MITSUBISHI ELECTRIC RESEARCH LABORATORIES
https://www.merl.com

Indirect Adaptive Model Predictive Control and its Application to Uncertain Linear Systems

Di Cairano, Stefano; Danielson, Claus

TR2020-115 July 22, 2020

Abstract

We consider constrained systems that are represented by uncertain models with unknown constant or slowly varying parameters. We propose an indirect adaptive model predictive control (IAMPC) approach where the prediction model can be adjusted during controller operation by a separately designed estimator that satisfies only a minimal set of assumptions. The controller guarantees constraint satisfaction despite the uncertainty in the parameters by means of a robust control invariant set, and input-to-state stability with respect to the estimation error by means of an appropriately designed method for adjusting the IAMPC prediction model and cost function based on the evolution of the parameter estimate. The controller has minimal computational overhead with respect to a nominal MPC and for the special case of uncertain linear systems, we obtain a constructive design procedure for the IAMPC which only solves quadratic programs during closed-loop control.

International Journal of Robust and Nonlinear Control

Received: Added at production | Revised: Added at production | Accepted: Added at production
DOI: xxx/xxxx

ARTICLE TYPE

Indirect Adaptive Model Predictive Control and its Application to Uncertain Linear Systems

Stefano Di Cairano* | Claus Danielson

[1]Mitsubishi Electric Research Laboratories, Cambridge, MA, USA

Correspondence
*Corresponding author,
Email: dicairano@ieee.org

Summary

We consider constrained systems that are represented by uncertain models with unknown constant or slowly varying parameters. We propose an indirect adaptive model predictive control (IAMPC) approach where the prediction model can be adjusted during controller operation by a separately designed estimator that satisfies only a minimal set of assumptions. The controller guarantees constraint satisfaction despite the uncertainty in the parameters by means of a robust control invariant set, and input-to-state stability with respect to the estimation error by means of an appropriately designed method for adjusting the IAMPC prediction model and cost function based on the evolution of the parameter estimate. The controller has minimal computational overhead with respect to a nominal MPC and for the special case of uncertain linear systems, we obtain a constructive design procedure for the IAMPC which only solves quadratic programs during closed-loop control.

KEYWORDS:
Model predictive control, Adaptive control, Constrained control, Uncertain systems

1 | INTRODUCTION

Both model-based and data-driven methods for control design have strengths and weaknesses. Model-based control design relies on first principle models that avoid the need for time consuming data collection campaigns and allow for reasoning on the system structure, yet they are almost never exact and may fail to capture some relevant system behaviors. Data-based control design represents all the behaviors that are visible from data, but it may require large amounts of data to capture infrequent or weakly observable behaviors, and most of the times obfuscates the internal structure of the system. There is a general agreement in the control system community that the ideal solution is to develop control design methods that allow for the exploitation of both the a-priori knowledge of the models and the operational information provided by data. However, combining these two approaches, which often are contrasting and even competing, is not straightforward.

A control design method that may significantly benefit from a synergy between model-based and data-driven methods is model predictive control (MPC)[1]. In MPC a prediction model is used to evaluate feasibility and performance of the sequence of actions to be selected by the controller. In standard (i.e., nominal) MPC the prediction model is assumed to be perfect, which allows for relatively simple verification of controller properties, such as constraint enforcement and closed-loop stability. However, in most applications some of the model parameters are uncertain at design time, especially when a controller is deployed to multiple instances of the plant, such as in automotive, factory automation, and aerospace applications[2,3], where the control algorithm and its auxiliary functions also need to have low complexity and computational effort, due to stringent cost, timing, and validation requirements. In several cases uncertainty is ignored, relying on the intrinsic robustness of MPC due to its use of feedback.

On the other hand, different approaches may be taken to explicitly deal with the model uncertainty and to retain some of the closed-loop guarantees, thus avoiding unexpected performance degradation and/or constraint violations[4].

Robust MPC methods is one class of approaches for dealing with model uncertainty, see, e.g.,[5,6,7,8], which retain certain stability and constraint satisfaction properties in closed-loop. In these approaches, uncertain parameters are often considered to be capable of changing arbitrarily, from one instant to the next one, which results in relatively cautious control. All these methods have limitations due to making only minimal assumptions about the uncertainty, for instance, being designed to only handle additive disturbances[7], requiring a very specific choice of cost function and terminal set[8], or imposing a significant increase in computational cost, due to solving linear matrix inequalities (LMIs) at each control step[5,6,9]. In fact, since the uncertainty is assumed to be constantly changing, robust MPC methods cannot exploit real-time adaptation based on acquired data for improving the performance.

When the uncertain parameters are unknown, but constant or slowly varying, an alternative approach is to learn their values based on data collected during operation, resulting in adaptive control techniques that ensure safe operation during the learning phase and improve performance as the parameters are learned. Adaptive MPC algorithms have been recently proposed based on different methods, such as a comparison model[10], min-max approaches with open-loop relaxations[11], learning of constant offsets[12], and set membership identification[13,14]. With the recent emphasis on machine learning applications in control, several approaches have been proposed for learning MPC, which achieve similar results to adaptive MPC by using machine learning techniques, such as iterative estimation of disturbance models based on Gaussian processes[15,16,17] or iterative construction of safe sets[18]. Another class of adaptive MPC algorithms focuses on “dual objective control”, i.e., controlling the uncertain system while providing sufficient excitation for identification, see, e.g.,[19,20,21,22,23,24,25], and references therein. In these approaches the cost function of MPC is usually augmented with a cost term related to the amount of information about the system that can be acquired by issuing a certain sequence of commands, and possibly by a model of the uncertainty update by the parameter estimation algorithm. Thus, in several of these approaches some knowledge of the parameter estimator is required and the MPC optimization problem often involve matrix terms and inequalities.

The goal of this paper is to develop an adaptive MPC design that allows for the model to be adjusted during operation while retaining constraint satisfaction and stability properties, and incurring only in a limited increase in computational overhead compared to nominal MPC. The model adjustment is achieved by a general estimator or learning algorithm, so that the same closed-loop properties are achieved regardless of how the parameters are estimated. This produces an indirect adaptive MPC (IAMPC) approach that achieves a separation principle between controller and parameter estimator design, which normally does not hold for MPC due to its predictive nature and the presence of constraints.

IAMPC combines model-based and data-based approaches by considering first principle models with specific uncertain parameters, so that the plant can be represented as a family of systems. By designing components of the optimal control problem to be dependent on the values of the learned parameters, the IAMPC can achieve robust constraint satisfactions through the computation of appropriate robust control invariant set, and input-to-state stability (ISS) with respect to the estimation error. By the definition of ISS, if the estimation improves, the performance of the control algorithm improves. Furthermore, if the estimated parameter converges to the correct value, the closed-loop will become asymptotically stable (AS). Rather than aiming to achieve finite time convergence by a specific estimator design, see, e.g.,[26]), here we will show that it is enough for the estimation error to become sufficiently small. Regardless of actually achieving AS and ISS, constraint satisfaction will hold even if the unknown parameters are continuously changing.

