

# An explicit dual control approach for constrained reference tracking of uncertain linear systems

Anilkumar Parsi, Andrea Iannelli, Roy S. Smith, *Fellow, IEEE*

**Abstract**—A finite horizon optimal tracking problem is considered for linear dynamical systems subject to parametric uncertainties in the state-space matrices and exogenous disturbances. A suboptimal solution is proposed using a model predictive control (MPC) based explicit dual control approach which enables active uncertainty learning. A novel algorithm for the design of robustly invariant online terminal sets and terminal controllers is presented. Set membership identification is used to update the parameter uncertainty online. A predicted worst-case cost is used in the MPC optimization problem to model the dual effect of the control input. The cost-to-go is estimated using contractivity of the proposed terminal set and the remaining time horizon, so that the optimizer can estimate future benefits of exploration. The proposed dual control algorithm ensures robust constraint satisfaction and recursive feasibility, and navigates the exploration-exploitation trade-off using a robust performance metric.

**Index Terms**—dual control, model predictive control, reference tracking, safe adaptive control, active learning

## I. INTRODUCTION

IN control problems involving unknown plants, the importance of balancing probing actions aimed at inferring properties of the system (exploration) and greedy decisions that maximize the performance based on the current knowledge (exploitation) has been well understood since the pioneering work in adaptive control [1]. Feldbaum [2] indicated that an optimal adaptive control law was characterized by the right combination of these two, often conflicting, properties, and coined the expression *adaptive dual control systems*. This optimal trade-off can be characterized by means of the principle of optimality, and thus the exact solution to the dual control problem is given by stochastic dynamic programming (DP) [3]. Because of its well-known curse of dimensionality, tractable solutions to this problem are not available except for trivial cases. This motivated approximate solutions which can be broadly divided into two groups, namely implicit and explicit dual control methods [4]. The former considers different types of approximations of the stochastic DP problems, e.g. using the wide-sense property from adaptive control [5], [6] or making use of approximate DP methods such as policy iteration [7]. To apply approximate DP algorithms to unknown

systems, a hyper-state is often considered, which augments the system's state with the vector of unknown parameters [8], [9]. The main issue generally associated with implicit approaches is scalability, as evidenced by their application to date to only very low-order problems. On the other hand, explicit methods relax the original optimal control problem (OCP) into a more tractable form, and then reformulate it so that the solution exhibits the desired dual properties. Even though the optimality gap with respect to the original optimal dual policy is generally unknown, this often represents a much more computationally tractable solution and thus is by far the most established of the two approaches [10], [11], [12], [13], [14], [15]. In an important class of explicit methods, the dual properties are introduced in an application-oriented fashion [16]. That is, probing actions that decrease a measure of the model uncertainty based on control-oriented measures (e.g. a robust performance criterion) are incentivized.

An important consideration for the control of uncertain systems is the guaranteed satisfaction of state and input constraints, which is also referred to as safety in literature [17]. Such safety guarantees can be achieved using model predictive control (MPC), an optimization-based technique where a control objective can be maximized while ensuring constraint satisfaction [18]. When the plant is unknown, robust approaches need to be employed to guarantee stability and constraint satisfaction in the face of the uncertainty [19], [20]. If the initial uncertainty is large, robustness will come at the cost of performance. Thus, it is beneficial to update the uncertainty set online using measurements gathered from the system. An adaptive MPC (AMPC) scheme was developed in [21] and [22] for regulation of linear systems described by state-space models subject to polytopic parametric uncertainty. It uses a tube MPC approach to ensure robust constraint satisfaction and updates the uncertainty using set-membership identification [23]. Adaptive MPC has also been proposed for nonlinear systems using a Lipschitz based method in [24], and a general framework has been developed in [25]. Nevertheless, there is no active uncertainty learning in these methods since actions are designed for the purpose of exploitation only. Another class of optimization-based methods for controlling unknown dynamical systems has recently emerged from the online learning community, where performance is commonly established by means of the concept of regret [26], [27]. In the case of unknown dynamics, these methods typically consider unconstrained control problems and learning is effectively passive.

This work was supported by the Swiss National Science Foundation under Grant 200021\_178890.

The authors are with the Automatic Control Laboratory, Swiss Federal Institute of Technology (ETH Zürich), 8092 Zurich, Switzerland (e-mail: aparsi/iannelli/rsmith@control.ee.ethz.ch).

In [28], we proposed an extension to [22] which introduces dual actions in AMPC by means of a predicted state tube while guaranteeing constraint satisfaction. The problem considered was again only a regulation one, and was implicitly framed in an infinite horizon setting. For linear systems with model uncertainties, regulating the state to the origin is easier than performing setpoint tracking. This is because the input setpoint is assumed to be known, thereby simplifying the design of MPC ingredients. Although this issue is partially addressed in the tracking MPC literature, most methods assume that the system is affected only by either additive disturbances [29], [30], [31], [32] or parametric uncertainties [33], [34]. Two related works are also [13], [14], which proposed a dual adaptive tracking MPC algorithm with uncertainty in the measurement equation only. Moreover, studying the dual effect of the control input is much more relevant in a finite horizon setting [15]. This is because there is limited time for learning the uncertainty and probing actions have decreasing rewards with time. Thus, estimating the benefits of exploratory control actions is difficult in this setting, compared to an infinite horizon setting where it is almost always beneficial to reduce model uncertainty. We therefore view finite horizon problems as a more meaningful, albeit difficult, domain in which the exploration-exploitation trade-off can be studied.

The goal of this paper is to propose an explicit dual control approach for finite horizon optimal tracking control of uncertain systems described by parametric state-space models. There are two main technical contributions. First, we propose a novel tracking adaptive MPC formulation in Section II for plants subject to parametric uncertainty in addition to exogenous disturbances. The proposed algorithm estimates the uncertain setpoints online and uses an online terminal set and a control law dependent on the estimated setpoints. The second contribution of the work is to leverage the viewpoint of MPC as an approximate DP method [17], [35] to obtain an application-oriented dual control algorithm to solve the finite horizon optimal control problem. That is, the proposed algorithm navigates the exploration-exploitation trade-off using a performance based metric as described in Section III.

To the best of the authors' knowledge, this is thus the first dual adaptive MPC scheme for optimal reference tracking of systems with exogenous disturbances and model mismatch in the dynamics.

### A. Problem formulation and methodology

Consider an uncertain, discrete-time, linear system of the form

$$x_{t+1} = A(\theta)x_t + B(\theta)u_t + w_t, \quad (1)$$

where  $x_t \in \mathbb{R}^n$  represents the state,  $u_t \in \mathbb{R}^m$  the control input and  $w_t \in \mathbb{R}^n$  the additive disturbance at the time step  $t$ . The state space matrices  $A(\theta)$  and  $B(\theta)$  have the parametric description

$$A(\theta) = A_0 + \sum_{i=1}^p A_i[\theta]_i, \quad B(\theta) = B_0 + \sum_{i=1}^p B_i[\theta]_i, \quad (2)$$

where  $\theta \in \mathbb{R}^p$  is an unknown constant parameter,  $[\theta]_i$  refers to the  $i^{\text{th}}$  element in  $\theta$  and  $\{A_i, B_i\}_{i=0}^p$  are known matrices

modeling structured uncertainty. The true value of  $\theta = \theta^*$  is known to lie inside the bounded set

$$\Theta := \{\theta \in \mathbb{R}^p | H_\theta \theta \leq h_\theta\}, \quad (3)$$

where  $H_\theta \in \mathbb{R}^{n_\theta \times p}$  and  $h_\theta \in \mathbb{R}^{n_\theta}$ . The disturbance  $w_t$  has unknown statistical properties, but always lies within the bounded set

$$\mathbb{W} := \{w \in \mathbb{R}^n | H_w w \leq h_w\}, \quad (4)$$

where  $H_w \in \mathbb{R}^{n_w \times n}$ ,  $h_w \in \mathbb{R}^{n_w}$ .

*Assumption 1:* The state of the system is perfectly measured at each time step.

The output to be tracked is given by

$$y_t = Cx_t, \quad (5)$$

where  $y_t \in \mathbb{R}^{n_y}$ . The states and inputs of the system are required to satisfy the constraints

$$\mathbb{Z} = \{(x_t, u_t) \in \mathbb{R}^n \times \mathbb{R}^m | Fx_t + Gu_t \leq \mathbf{1}\}, \quad (6)$$

where  $\mathbb{Z}$  is a compact set, and  $F \in \mathbb{R}^{n_c \times n}$  and  $G \in \mathbb{R}^{n_c \times m}$  are given matrices. Given an initial state  $\hat{x}$ , the ideal objective would be to compute the control policy  $\Pi := \{u_t = \pi_t(x_t)\}_{t=0}^T$  which solve the following finite horizon optimal control problem (FHOCP)

$$\min_{\Pi} \max_{w_t} \sum_{t=0}^T \|Q(y_t - r_t)\|_{\infty} + \|R(\pi_t(x_t) - u_t^*)\|_{\infty}, \quad (7a)$$

$$\text{s.t. } A(\theta^*)x_t + B(\theta^*)\pi_t(x_t) + w_t = x_{t+1}, \quad (7b)$$

$$Fx_t + G\pi_t(x_t) \leq \mathbf{1}, \quad (7c)$$

$$y_t = Cx_t, \quad (7d)$$

$$A(\theta^*)x_t^* + B(\theta^*)u_t^* = x_{t+1}^*, \quad (7e)$$

$$Cx_t^* = r_t, \quad \forall t \in \mathbb{N}_0^T, \quad (7f)$$

$$x_0 = \hat{x}, \quad x_0^* = \hat{x}, \quad (7g)$$

where  $T$  is the length of the finite control horizon,  $Q \in \mathbb{R}^{n_c \times n_c}$  and  $R \in \mathbb{R}^{m \times m}$  are positive definite matrices, and  $\mathbb{N}_0^T$  denotes the set of integers from 0 to  $T$ . In addition,  $\{r_t\}_{t=0}^T$  is a known reference trajectory for the output,  $\{u_t^*\}_{t=0}^T$  is a sequence of unknown input setpoints, and  $\{x_t^*\}_{t=0}^T$  represents an unknown desired state trajectory satisfying (7e),(7f). For legibility, the state evolution predicted by the FHOCP is represented using the same variables  $\{x_t\}_{t=0}^T$  used for the evolution of the true state in (1). Solving (7) is difficult because  $\theta^*$  is not known, which results in an uncertain cost function (7a), state evolution (7b) and setpoints (7e). Hence, the objective of this work is to compute suboptimal policies to minimize the cost (7a) for the uncertain system (1) while satisfying the constraints (6). Without any loss in robustness of constraint satisfaction, the setpoint can be estimated using a nominal parameter estimate  $\hat{\theta}$ . In order to ensure robustness of constraint satisfaction in face of the uncertainty, one can solve the robust optimization

problem

$$\begin{aligned}
\min_{\Pi} \max_{\substack{\theta \in \Theta, \\ w_t \in \mathbb{W}}} & \sum_{t=0}^T \|Q(y_t - r_t)\|_{\infty} + \|R(\pi_t(x_t) - u_t^*)\|_{\infty}, \\
\text{s.t.} & A(\theta)x_t + B(\theta)\pi_t(x_t) + w_t = x_{t+1}, \\
& Fx_t + G\pi_t(x_t) \leq \mathbf{1}, \\
& y_t = Cx_t, \\
& A(\bar{\theta})x_t^* + B(\bar{\theta})u_t^* = x_{t+1}^*, \\
& Cx_t^* = r_t, \quad \forall t \in \mathbb{N}_0^T, \\
& x_0 = \overset{\circ}{x}, \quad x_0^* = \overset{\circ}{x}.
\end{aligned} \tag{8}$$

Nevertheless, the robustness guarantees provided by (8) could also result in conservatism when  $\Theta$  is large. This conservatism can be reduced by using the measurements from the system to reduce the size of the uncertainty set (3). To this aim, set membership identification [23] is used in this work (Section II-A). In this approach, the uncertainty set  $\Theta$  is updated by constructing non-falsified parameter sets from state measurements and control inputs. This results in a dual effect for the control, where the control inputs affect both tracking quality and identification performance. It is well known that such problems can be highly non-convex, even for scalar systems with small control horizons  $T$  [36]. For this reason, the control problem (8) is reformulated as described below.

The control policies  $\pi_t(x_t)$  are restricted to be time invariant and affine in the state variable, and are formulated by using an estimate of the state and input setpoint, and a prestabilizing feedback gain (Section II-B). At each time step  $k$ , a feasible input sequence for a shorter time interval  $[k, k + N - 1]$  (called the prediction horizon, with length  $N$ ) is computed using MPC. In addition, a tube MPC approach [37] is used to construct a robust state tube consisting of all the states that the system could reach in the prediction horizon under the defined control policy and an updated parameter set (Section II-C). Tube MPC controllers ensure that the last set of the state tube lies inside a robust invariant set called the terminal set, which is in general computed offline. In Section II-D, an online algorithm to design robustly invariant terminal sets is proposed, which uses the updated parameter set to improve the tracking performance.

