_{des} &= \arcsin(u_{\mathcal{L}_1,x}\cos(\psi) + u_{\mathcal{L}_1,y}\sin(\psi)) \\ \dot{z}_{des} &= u_{\mathcal{L}_1,z}\,, \end{aligned}$$

where ψ is the current yaw angle. The desired yaw angle ψ_{des} is set to zero and controlled through a separate proportional controller.

B. Experimental Setup

The experiments were performed using the Parrot Bebop 2 quadrotor. An overhead motion capture camera system was used to obtain position, velocity, rotation, and rotational velocity measurements. Let $r_{2,j}(k)$ be the desired translational positions and $y_{2,j}(k)$ be the measured quadrotor positions in the $j = x, y, z$ directions, respectively. We propose five different trajectories to test our approach, as shown in Fig. 3. To quantify the controller performance, we define the average position error as:

$$e = \frac{1}{N}\sum_{k=1}^{N}\sqrt{e_x^2(k) + e_y^2(k) + e_z^2(k)}\,, \tag{25}$$

where $e_j(k) = r_{2,j}(k) - y_{2,j}(k)$, N represents the trajectory length, and $j = x, y, z$.

The matrices of the cost function in (22) are defined as $Q = qI$, $R = rI$ and $S = sI$ where $q = 17$, $r = 0.08$ and $s = 0.02$, and I is the identity matrix. Since the objective is to improve the trajectory tracking performance, a significantly larger weight is given to this part in the cost function.

C. $\mathcal{L}_1$ Adaptive Control Performance Bounds

In this subsection, we experimentally verify that *(i)* the estimation error $\tilde{y}$ remains bounded, and *(ii)* the actual system remains close to the ideal linear system for the extended $\mathcal{L}_1$ adaptive controller described in [16] and shown in Fig. 2. To do this, we use the reference input r_2 to command the quadrotor to hover for 1.5 seconds after which it starts moving in a 3D straight line. Fig. 4 shows, on the left side, the estimation error $\hat{y}_{1,i} - y_{1,i}$ for each axis $i = x, y, z$. It is clear that for all three axes the estimation error remains bounded. Moreover, using the ideal linear reference model (23), we calculate the trajectory $y_{2,id}$ that the ideal reference system would follow if the reference input r_2 was applied. The right side of Fig. 4 shows in red the trajectory of the ideal reference system $y_{2,id}$ and in blue the actual trajectory that the quadrotor followed y_2, when the input r_2 is applied. The actual system does behave close to the ideal reference system.

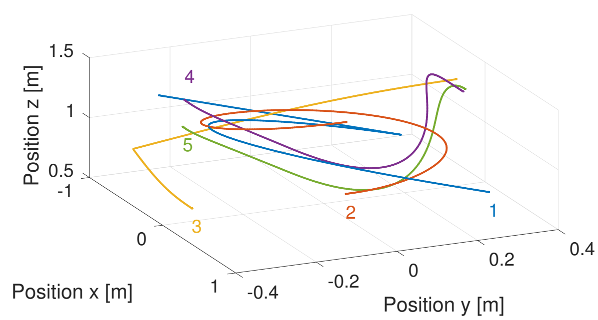

Fig. 3. The five different trajectories that are used to test the MPC-$\mathcal{L}_1$ control framework.

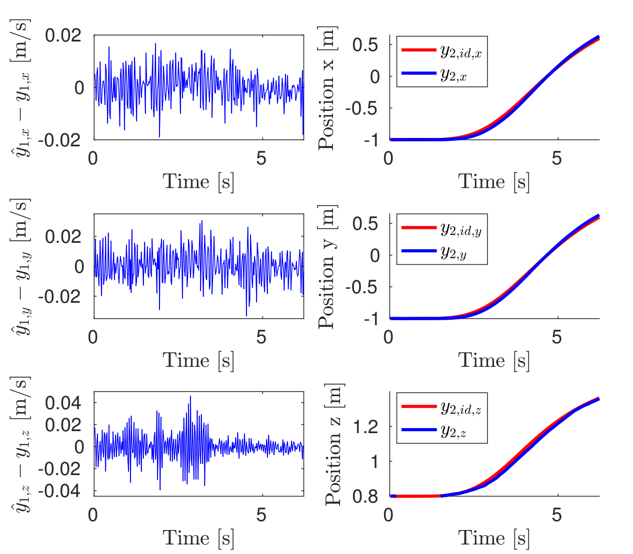

Fig. 4. On the left, the estimation error $\hat{y}_{1,i} - y_{1,i}$ for each axis $i = x, y, z$ is shown. The estimation error remains bounded for the length of the trajectory, as suggested by (14). On the right, the ideal trajectory followed by the ideal system (23) (shown in red) is compared to the actual trajectory followed by the quadrotor (shown in blue).