After deriving general results for nonlinear systems, we introduce a constructive design procedure for uncertain linear systems, that we re-formulate as uncertain polytopic systems, i.e., systems whose unknown system matrix lie in a matrix polytope. The linear IAMPC uses a parameter-dependent quadratic terminal cost which is designed offline using parameter-dependent Lyapunov function (pLF)[27]. The pLF and its corresponding stabilizing control law are used to achieve ISS and to design the terminal set. Robust constraint satisfaction in presence of parameter estimation error is achieved by using a robust control invariant (RCI) set designed for a polytopic linear difference inclusions (pLDI)[28]. The proposed IAMPC allows uncertainty in the system dynamics, as opposed to additive disturbances or offsets as in[7,12], yet it only requires solving quadratic programs (QPs), as opposed to robust MPC methods that require the online solution of LMIs[5,6,9], which is often intractable in microcontrollers for applications such as those in[2,3].

The paper is structured as follows[1]. After defining the problem in Section 2, in Section 3 we present conditions on the IAMPC design that provide the desired closed-loop properties of the system for the general nonlinear case. In Section 4 we specialize the

[1]Preliminary results related to this research were presented in[29,30,31]. In this paper we generalize those results to nonlinear systems, which also allows to generalize the design for the linear case in multiple directions, and results in more general proofs of the closed-loop properties, with fewer conditions and assumptions.

IAMPC design to uncertain polytopic systems, and describe a design process that achieves the previously introduced conditions that guarantee the desired properties. In Section 5 we discuss further extensions to the IAMPC design for uncertain linear systems that allow us to tackle problems that are relevant in practice. In Section 6 we demonstrate the operation of the method in two case studies. First, an illustrative uncertain second order system for which the algorithm properties are simple to visualize. Second, a more realistic application to air conditioning system control, which suggests that the method is practical in real applications. Conclusions and future developments are discussed in Section 7.

1.1 | Notation

$\mathbb{R}$, $\mathbb{R}_{0+}$, $\mathbb{R}_{+}$, $\mathbb{Z}$, $\mathbb{Z}_{0+}$, $\mathbb{Z}_{+}$ denote the sets of real, nonnegative real, positive real, and integer, nonnegative integer, positive integer numbers. We denote interval of numbers using notations as $\mathbb{Z}_{[a,b)} = \{z \in \mathbb{Z} : a \leq z < b\}$. $\mathrm{co}\{\mathcal{X}\}$ denotes the convex hull of the set $\mathcal{X}$. For vectors, inequalities are intended componentwise, while for matrices indicate (semi)definiteness, and $\lambda_{\min}(Q)$ $\lambda_{\max}(Q)$ denote the smallest and largest eigenvalues of the matrix Q, respectively. By $[x]_i$ we denote the i-th component of vector x,the stacking of two vectors by $(x, y) = [x'\ y']'$ and by I and 0 the identity and the "all-zero" matrices of appropriate dimension. $\|\cdot\|_p$ denotes the p-norm, and $\|\cdot\| = \|\cdot\|_2$. For a discrete-time signal $x \in \mathbb{R}^n$ with sampling period T_s, $x(t)$ is the state a sampling instant t, i.e., at time $T_s t$, $x_{k|t}$ denotes the predicted value of x at sample $t+k$, i.e., $x(t+k)$, based on data at sample t, and $x_{0|t} = x(t)$. A function $\alpha : \mathbb{R}_{0+} \to \mathbb{R}_{0+}$ is of class $\mathcal{K}$ if it is continuous, strictly increasing and $\alpha(0) = 0$; if in addition $\lim_{c\to\infty} \alpha(c) = \infty$, α is of class $\mathcal{K}_\infty$.

1.2 | Preliminary Definitions and Results

We provide the following standard definitions and results, which are discussed at length for instance in [1, Appendix B].

Definition 1. Given $x(t+1) = f(x(t), w(t))$, $x \in \mathbb{R}^n$, $w \in \mathcal{W} \subseteq \mathbb{R}^d$, a set $\mathcal{S} \subset \mathbb{R}^n$ is robust positive invariant (RPI) for f iff for all $x \in \mathcal{S}$, $f(x, w) \in \mathcal{S}$, for all $w \in \mathcal{W}$. If $w = \{0\}$, $\mathcal{S}$ is simply called positive invariant (PI). □

Definition 2. Given $x(t+1) = f(x(t), u(t), w(t))$, $x \in \mathbb{R}^n$, $u \in \mathcal{U} \subseteq \mathbb{R}^m$, $w \in \mathcal{W} \subseteq \mathbb{R}^d$, a set $\mathcal{S} \subset \mathbb{R}^n$ is robust control invariant (RCI) for f iff for all $x \in \mathcal{S}$, there exists $u \in \mathcal{U}$ such that $f(x, u, w) \in \mathcal{S}$, for all $w \in \mathcal{W}$. If $w = \{0\}$, $\mathcal{S}$ is simply called control invariant (CI). □

Definition 3. Given $x(t+1) = f(x(t))$, $x \in \mathbb{R}^n$, and a PI set $\mathcal{S}$ for f, $0 \in \mathcal{S}$, a function $\mathcal{V} : \mathbb{R}^n \to \mathbb{R}_{0+}$ such that there exists $\alpha_1, \alpha_2, \alpha_\Delta \in \mathcal{K}_\infty$ such that $\alpha_1(\|x\|) \leq \mathcal{V}(x) \leq \alpha_2(\|x\|)$, $\mathcal{V}(f(x)) - \mathcal{V}(x) \leq -\alpha_\Delta(\|x\|)$ for all $x \in \mathcal{S}$ is a Lyapunov function for f in $\mathcal{S}$. □

Definition 4. Given $x(t+1) = f(x(t), w(t))$, $x \in \mathbb{R}^n$, $w \in \mathcal{W} \subseteq \mathbb{R}^d$, and a RPI set $\mathcal{S}$ for f, $0 \in \mathcal{S}$, a function $\mathcal{V} : \mathbb{R}^n \to \mathbb{R}_+$ such that there exists $\alpha_1, \alpha_2, \alpha_\Delta \in \mathcal{K}_\infty$ and $\gamma \in \mathcal{K}$ such that $\alpha_1(\|x\|) \leq \mathcal{V}(x) \leq \alpha_2(\|x\|)$, $\mathcal{V}(f(x)) - \mathcal{V}(x) \leq -\alpha_\Delta(\|x\|) + \gamma(\|w\|)$ for all $x \in \mathcal{S}$, $w \in \mathcal{W}$ is an input-to-state stable (ISS) Lyapunov function for f in $\mathcal{S}$ with respect to w. □