To enable dual control, the effect of control input on the identification is modeled in Section III-A, and a predicted state tube is constructed in Section III-B. The robust cost function in (8) is then approximated by a worst-case cost over the predicted state tube (Section III-C). This captures the dual effect of the control using an application-oriented metric [16]. Moreover, the cost to be incurred after the end of the prediction horizon (hereafter called cost-to-go) is approximated by using the remaining time in the control horizon and the contractivity of the terminal set. This allows a novel trade-off between exploration and exploitation which will depend on, among other things, the number of time steps remaining in the control horizon. The dual adaptive MPC (AMPC) algorithm and its properties are discussed in Section IV.

## B. Notation

The sets of real numbers and non-negative real numbers are denoted by  $\mathbb{R}$  and  $\mathbb{R}_{\geq 0}$  respectively. For a vector  $b$ ,  $\|b\|_k$  represents the  $k$ -norm for  $k \in \{2, \infty\}$ . The  $i^{\text{th}}$  row of a matrix  $A$  is denoted by  $[A]_i$ , and  $\otimes$  represents the Kronecker product. The dimensions of matrices and vectors are not always specified when they can be inferred from the context. For any real scalar-valued function  $J$ ,  $\max_{h \in \mathbb{H}} J(h)$  refers to the maximization of  $J$  over the set  $\mathbb{H}$ . The Minkowski sum of two sets  $A$  and  $B$  is denoted by  $A \oplus B$ , and  $\mathbf{1}$  denotes a column vector of appropriate length whose elements are equal to 1. The convex hull of the elements of a set  $S$  is represented by  $\text{co}\{S\}$ . The notation  $a_{l|k}$  denotes the value of  $a$  at time step  $k+l$  computed at the time step  $k$ . The identity matrix of size  $n \times n$  is denoted by  $I_n$ .

## II. TRACKING FORMULATION FOR AMPC

In this section, the constraints in FHOC (8) are reformulated using MPC and set membership identification is introduced to reduce the size of the uncertainty set online. An existing tube MPC approach is modified to enable setpoint tracking by defining the control law as a function of the reference trajectory and the estimates of the system's uncertain parameters. Offline and online designs of terminal sets are then proposed.

### A. Parameter identification

Set-membership identification is a technique used to identify systems affected by bounded noise with unknown statistical properties [23]. In this method, the measurement data is used to update a set of feasible parameters, denoted as  $\Theta_t$ . In order to ensure that the number of constraints in  $\Theta_t$  does not increase at every time step, it is defined as

$$\Theta_t := \{\theta \in \mathbb{R}^p \mid H_{\theta}\theta \leq h_{\theta_t}\}, \tag{9}$$

where  $h_{\theta_t}$  initialized with  $h_{\theta}$ , and updated online using measurement data. To perform the update, a set of non-falsified parameters is constructed using the last  $\tau$  measurements as

$$\begin{aligned}
\Delta_t & := \left\{ \theta \in \mathbb{R}^p \mid -H_w D_i \theta \leq h_w + H_w d_{i+1}, \forall i \in \mathbb{N}_{t-\tau}^{t-1} \right\} \\
& = \{ \theta \in \mathbb{R}^p \mid H_{\Delta} \theta \leq h_{\Delta} \},
\end{aligned} \tag{10}$$

where  $D_i := D(x_i, u_i) \in \mathbb{R}^{n \times p}$  and  $d_{i+1} \in \mathbb{R}^n$  are

$$\begin{aligned}
D_i & = [A_1 x_i + B_1 u_i, \dots, A_p x_i + B_p u_i], \\
d_{i+1} & := A_0 x_t + B_0 u_t - x_{i+1}, \quad \forall i \in \mathbb{N}_{t-\tau}^{t-1}.
\end{aligned} \tag{11}$$

Using (10),  $\Theta_t$  is updated such that

$$\Theta_t \supseteq \Theta_{t-1} \cap \Delta_t \tag{12}$$

is satisfied. This is ensured by calculating  $h_{\theta_t}$  according to the  $n_{\theta}$  linear programs ( $\forall i \in \mathbb{N}_1^{n_{\theta}}$ )

$$\begin{aligned}
[h_{\theta_t}]_i & = \max_{\theta} [H_{\theta}]_i \theta \\
\text{s.t.} \quad & \begin{bmatrix} H_{\theta} \\ H_{\Delta} \end{bmatrix} \theta \leq \begin{bmatrix} h_{\theta_{t-1}} \\ h_{\Delta} \end{bmatrix}.
\end{aligned} \tag{13}$$

The optimization problems in (13) ensure that the set update satisfies  $\Theta_t \subseteq \Theta_{t-1}$ , because the maximum value of the objective is upper bounded by  $[h_{\theta_{t-1}}]_i$  in the constraints. Thus, the parameter identification in (13) results in sets  $\Theta_t$  of non-increasing size. Note that the quality of identification improves by increasing the number of hyperplanes chosen to represent the set  $\Theta_t$ . However, this results in an increase in the computational complexity of the MPC optimization problem later formulated in Section IV.

In addition, a parameter estimate  $\bar{\theta}_k$  is computed, which will be used to define the control law and the setpoints in the cost function. At each time step, this estimate is computed using the center of the maximum volume  $\ell_2$ -norm ball that can be inscribed inside the set  $\Theta_k$ , also known as the Chebyshev center of the set [38]. The estimate,  $\bar{\theta}_k$ , is the optimal value of  $\theta_c$  in the following quadratic program

$$\begin{aligned} \max_{r_c, \bar{\theta}_k} \quad & r_c - \mu \|\theta_c - \bar{\theta}_{k-1}\|_2^2, \\ \text{s.t.} \quad & [H\theta]_i \theta_c + r_c \|[H\theta]_i\|_2 \leq h_{\theta_k}, \quad \forall i \in \mathbb{N}_1^{n_\theta}, \end{aligned} \quad (14)$$

where the second term in the cost function ensures that the problem has a unique solution, and the parameter  $\mu > 0$  must be chosen to trade-off rate of change in the parameter estimate with the distance from the Chebyshev center of the set.

*Remark 1:* The Chebyshev center of a set reduces the deviation of the estimate from the true parameter based on all the points in parameter set. Alternatively, one can use a least mean squares or a recursive least squares filter [39] to obtain the parameter estimate.

## B. Control policy parameterization

In order to formulate a computationally tractable optimization problem, a receding horizon approach is used, where at each time step  $k$ , a feasible input sequence is computed for the prediction horizon  $[k, k+N-1]$ . Note that  $k$  will be used as the time index while formulating the MPC optimization problem at time step  $k$ . Within the prediction horizon, the control policies  $\pi_k(x_k)$  are restricted to be affine functions of  $x_k$ , with the linear term being time-invariant. The policies also depend on estimates of the state and input setpoints, which are computed using the parameter estimate  $\bar{\theta}_k$ .

In order to estimate the state and input setpoints  $(x_k^*, u_k^*)$  along the prediction horizon, the following set of linear constraints can be defined along the lines of (7e),(7f)

$$\begin{bmatrix} A(\bar{\theta}_k) & B(\bar{\theta}_k) \\ C & 0 \end{bmatrix} \begin{bmatrix} \bar{x}_{l|k,k} \\ \bar{u}_{l|k,k} \end{bmatrix} = \begin{bmatrix} \bar{x}_{l+1|k,k} \\ r_{k+l} \end{bmatrix}, \quad \forall l \in \mathbb{N}_0^{N-1}, \quad (15)$$

where the subscripts in  $\bar{x}_{l|k,k}$  denote that the estimate of the setpoint at time step  $k+l$  (first subscript) was computed using the parameter estimate  $\bar{\theta}_k$  (second subscript). The unknown setpoints  $(x_{l+k}^*, u_{l+k}^*)$  can be approximated by  $(\bar{x}_{l|k,k}, \bar{u}_{l|k,k})$  within the MPC horizon, i.e., for  $l \in \mathbb{N}_0^{N-1}$ . In addition, the setpoints at the end of the prediction horizon are defined as

$$\begin{bmatrix} A(\bar{\theta}_k) - I_n & B(\bar{\theta}_k) \\ C & 0 \end{bmatrix} \begin{bmatrix} \bar{x}_{N|k,k} \\ \bar{u}_{N|k,k} \end{bmatrix} = \begin{bmatrix} 0 \\ r_{k+N} \end{bmatrix}. \quad (16)$$

The constraints (16) define the setpoints  $(\bar{x}_{N|k,k}, \bar{u}_{N|k,k})$  as an equilibrium state and input, which will be used in the design of online terminal sets in Section II-D.

*Assumption 2:* The system satisfies  $m \geq n_y$ , i.e., the system has at least as many inputs as the desired outputs to be tracked. Additionally, for all  $\theta \in \Theta$ , the system satisfies

$$\begin{aligned} \text{rank} \left( \begin{bmatrix} A_{\text{st}}(\theta) & B_{\text{st}}(\theta) \\ C_{\text{st}} & 0 \end{bmatrix} \right) &= N(n+n_y), \\ \text{rank} \left( \begin{bmatrix} A(\theta) - I_n & B(\theta) \\ C & 0 \end{bmatrix} \right) &= n+n_y, \end{aligned} \quad (17)$$

where the matrices  $A_{\text{st}}(\theta) \in \mathbb{R}^{Nn \times Nn}$ ,  $B_{\text{st}}(\theta) \in \mathbb{R}^{Nn \times Nm}$  and  $C_{\text{st}} \in \mathbb{R}^{Nn_y \times Nn_y}$  are defined as

$$\begin{aligned} A_{\text{st}}(\theta) &= \begin{bmatrix} A(\theta) & -I_n & 0 & \dots & 0 \\ 0 & A(\theta) & -I_n & \dots & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & \dots & A(\theta) & -I_n & \\ 0 & \dots & \dots & & A(\theta) \end{bmatrix}, \\ B_{\text{st}}(\theta) &= I_N \otimes B(\theta), \quad C_{\text{st}} = I_N \otimes C, \end{aligned} \quad (18)$$

and  $\otimes$  denotes the Kronecker product operator.

Note that Assumption 2 is made so that there exists a state and input setpoint sequence satisfying (15), (16) for any given reference trajectory and any parameter in the set  $\Theta_0$ . Similar assumptions are made in tracking MPC literature considering systems with model uncertainty in the system matrices [33].

The control inputs in the prediction horizon are now parameterized as

$$u_{l|k}(x) = K(x - \bar{x}_{l|k,k}) + v_{l|k}, \quad \forall l \in \mathbb{N}_0^{N-1}, \quad (19)$$

where  $\{v_{l|k}\}_{l=0}^{N-1}$  are decision variables in the MPC optimization problem and  $K \in \mathbb{R}^{m \times n}$  is a prestabilizing feedback matrix which satisfies the following assumption.

*Assumption 3:* The parameter set  $\Theta_0$  is such that there exists a feedback gain  $K$  which asymptotically stabilizes  $A_{\text{cl}}(\theta) = A(\theta) + B(\theta)K$ ,  $\forall \theta \in \Theta$ .

Such a stabilizing feedback gain  $K$  can be computed using standard robust control techniques, for example, as suggested in [40].

## C. Robust state tube

The robust tube MPC approach proposed in [21] is now modified for the tracking problem using the setpoints  $\bar{x}_{l|k,k}$  in the prediction horizon. A robust state tube is constructed to contain all the states that the system can reach under the control law (19), for all realizations of  $\theta \in \Theta_k$  and  $w \in \mathbb{W}$ . Specifically, the state tube  $\{\mathbb{X}_{l|k}\}_{l=0}^N$  satisfies

$$\begin{aligned} \mathbb{X}_{0|k} &\ni \{x_k\}, \quad \mathbb{X}_{l+1|k} \supseteq A(\theta)\mathbb{X}_{l|k} \oplus B(\theta)u_{l|k} \oplus \mathbb{W}, \\ &\forall \theta \in \Theta_k, \quad l \in \mathbb{N}_0^{N-1}. \end{aligned} \quad (20)$$

To ensure that the representation of the sets  $\mathbb{X}_{l|k}$  is tractable, they are defined as a sequence of homothetic sets [37], that is, as a translation and scaling of a predefined polytope  $\mathbb{X}_0$ . This polytope is designed offline, and has the description

$$\mathbb{X}_0 := \{x | H_x x \leq \mathbf{1}\} = \text{co}\{x^1, x^2, \dots, x^q\}, \quad (21)$$

where  $H_x \in \mathbb{R}^{n_x \times n}$  and  $\{x^1, x^2, \dots, x^q\}$  are the vertices of  $\mathbb{X}_0$ . In the online optimization, the center of  $\mathbb{X}_0$  is translated using  $z_{l|k} \in \mathbb{R}^n$  and its size is scaled by  $\alpha_{l|k} \in \mathbb{R}_{\geq 0}$  to define

the set  $\mathbb{X}_{l|k}$ . Thus, the propagation of the state tube is defined by the optimization variables  $\{z_{l|k}, \alpha_{l|k}\}_{l=0}^N$  according to

$$\begin{aligned}\mathbb{X}_{l|k} &= \{z_{l|k}\} \oplus \alpha_{l|k} \mathbb{X}_0 = \{x | H_x(x - z_{l|k}) \leq \alpha_{l|k} \mathbf{1}\} \\ &= \{z_{l|k}\} \oplus \alpha_{l|k} \text{co}\{x^1, x^2, \dots, x^q\}.\end{aligned}\quad (22)$$

Using the above parameterization, the constraints (6) and set dynamics (20) can be reformulated as linear constraints in terms of the optimization variables. For this purpose, the following notation is used

$$\begin{aligned}x_{l|k}^j &= z_{l|k} + \alpha_{l|k} x^j, & u_{l|k}^j &= K(x_{l|k}^j - \bar{x}_{l|k,k}) + v_{l|k}, \\ D_{l|k}^j &= D(x_{l|k}^j, u_{l|k}^j), & d_{l|k}^j &= A_0 x_{l|k}^j + B_0 u_{l|k}^j - z_{l+1|k},\end{aligned}\quad (23)$$

where  $j \in \mathbb{N}_1^q, l \in \mathbb{N}_0^{N-1}$ . Note that unlike the definition in (11) where  $D_t$  and  $d_t$  are functions of known states and inputs, the quantities  $D_{l|k}^j$  and  $d_{l|k}^j$  linearly depend on the optimization variables. Additionally, the vectors  $\bar{f}$  and  $\bar{w}$  are computed offline such that for  $i \in \mathbb{N}_1^{n_c}, j \in \mathbb{N}_1^{n_x}$ ,

$$[\bar{f}]_i = \max_{x \in \mathbb{X}_0} [F + GK]_i x, \quad [\bar{w}]_j = \max_{w \in \mathbb{W}} [H_x]_j w. \quad (24)$$

The following proposition reformulates the robust constraints and set-dynamics as affine constraints.