D. Predictive Performance

We first demonstrate the benefit of the predictive controller component. To do this, we compare the tracking performance obtained with the proposed MPC-$\mathcal{L}_1$ approach to that ob-

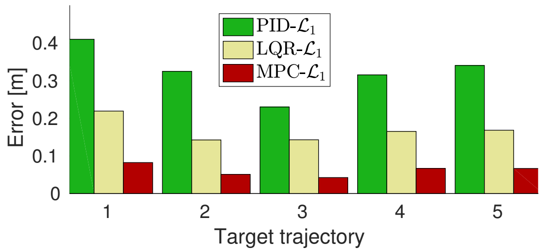

Fig. 5. Average position tracking errors over five different trajectories for each of the three controllers: PID-$\mathcal{L}_1$, LQR-$\mathcal{L}_1$ and MPC-$\mathcal{L}_1$. The PID- and LQR-based controllers use feedback information only; hence, the tracking errors are larger. The predictive controller is able to optimize the reference input based on future information of the desired trajectory. Consequently, the error achieved is significantly smaller.

tained with a PID-$\mathcal{L}_1$ and LQR-$\mathcal{L}_1$ approach. The PID-$\mathcal{L}_1$ framework uses a proportional-integral-derivative (PID) controller to modify the input r_2 of the underlying $\mathcal{L}_1$ adaptive controller. The LQR-$\mathcal{L}_1$ framework uses an infinite-horizon linear-quadratic regulator (LQR) with integral component to modify the input r_2. The integral component in the LQR is used to decrease the steady-state error. All three approaches MPC-$\mathcal{L}_1$, PID-$\mathcal{L}_1$ and LQR-$\mathcal{L}_1$ use the same underlying extended $\mathcal{L}_1$ adaptive controller. The MPC-$\mathcal{L}_1$ uses the cost function specified in Section IV-B. The outer PID and LQR controllers were tuned in experiment to maximize the tracking performance for trajectory 4 (purple) in Fig. 3. It is important to note that neither the PID-$\mathcal{L}_1$ nor the LQR-$\mathcal{L}_1$ controllers take into account future desired outputs when computing the input to the underlying $\mathcal{L}_1$ adaptive controller.

We compare the three controllers on the five test trajectories shown in Fig. 3. The average position tracking errors for each controller and trajectory are shown in Fig. 5. The PID- and LQR-based controllers use only feedback information to correct for tracking errors and have significantly larger tracking errors than the proposed MPC-$\mathcal{L}_1$ framework. This is better observed in Fig. 6, where the positions followed by the quadrotor for each of the three controllers, PID-$\mathcal{L}_1$, LQR-$\mathcal{L}_1$ and MPC-$\mathcal{L}_1$, when tracking trajectory 4 (purple) in Fig. 3 are shown. The feedback-only controllers show a significant tracking delay, see the dark blue and black lines. The predictive component of the proposed MPC-$\mathcal{L}_1$ framework optimizes over the prediction horizon and achieves a better performance, as shown by the cyan line in Fig. 6. This shows the benefit of the predictive MPC component of the proposed approach.

E. Robustness to Disturbances

To assess the robustness to disturbances, the proposed framework is compared to an MPC-PID framework. The MPC-PID framework uses an MPC to modify the input to an underlying PID controller. The system model used in the MPC is obtained by applying a step input to the x, y, and z directions separately and characterizing the system response when the quadrotor is controlled by the PID controller. The system is assumed to be a second-order, linear system in each direction. We first use the MPC-PID and MPC-$\mathcal{L}_1$ frameworks to track each of the five desired trajectories. The cost functions used in the MPC-PID and MPC-$\mathcal{L}_1$ frameworks are the same.

The average position errors for the MPC-PID and MPC-$\mathcal{L}_1$ approaches are shown in Fig. 7 in dark blue and dark red, respectively. The system model obtained through the step response experiments for the MPC-PID framework is not accurate enough to describe the system behavior well. We found that tuning this system model for each trajectory could improve performance for that trajectory; however, doing so affected the performance of the other trajectories. In order to fairly compare both approaches, the model used in the MPC-PID framework is kept constant across trajectories.