Result 1. Given $x(t+1) = f(x(t))$, $x \in \mathbb{R}^n$, and a PI $\mathcal{S}$ for f, $0 \in \mathcal{S}$, if there exists a Lyapunov function for f in $\mathcal{S}$, the origin is asymptotically stable (AS) for f with domain of attraction $\mathcal{S}$. Given $x(t+1) = f(x(t), w(t))$, $x \in \mathbb{R}^n$, $w \in \mathcal{W} \subseteq \mathbb{R}^d$, and a RPI $\mathcal{S}$ for f, $0 \in \mathcal{S}$, if there exists a ISS Lyapunov function for f in $\mathcal{S}$, the origin is ISS for f with respect to w with domain of attraction $\mathcal{S}$. □

2 | PROBLEM DEFINITION

Consider the state and input admissible sets, $\mathcal{X} \subseteq \mathbb{R}^n$, $\mathcal{U} \subseteq \mathbb{R}^m$, with origin in their interior, $0 \in \mathrm{int}(\mathcal{X})$, $0 \in \mathrm{int}(\mathcal{U})$, and a system represented by

$$x(t+1) = \bar{f}_\vartheta(x(t), u(t)) \tag{1}$$

where $x \in \mathcal{X}$ is the state vector, $u \in \mathcal{U}$ is the input vector, and $\{\bar{f}_\vartheta\}_{\vartheta\in\Theta}$ is a family of functions selected by the vector of physical parameters $\vartheta \in \Theta \subset \mathbb{R}^p$, where Θ is bounded. Here, the physical parameter vector is assumed to be constant or slowly varying with respect to the plant dynamics, and we denote its actual value by $\bar{\vartheta}$ according to (1).

Our goal is to stabilize the uncertain system (1) and enforce its state and input constraints using MPC. Using a family of prediction models $\{f_\xi\}_{\xi\in\Xi}$, parametrized by the vector $\xi \in \Xi$ (that may or may not be the same as $\vartheta \in \Theta$) MPC solves the following finite horizon optimal control problem in a receding horizon fashion

$$\mathcal{V}_{\xi_N(t)}(x(t)) = \min_{U_N(t)} F_{\xi_{N|t}}(x_{N|t}) + \sum_{k=0}^{N-1} L(x_{k|t}, u_{k|t}) \tag{2a}$$

$$\text{s.t. } x_{k+1|t} = f_{\xi_{k|t}}(x_{k|t}, u_{k|t}) \tag{2b}$$

$$(x_{k|t}, u_{k|t}) \in \mathcal{C}_{xu} \subseteq \mathcal{X} \times \mathcal{U} \tag{2c}$$

$$x_{N|t} \in \mathcal{X}_N \tag{2d}$$

$$x_{0|t} = x(t) \tag{2e}$$

where $\mathcal{C}_{xu} \subseteq \mathcal{X} \times \mathcal{U}$ is an admissible joint (sub)set of states and inputs, F_ξ, L are the terminal and stage cost, respectively, $\mathcal{X}_N$ is the terminal set, N is the prediction horizon, and $\mathcal{V}_{\xi_N} : \mathcal{X} \to \mathbb{R}_{0+}$ is the value function. The uncertainty parametrization in the family of prediction models $\{f_\xi\}_{\xi\in\Xi}$ and the family of system models $\{\bar{f}_\vartheta\}_{\vartheta\in\Theta}$ is different, but they share the same state and input vectors. The parameter $\xi \in \Xi$ is allowed to be time-varying since it represents an estimate of $\bar{\vartheta}$, and hence changes over time. The MPC law is

$$u(t) = \kappa_{\text{MPC}}(\xi_N(t), x(t)) = u^*_{0|t} \tag{3}$$

where $\xi_N(t) = [\xi_{0|t} \ \ldots \xi_{N|t}]$, and $U^*_N(t) = [u^*_{0|t} \ \ldots u^*_{N-1|t}]$ is the optimal solution of (2). The control law (3) is designed by constructing $\mathcal{C}_{xu}$, $\mathcal{X}_N$, F_ξ, and N in (2).

If $\bar{\vartheta}$ is perfectly known at design time, (i.e., we can set $f_{\xi_{k|t}} = \bar{f}_{\bar{\vartheta}}$, for all $k \in \mathbb{Z}_{[0,N-1]}$, $t \in \mathbb{Z}_{0+}$), (2) is a nominal MPC whose design amounts to selecting $F_\xi = F$ and $\mathcal{X}_N$, while setting $\mathcal{C}_{xu} = \mathcal{X} \times \mathcal{U}$ and choosing any $N \in \mathbb{Z}_{0+}$. On the other hand, if $\bar{\vartheta}$ is not known and hence ξ is uncertain, there are different options. If the physical parameters $\bar{\vartheta}$ are constantly changing, robust MPC can be used. For instance, tube MPC[7] divides the problem into two parts, disturbance compensation by a feedback gain and control of a nominal system by MPC with input and state constraints tightened based on the worst case closed-loop disturbance, $\mathcal{C}_{xu} \subset \mathcal{X} \times \mathcal{U}$.

When $\bar{\vartheta}$ is constant or only slowly changing, as we consider here, an alternative approach is to estimate $\bar{\vartheta}$. While after $\bar{\vartheta}$ is correctly estimated (2) becomes again a standard MPC, one needs to account for the behavior during the learning transient[11], especially in terms of the enforcement of the constraints and control performance with respect to the estimation error. In this case, the MPC design amounts to determining an update law for the prediction model parameters $\xi_N(t)$ and for other elements in (2) accordingly, such as the terminal cost F_ξ, while ensuring through the design of $\mathcal{C}_{xu}$, $\mathcal{X}_N$, N that the constraints are satisfied and (2) remains feasible throughout the learning transient. Finally, while there may be some benefits from tailoring the MPC adaptation to a specific estimator[26], in order to achieve a modular and flexible design, it may be preferable not to restrict the MPC design to a specific algorithm, but rather to define an update strategy for $\xi_{t|k}$ in (2) based only on the sequence of estimates of $\bar{\vartheta}$, which does not impose requirements on the estimation algorithm used.

Thus, the overall problem tackled in this paper is stated as follows.

Problem 1. Given a suitable family of prediction models $\{f_\xi\}_{\xi\in\Xi}$ and an estimator producing the sequence of estimates $\{\vartheta(t)\}_t$, design an update law $\xi_N(t) = \mathcal{H}(\{\vartheta(t)\}_t)$ and F_ξ, N, $\mathcal{X}_N$ and $\mathcal{C}_{xu}$ in (2) so that the controller (3) achieves: (i) robust satisfaction of the constraints, (ii) input-to-state stability of the closed-loop with respect to estimation error, (iii) computational load comparable to a (non-adaptive) MPC, and guaranteed convergence of the runtime numerical algorithms. □

In Problem 1, (i) is concerned with the system safety in terms of constraints satisfaction, (ii) is concerned with system performance during the transient and after convergence of the estimate, (iii) is concerned with computational requirements for enabling the application to domains with stringent real-time requirements and limited computational capabilities[32,3].