*Proposition 1 ([21, Proposition 9]):* Let the state tube  $\{\mathbb{X}_{l|k}\}_{l=0}^N$  be parameterized according to (22). Then, the constraints (6) and set-dynamics (20) are satisfied if and only if  $\forall j \in \mathbb{N}_1^q, l \in \mathbb{N}_0^{N-1}$ , there exist matrices  $\Lambda_{l|k}^j \in \mathbb{R}_{\geq 0}^{n_x \times n_\theta}$  with non-negative elements such that

$$(F + GK)z_{l|k} + Gv_{l|k} - GK\bar{x}_{l|k,k} + \alpha_{l|k}\bar{f} \leq \mathbf{1}, \quad (25a)$$

$$-H_x z_{0|k} - \alpha_{0|k} \mathbf{1} \leq -H_x x_k, \quad (25b)$$

$$\Lambda_{l|k}^j h_{\theta_k} + H_x d_{l|k}^j - \alpha_{l+1|k} \mathbf{1} \leq -\bar{w}, \quad (25c)$$

$$H_x D_{l|k}^j = \Lambda_{l|k}^j H_\theta. \quad (25d)$$

#### D. Terminal set

The system's state must not be driven into the regions of the state space which could result in constraint violations after the prediction horizon. This can be guaranteed by ensuring the MPC algorithm is recursively feasible, which is generally achieved using a robustly invariant terminal set and a terminal control law.

*Definition 1 (Robust Invariance [20]):* A set  $\mathbb{X}$  is said to be robustly invariant under the dynamics (1) and a terminal control law  $\pi(x)$  if  $A(\theta)x + B(\theta)\pi(x) + w \in \mathbb{X}$  and  $(x, \pi(x)) \in \mathbb{Z}$  for all  $x \in \mathbb{X}, \theta \in \Theta$  and  $w \in \mathbb{W}$ .

In this paper, we use a similar approach to that followed in [28] and impose terminal constraints to make sure that  $\mathbb{X}_{N|k}$  lies inside an invariant set. As the controller in [28] is designed to perform regulation, the center  $z_{N|k}$  was set as the origin and  $\alpha_{N|k}$  is bounded by a precomputed constant  $\bar{\alpha}$ . However, a different strategy must be employed for reference tracking to account for the uncertainty in the input setpoint. We propose two different methods, offline and online, for the design of terminal sets.

**1) Offline terminal set design:** To design the terminal set offline, the initial parameter estimate  $\bar{\theta}_0$  is used to generate setpoint estimates  $(\bar{x}_{N|k,0}, \bar{u}_{N|k,0})$  for  $k \in \mathbb{N}_0^{T-N}$ . These setpoints might not lie in the feasible region  $\mathbb{Z}$ . For this reason, the terminal set to be used at time step  $k$  is defined using an artificial setpoint  $(\tilde{x}_{N|k,0}, \tilde{u}_{N|k,0})$  and an upper bound  $\tilde{\alpha}_{N|k} \in \mathbb{R}_{\geq 0}$  on  $\alpha_{N|k}$ . The size of the terminal set is thus specified by  $\tilde{\alpha}_{N|k}$ , which is large when the setpoint  $(\tilde{x}_{N|k,0}, \tilde{u}_{N|k,0})$  is far from the constraints (6). The terminal control law is chosen to be of the form  $\pi^T(x) = K(x - \tilde{x}_{N|k}) + \tilde{u}_{N|k}$ . Hence  $\tilde{x}_{N|k,0}, \tilde{u}_{N|k,0}$  and  $\tilde{\alpha}_{N|k}$  are computed offline by solving the following  $T-N+1$  quadratic programs (for  $k \in \mathbb{N}_0^{T-N}$ )

$$\begin{aligned}\min_{\substack{\tilde{x}_{N|k,0}, \tilde{u}_{N|k,0}, \tilde{\alpha}_{N|k}, \\ \tilde{x}_{N|k,0}, \tilde{u}_{N|k,0}}} & \|\tilde{x}_{N|k,0} - \bar{x}_{N|k,0}\|_2^2 + \\ & \|\tilde{u}_{N|k,0} - \bar{u}_{N|k,0}\|_2^2 - \xi \tilde{\alpha}_{N|k}, \\ \text{s.t.} & \begin{bmatrix} A(\bar{\theta}_0) - I & B(\bar{\theta}_0) \\ C & 0 \end{bmatrix} \begin{bmatrix} \tilde{x}_{N|k,0} \\ \tilde{u}_{N|k,0} \end{bmatrix} = \begin{bmatrix} 0 \\ r_{k+N} \end{bmatrix},\end{aligned}\quad (26a)$$

$$F(\tilde{x}_{N|k,0} + \tilde{\alpha}_{N|k} x^j) + G(\tilde{u}_{N|k,0} + \tilde{\alpha}_{N|k} K x^j) \leq \mathbf{1}, \quad (26b)$$

$$\begin{aligned}H_x(A(\theta^i)(\tilde{x}_{N|k,0} + \tilde{\alpha}_{N|k} x^j) + \\ B(\theta^i)(\tilde{u}_{N|k,0} + \tilde{\alpha}_{N|k} K x^j) - \tilde{x}_{N|k,0}) + \bar{w} \leq \tilde{\alpha}_{N|k} \mathbf{1},\end{aligned}\quad (26c)$$

$\forall i \in \mathbb{N}_1^{q_\theta}, j \in \mathbb{N}_1^q,$

where the cost function tries to minimize the distance between the artificial and estimated setpoints, and the constant  $\xi > 0$  trades off this distance with the size of the terminal set. Note that  $\{\theta^i\}_{i=1}^{q_\theta}$  represent the vertices of the set  $\Theta_0$ . The constraints (26b) ensure that the terminal set is within  $\mathbb{Z}$ , and (26c) imposes the robust invariance condition. The set  $\mathbb{X}_{N|k}$  is constrained to lie inside the invariant region by setting:

$$z_{N|k} = \tilde{x}_{N|k,0}, \quad \alpha_{N|k} \leq \tilde{\alpha}_{N|k}. \quad (27)$$

Although the terminal sets designed in (26) account for the reference trajectory and can be tuned to find the largest terminal sets, such an offline design would be conservative for two reasons. The first reason is that the identification step (13) updates the parameter set and can result in a larger terminal region, which is not leveraged in this approach. The second reason is that choosing the size of the terminal set a priori can lead to poor performance. For example, choosing a small  $\xi$  results in a small terminal set and reduces the feasible region of MPC.

**2) Online terminal set design:** The reasons discussed above motivate an online design where  $z_{N|k}$  is set equal to an optimization variable  $\tilde{x}_{N|k} \in \mathbb{R}^n$ , whose distance from  $\bar{x}_{N|k,k}$  can be minimized by the MPC optimizer. Additionally,  $\alpha_{N|k}$  is upper bounded by a scalar optimization variable  $\tilde{\alpha}_{N|k} \in \mathbb{R}_{\geq 0}$ , which is constrained such that the terminal set  $\mathbb{X}_{N|k}^T := \tilde{x}_{N|k} \oplus \tilde{\alpha}_{N|k} \mathbb{X}_0$  is robustly invariant for the system (1) under the terminal control law

$$\pi_k^T(x) = K(x - \tilde{x}_{N|k}) + \tilde{u}_{N|k} \quad (28)$$

where  $\tilde{u}_{N|k} \in \mathbb{R}^m$  is an optimization variable. Using the notation

$$\begin{aligned}\tilde{x}_{N|k}^j &= \tilde{x}_{N|k} + \tilde{\alpha}_{N|k} x^j, & \tilde{u}_{N|k}^j &= K(x_{N|k}^j - \tilde{x}_{N|k}) + \tilde{u}_{N|k}, \\ D_{N|k}^j &= D(\tilde{x}_{N|k}^j, \tilde{u}_{N|k}^j), & d_{N|k}^j &= A_0 \tilde{x}_{N|k}^j + B_0 \tilde{u}_{N|k}^j - \tilde{x}_{N|k},\end{aligned}\quad (29)$$

for  $j \in \mathbb{N}_1^q$ , robust invariance of  $\mathbb{X}_{N|k}^T$  can be formulated as

$$F\tilde{x}_{N|k} + G\tilde{u}_{N|k} + \tilde{\alpha}_{N|k}\bar{f} \leq \mathbf{1}, \quad (30a)$$

$$\tilde{\Lambda}_{N|k}^j h_{\theta_k} + H_x d_{N|k}^j - \tilde{\alpha}_{N|k}\mathbf{1} \leq -\bar{w}, \quad (30b)$$

$$H_x D_{N|k}^j = \tilde{\Lambda}_{N|k}^j H_{\theta}, \quad (30c)$$

where  $j \in \mathbb{N}_1^q$  and  $\tilde{\Lambda}_{N|k}^j \in \mathbb{R}_{\geq 0}^{n_x \times n_{\theta}}$  are Lagrange multiplier variables, similar to (25). The feasibility of the constraints in (30) will be addressed in Proposition 2. The optimization variables defining  $\mathbb{X}_{N|k}$  are constrained as

$$z_{N|k} = \tilde{x}_{N|k}, \quad \alpha_{N|k} \leq \tilde{\alpha}_{N|k}. \quad (31)$$

Thus, the online terminal sets guarantee robust invariance for parameters in  $\Theta_k$  instead of  $\Theta$ , and also provide flexibility to increase their size depending on the feasibility of the reference trajectory. In view of these advantages, the subsequent sections formulate the MPC problem using online terminal sets.

*Remark 2:* Note that a terminal set is used in the MPC approximation of (7) for the time steps  $k$  where  $k+N < T$  in order to retain feasibility after time step  $k+N$ . When  $k+N \geq T$ , i.e., towards the end of the control horizon, the terminal constraints can be omitted from the MPC optimization problem and the value of  $N$  can be appropriately shortened.

**3) Design of  $\mathbb{X}_0$ :** An important aspect of the MPC algorithm is the design of  $\mathbb{X}_0$ , which is used to propagate the set dynamics as well as parameterize the terminal set. Thus, an essential design condition for  $\mathbb{X}_0$  is that (30) can admit a feasible solution. Using Assumption 4, the following proposition shows that the feasibility of the terminal constraints can be guaranteed.

*Assumption 4:* The set  $\mathbb{X}_0$  is such that there exists a scalar  $\lambda_c \in [0, 1)$  satisfying

$$H_x(A(\theta)x + B(\theta)Kx) \leq \lambda_c \mathbf{1}, \quad \forall \theta \in \Theta, x \in \mathbb{X}_0, \quad (32a)$$

$$\lambda_c \mathbf{1} \leq \mathbf{1} - \bar{w}\bar{f} \quad (32b)$$

where  $\bar{f} = \max_i [\bar{f}]_i, i \in \mathbb{N}_1^{n_c}$ .

The conditions in (32a) imply that the set  $\mathbb{X}_0$  is  $\lambda$ -contractive with contractivity factor  $\lambda_c$  for the undisturbed dynamics of the system (1) under the control law  $u = Kx$ .

*Proposition 2:* If Assumption 4 is satisfied, then the terminal conditions (30) admit a feasible solution.