Next, using a fan we introduce a wind disturbance at different points in each trajectory. The resulting average errors when the fan disturbance is applied are shown in Fig. 7 in light blue and light red for the MPC-PID and MPC-$\mathcal{L}_1$ frameworks, respectively. The proposed MPC-$\mathcal{L}_1$ framework is able to keep approximately the same performance when a disturbance is applied since the underlying $\mathcal{L}_1$ adaptive controller is able to compensate for it. The MPC-PID approach performs generally worse when wind is applied compared to without wind. Only for trajectory 3, the disturbance applied to the system steers the system in the direction that reduces the tracking trajectory error. Nevertheless, the proposed framework has, on average, a significantly smaller increase in the error when the disturbance is applied.

V. CONCLUSIONS

In this paper, we introduce a novel adaptive MPC framework to improve trajectory tracking performance in the presence of disturbances and partially unknown dynamics. The framework relies on an $\mathcal{L}_1$ adaptive controller to keep the system close to a predefined linear system behavior, despite the presence of disturbances and changing dynamics. The MPC is used to calculate an optimal reference input to the underlying $\mathcal{L}_1$ adaptive controller based on the reference model from the $\mathcal{L}_1$ adaptive controller. Finally, we validated the proposed approach through experiments on a quadrotor. In experiments, the proposed method improves the tracking performance of the system as compared to non-predictive approaches (PID-$\mathcal{L}_1$ and LQR-$\mathcal{L}_1$) and a non-adaptive approach (MPC-PID), even in the presence of strong wind disturbances. The proposed approach is able to effectively control nonlinear systems with a computationally-efficient, adaptive, linear MPC approach.

REFERENCES

[1] R. Skelton, “Model error concepts in control design,” *International Journal of Control*, vol. 49, no. 5, pp. 1725–1753, 1989.

[2] M. Morari and J. H. Lee, “Model predictive control: past, present and future,” *Computers & Chemical Engineering*, vol. 23, no. 4, pp. 667–682, 1999.

[3] S. Skogestad and I. Postlethwaite, *Multivariable feedback control: analysis and design*. Wiley New York, 2007, vol. 2.

[4] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. Scokaert, “Constrained model predictive control: stability and optimality,” *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

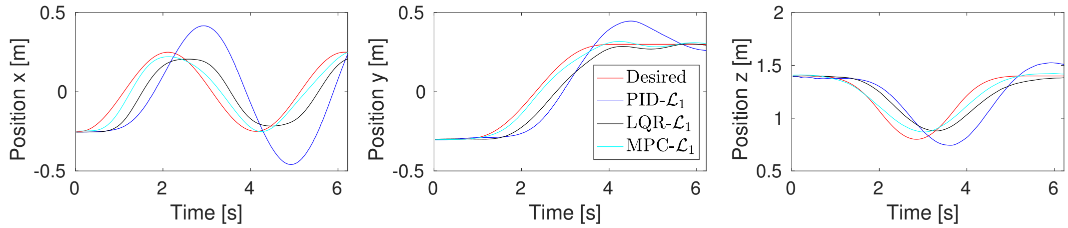

Fig. 6. Actual position of the quadrotor tracking the purple trajectory in Fig. 3 for each of the three controllers: PID-$\mathcal{L}_1$, LQR-$\mathcal{L}_1$ and MPC-$\mathcal{L}_1$. The feedback-only controllers PID-$\mathcal{L}_1$ and LQR-$\mathcal{L}_1$ present tracking delay. The proposed MPC-$\mathcal{L}_1$ framework reduces the tracking error as it is able to include future desired outputs in the calculation of the optimal input.

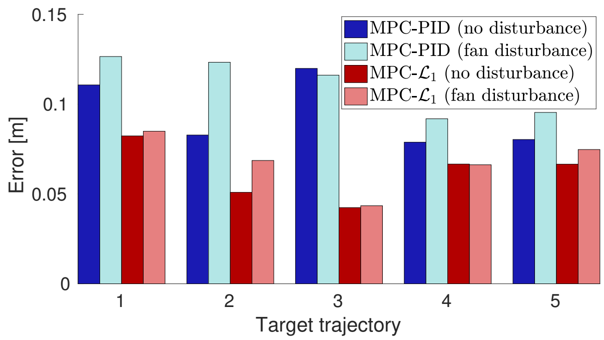

Fig. 7. Average position tracking errors over five different trajectories when using the MPC-PID and MPC-$\mathcal{L}_1$ controllers. The cases when no disturbance is applied to the system are depicted in dark blue (MPC-PID) and dark red (MPC-$\mathcal{L}_1$). A fan is used to apply a wind disturbance to the system and the resulting errors are depicted in light blue (MPC-PID) and light red (MPC-$\mathcal{L}_1$). The proposed MPC-$\mathcal{L}_1$ framework is able to compensate for the disturbance applied and has a performance similar to the case when no disturbance is applied.