Problem 1 requires robust constraint satisfaction (as in[5,6,7]) and ISS, i.e., a proportional effect of the estimation error on the closed-loop Lyapunov function. The rationale for seeking ISS rather than robust stability as in[5,6,9] is that, when the unknown parameters do not change or change slowly, a "well designed" estimator should eventually converge to the correct value, and hence, by the ISS definition, the closed-loop becomes AS without imposing the often conservative conditions required for robust stability. Furthermore, ISS holds regardless of the estimator convergence. Robust constraint satisfaction must also hold in presence of estimation error, which can be continuously changing especially during the learning transient, and if the parameters change after their values have been learned. Thus, the method for solving Problem 1 must be robust with respect to the safety aspects, i.e., the constraints, and use data to improve the performance aspects, via the cost function that determines the ISS Lyapunov function.

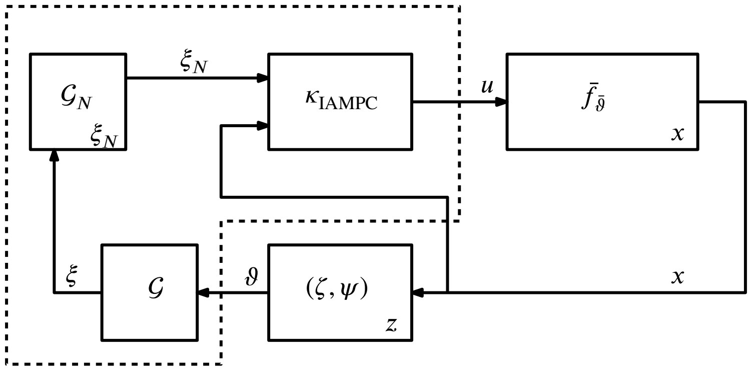


FIGURE 1 Overall architecture of the Indirect Adaptive MPC. The design for the blocks within the dashed lines is described here, while the parameter estimator (ζ, ψ) can be designed independently.

This paper focuses on a control design that is independent from the estimator design, as in Problem 1 a generic estimator is considered, the dependency of the closed-loop performance on estimation error is captured by the ISS gain. In the next Section we will show how the estimator design can be left free of restrictive conditions by an appropriate design of the interface between the estimator and the controller. However, since we rely on the estimator eventually producing a correct or improved estimate, our design seeks ISS rather than robust stabilization. This avoids the need, present instead in tube MPC, to divide the control problem into two parts, rejection of disturbances by a feedback controller and control of the nominal system subject to tightened constraints. Consequently, our method does not require designing a disturbance rejection controller and its invariant set, and allocating the resources, e.g., the control input range and the state constraints, among disturbance rejection and control of the nominal system. Thus, we avoid any loss of performance due to such control allocation, which is likely to occur when the tube MPC uses fixed disturbance invariant sets, or restrictive parameterizations for these.

While (i) and (ii) in Problem 1 can be achieved in a generic setting as shown in the next section, (iii), which is often necessary for practical implementation, imposes some restrictions on the plant (1). In particular, after deriving general conditions for (i) and (ii) based on the general model (1), we propose a design that also achieves (iii) for the specific class of linear systems subject to bounded uncertain parameters in the state transition matrix, and polyhedral constraints

$$x(t+1) = \bar{A}(\vartheta)x(t) + Bu(t) \tag{4a}$$

$$(x(t), u(t)) \in \mathcal{X} \times \mathcal{U} \tag{4b}$$

$$\vartheta \in \Theta, \tag{4c}$$

where Θ is bounded and $\mathcal{X}$, $\mathcal{U}$ are polytopes. Later, we will also discuss the extension to when B is uncertain, i.e., $B = B(\vartheta)$.

3 | INDIRECT ADAPTIVE MODEL PREDICTIVE CONTROL

In this section, we first describe the general architecture of the proposed Indirect Adaptive MPC (IAMPC), which is shown in Figure 1. Then, we discuss the closed-loop properties achieved under appropriate conditions on the components designed in this paper, namely, $\mathcal{G}$, $\mathcal{G}_N$, κ_{IAMPC} in Figure 1, that hold for an independently designed estimator, namely, (ζ, ψ) in Figure 1. Different ways to satisfy such conditions will be presented in later sections.

3.1 | IAMPC Control Architecture

The controller uses a prediction model represented by the family of functions $\{f_\xi\}_{\xi \in \Xi}$ parametrized by an abstract parameter vector $\xi \in \Xi \subset \mathbb{R}^\ell$, possibly different from $\vartheta \in \Theta \subset \mathbb{R}^p$,

$$x(t+1) = f_\xi(x(t), u(t)). \tag{5}$$

where the set Ξ is bounded. The relation between ϑ and ξ is defined by a, possibly multivalued, mapping $\Phi : \Theta_e \to 2^\Xi$, where $\Theta_e \supseteq \Theta$ is the set of allowed values of the estimates for the parameter vector which includes, but it is not restricted to, the set of allowed physical values, Θ. According to (1), we denote the correct value of the physical parameter by $\bar{\vartheta}$, and by $\bar{\xi}$ a corresponding correct value of the virtual parameter, i.e., $\bar{\xi} \in \Phi(\bar{\vartheta})$. The parametric model (5) is consistent with the physical model (1) in the sense that given $\bar{\vartheta} \in \Theta$, for any $\xi \in \Phi(\bar{\vartheta})$, $\bar{f}_{\bar{\vartheta}}(x,u) = f_{\xi}(x,u)$ for all $x \in \mathcal{X}$, $u \in \mathcal{U}$. This implies that for all $\xi_1, \xi_2 \in \Phi(\bar{\vartheta})$, $f_{\xi_1}(x,u) = f_{\xi_2}(x,u)$ for all $x \in \mathcal{X}$, $u \in \mathcal{U}$.