*Proof:* The contractivity of  $\mathbb{X}_0$  (32a) implies

$$H_x A_{cl}(\theta)x^j \leq \lambda_c \mathbf{1}, \quad \theta \in \Theta, j \in \mathbb{N}_1^q. \quad (33)$$

Using Definition 1, the conditions for robust invariance of  $\mathbb{X}_{N|k}^T$  can be written as

$$Fx + G\pi_{N|k}^T(x) \leq \mathbf{1}, \quad (34a)$$

$$H_x(A(\theta)x + B(\theta)\pi_{N|k}^T(x) + w - \tilde{x}_{N|k}) \leq \tilde{\alpha}_{N|k}\mathbf{1}, \quad (34b)$$

$$\forall x \in \mathbb{X}_{N|k}, \theta \in \Theta, w \in \mathbb{W}.$$

Substituting the control law (28) and using (24), (34a) can be rewritten as (30a). Similarly, (34b) can be reformulated as

$$H_x(A_{cl}(\theta)(x - \tilde{x}_{N|k}) + A(\theta)\tilde{x}_{N|k} + B(\theta)\tilde{u}_{N|k}) \leq \tilde{\alpha}_{N|k}\mathbf{1}, \quad (35a)$$

$$\begin{aligned} & -\tilde{x}_{N|k}) + \bar{w} \leq \tilde{\alpha}_{N|k}\mathbf{1}, \\ \iff & H_x(A_{cl}(\theta)(\tilde{\alpha}_{N|k}x^j) + A(\theta)\tilde{x}_{N|k} + B(\theta)\tilde{u}_{N|k}) \\ & -\tilde{x}_{N|k}) + \bar{w} \leq \tilde{\alpha}_{N|k}\mathbf{1}, \quad \forall j \in \mathbb{N}_1^q. \end{aligned} \quad (35b)$$

Note that (35b) is equivalent to the pair of equations (30b),(30c), where the Lagrange multipliers  $\tilde{\Lambda}_{N|k}^j$  are used to reformulate a maximization over  $\theta$  to ensure robustness. The equivalence holds due to strong duality. Using (33), the condition (35b) is always satisfied if

$$\bar{w} + H_x(A(\theta)\tilde{x}_{N|k} + B(\theta)\tilde{u}_{N|k} - \tilde{x}_{N|k}) \leq \tilde{\alpha}_{N|k}(\mathbf{1} - \lambda_c). \quad (36)$$

It can be seen that the conditions (30a) and (36) admit  $\tilde{x}_{N|k} = \tilde{u}_{N|k} = 0$ , and  $\tilde{\alpha}_{N|k} = 1/\bar{f}$  as a feasible solution. ■

Although the above proof provides a feasible terminal set with center at the origin, the parameter  $\lambda_c$  is a design variable which can be tuned to allow other feasible solutions. The offline design procedure to be followed is described in Algorithm 1. Given the system dynamics, a control gain  $K$  is first designed. Then, a  $\lambda$ -contractive set  $\mathbb{X}_0$  with a chosen  $\lambda_c$  is constructed using the algorithm given in [41, p. 3]. An important point to consider is that the design of  $\mathbb{X}_0$  affects the control performance through the choice of  $\lambda_c$ , and the number of vertices  $q$  and hyperplanes  $n_x$  in  $\mathbb{X}_0$  affect the computational complexity as seen in (25).

---

#### Algorithm 1 Offline design procedure

---

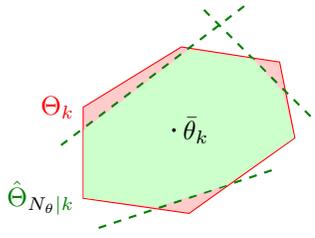
**Input:**  $\Theta$ , **Output:**  $K, \mathbb{X}_0$

- 1: Compute robustly stabilizing gain  $K$  (e.g., see [40])
  - 2: Choose a contractivity rate  $\lambda_c \in [0, 1)$
  - 3: Compute  $\mathbb{X}_0$  as suggested in [41, p. 3]
  - 4: Compute the constants  $\bar{w}, \bar{f}, \bar{f}$
  - 5: **if**  $\lambda_c \leq 1 - \bar{w}\bar{f}$  **then**
  - 6:     **STOP**
  - 7: **else**
  - 8:     Return to step 1, redesign  $K, \mathbb{X}_0$
  - 9: **end if**
- 

*Remark 3:* Note that the design of polytopic contractive sets is a difficult problem, which has been extensively studied for linear systems [42], [43], [44]. The upper bound (32b) in Assumption 4 implies that the condition in step 5 of Algorithm 1 is satisfied. If this condition is not satisfied, one must redesign  $K$  and choose a new  $\lambda_c$ . Even though there is no systematic way to ensure an improved design at the next iteration, one heuristic that worked in practice was to choose smaller  $\lambda_c$ , and impose  $\lambda$ -contractivity on the ellipsoidal region stabilized by  $K$  [40].

### III. DUAL CONTROL USING CONSTRAINT PREDICTIONS

The robust state tube defined in Section II ensures that the constraints are satisfied for any parameter in  $\Theta_k$ . This section models the effect of the control inputs on the identification scheme to enable dual control in closed loop. For this purpose, we define predicted variables and sets which are analogous to the ones defined in Section II. These predicted quantities are denoted with a hat ( $\hat{\cdot}$ ).



**Fig. 1:** The parameter set  $\Theta_k$  is bounded by the constraints shown in red, and the parameter estimate  $\bar{\theta}_k$  lies inside this set. The predicted bounds are shown as dashed green lines, which depend on the control input variables  $v_{l|k}$ . The green shaded region shows the resulting predicted parameter set  $\hat{\Theta}_{N_\theta|k}$ .

### A. Predicted parameter set

The parameter identification step (12) connects the control inputs to the parameter sets in closed loop. This link can be leveraged within the MPC problem by predicting the future state measurements as  $\hat{x}_{l|k}$  according to

$$\begin{aligned} \hat{x}_{l+1|k} &= A(\bar{\theta}_k)\hat{x}_{l|k} + B(\bar{\theta}_k)\hat{u}_{l|k}, \\ \hat{u}_{l|k} &= K(\hat{x}_{l|k} - \bar{x}_{l|k,k}) + v_{l|k}, \quad \forall l \in \mathbb{N}_0^{N_\theta-1} \\ \hat{x}_{0|k} &= x_k, \end{aligned} \quad (37)$$

where  $\bar{\theta}_k$  is the parameter estimated by (14), and  $N_\theta \in \mathbb{N}_2^N$  is the *lookahead* horizon. Using these predicted measurements, the predicted non-falsified sets  $\{\hat{\Delta}_{l|k}\}_{l=1}^{N_\theta}$  are defined as

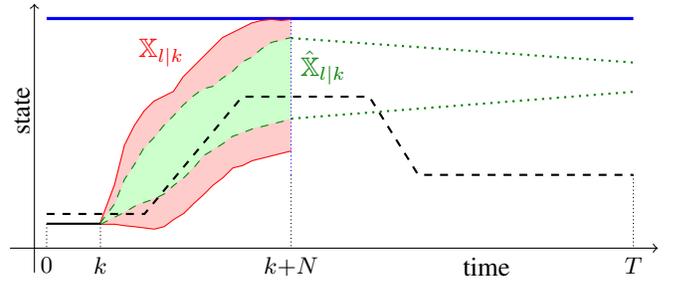
$$\hat{\Delta}_{l|k} := \{\theta \mid \hat{x}_{t+1|k} - A(\theta)\hat{x}_{t|k} - B(\theta)\hat{u}_{t|k} \in \mathbb{W}, \forall t \in \mathbb{N}_0^{l-1}\}. \quad (38)$$

The hyperplanes defining the sets  $\hat{\Delta}_{l|k}$  depend affinely on the control input variables  $v_{l|k}$  as seen in (37). This dependence is omitted for clarity of presentation. A sequence of predicted parameter sets  $\{\hat{\Theta}_{l|k}\}_{l=1}^{N_\theta} \subseteq \Theta_k$  can now be defined as

$$\hat{\Theta}_{l|k} := \Theta_k \cap \hat{\Delta}_{l|k} = \{\theta \in \mathbb{R}^p \mid \hat{H}_{\theta_{l|k}} \theta \leq \hat{h}_{\theta_{l|k}}\}. \quad (39)$$

An illustration of the parameter estimate, the predicted constraints and parameter set is shown in Fig 1.

*Remark 4:* Note that the lookahead horizon  $N_\theta$  must have a minimum value of 2. This is because, when  $N_\theta = 1$  the predicted non-falsified set  $\hat{\Delta}_{N_\theta|k}$  is described by the first control input  $u_k$  and the known state measurement  $x_k$ . This means that, if some uncertain parameters only affect the matrix  $A(\theta)$ , the predicted constraints on these parameters cannot be characterized as a function of the MPC optimization variables. As a result, there would be no dual action of the control input for these parameters. Additionally, it can be seen that as  $N_\theta$  increases, the length of the input sequence affecting the parameter estimation increases, allowing for improved identification. However, this also results in higher computational costs due to the non-convex constraints in the optimization problem, as seen in Section III-B.



**Fig. 2:** Evolution of the robust state tube  $\mathbb{X}_{l|k}$  (red, solid) and the predicted state tube  $\hat{\mathbb{X}}_{l|k}$  (green, dashed). The reference trajectory (black, dashed) and the state constraint (blue, solid) are also shown. The cost function is predicted using  $\hat{\mathbb{X}}_{l|k}$ , and cost-to-go is estimated using  $\lambda$ -contractivity (green, dotted).

### B. Predicted state tube

In order to capture the effect of  $u_k$  on the performance via the cost, a predicted state tube is constructed such that

$$\begin{aligned} \hat{\mathbb{X}}_{0|k} &\ni \{x_k\}, \\ \hat{\mathbb{X}}_{l+1|k} &\supseteq A(\theta)\hat{\mathbb{X}}_{l|k} \oplus B(\theta)u_{l|k} \oplus \mathbb{W}, \quad \forall \theta \in \hat{\Theta}_{l|k}, l \in \mathbb{N}_0^{N_\theta-1} \\ \hat{\mathbb{X}}_{l+1|k} &\supseteq A(\theta)\hat{\mathbb{X}}_{l|k} \oplus B(\theta)u_{l|k} \oplus \mathbb{W}, \quad \forall \theta \in \hat{\Theta}_{N_\theta|k}, l \in \mathbb{N}_{N_\theta}^{N-1}. \end{aligned} \quad (40)$$

Note that the same control law (19) is used to propagate the robust and predicted state tubes. However, the parameter sets in (20) and (40) are different. The sets  $\hat{\mathbb{X}}_{l|k} = \{\hat{z}_{l|k}\} \oplus \hat{\alpha}_{l|k} \mathbb{X}_0$  are parameterized using  $\hat{z}_{l|k} \in \mathbb{R}^n$ ,  $\hat{\alpha}_{l|k} \in \mathbb{R}_{\geq 0}$  for all  $l \in \mathbb{N}_0^{N-1}$ . The following notation is now defined for  $j \in \mathbb{N}_1^q$ ,  $l \in \mathbb{N}_0^{N-1}$

$$\begin{aligned} \hat{x}_{l|k}^j &= \hat{z}_{l|k} + \hat{\alpha}_{l|k} x^j, & \hat{d}_{l|k}^j &= A_0 x_{l|k}^j + B_0 u_{l|k}^j - \hat{z}_{l+1|k}, \\ \hat{D}_{l|k}^j &= D(\hat{x}_{l|k}^j, \hat{u}_{l|k}^j), & \hat{u}_{l|k}^j &= K(\hat{x}_{l|k}^j - \bar{x}_{l|k,k}) + v_{l|k}. \end{aligned} \quad (41)$$

The next proposition formulates the dynamics of the predicted state tube as affine constraints.

*Proposition 3:* The predicted state tube  $\{\hat{\mathbb{X}}_{l|k}\}_{l=0}^N$  satisfies the set-dynamics (40) if and only if, for all  $j \in \mathbb{N}_1^q$  and  $l \in \mathbb{N}_0^{N-1}$  there exists  $\hat{\Lambda}_{l|k}^j \in \mathbb{R}_{\geq 0}^{n_x \times (n_\theta + n_w)}$  such that

$$-H_x \hat{z}_{0|k} - \hat{\alpha}_{0|k} \mathbf{1} \leq -H_x x_k, \quad (42a)$$

$$\hat{\Lambda}_{l|k}^j \hat{h}_{\theta_{l|k}} + H_x \hat{d}_{l|k}^j - \hat{\alpha}_{l+1|k} \mathbf{1} \leq -\bar{w}, \quad (42b)$$

$$H_x \hat{D}_{l|k}^j = \hat{\Lambda}_{l|k}^j \hat{H}_{\theta_{l|k}}. \quad (42c)$$

The constraints (42b) and (42c) are bilinear in the optimization variables, because the hyperplanes defining  $\hat{\Delta}_{l|k}$  are affinely dependent on the control input variables  $v_{l|k}$ , as seen in (37),(38). An illustration of the robust and predicted state tubes is shown in Fig. 2. It can be seen that predicted state tube always lies within the robust state tube ( $\hat{\mathbb{X}}_{l|k} \subseteq \mathbb{X}_{l|k}$ ), because  $\hat{\Theta}_{l|k} \subseteq \Theta_k$  according to (39).