[5] D. L. Marruedo, T. Alamo, and E. Camacho, "Input-to-state stable MPC for constrained discrete-time nonlinear systems with bounded additive uncertainties," in *Proc. of the IEEE Conference on Decision and Control (CDC)*, vol. 4, 2002, pp. 4619–4624.

[6] H. Fukushima, T.-H. Kim, and T. Sugie, "Adaptive model predictive control for a class of constrained linear systems based on the comparison model," *Automatica*, vol. 43, no. 2, pp. 301–308, 2007.

[7] V. Adetola, D. DeHaan, and M. Guay, "Adaptive model predictive control for constrained nonlinear systems," *Systems & Control Letters*, vol. 58, no. 5, pp. 320–326, 2009.

[8] T. Wang, H. Gao, and J. Qiu, "A combined adaptive neural network and nonlinear model predictive control for multirate networked industrial process control," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 2, pp. 416–425, 2016.

[9] I. Lenz, R. A. Knepper, and A. Saxena, "DeepMPC: Learning deep latent features for model predictive control." in *Robotics: Science and Systems*, 2015.

[10] C. J. Ostafew, A. P. Schoellig, and T. D. Barfoot, "Robust constrained learning-based NMPC enabling reliable mobile robot path tracking," *International Journal of Robotics Research*, vol. 35, no. 13, pp. 1547–1563, 2016.

[11] C. D. McKinnon and A. P. Schoellig, "Experience-based model selection to enable long-term, safe control for repetitive tasks under changing conditions," in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, accepted.

[12] N. Hovakimyan and C. Cao, *$\mathcal{L}_1$ Adaptive Control Theory: Guaranteed Robustness with Fast Adaptation*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2010.

[13] R. W. Beard, N. B. Knoebel, C. Cao, N. Hovakimyan, and J. S. Matthews, "An L1 adaptive pitch controller for miniature air vehicles," in *Proc. of the AIAA Guidance, Navigation, and Control Conference*, 2006, pp. 5788–5794.

[14] I. Kaminer, O. Yakimenko, V. Dobrokhodov, A. Pascoal, N. Hovakimyan, V. Patel, C. Cao, and A. Young, "Coordinated path following for time-critical missions of multiple UAVs via L1 adaptive output feedback controllers," in *Proc. of the AIAA Guidance, Navigation, and Control Conference*. AIAA 2007-6409, 2007, 2007.

[15] Z. Zuo and P. Ru, "Augmented L1 adaptive tracking control of quad-rotor unmanned aircrafts," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 50, no. 4, pp. 3090–3101, 2014.

[16] K. Pereida, R. R. Duivenvoorden, and A. P. Schoellig, "High-precision trajectory tracking in changing environments through $\mathcal{L}_1$ adaptive feedback and iterative learning," in *Proc. of the IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 344–350.

[17] I. M. Gregory, C. Cao, E. Xargay, N. Hovakimyan, and X. Zou, "$\mathcal{L}_1$ adaptive control design for NASA AirSTAR flight test vehicle," in *Proc. of the AIAA Guidance, Navigation, and Control Conference*. AIAA 2009-5738, 2009.

[18] V. V. Patel, C. Cao, N. Hovakimyan, K. A. Wise, and E. Lavretsky, "$\mathcal{L}_1$ adaptive controller for tailless unstable aircraft," in *Proc. of the American Control Conference (ACC)*, 2007, pp. 5272–5277.

[19] S. Mallikarjunan, B. Nesbitt, E. Kharisov, E. Xargay, N. Hovakimyan, C. Cao *et al.*, "L1 adaptive controller for attitude control of multirotors," in *Proc. of the AIAA Guidance, Navigation, and Control Conference*, 2012, pp. AIAA 2012–4831, 2012.

[20] G. Marafioti, R. R. Bitmead, and M. Hovd, "Persistently exciting model predictive control," *International Journal of Adaptive Control and Signal Processing*, vol. 28, no. 6, pp. 536–552, 2014.

[21] B. Michini and J. P. How, "$\mathcal{L}_1$ adaptive control for indoor autonomous vehicles: Design process and flight testing," in *Proc. of the AIAA Guidance, Navigation, and Control Conference*. AIAA 2009-5754, 2009.