The actual parameter vector $\bar{\vartheta}$ is estimated by a general estimator

$$z(t+1) = \zeta(z(t), x(t), u(t)) \tag{6a}$$

$$\vartheta(t) = \psi(z(t), x(t)), \tag{6b}$$

where $z \in \mathcal{Z} \subseteq \mathbb{R}^e$ is the estimator (internal) state and $\vartheta \in \Theta_e$ is the current estimate. Here, no significant restrictions are imposed on the estimator (6), as the functions ζ and ψ, and the domain set $\mathcal{Z}$ are general. Given ϑ, the function $\mathcal{G} : \Theta_e \to \Xi$ selects a specific value for $\xi \in \Phi(\vartheta)$,

$$\xi = \mathcal{G}(\vartheta) \in \Phi(\vartheta). \tag{7}$$

Let $\xi_N \in \Xi^{N+1}$ denote a sequence of length $N+1$ of values for ξ, and $\xi_N(t) = [\xi_{0|t}, \ldots, \xi_{N|t}] \in \Xi^{N+1}$ denote such a sequence from time t to $t+N$. At any time step such a sequence is updated by the function $\mathcal{G}_N : \Xi^{N+1} \times \Xi \to \Xi^{N+1}$ based on its previous value and the current updated estimate $\xi(t) = \mathcal{G}(\vartheta(t)) \in \Phi(\vartheta(t))$,

$$\xi_N(t) = \mathcal{G}_N(\xi_N(t-1), \xi(t)) = \mathcal{G}_N(\xi_N(t-1), \mathcal{G}(\vartheta(t))). \tag{8}$$

Note that (7), (8) implement the function of $\mathcal{H}$ in Problem 1.

The system input is determined by the control law

$$u(t) = \kappa_{\text{IAMPC}}(\xi_N(t), x(t)) = u^*_{0|t} \tag{9}$$

where $\xi_N(t) = [\xi_{0|t} \; \ldots \xi_{N|t}] \in \Xi^{N+1}$ is the sequence of prediction model parameters, not necessarily constant, and $U_N(t)^* = [u^*_{0|t}, \ldots, u^*_{N-1|t}]$ is the optimizer of (2) where, in addition, $F_{\xi}(x) \geq \alpha_F(\|x\|)$, for all $\xi \in \Xi$, $\alpha_m(\|x\|) \leq L(x,u) \leq \alpha_L(\|x\|)$, for all $u \in \mathcal{U}$, and $\alpha_F, \alpha_L, \alpha_m \in \mathcal{K}_\infty$. We assume that the value function $\mathcal{V}_{\xi_N}$ or the closed-loop (5), (9) are (uniformly) continuous[33].

Remark 1. In (2), $\xi_{N|t}$ affects the terminal cost, while $\xi_{0|t}, \ldots, \xi_{N-1|t}$ only effects the prediction dynamics. This means that when comparing the MPC solutions in two subsequent time steps t and $t+1$, the change in the terminal cost is determined by $\xi_{N|t}$ and $\xi_{N|t+1}$, while the feasibility of the constraints is determined by $\xi_{0|t}, \ldots, \xi_{N-1|t}$ and $\xi_{0|t+1}, \ldots, \xi_{N-1|t+1}$.

3.2 | Closed-loop Properties

In this section, we establish suitable conditions on the IAMPC (7), (8), (9) that produce the desired closed-loop properties for an independently designed estimator (6), in the case of a general plant (1) and general prediction model (5). In the following section we will constructively show how to meet these conditions for the class of systems (4).

For describing the design and proving the properties of IAMPC, we follow a two steps approach. First, we consider design and properties of the nominal case, when the estimated parameter vector is always equal to the actual one, yet possibly time-varying, which account for the estimate update. Then, we include the effects of the parameter uncertainty in its estimate, which is the actual focus of this work, by modifying the design and analysis accordingly. Thus, we start from the case when the physical parameter vector is time-varying but the estimation error is zero, and then we address the actual IAMPC objective when the physical parameter vector is constant, but the parameter estimate is being updated and the estimation error is in general non-zero.

Consider the following conditions

Conditions 1. (Nominal stability)

N1 Terminal set and controller condition. There exist $\mathcal{X}_N \subseteq \mathcal{X}$, and a terminal controller $\kappa : \mathbb{R}^n \times \Xi \to \mathbb{R}^m$ such that

$$\forall x \in \mathcal{X}_N, x_0 \in \mathcal{X}, z_0 \in \mathcal{Z},\ \xi \in \Phi(\psi(z_0, x_0)), \qquad f_{\xi}(x, \kappa(x,\xi)) \in \mathcal{X}_N,\ (x, \kappa(x,\xi)) \in \mathcal{C}_{xu} \subseteq \mathcal{X} \times \mathcal{U}. \tag{10}$$

N2 Terminal cost condition. The cost function terms F_{ξ}, L satisfy

$$\begin{aligned} &\forall x \in \mathcal{X}_N,\ \xi \in \Xi,\ x_0 \in \mathcal{X},\ z_0 \in \mathcal{Z}, \\ &F_{\xi^+}(f_{\xi}(x, \kappa(x,\xi))) - F_{\xi}(x) + L(x, \kappa(x,\xi)) \leq 0,\ \forall \xi^+ \in \Phi(\psi(z_0, x_0)). \end{aligned} \tag{11}$$

In (10) and (11), the state vector x_0 used to compute the virtual parameter vector estimate ξ is allowed to be different from the state vector x used in the terminal cost, controller, and set. This is necessary for use in a predictive controller where the estimate is based on the current data, while the terminal cost, controller and set are applied to a future predicted state.

Define $\Delta\mathcal{V}_{\xi_N(t)}(x(t)) = \mathcal{V}_{\xi_N(t+1)}(f_{\xi(t)}(x(t), u(t))) - \mathcal{V}_{\xi_N(t)}(x(t))$, where $\xi_N(t+1) = \mathcal{G}_N(\xi_N(t), \xi(t))$. As a first step towards assessing the properties of IAMPC under uncertainty, the following lemma proves that (5) in closed-loop with (9) is asymptotically stable in the nominal case, i.e., $\bar{f}_{\bar{\vartheta}} = f_\xi$ where the parameter is possibly time-varying, but always known ahead of time over the entire prediction horizon N, $\xi_{t|k} = \xi(t+k)$.

Lemma 1. Let N1, N2 hold. Let $\xi_{t|k} = \xi(t+k)$ for $k = 0, \dots, N-1$. Let (2) be feasible at time $\bar{t} \in \mathbb{R}_{0+}$ i.e. $(x(\bar{t}), \xi_N(\bar{t})) \in \mathcal{X}^F_\Xi$ where $\mathcal{X}^F_\Xi = \{(x, \xi_N) \in \mathcal{X} \times \Xi^N \,:\, \mathcal{V}_{\xi_N}(x) < \infty\} \neq \emptyset$ is the feasible set of (2). Then, for all $t > \bar{t}$ and all $\{\xi_N(t)\}_t$ obtained from (7) there exists $\alpha_1, \alpha_2, \alpha_\Delta \in \mathcal{K}_\infty$ such that

$$\alpha_1(\|x\|) \le \mathcal{V}_{\xi_N(t)}(x(t)) \le \alpha_2(\|x\|) \tag{12a}$$

$$\Delta\mathcal{V}_{\xi_N(t)}(x(t)) \le -\alpha_\Delta(\|x\|), \tag{12b}$$

holds for the closed-loop system (5) and (9) and hence the closed-loop is AS. □

The proof of Lemma 1 is in Appendix A.