### C. Predicted cost function

The predicted state tube captures the effect of the control input on the resulting uncertainty as well as the dynamics. However, its dependence on the parameter estimate  $\bar{\theta}_k$  means that it cannot guarantee the trajectories to lie within the constraint set. A popular strategy used in safe-learning literature is to decouple the robustness and learning [45]. This approach

is used here, whereby the constraint satisfaction is ensured by the robust state tube, and the predicted state tube is used to define the AMPC cost function as the sum of stage costs and a terminal cost,

$$\Omega(\{\hat{\mathbb{X}}_{l|k}, \bar{x}_{l|k,k}, \bar{u}_{l|k,k}\}_{l=0}^N, \{v_{l|k}\}_{l=0}^{N-1}, \tilde{u}_{N|k}) := \sum_{l=0}^{N-1} J(\hat{\mathbb{X}}_{l|k}, \bar{x}_{l|k,k}, \bar{u}_{l|k,k}, v_{l|k}) + J_{\mathcal{T}}(\hat{\mathbb{X}}_{N|k}, \bar{x}_{N|k,k}, \bar{u}_{N|k,k}, \tilde{u}_{N|k}). \quad (43)$$

The stage cost  $J(\hat{\mathbb{X}}_{l|k}, \bar{x}_{l|k,k}, \bar{u}_{l|k,k}, v_{l|k})$  is defined as

$$J(\hat{\mathbb{X}}_{l|k}, \bar{x}_{l|k,k}, \bar{u}_{l|k,k}, v_{l|k}) = \max_{x \in \hat{\mathbb{X}}_{l|k}} \|Q(Cx - r_{k+l})\|_{\infty} + \|R(K(x - \bar{x}_{l|k,k}) + v_{l|k} - \bar{u}_{l|k,k})\|_{\infty}, \quad (44)$$

where the maximization over the predicted state tube is used to approximate the worst-case performance due to the uncertainty and disturbance. Using an epigraph reformulation, (44) can be described as a linear cost function [38]. The terminal cost function  $J_{\mathcal{T}}(\hat{\mathbb{X}}_{N|k}, \bar{x}_{N|k,k}, \bar{u}_{N|k,k}, \tilde{u}_{N|k})$  must capture the cost-to-go from time step  $k + N$  to  $T$ . This is approximated using the  $\lambda$ -contractivity of the terminal set as

$$J_{\mathcal{T}}(\hat{\mathbb{X}}_{N|k}, \bar{x}_{N|k,k}, \bar{u}_{N|k,k}, \tilde{u}_{N|k}) = \max_{x \in \hat{\mathbb{X}}_{N|k}} \beta(k + N) \left( \|Q(Cx - r_{k+N})\|_{\infty} + \|R(K(x - \bar{x}_{N|k,k}) + \tilde{u}_{N|k} - \bar{u}_{N|k,k})\|_{\infty} \right), \quad (45)$$

where  $\beta(k) = (1 - \lambda_c^{T-k}) / (1 - \lambda_c)$  is a time dependent factor. The cost-to-go is approximated using  $\beta(k)$  under the assumption that the reference trajectory does not change and that the system is undisturbed in the time horizon  $[k + N, T]$ . Thus, the cost function defined in (43) estimates the performance benefits of exploration within the prediction horizon in the stage cost (44), and approximates the benefits after the prediction horizon in the terminal cost (45) by using the contractivity and remaining length of the control horizon. Moreover, the cost function (43) enables application-oriented dual control, since the optimizer can trade-off exploration with exploitation based on the estimated improvement of a robust performance metric.

*Remark 5:* The optimal control problem (7) uses an infinity norm in the cost function, which enables to describe the MPC worst-case cost (43) as a linear function using epigraph reformulation. Instead, if a more standard quadratic cost is considered in (7) and in MPC, the maximization of a quadratic function over the predicted state tube sets can be described using second order cone constraints.

#### IV. DUAL ADAPTIVE MPC

In this section, we formulate the dual adaptive MPC algorithm using the robust and predicted state tubes described in Sections II and III. An overview of the different parts of the algorithm is shown in Figure 3. The two main contributions of the paper can also be seen in the figure, as (i) improvements to tracking MPC using the blocks from Section II and (ii) the proposition of a novel application-oriented dual control scheme using the blocks from Section III.

#### A. Algorithm

The decision variables of the MPC optimization problem are

$$\gamma_k = \left\{ \begin{array}{l} \{z_{l|k}, \alpha_{l|k}\}_{l=0}^N, \{\hat{z}_{l|k}, \hat{\alpha}_{l|k}\}_{l=0}^N, \\ \{v_{l|k}, \{\Lambda_{l|k}^j\}_{j=1}^q, \{\hat{\Lambda}_{l|k}^j\}_{j=1}^q\}_{l=0}^{N-1}, \\ \{\bar{x}_{l|k,k}, \bar{u}_{l|k,k}\}_{l=0}^N, \tilde{x}_{N|k}, \tilde{u}_{N|k}, \tilde{\alpha}_{N|k}, \{\tilde{\Lambda}_{N|k}^j\}_{j=1}^q \end{array} \right\}, \quad (46)$$

and the optimization problem can be written as

$$\min_{\gamma_k} \Omega(\{\hat{\mathbb{X}}_{l|k}, \bar{x}_{l|k,k}, \bar{u}_{l|k,k}\}_{l=0}^N, \{v_{l|k}\}_{l=0}^{N-1}, \tilde{u}_{N|k})$$

$$\text{s.t. } (F + GK)z_{l|k} + Gv_{l|k} + \alpha_{l|k}\bar{f} \leq \mathbf{1}, \quad (47a)$$

$$-H_x z_{0|k} - \alpha_{0|k}\mathbf{1} \leq -H_x x_k, \quad (47b)$$

$$\Lambda_{l|k}^j h_{\theta_k} + H_x d_{l|k}^j - \alpha_{l+1|k}\mathbf{1} \leq -\bar{w}, \quad (47c)$$

$$H_x D_{l|k}^j = \Lambda_{l|k}^j H_{\theta}, \quad (47d)$$

$$-H_x \hat{z}_{0|k} - \hat{\alpha}_{0|k}\mathbf{1} \leq -H_x x_k, \quad (47e)$$

$$\hat{\Lambda}_{l|k}^j \hat{h}_{\theta_{l|k}} + H_x \hat{d}_{l|k}^j - \hat{\alpha}_{l+1|k}\mathbf{1} \leq -\bar{w}, \quad (47f)$$

$$H_x \hat{D}_{l|k}^j = \hat{\Lambda}_{l|k}^j \hat{H}_{\theta_{l|k}}, \quad (47g)$$

$$\begin{bmatrix} A(\bar{\theta}_k) & B(\bar{\theta}_k) \\ C & 0 \end{bmatrix} \begin{bmatrix} \bar{x}_{l|k,k} \\ \bar{u}_{l|k,k} \end{bmatrix} = \begin{bmatrix} \bar{x}_{l+1|k,k} \\ r_{l|k} \end{bmatrix}, \quad (47h)$$

$$(F + GK)\tilde{x}_{N|k} + G\tilde{u}_{N|k} + \tilde{\alpha}_{N|k}\bar{f} \leq \mathbf{1}, \quad (47i)$$

$$\tilde{\Lambda}_{N|k}^j h_{\theta_k} + H_x \tilde{d}_{N|k}^j - \tilde{\alpha}_{N|k}\mathbf{1} \leq -\bar{w}, \quad (47j)$$

$$H_x D_{N|k}^j = \tilde{\Lambda}_{N|k}^j H_{\theta}, \quad (47k)$$

$$z_{N|k} = \bar{x}_{N|k,k}, \quad \alpha_{N|k} \leq \tilde{\alpha}_{N|k}, \quad (47l)$$

$$\begin{bmatrix} A(\bar{\theta}_k) - I & B(\bar{\theta}_k) \\ C & 0 \end{bmatrix} \begin{bmatrix} \tilde{x}_{N|k,k} \\ \tilde{u}_{N|k,k} \end{bmatrix} = \begin{bmatrix} 0 \\ r_{k+N} \end{bmatrix}, \quad (47m)$$

$$\alpha_{l|k} \geq 0, \quad \hat{\alpha}_{l|k} \geq 0, \quad \tilde{\alpha}_{N|k} \geq 0, \quad (47n)$$

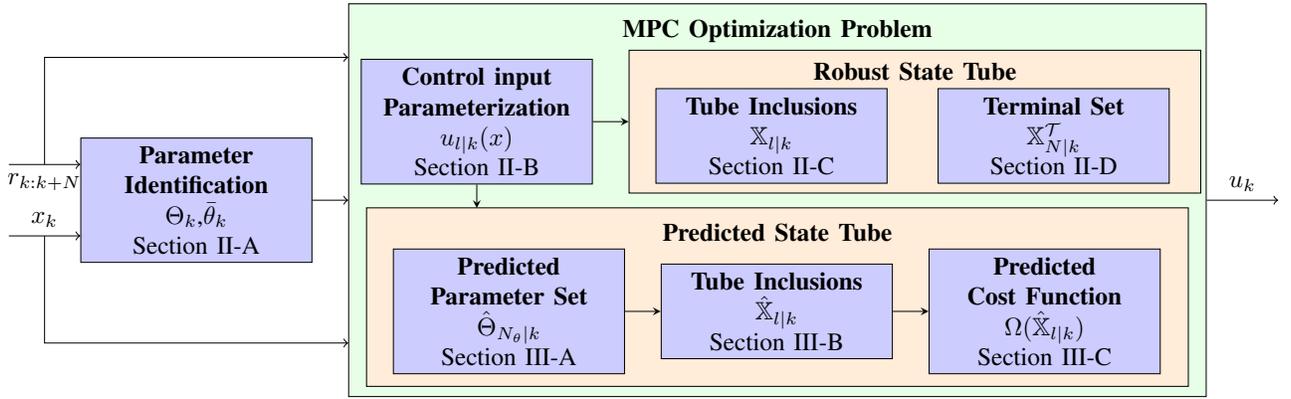
$$\Lambda_{l|k}^j \geq 0, \quad \hat{\Lambda}_{l|k}^j \geq 0, \quad \tilde{\Lambda}_{N|k}^j \geq 0, \quad (47o)$$

$$\forall j \in \mathbb{N}_1^q, l \in \mathbb{N}_0^{N-1}.$$

The cost function of the optimization problem is defined in (43). The constraints (47a) model the state and input constraints, (47b)-(47d) propagate the robust state tube according to Proposition 1. The constraints (47e)-(47g) propagate the predicted state tube as defined in Proposition 3, and (47h) computes the setpoint estimates used in the stage cost and input parameterization as stated in (15). The terminal constraints from (30) are reproduced in (47i)-(47l), and (47m) estimates the terminal setpoint as defined in (16). The optimization problem (47) is a nonconvex bilinear program due to the bilinear constraints (47f), (47g).

The dual AMPC algorithm that can be used to solve (7) is described in Algorithm 2. The prestabilizing gain  $K$  and the state tube shape  $\mathbb{X}_0$  are designed offline using Algorithm 1. The value of  $h_{\theta_k}$  is initialized according to (3), and an initial guess is used for  $\bar{\theta}_0$ .

*Remark 6:* An important property of tracking MPC algorithms is to guarantee offset free tracking. Nevertheless, such a guarantee can only be proven asymptotically for systems affected by disturbances and model uncertainties [29], [33]. This cannot be achieved in a finite horizon setting, such as the one considered in this paper. However, the modifications



**Fig. 3:** A schematic highlighting the different parts of the proposed dual adaptive MPC algorithm. Each block shows the main variable, set or function computed in that block. The measurements  $x_k$  are obtained from the system, and the control inputs  $u_k$  computed by the optimization problem are applied to the system.

### Algorithm 2 Reference tracking using dual AMPC

**Input:**  $\bar{x}, r_k$

**Offline Design**  $K, \mathbb{X}_0$  using Algorithm 1

**Online**

- 1:  $k \leftarrow 1$
- 2: **repeat**
- 3:   Obtain the measurement  $x_k$
- 4:   Construct  $\Delta_k$  according to (10)
- 5:   Update  $h_{\theta_k}$  using (13) and compute  $\bar{\theta}_k$  using (14)
- 6:   Solve optimization problem (47)
- 7:   Apply the control input  $u_k = K(x_k - \bar{x}_{k,k}) + v_{0|k}$
- 8:    $k \leftarrow k + 1$
- 9: **until**  $k \leftarrow T$

proposed in Section II are not restricted to finite horizon problems, and can be used along with AMPC to achieve offset free tracking. For such an objective, the exploration based cost function must be replaced by a certainty equivalence one, based on the parameter  $\bar{\theta}_k$  which is updated using a stable observer. Additionally, the algorithm must ensure that the estimate satisfies  $\bar{\theta}_k \rightarrow \theta^*$  asymptotically, using conditions such as persistence of excitation [22].

### B. Recursive feasibility

One of the main objectives of the proposed dual AMPC method is to ensure robust constraint satisfaction while navigating the exploration-exploitation trade-off.

*Theorem 1:* Let Assumptions 2, 3 and 4 be satisfied, and the optimization problem (47) be feasible for the initial time step  $k = 0$ . The trajectories of the closed loop system using the dual AMPC algorithm (2) will satisfy the constraints (6), and the optimization problem (47) will remain feasible for all  $k \in \mathbb{N}_1^T$ .

*Proof:* The recursive feasibility of (47) will be proven by induction. The feasibility at time step  $k = 0$  is assumed. Let the solution of (47) at time step  $k$  be denoted as  $\gamma_k^*$ , where the superscript  $*$  is added to all the variables in (46). Moreover, denote the robust state tube as  $\{\mathbb{X}_{l|k}^*\}_{l=0}^N$  and the terminal set as  $\mathbb{X}_{N|k}^T$ . Using these variables, a feasible solution can be computed at the time step  $k + 1$ .

Because the robust state tube also satisfies the set-dynamics of the predicted state tube (40), one only needs to find a feasible solution for the input variables  $\{v_{l|k+1}\}_{l=0}^{N-1}$ , robust state tube variables  $\{z_{l|k+1}, \alpha_{l|k+1}\}_{l=0}^N, \{\{\Lambda_{l|k+1}^j\}_{j=1}^q\}_{l=0}^{N-1}$ , and terminal set variables  $\tilde{x}_{N|k+1}, \tilde{u}_{N|k+1}, \tilde{\alpha}_{N|k+1}, \{\tilde{\Lambda}_{N|k+1}^j\}_{j=1}^q$ .

Before the optimization problem (47) is solved at the next time step, the parameter estimate  $\bar{\theta}_k$  and set  $\Theta_k$  are updated to  $\bar{\theta}_{k+1}$  and  $\Theta_{k+1}$  respectively. Let  $\{\tilde{x}_{l|k+1, k+1}, \tilde{u}_{l|k+1, k+1}\}_{l=0}^N$  be a set of feasible solutions to the equations (15) and (16) calculated at time step  $k + 1$ , which exists due to Assumption 2. Consider the sequence of input variables for  $l \in \mathbb{N}_1^N$

$$v_{l-1|k+1} = v_{l|k}^* + K(\tilde{x}_{l-1|k+1} - \tilde{x}_{l|k}^*). \quad (48)$$

This results in the control law

$$\begin{aligned} u_{l-1|k+1}(x) &= K(x - \tilde{x}_{l-1|k+1}) + v_{l-1|k+1} \\ &= K(x - \tilde{x}_{l|k}^*) + v_{l|k}^*, \quad \forall l \in \mathbb{N}_1^N, \end{aligned} \quad (49)$$

which is equivalent to the optimal control law computed at time step  $k$ . Similarly, by defining  $v_{N-1|k+1} = \tilde{u}_{N|k}^* + K(\tilde{x}_{N-1|k+1} - \tilde{x}_{N|k}^*)$ , the terminal control law computed at the time step  $k$  can be used to compute a feasible solution for  $v_{N-1|k+1}$ . Because the updated parameter set satisfies  $\Theta_{k+1} \subseteq \Theta_k$ , the robust state tube variables  $\{z_{l|k+1}, \alpha_{l|k+1}, \{\Lambda_{l|k+1}^j\}_{j=1}^q\}_{l=0}^{N-1}$  can be computed by setting

$$\mathbb{X}_{l-1|k+1} = \mathbb{X}_{l|k}^*, \quad \forall l \in \mathbb{N}_1^N. \quad (50)$$

The robust invariance of the set  $\mathbb{X}_{N|k}^T$  implies that

$$\mathbb{X}_{N|k}^T \supseteq A_{cl}(\theta) \mathbb{X}_{N|k}^* \oplus B(\theta) v_{N|k}^* \oplus \mathbb{W}. \quad (51)$$

Thus, the values of the robust state tube variables  $\{z_{N|k+1}, \alpha_{N|k+1}\}$  are computed by setting  $\mathbb{X}_{N|k+1} = \mathbb{X}_{N|k}^T$ . Additionally, robust invariance of  $\mathbb{X}_{N|k}^T$  also implies that the terminal set variables  $\tilde{x}_{N|k+1}, \tilde{u}_{N|k+1}, \tilde{\alpha}_{N|k+1}, \{\tilde{\Lambda}_{N|k+1}^j\}_{j=1}^q$  can remain unchanged from the optimal values at time step  $k$ .

Using the above procedure, a feasible solution to (47) can be constructed for the time step  $k + 1$ , proving the recursive feasibility of Algorithm 2 by induction. Finally, the constraints (6) are always satisfied because the robust state tube lies within the constraint set, as shown in Proposition 1. ■

*Remark 7:* It can be seen that the optimization problem (47) is recursively feasible due to the flexibility provided by the online terminal sets. This is because even if  $r_{k+1} \neq r_k$ , the optimizer can choose to center the terminal set at  $\tilde{x}_{N|k+1} = \tilde{x}_{N|k}^*$  to ensure constraint satisfaction. This would not be possible with the offline design of terminal sets presented in (26) and (27).

### C. Tube inclusion approximation

It can be seen that the number of constraints and variables involved in the optimization problem (47) depends on the number of vertices of  $\mathbb{X}_0$ , i.e.,  $q$ . This means that the computational complexity of the optimization problem can grow combinatorially with the state dimension. This drawback of robust adaptive MPC has been addressed either by choosing a restrictive structure on the  $\Theta_k$  matrix [46], or using a distributed optimization approach by imposing structure on the system dynamics [47]. In this work, we propose a novel way to limit the optimization problem size by parameterizing the Lagrange multiplier variables  $\Lambda_{l|k}^j, \hat{\Lambda}_{l|k}^j, \tilde{\Lambda}_{N|k}^j$ .

Define the matrices  $E_i = [A_i + B_i K \quad -B_i K \quad B_i]$ ,  $\forall i \in \mathbb{N}_0^p$  and vectors  $e_{l|k} = [z_{l|k}^\top \quad \bar{x}_{l|k,k}^\top \quad v_{l|k}^\top]^\top$ ,  $\forall l \in \mathbb{N}_0^{N-1}$ . At each time step, before solving the MPC optimization problem, the matrices  $\Lambda_k^j \in \mathbb{R}^{n_x \times n_\theta}$ ,  $\forall j \in \mathbb{N}_1^q$  are computed such that

$$\Lambda_k^j H_\theta = H_x D(x^j, Kx^j). \quad (52)$$

Then, the following quantities are defined

$$\underline{\Lambda}_k = \min_{j \in \mathbb{N}_1^q} \Lambda_k^j, \quad \bar{\Lambda}_k = \max_{j \in \mathbb{N}_1^q} \Lambda_k^j h_{\theta_k} + H_x (A_0 + B_0 K) x^j, \quad (53)$$

where the minimization and maximization are performed elementwise. Then, the following lemma reformulates the vertex-wise tube inclusion constraints so that they can be applied on the state tube variables.

*Proposition 4:* Let the matrices  $\Lambda_{l|k}$  satisfy

$$\Lambda_{l|k} + \alpha_{l|k} \underline{\Lambda}_k \geq 0, \quad (54a)$$

$$\Lambda_{l|k} H_\theta = H_x [E_1 \ E_2 \ \dots \ E_p] \otimes e_{l|k}, \quad (54b)$$

$$\Lambda_{l|k} h_{\theta_k} + \alpha_{l|k} \bar{\Lambda}_k + H_x E_0 e_{l|k} - H_x z_{l+1|k} - \alpha_{l+1|k} \mathbf{1} \leq -\bar{w}. \quad (54c)$$

Then  $\Lambda_{l|k}^j = \Lambda_{l|k} + \alpha_{l|k} \Lambda_k^j$  satisfies the constraints (47c), (47d)  $\forall j \in \mathbb{N}_1^q$ .

*Proof:* First, consider the constraints (47d), where the term  $H_x D_{l|k}^j$  can be written as

$$H_x D_{l|k}^j = H_x [E_1 \ E_2 \ \dots \ E_p] \otimes e_{l|k} + \alpha_{l|k} H_x D(x^j, Kx^j)$$

Thus, the vertex dependent part of the equality constraint can be separated from the vertex-independent part. The former is ensured by (52), and the latter by (54b). Then, consider the inequality constraints (47c). Here, the term  $H_x d_{l|k}^j$  can be written as

$$H_x d_{l|k}^j = H_x E_0 e_{l|k} - H_x z_{l+1|k} + \alpha_{l|k} H_x (A_0 + B_0 K) x^j$$

When the parameterization  $\Lambda_{l|k}^j = \Lambda_{l|k} + \alpha_{l|k} \Lambda_k^j$  is applied to (47c), it can be seen that  $\alpha_{l|k} \bar{\Lambda}_k$  bounds the effect of the vertex

dependent terms. Thus, the inequality (54c) ensures (47c) for all the vertices of the state tube. ■

Thus, by using the proposed parameterization, the robust state tube can be constructed using fewer optimization variables and constraints as shown in (54), instead of using the constraints (47c) and (47d) defined at each vertex of the state tube. A similar approach can be used to parameterize  $\tilde{\Lambda}_{N|k}^j$  as  $\tilde{\Lambda}_{N|k} + \alpha_{N|k} \Lambda_k^j$ , thereby replacing the terminal constraints (47j), (47k) with

$$\tilde{\Lambda}_{N|k} + \alpha_{N|k} \underline{\Lambda}_k \geq 0, \quad (55a)$$

$$\tilde{\Lambda}_{N|k} H_\theta = H_x [E_1 \ E_2 \ \dots \ E_p] \otimes \tilde{e}_{N|k}, \quad (55b)$$

$$\tilde{\Lambda}_{N|k} h_{\theta_k} + \alpha_{N|k} \bar{\Lambda}_k + H_x E_0 \tilde{e}_{N|k} - H_x \tilde{x}_{N|k} - \alpha_{l+1|k} \mathbf{1} \leq -\bar{w}, \quad (55c)$$

where  $\tilde{e}_{N|k} = [\tilde{x}_{N|k}^\top \quad \tilde{x}_{N|k}^\top \quad \tilde{u}_{N|k}^\top]^\top$ .

Finally, the Lagrange multipliers defining the predicted state tube are parameterized as

$$\hat{\Lambda}_{l|k}^j = [\hat{\Lambda}_{l|k} + \alpha_{l|k} \Lambda_k^j \quad \hat{\Lambda}_{l|k}^\Delta]. \quad (56)$$

It can be seen that the first  $n_\theta$  columns of  $\hat{\Lambda}_{l|k}^j$  are parameterized using the precomputed  $\Lambda_k^j$ , while the predicted constraints in  $\hat{\Lambda}_{l|k}$  will be multiplied by  $\hat{\Lambda}_{l|k}^\Delta$ . Using (56), the tube inclusion constraints (47f) and (47g) can be replaced by the sufficient conditions

$$\hat{\Lambda}_{l|k} + \alpha_{l|k} \underline{\Lambda}_k \geq 0, \quad \hat{\Lambda}_{l|k}^\Delta \geq 0, \quad (57a)$$

$$[\hat{\Lambda}_{l|k} \quad \hat{\Lambda}_{l|k}^\Delta] \hat{H}_{\theta_{l|k}} = H_x [E_1 \ E_2 \ \dots \ E_p] \otimes \hat{e}_{N|k}, \quad (57b)$$

$$[\hat{\Lambda}_{l|k} \quad \hat{\Lambda}_{l|k}^\Delta] \hat{h}_{\theta_{l|k}} + \hat{\alpha}_{l|k} \bar{\Lambda}_k + H_x E_0 \hat{e}_{l|k} - H_x \hat{z}_{l+1|k} - \hat{\alpha}_{l+1|k} \mathbf{1} \leq -\bar{w}. \quad (57c)$$

The sufficiency of 57 follows from a similar proof as in Proposition 4. Thus the number of nonconvex constraints in the optimization problem do not increase with the number of vertices in the state tube set  $\mathbb{X}_0$ .

The amount of conservatism induced by the proposed sufficient conditions depends on the matrices  $\Lambda_k^j$  computed according to (52). Since the matrix  $H_\theta$  describes the bounded set  $\Theta_k$ , it has more rows than columns. Thus, the choice of  $\Lambda_k^j$  is non-unique based on (52) alone. The quantities  $\underline{\Lambda}_k, \bar{\Lambda}_k$  can be used to optimize over the feasible values of  $\Lambda_k^j$ . As seen in (53) and (54a), negative values in  $\Lambda_k^j$  result in a minimum positive magnitude for  $\Lambda_{l|k}$ , thereby growing the state tube quickly as seen in (55c). In addition, large values for  $\bar{\Lambda}_k$  directly influence the growth of the state tube in (55c). The following optimization problem is solved at each time step to compute the matrices  $\Lambda_k^j$ :

$$\begin{aligned} \min_{\{\Lambda_k^j\}} \quad & \|\bar{\Lambda}_k\|_2 + \mu_\theta \|\Lambda_k^j\|_{\max}, \\ \text{s.t.} \quad & \Lambda_k^j H_\theta = H_x D(x^j, Kx^j), \end{aligned} \quad (58)$$

$$\bar{\Lambda}_k \geq \Lambda_k^j h_{\theta_k} + H_x (A_0 + B_0 K) x^j, \quad \forall j \in \mathbb{N}_1^q,$$

where  $\mu_\theta$  is a tuning parameter which must be chosen such that the precomputed  $\Lambda_k^j$  matrices do not have large negative numbers or result in large  $\bar{\Lambda}_k$ .

## V. SIMULATION STUDIES

In this section, we illustrate the advantages of the proposed dual adaptive MPC scheme using comparative simulations on two systems. The first example considers a 2 state, 2 input system with 2 uncertain parameters. The second example considers a 6 state, 3 input system with 5 uncertain parameters in the model. The code to simulate both the examples is available in the online repository [48].

### A. Two state example

For the first example, the model of the system is parameterized by the matrices

$$\begin{aligned} A_0 &= \begin{bmatrix} 0.85 & 0.5 \\ 0.2 & 0.7 \end{bmatrix}, & A_1 &= \begin{bmatrix} 0.1 & 0 \\ 0 & 0.2 \end{bmatrix}, & A_2 &= \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \\ B_0 &= \begin{bmatrix} 1 & 0.4 \\ 0.2 & 0.6 \end{bmatrix}, & B_1 &= \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, & B_2 &= \begin{bmatrix} 0 & 0.2 \\ 0 & 0.35 \end{bmatrix}, \end{aligned} \quad (59)$$

where the state and input matrices are affected by different uncertain parameters. The outputs to be tracked are the two state variables, and thus a target state trajectory is given by  $r_k$ . In addition, because  $m = n_y = 2$  the constraint (15) defines a unique input setpoint. The uncertainty is defined by the parameter set  $\Theta = \{\theta \in \mathbb{R}^2 \mid \|\theta\|_\infty \leq 1\}$ . Note that the parameter  $[\theta]_2$  is affecting the second input channel only, and its relative effect on  $B(\theta)$  is larger compared to the effect of  $[\theta]_1$  on  $A(\theta)$ . The constraint set  $\mathbb{Z}$  is defined as

$$\mathbb{Z} = \{(x, u) \in \mathbb{R}^{2 \times 2} \mid \|x\|_\infty \leq 3, \|u\|_\infty \leq 2\},$$

and the disturbance set is  $\mathbb{W} := \{w \in \mathbb{R}^2 \mid \|w\|_\infty \leq 0.1\}$ . The cost function is defined by the matrices  $Q = 2I_2$  and  $R = I_2$ .