Next, building on the results for the nominal case, we consider the impact of the parameter vector $\bar{\vartheta}$ being constant but initially unknown, and we introduce conditions that ensure ISS with respect to the estimation error of the plant (1) in closed-loop with the IAMPC (9). Since $\Phi(\bar{\vartheta})$ is potentially multivalued, we define the estimation error as

$$\varepsilon(\xi, \bar{\vartheta}) = \min_{\bar{\xi}} \|\xi - \bar{\xi}\| \tag{13a}$$

$$\text{s.t. } \bar{\xi} \in \Phi(\bar{\vartheta}) \tag{13b}$$

where the optimizer $\bar{\xi}^*$ of (13) is the virtual parameter vector corresponding to $\bar{\vartheta}$ that minimizes the distance from the current virtual parameter estimate ξ, and, for brevity, $\varepsilon(\xi, \bar{\vartheta}) = \|\xi - \bar{\xi}^*\| = \|\tilde{\xi}\|$, where $\tilde{\xi} \in \tilde{\Xi}(\xi)$ and $\tilde{\Xi}(\xi) = \{\tilde{\xi} \in \mathbb{R}^\ell \,:\, \exists \bar{\xi} \in \Xi,\ \tilde{\xi} = \bar{\xi} - \xi\}$. The estimation error (13) is used to only prove ISS, but need not be computed in practice.

Consider the additional conditions

Conditions 2. (Robustness)

R1 Robust control invariance.

$$\exists C_x \subseteq \mathcal{X} \,:\, \forall x \in \mathcal{X},\ \exists u \in \mathcal{U},\ f_\xi(x, u) \in C_x,\ \forall \xi \in \Xi \tag{14}$$

R2 Terminal set reachability. For $x_{k+1} = f_{\xi_k}(x_k, u_k)$,

$$\forall x \in C_x, \xi_N \in \Xi^N,\ \exists U_N \in \mathcal{U}^N \,:\, (x_k, u_k) \in C_{xu}, \forall k \in \mathbb{Z}_{[0,N-1]},\ x_N \in \mathcal{X}_N, \tag{15}$$

R3 Value function bound.

$$\exists \varsigma \in \mathcal{K}_\infty \,:\, |\mathcal{V}_{\xi_N}(x_1) - \mathcal{V}_{\xi_N}(x_2)| \le \varsigma(\|x_1 - x_2\|) \tag{16}$$

R4 Error bounding function. There exists $\gamma, \bar{\gamma}_1, \bar{\gamma}_2 \in \mathcal{K}_\infty$, $\varepsilon \in \mathbb{R}_+$, $c \in \mathbb{R}_{(0,1)}$ such that

$$\|\varepsilon_x\| = \|f_{\bar{\xi}}(x, u) - f_\xi(x, u)\| \le \gamma(\|\tilde{\xi}\| \|x\|) \tag{17a}$$

$$\bar{\gamma}(a + b) \le \bar{\gamma}_1(a) + \bar{\gamma}_2(b) \tag{17b}$$

$$\bar{\gamma}_1(\varepsilon s^2) \le c\alpha_\Delta(s),\ \forall s \in [0, \max_{x \in C_x} \|x\|] \tag{17c}$$

where $\bar{\gamma} = \varsigma \circ \gamma$.

It is worth noting the different role played by ξ and ξ_N in conditions R1 and R2. The former, i.e., (14), is a robustness condition requiring the existence of an input that enforces the condition for all $\xi \in \Xi$. The latter, i.e., (15), is a parametric condition requiring that given any sequence $\xi_N \in \Xi$, there exists a sequence of inputs enforcing the condition for that specific ξ_N. Such difference is reflected in the different position of the universal quantifiers. In R4, (17a) requires some smoothness of the prediction error around the equilibrium and the uncertainty to be multiplicative to the state; (17b) simply decomposes a class-$\mathcal{K}$ function of a sum of vectors, into the sum of class-$\mathcal{K}$ functions acting on single vectors, which is simple to satisfy[34]; and (17c) requires that the class-$\mathcal{K}$ function bounding the value function decrease along the trajectory (12b) grows at least quadratically around the origin.

Conditions 3. (Parameter update)

U1 Prediction consistency. $\mathcal{G}_N$ in (8) is such that for every $t \in \mathbb{R}_+$, $\xi_{k|t} = \xi_{k+1|t-1}$, $k \in \mathbb{Z}_{[0,N-1]}$, where $\xi_N(t) \in \Xi^{N+1}$.

Theorem 1. Let N1–N2, R1–R4, U1 hold. Let $\mathcal{C}_{xu} = \{(x,u) \in \mathcal{C}_x \times \mathcal{U} : f_\xi(x,u) \in \mathcal{C}_x, \forall \xi \in \Xi\}$. There exists an RPI set $\mathcal{X}_0 \subseteq \mathcal{X}$ for the closed-loop such that for all $x(t) \in \mathcal{X}_0$ the closed-loop recursively satisfies the constraints and

$$\alpha_1(\|x(t)\|) \leq \mathcal{V}_{\xi_N(t)}(x(t)) \leq \alpha_2(\|x(t)\|) \tag{18a}$$

$$\Delta\mathcal{V}_{\xi_N(t)}(x(t)) \leq -c_\Delta\alpha_\Delta(\|x(t)\|) + \tilde{\gamma}(\|\tilde{\xi}_{0|t}\|), \tag{18b}$$

where $c_\Delta \in (0,1)$, i.e., the system is ISS in $\mathcal{X}_0$ with respect to the estimation error $\|\tilde{\xi}_{0|t}\| = \varepsilon(\xi_{0|t}, \bar{\vartheta})$ defined in (13).

Proof. First we can prove that if $x(\bar{t}) \in \mathcal{C}_x$ then (2) is feasible, and remains feasible for any $t \geq \bar{t}$, i.e., the controller (9) continues operating. By R2 if $x \in \mathcal{C}_x$, for all $\xi_N \in \Xi^N$, there exists U_N such that $x_N \in \mathcal{X}_N$ and $(x_k, u_k) \in \mathcal{C}_{xu}$. By R1, if $(x_k, u_k) \in \mathcal{C}_{xu}$, then $u \in \mathcal{U}$ and $f_\xi(x,u) \in \mathcal{C}_x \subseteq \mathcal{X}$, $\forall \xi \in \Xi$. Thus, if $x(\bar{t}) \in \mathcal{C}_x \subseteq \mathcal{X}$, then $u(\bar{t}) \in \mathcal{U}$, and $x(\bar{t}+1) \in \mathcal{C}_x \subseteq \mathcal{X}$, and the reasoning can be repeated for all future time steps $t > \bar{t}$.