The controller using Algorithm 2 is referred to as dual AMPC (DAMPC) in this section. The performance of DAMPC is compared to the adaptive MPC algorithm proposed in [21], referred to as passive AMPC (PAMPC). This is because PAMPC updates the parameter set using set membership, but does not model the dual effect of the control input in the optimization problem. This makes the uncertainty learning passive. The PAMPC controller was proposed in [21] for a regulation task. For the sake of comparison, the algorithm is modified to enable setpoint tracking using the offline terminal set approach presented in Section II-D. In addition, the PAMPC controller also uses (15) to estimate the setpoints in the MPC cost function. Both the controllers are used to track a piecewise constant reference trajectory with five different setpoints for  $T = 100$  time steps, and use a prediction horizon of  $N = 8$  steps. The DAMPC controller used a lookahead horizon  $N_\theta = 8$  steps. The control gain  $K$  is designed such that the unit hyperbox in 2 dimensions satisfies Assumption 4, with  $\lambda_c = 0.96$ . This implies  $n_x=4$  and  $q=4$  for the chosen  $\mathbb{X}_0$ . The simulations are performed for 50 different realizations of the parameter  $\theta^* \in \Theta$ . For each realization of  $\theta^*$ , 4 simulations with different disturbance sequences  $\{w_k\}_{k=0}^T \in \mathbb{W}$  are performed. The values of  $\theta^*$  and  $\{w_k\}_{k=0}^T$  are generated at random with a uniform distribution, and the same parameters and disturbance sequences are used in closed loop for the two controllers. Moreover, a third controller is also designed with perfect knowledge of  $\theta^*$ . This controller,

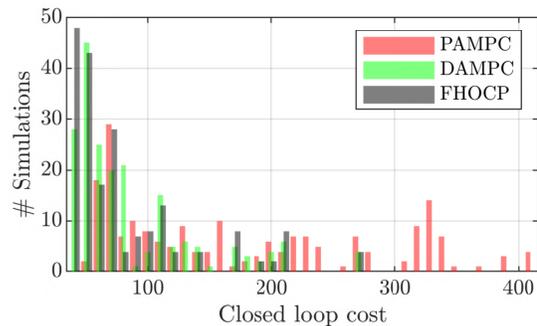


Fig. 4: Comparison of closed loop costs over 200 simulations with different realizations of uncertainty and disturbance.

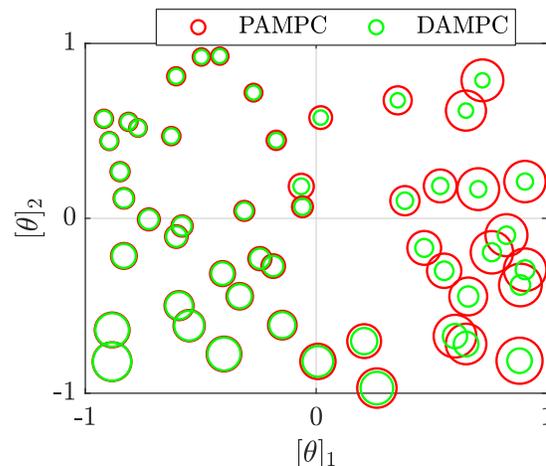


Fig. 5: Distribution of costs as a function of the true parameter, averaged over the disturbance realizations. The center of each circle represents the true parameter, and the area of the circle is proportional to the averaged cost.

called the FHOCP controller, computes an input sequence for the whole remaining control horizon (from  $k$  to  $T$ ) instead of the a shorter prediction horizon. Moreover, the controller also constructs a state tube using  $\mathbb{X}_0$  to be robust against the additive disturbance. This controller is used to quantify the loss of performance caused by the exploratory inputs applied to the system in order to identify the parameters. All the controllers are initialized with the same initial guess  $\theta_0 = [0.1, 0.1]^T$ , and the initial state of the system is at the origin.

The closed loop costs achieved by the controllers for the 200 simulations are compared in Fig. 4, where the frequency of occurrence of the costs is shown on y-axis. The average closed loop cost achieved by the FHOCP controller is 75.5, DAMPC is 81.5 and PAMPC is 172.1. It can be seen from the distribution of costs that the FHOCP controller results in a slightly improved performance over DAMPC, especially in the simulations with low closed loop costs. In order to provide more insight on the variance of the costs, the distribution of the costs achieved by PAMPC and DAMPC is plotted as a function of  $\theta^*$  in Fig. 5, where the costs are averaged over the 4 realizations of  $w_k$  for each  $\theta^*$ . In this figure, each circle is centered at the true parameter and its area represents the averaged closed loop cost for that parameter. The figure can be analyzed in four quadrants. In the upper-

left quadrant, both controllers show similar performance and relatively small costs. This represents a region where passive exploration is sufficient to reduce the uncertainty and track the given trajectory. In the upper- and lower-right quadrants, the DAMPC algorithm shows noticeably smaller costs. This indicates that systems from this region of parameter space benefit from active uncertainty learning. Finally, most systems from the lower-left quadrant have relatively higher costs for both algorithms. This is because some of the true input setpoints  $u_k^*$  lie outside the feasible region for these systems.

The closed loop trajectories of the system for one realization of the true parameters ( $\theta^* = [-0.06, 0.07]^T$ ) are shown in Fig. 6. Only the first 60 time steps of the simulation are shown, because both DAMPC and PAMPC controllers identify the parameters and have a similar response after this time for this system. For the DAMPC controller, three different values of  $N_\theta$  were used, and the trajectories are labeled as  $D_2, D_5, D_8$  for  $N_\theta=2, 5, 8$  respectively. It can be seen that all the DAMPC controllers excite the system before  $t = 20$ , although the reference to be tracked is 0 and the system is already close to the origin. This is an exploratory input signal, which the controller activates immediately after a change of reference setpoint is seen in the prediction horizon. The  $D_2$  controller uses large input signals to identify the plant, since it has assumed a short experiment length for identification. The  $D_5$  and  $D_8$  controllers use smaller excitation signals, and achieve lower closed loop costs. The PAMPC controller does not use any exploratory excitation in the initial part of the trajectory, as expected. This means that the uncertainty set used by the PAMPC controller when the setpoint change occurs is difficult to ensure robustness compared to the one used by the DAMPC controllers, and it results in a poor setpoint tracking performance. Moreover, it can be seen that due to the large uncertainty in the second input channel, the PAMPC controller does not fully use  $[u]_2$  until  $t = 34$  when passive exploration has considerably reduced the uncertainty.

The simulations were performed using an Intel Xeon Gold 5118 processor with 2GB RAM. All the optimization problems were implemented using YALMIP [49]. The optimization problem of the PAMPC algorithm is a linear program, and was solved using MOSEK [50]. The average computation times for solving the PAMPC optimization problem was 0.5s. The DAMPC optimization problem is nonconvex, due to the bilinear equalities and inequalities in (47). This optimization problem was solved using IPOPT [51], which uses an interior-point based line-search algorithm. The optimization problems were warm-started with the solution obtained at the previous time step. The average computation times for solving the DAMPC algorithm for  $N_\theta = 2, 5, 8$  were 0.5s, 1.1s, 1.8s respectively. The average values of the relative increase in closed loop cost, of DAMPC compared to FHOC were 16.3%, 11.6%, 10.8% for  $N_\theta = 2, 5, 8$  respectively. Thus, it can be seen that increasing  $N_\theta$  improves performance and also increases the computational cost.

### B. Six state example

As a second simulation example, we consider a mass-spring damper system with three masses  $m_1, m_2, m_3$ , each weighing

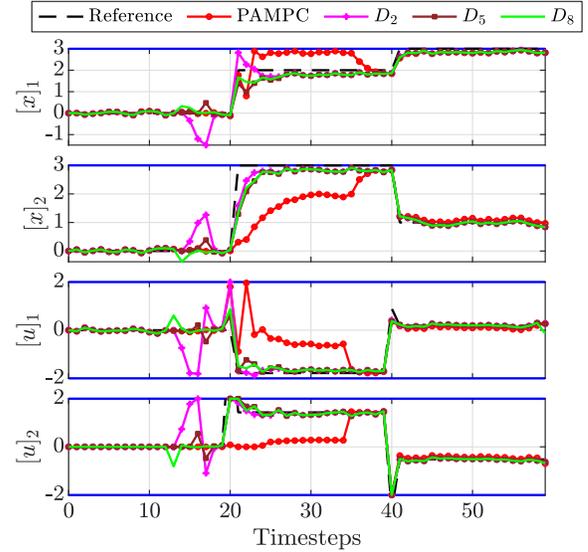


Fig. 6: Comparison of trajectories for PAMPC and DAMPC (with  $N_\theta = 2, 5, 8$ ) for the first 60 timesteps. The true state and input setpoints are shown as a dashed line, and the constraints are shown in blue. Recall that the controllers do not have knowledge of the true input setpoints.

1kg and connected along a line by two springs and dampers. Using the position and velocity of each mass as the state of the system, the state matrix can be written as

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -k_{12} & -c_{12} & k_{12} & c_{12} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ k_{12} & c_{12} & -k_{12}-k_{23} & -c_{12}-c_{23} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & k_{23} & c_{23} & -k_{23} & -c_{23} \end{bmatrix},$$

and the input matrix as

$$B = \begin{bmatrix} 0 & F_g & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & F_g & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & F_g \end{bmatrix}^T$$

where  $k_{ij}$  and  $c_{ij}$  represent the spring constant and damping coefficient of the elements connecting masses  $i$  and  $j$  respectively. In addition,  $F_g$  is an actuator gain on the control input to give the force acting on each mass. The spring constants have the nominal values  $k_{12} = 3.2Nm^{-1}$ ,  $k_{23} = 5.8Nm^{-1}$  and an uncertainty of  $\pm 10\%$  of their nominal value. The damping coefficients are given as  $c_{12} = 2.3Nsm^{-1}$ ,  $c_{23} = 4.5Nsm^{-1}$  with a  $\pm 5\%$  uncertainty. The actuator gain  $F_g = 6.4$  with a  $\pm 7\%$  uncertainty. The system dynamics are discretized using Euler discretization with a sampling time of 0.1s to preserve the uncertainty structure. The resulting model has 6 states (positions and velocities of the masses) and 3 inputs. Additionally, an additive disturbance with a maximum magnitude of 0.05 is acting on each state of the system. The outputs to be tracked are the positions of the masses, and a reference trajectory of 75 timesteps is considered. The states and inputs of the system are constrained to be within the bounds  $\pm 5$ . The cost function is defined by the matrices  $Q = 2I_3$  and  $R = I_3$ .

The tracking performance with PAMPC and DAMPC controllers is shown in Figure 7, where the highest and lowest

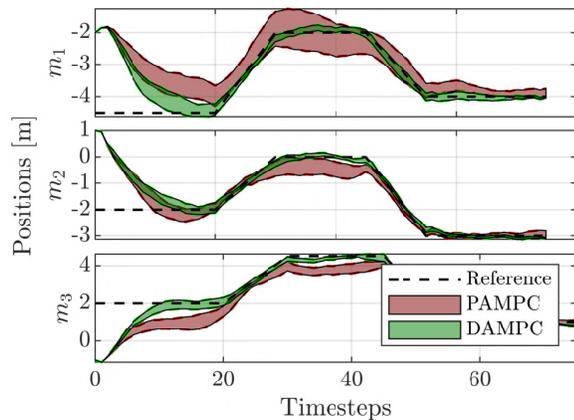


Fig. 7: Reference tracking performance of the DAMPC and PAMPC controllers when applied to the mass spring damper example. For each mass, the highest and lowest values of positions at each timestep over 150 simulations are plotted.

values of the positions at each timestep over 150 simulations (50 random realizations of  $\theta^*$ , each with 3 random realizations of  $w_k$ ) are plotted. The DAMPC controller has a prediction horizon of  $N=6$  steps and lookahead horizon  $N_\theta=4$ . The set  $\mathbb{X}_0$  is designed using Algorithm 1, and is defined by 24 hyperplanes and 322 vertices. The system is affected by 5 parameters (2 springs, 2 dampers and 1 actuator gain), and the uncertainty set  $\Theta_k$  is represented as a hyperbox in 5 dimensions. Due to the large size of the resulting optimization problem, it was chosen to use the tube inclusion approximations proposed in Section IV-C. The PAMPC controller also uses a prediction horizon of  $N=6$  steps. It can be seen that the DAMPC controller tracks the reference better, as shown by the closeness of the upper and lower bounds to the reference trajectory.

The closed loop cost achieved by the DAMPC controller is 85.4, and that of PAMPC is 135.6. This improved performance comes at the cost of computational complexity. The simulations were performed on a Intel Xeon E3-1585Lv5 processor with 6GB RAM. The average computation time to solve the DAMPC optimization problem was 110.7s and that of the PAMPC optimization problem is 105.4s. Note that the DAMPC optimization problem was solved using the tube inclusion approximations proposed in Section IV-C, while the PAMPC optimization problem was setup using tube inclusion constraints at each vertex.

## VI. CONCLUSIONS

In this paper, we propose an explicit dual control approach to approximate a finite horizon optimal tracking problem for uncertain linear systems. In order to learn the uncertainty, an adaptive MPC scheme is used. A tracking formulation is proposed to enable the computation of online terminal sets which are robustly invariant. This is particularly beneficial for adaptive MPC algorithms, since the uncertainty set is reduced online. In addition, the cost function uses a predicted state tube in order to model the dual effect of the control input, and the contractivity of the proposed terminal sets is used to approximate the cost-to-go of the optimal control problem. The

proposed algorithm guarantees recursive feasibility in the face of changing reference trajectories and parameter estimates. The resulting optimization problem is nonconvex due to terms in the constraints, and its size can grow combinatorially with the state dimension of the system. A novel tube inclusion approximation is proposed to prevent this growth, which can also be applied to other adaptive MPC schemes in literature. Two simulation studies highlight the performance improvement achieved by the proposed algorithm over an adaptive MPC approach which does not model the dual control effect.

An interesting direction for future research is to extend the algorithm for time varying systems, where exploration would be beneficial throughout the control horizon. This is not the case for time invariant systems, since exploration can be stopped once the uncertainty set has been reduced to a small size. Another promising direction is to implement the algorithm using ellipsoidal sets to parameterize the state tube, which would simplify the offline design of contractive sets.

## REFERENCES

- [1] K. Åström and B. Wittenmark, *Adaptive Control*. Dover Publications, 2008.
- [2] A. Feldbaum, “Dual control theory. I-IV,” *Avtomatika i Telemekhanika*, vol. 21, no. 9, pp. 1240–1249, 1960-61.
- [3] A. Mesbah, “Stochastic model predictive control with active uncertainty learning: A survey on dual control,” *Annual Reviews in Control*, vol. 45, pp. 107–117, 2018.
- [4] N. Filatov and H. Unbehauen, “Survey of adaptive dual control methods,” *IEE Proceedings - Control Theory and Applications*, vol. 147, pp. 118–128, 2000.
- [5] E. Tse and Y. Bar-Shalom, “An actively adaptive control for linear systems with random parameters via the dual control approach,” *IEEE Transactions on Automatic Control*, vol. 18, no. 2, pp. 109–117, 1973.
- [6] D. S. Bayard and A. Schumitzky, “Implicit dual control based on particle filtering and forward dynamic programming,” *International Journal of Adaptive Control and Signal Processing*, vol. 24, pp. 155–177, 2010.
- [7] W. B. Powell, *Approximate Dynamic Programming: Solving the Curses of Dimensionality (Wiley Series in Probability and Statistics)*. Wiley-Interscience, 2007.
- [8] J. M. Lee and J. H. Lee, “An approximate dynamic programming based approach to dual adaptive control,” *Journal of Process Control*, vol. 19, no. 5, pp. 859–864, 2009.
- [9] E. Arcari, L. Hewing, and M. N. Zeilinger, “An approximate dynamic programming approach for dual stochastic model predictive control,” *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 8105–8111, 2020.
- [10] R. Milito, C. Padilla, R. Padilla, and D. Cadorin, “An innovations approach to dual control,” *IEEE Transactions on Automatic Control*, vol. 27, no. 1, pp. 132–137, 1982.
- [11] L. Pronzato, C. Kulcsar, and E. Walter, “An actively adaptive control policy for linear models,” *IEEE Transactions on Automatic Control*, vol. 41, no. 6, pp. 855–858, 1996.
- [12] A. Weiss and S. Di Cairano, “Robust dual control MPC with guaranteed constraint satisfaction,” in *53rd IEEE Conference on Decision and Control*, 2014.
- [13] T. A. N. Heirung, B. E. Ydstie, and B. Foss, “Dual adaptive model predictive control,” *Automatica*, vol. 80, pp. 340–348, 2017.
- [14] R. Soloperto, J. Köhler, M. A. Müller, and F. Allgöwer, “Dual adaptive MPC for output tracking of linear systems,” in *2019 IEEE 58th Conference on Decision and Control (CDC)*. IEEE, 2019, pp. 1377–1382.
- [15] A. Iannelli and R. S. Smith, “A Multiobjective LQR Synthesis Approach to Dual Control for Uncertain Plants,” *IEEE Control Systems Letters*, vol. 4, no. 4, pp. 952–957, 2020.
- [16] C. A. Larsson, A. Ebadat, C. R. Rojas, X. Bombois, and H. Hjalmarsson, “An application-oriented approach to dual control with excitation for closed-loop identification,” *European Journal of Control*, vol. 29, pp. 1–16, 2016.
- [17] L. Hewing, K. P. Wabersich, M. Menner, and M. N. Zeilinger, “Learning-based model predictive control: Toward safe learning in control,” *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 3, pp. 269–296, 2020.

- [18] F. Borrelli, A. Bemporad, and M. Morari, *Predictive control for linear and hybrid systems*. Cambridge University Press, 2017.
- [19] W. Langson, I. Chrysochoos, S. Raković, and D. Q. Mayne, “Robust model predictive control using tubes,” *Automatica*, vol. 40, no. 1, pp. 125–133, 2004.
- [20] B. Kouvaritakis and M. Cannon, *Model Predictive Control: Classical, Robust and Stochastic*. Springer International Publishing, 2015.
- [21] M. Lorenzen, F. Allgöwer, and M. Cannon, “Adaptive model predictive control with robust constraint satisfaction,” *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 3313–3318, 2017.
- [22] M. Lorenzen, M. Cannon, and F. Allgöwer, “Robust MPC with recursive model update,” *Automatica*, vol. 103, 2019.
- [23] M. Milanese and A. Vicino, “Optimal estimation theory for dynamic systems with set membership uncertainty: an overview,” *Automatica*, vol. 27, no. 6, pp. 997–1009, 1991.
- [24] V. Adetola and M. Guay, “Robust adaptive MPC for constrained uncertain nonlinear systems,” *International Journal of Adaptive Control and Signal Processing*, vol. 25, no. 2, pp. 155–167, 2011.
- [25] J. Köhler, P. Kötting, R. Soloperto, F. Allgöwer, and M. A. Müller, “A robust adaptive model predictive control framework for nonlinear uncertain systems,” *International Journal of Robust and Nonlinear Control*, 2020.
- [26] A. Cohen, T. Koren, and Y. Mansour, “Learning linear-quadratic regulators efficiently with only  $\sqrt{T}$  regret,” in *Proceedings of the 36th International Conference on Machine Learning*, 2019, pp. 1300–1309.
- [27] E. Hazan, S. Kakade, and K. Singh, “The nonstochastic control problem,” in *Proceedings of the 31st International Conference on Algorithmic Learning Theory*, vol. 117, 2020, pp. 408–421.
- [28] A. Parsi, A. Iannelli, and R. S. Smith, “Active exploration in adaptive model predictive control,” in *59th IEEE Conference on Decision and Control (CDC)*, 2020.
- [29] U. Maeder, F. Borrelli, and M. Morari, “Linear offset-free model predictive control,” *Automatica*, vol. 45, no. 10, pp. 2214–2222, 2009.
- [30] M. N. Zeilinger, M. Morari, and C. N. Jones, “Soft constrained model predictive control with robust stability guarantees,” *IEEE Transactions on Automatic Control*, vol. 59, no. 5, pp. 1190–1202, 2014.
- [31] S. Di Cairano and F. Borrelli, “Reference tracking with guaranteed error bound for constrained linear systems,” *IEEE Transactions on Automatic Control*, vol. 61, no. 8, pp. 2245–2250, 2015.
- [32] M. Pereira, D. M. de la Pena, D. Limón, I. Alvarado, and T. Alamo, “Robust model predictive controller for tracking changing periodic signals,” *IEEE Transactions on Automatic Control*, vol. 62, no. 10, pp. 5343–5350, 2016.
- [33] G. Pannocchia, “Robust model predictive control with guaranteed set-point tracking,” *Journal of Process control*, vol. 14, pp. 927–937, 2004.
- [34] B. Zhu and X. Xia, “Adaptive model predictive control for unconstrained discrete-time linear systems with parametric uncertainties,” *IEEE Transactions on Automatic Control*, vol. 61, no. 10, pp. 3171–3176, 2015.
- [35] D. P. Bertsekas, “Dynamic programming and suboptimal control: A survey from ADP to MPC,” *European Journal of Control*, vol. 11, no. 4, pp. 310–334, 2005.
- [36] E. D. Klenke and P. Hennig, “Dual control for approximate Bayesian reinforcement learning,” *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 4354–4383, 2016.
- [37] S. V. Raković, B. Kouvaritakis, R. Findeisen, and M. Cannon, “Homothetic tube model predictive control,” *Automatica*, vol. 48, no. 8, pp. 1631–1638, 2012.
- [38] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.
- [39] S. O. Haykin, *Adaptive Filter Theory*. Pearson, 2014.
- [40] M. V. Kothare, V. Balakrishnan, and M. Morari, “Robust constrained model predictive control using linear matrix inequalities,” *Automatica*, vol. 32, no. 10, pp. 1361–1379, 1996.
- [41] A. Didier, A. Parsi, J. Coulson, and R. S. Smith, “Robust Adaptive Model Predictive Control of Quadrotors,” in *European Control Conference*, 2021.
- [42] I. Kolmanovskiy and E. G. Gilbert, “Theory and computation of disturbance invariant sets for discrete-time linear systems,” *Mathematical problems in engineering*, vol. 4, no. 4, pp. 317–367, 1998.
- [43] F. Blanchini and S. Miani, *Set-theoretic methods in control*. Springer, 2008.
- [44] D. Rubin, H.-N. Nguyen, and P.-O. Gutman, “Computation of polyhedral positive invariant sets via linear matrix inequalities,” in *2018 European Control Conference (ECC)*. IEEE, 2018, pp. 2941–2946.
- [45] A. Aswani, H. Gonzalez, S. S. Sastry, and C. Tomlin, “Provably safe and robust learning-based model predictive control,” *Automatica*, vol. 49, no. 5, pp. 1216–1226, 2013.
- [46] J. Köhler, E. Andina, R. Soloperto, M. A. Müller, and F. Allgöwer, “Linear robust adaptive model predictive control: Computational complexity and conservatism,” in *58th IEEE Conference on Decision and Control (CDC)*. IEEE, 2019, pp. 1383–1388.
- [47] A. Parsi, A. Aboudonia, A. Iannelli, J. Lygeros, and R. S. Smith, “A distributed framework for linear adaptive MPC,” in *60th IEEE Conference on Decision and Control (CDC)*, 2021.
- [48] A. Parsi. (2021) Supplementary material for the journal paper titled “An explicit dual control approach for constrained reference tracking of uncertain linear systems”. [Online]. Available: <https://doi.org/10.3929/ethz-b-000518264>
- [49] J. Löfberg, “YALMIP : A toolbox for modeling and optimization in MATLAB,” in *In Proceedings of the CACSD Conference*, Taipei, 2004.
- [50] M. ApS, *The MOSEK optimization toolbox for MATLAB manual. Version 9.0.*, 2019. [Online]. Available: <http://docs.mosek.com/9.0/toolbox/index.html>
- [51] A. Wächter and L. T. Biegler, “On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming,” *Mathematical programming*, vol. 106, no. 1, pp. 25–57, 2006.



**Anilkumar Parsi** completed his B.Tech (Hons) in Mechanical Engineering at the Indian Institute of Technology Madras in 2016. He received an M.Sc. in Mechanical Engineering from ETH Zürich in 2019. He is currently a Ph.D. student in the Automatic Control Laboratory at ETH Zürich. His research interests include model predictive control, robust and learning based control.



control systems, data-driven methods, and dynamical systems theory.

**Andrea Iannelli** completed the B.Eng. and M.Sc. degrees in Aerospace Engineering at the University of Pisa and he received his PhD from the University of Bristol. In his doctoral studies he worked on the reconciliation between robust control and dynamical systems theory, with application to linear and nonlinear aeroelastic problems. He is currently a postdoctoral researcher in the Automatic Control Laboratory at ETH Zürich. His research interests include modeling, analysis and synthesis of uncertain



control and power system design. His research interests involve the modeling, identification, and control of uncertain systems. Particular control application domains of interest include chemical processes, flexible structure vibration, spacecraft and vehicle formations, aerodynamic control of kites, automotive engines, Mars aeromaneuvering entry design, building energy control, and thermoacoustic machines. He is a Fellow of the IEEE and the IFAC, an Associate Fellow of the AIAA, and a member of SIAM.

**Roy S. Smith** is a professor of Electrical Engineering at the Swiss Federal Institute of Technology (ETH), Zürich. Prior to joining ETH in 2011, he was on the faculty of the University of California, Santa Barbara, from 1990 to 2010. His Ph.D. is from the California Institute of Technology (1990) and his undergraduate degree is from the University of Canterbury (1980) in his native New Zealand. He has been a long-time consultant to the NASA Jet Propulsion Laboratory and has industrial experience in automotive