Bounds (18a) can be proved exactly as in Lemma 1, since the value function is the same, and the bounds are not affected by the estimation error. Also, the assumptions in Lemma 1 hold thanks to condition U1.

As for the ISS inequality (18b), the change of the value function can be split as

$$\begin{aligned}\Delta\mathcal{V}_{\xi(t)}(x(t)) &= \mathcal{V}_{\xi(t+1)}\big(f_{\bar{\xi}(t)}(x(t),u(t))\big) - \mathcal{V}_{\xi(t)}\big(x(t)\big)\\ &= \underbrace{\mathcal{V}_{\xi(t+1)}\big(f_{\xi(t)}(x(t),u(t))\big) - \mathcal{V}_{\xi(t)}\big(x(t)\big)}_{-\alpha_\Delta(\|x(t)\|)} + \underbrace{\mathcal{V}_{\xi(t+1)}\big(f_{\bar{\xi}(t)}(x(t),u(t))\big) - \mathcal{V}_{\xi(t+1)}\big(f_{\xi(t)}(x(t),u(t))\big)}_{\varsigma(\|f_{\bar{\xi}(t)}(x(t),u(t)) - f_{\xi(t)}(x(t),u(t))\|)},\end{aligned}$$

where $\mathcal{V}_{\xi(t+1)}\big(f_{\xi(t)}(x(t),u(t))\big) - \mathcal{V}_{\xi(t)}\big(x(t)\big) \leq -\alpha_\Delta(\|x(t)\|)$ by Lemma 1 and $\mathcal{V}_{\xi(t+1)}\big(f_{\bar{\xi}(t)}(x(t),u(t))\big) - \mathcal{V}_{\xi(t+1)}\big(f_{\xi(t)}(x(t),u(t))\big) \leq \varsigma(\|f_{\bar{\xi}(t)}(x(t),u(t)) - f_{\xi(t)}(x(t),u(t))\|)$ by R3. By R4, where $\bar{\gamma} = \varsigma\circ\gamma$,

$$\begin{aligned}\Delta\mathcal{V}_{\xi(t)}(x(t)) &\leq -\alpha_\Delta(\|x(t)\|) + \varsigma(\|f_{\bar{\xi}(t)}(x(t),u(t)) - f_{\xi(t)}(x(t),u(t))\|)\\ &\leq -\alpha_\Delta(\|x(t)\|) + \varsigma(\gamma(\|\tilde{\xi}\|\|x(t)\|)) = -\alpha_\Delta(\|x(t)\|) + \bar{\gamma}(\|\tilde{\xi}\|\|x(t)\|).\end{aligned} \tag{19}$$

Using a generalization of Young's inequality (also known as "Peter & Paul Theorem") and R4

$$\Delta\mathcal{V}_{\xi(t+1)} \leq -\alpha_\Delta(\|x\|) + \bar{\gamma}(\frac{1}{2\varepsilon}\|\tilde{\xi}_{0|t}\|^2 + \frac{1}{2}\varepsilon\|x\|^2) \leq -\alpha_\Delta(\|x\|) + \bar{\gamma}_2(\frac{1}{2\varepsilon}\|\tilde{\xi}_{0|t}\|^2) + \bar{\gamma}_1(\frac{1}{2}\varepsilon\|x\|^2),$$

and hence, using again R4, there exists c and ε such that

$$\Delta\mathcal{V}_{\xi(t+1)} \leq -\alpha_\Delta(\|x\|) + \bar{\gamma}_2(\frac{1}{2\varepsilon}\|\tilde{\xi}_{0|t}\|^2) + c\alpha_\Delta(\frac{1}{2}\|x\|) \leq -(1-c)\alpha_\Delta(\|x\|) + \tilde{\gamma}_2(\|\tilde{\xi}_{0|t}\|)$$

where $\tilde{\gamma}_2 \in \mathcal{K}_\infty$. Thus, the statement is proved with $\mathcal{X}_0 = \mathcal{C}_x$ and hence the system is ISS with respect to the estimation error $\|\tilde{\xi}_{0|t}\| = \varepsilon(\xi_{0|t}, \bar{\vartheta})$. □

The following corollary shows that for small yet non-zero estimation error the closed-loop system is AS.

Corollary 1. Let the assumptions of Theorem 1 hold. If there exists $\bar{\mathcal{X}}_0 \subseteq \mathcal{X}_0$, $c_1 \in \mathbb{R}_{(0,1)}$, $c_2, c_3, c_4 \in \mathbb{R}_+$, such that for all $x \in \bar{\mathcal{X}}_0$, $\|\mathcal{V}_{\xi_N}(x_1) - \mathcal{V}_{\xi_N}(x_2)\| \leq c_3\mathcal{V}_{\xi_N}(x_1 - x_2) + c_4\mathcal{V}_{\xi_N}(x_2)$ and $c_3\alpha_2\circ\gamma(c_2\|x\|) + c_4\alpha_2(\|x\|) \leq c_1\alpha_\Delta(\|x\|)$, then there exists $\bar{c}_\Delta \in \mathbb{R}_{(0,1)}$, $\hat{c}_2 \leq c_2$ such that if $\|\tilde{\xi}\| \leq \hat{c}_2$,

$$\Delta\mathcal{V}_{\xi_N(t)}(x(t)) \leq -\bar{c}_\Delta\alpha_\Delta(\|x\|),\ \forall x \in \hat{\mathcal{X}}_0,$$

for some $\hat{\mathcal{X}}_0 \subseteq \bar{\mathcal{X}}_0$. □

The proof of Corollary 1 is in Appendix A.

The following result, which follows directly from Theorem 1, summarizes the developments of this section as a solution for Problem 1.

Result 2. Under Conditions N1–N2, R1–R4, U1 the IAMPC (7), (8), (9) solves (i) and (ii) of Problem 1.

Remark 2. The IAMPC architecture provides a separation principle between MPC and parameter estimation, meaning that one can disjoin the design of a stabilizing controller and a convergent estimator and still achieve overall stability properties and constraint satisfaction, despite the estimation error. The key ingredients to achieve that are the parameterization of the terminal cost (11), the enforcement of the robust control invariant (14) in the IAMPC optimal control problem, and the appropriate construction of the parameter update (7) and predictor (8) that interface IAMPC with the estimator (6).

As our goal is to achieve the separation discussed in Remark 2, we did not pursue a dual objective control approach [25,24,19] since we do not want to require any knowledge on the estimation algorithm, as often needed to select the optimal input sequence for parameter uncertainty reduction. We also do not want to have to include matrix inequalities in the optimization problem [25], because of the induced increase in computational burden.

As for (iii) in Problem 1, convergence of the numerical algorithm cannot be guaranteed in the general case of nonlinear plant (1), as the optimization problems may be non-convex, and, similarly, the structure and complexity of the cost function and constraint sets are difficult to predict, making it impossible to quantify the computational load.

Next we show a constructive method to design of IAMPC for uncertain linear systems such that the conditions N1-N2, R1-R4, U1 are satisfied and the properties $(i)-(iii)$ holds for the closed-loop system, reaching a full solution of Problem 1.

4 | IAMPC FOR POLYTOPIC SYSTEMS

Now, we consider the class of discrete-time linear uncertain constrained systems (4) with sampling period T_s, which, as discussed for instance in [5], can be re-parametrized as the uncertain polytopic system

$$x(t+1) = \sum_{i=1}^{\ell}[\xi(t)]_i A_i x(t) + Bu(t), \tag{21a}$$

$$(x,u) \in \mathcal{X} \times \mathcal{U} \tag{21b}$$

$$\xi(t) = \bar{\xi} \tag{21c}$$

where $\xi \in \Xi$ is the abstract parameter vector taking the place of the physical parameter vector ϑ, and $\Xi = \{\xi \in \mathbb{R}^\ell : \sum_{i=1}^{\ell}[\xi]_i = 1,\ [\xi]_i \geq 0,\ \forall i \in \mathbb{Z}_{[1,\ell]}\}$ is the unit simplex in $\mathbb{R}^\ell$, $A_i \in \mathbb{R}^{n\times n}$, $i \in \mathbb{Z}_{[1,\ell]}$ and B are known matrices of appropriate size, and $\mathcal{X} \subseteq \mathbb{R}^n$, $\mathcal{U} \subseteq \mathbb{R}^m$ are polytopic admissible regions of states and inputs. By (21c) we highlight that here we consider the case where the parameters are unknown but constant or slowly varying. Considering only (21a), (21b), when $\xi(t)$ is known we obtain the linear parameter varying (LPV) system of the form used, e.g., in [27], while when $\xi(t)$ is unknown, we obtain the general uncertain polytopic system where the dynamics can change at any step, see, e.g., [6]. We call ξ the convex combination vector, since it describes a convex combination of the vertex systems $f^{(i)}(x,u) = A_i x + Bu$, $i \in \mathbb{Z}_{[1,\ell]}$.

Remark 3. The trajectories produced by (21) are a subset of those of the pLDI

$$x(t+1) \in \mathrm{co}\{A_i x(t) + Bu(t)\}_{i=1}^{\ell}. \tag{22}$$

If $\xi(t)$ is continuously varying over the simplex Ξ, then the pLDI (21a) is equivalent to (22).

Using (21) with unknown $\bar{\xi}$, the MPC (2) can be formulated as

$$\mathcal{V}_{\xi_N(t)}(x(t)) = \min_{U_t}\ x'_{N|t}\mathcal{P}(\xi_{N|t})x_{N|t} + \sum_{k=0}^{N-1} x'_{k|t}Qx_{k|t} + u'_{k|t}Ru_{k|t} \tag{23a}$$

$$\text{s.t.}\ \ x_{k+1|t} = \sum_{i=1}^{\ell}[\xi_{k|t}]_i A_i x_{k|t} + Bu_{k|t} \tag{23b}$$

$$(x_{k|t}, u_{k|t}) \in \mathcal{C}_{xu} \subseteq \mathcal{X} \times \mathcal{U} \tag{23c}$$

$$x_{N|t} \in \mathcal{X}_N \tag{23d}$$

$$x_{0|t} = x(t), \tag{23e}$$

where $N \in \mathbb{R}_+$, $Q \in \mathbb{R}^{n\times n}$, $R \in \mathbb{R}^{m\times m}$, $Q, R > 0$, $\mathcal{P}(\xi) > 0 \in \mathbb{R}^{n\times n}$, for all $\xi \in \Xi$.

Consider the dynamics (21a), we define the parameter-dependent (linear) control law

$$u = \kappa(\xi)x = \left(\sum_{i=1}^{\ell}[\xi]_i K_i\right)x, \tag{24}$$

and the parameter-dependent (quadratic) function

$$\mathcal{V}(x,\xi) = x'\mathcal{P}(\xi)x = x'\left(\sum_{i=1}^{\ell}[\xi]_i P_i\right)x, \tag{25}$$

where $P_i > 0$, $i \in \mathbb{Z}_{[1,\ell]}$.

Definition 5 ([27]). A function (25) such that $\mathcal{V}(x(t+1),\xi(t+1)) - \mathcal{V}(x(t),\xi(t)) \le 0$, for all $\xi(t),\xi(t+1) \in \Xi$, where equality holds only if $x = 0$, is a parameter-dependent Lyapunov function (pLF) for (21a) in closed-loop with (24). □

By[27,6,35], given $Q > 0 \in \mathbb{R}^{n\times n}$, $R > 0 \in \mathbb{R}^{m\times m}$, any solution $G_i, S_i \in \mathbb{R}^{n\times n}$, $S_i > 0$, $E_i \in \mathbb{R}^{m\times n}$, $i \in \mathbb{Z}_{[1,\ell]}$, of

$$\begin{bmatrix} G_i + G_i' - S_i & (A_iG_i + B_iE_i)' & E_i' & G_i' \\ (A_iG_i + B_iE_i) & S_j & 0 & 0 \\ E_i & 0 & R^{-1} & 0 \\ G_i & 0 & 0 & Q^{-1} \end{bmatrix} > 0, \ \ \forall i,j \in \mathbb{Z}_{[1,\ell]}. \tag{26}$$

with $B_i = B$ for all $i \in \mathbb{Z}_{[1,\ell]}$ satisfies

$$\mathcal{V}(x(t+1),\xi(t+1)) - \mathcal{V}(x(t),\xi(t)) \le -x(t)'(Q + \kappa(\xi(t))'R\kappa(\xi(t)))x(t)', \ \forall \xi(t),\xi(t+1) \in \Xi \tag{27}$$

for the pLF (25) and closed-loop dynamics (21a), (24) where $P_i = S_i^{-1}$, $K_i = E_iG_i^{-1}$ for $i \in \mathbb{Z}_{[1,\ell]}$.

Assumption 1. For the given A_i, $i \in \mathbb{Z}_{[1,\ell]}$, B, Q, R, (26) admits a feasible solution

The LMI (26) is a relaxation of those in[5,6,9] since it allows for a parameter-dependent Lyapunov function and a parameter-dependent linear control law. Thus, Assumption 1 is a relaxation of, and implied by, the existence of an (unconstrained) stabilizing linear control law for (21a) when $\xi(t) \in \Xi$ is not known, see, e.g.,[5,6,9]. Indeed, if the vertex systems are such that the uncertainty is too large, (26) may be infeasible, same as when robustly stabilizing controller for an uncertain system does not exist. However, since (26) is used here only for design, as opposed to computation of the control input in[5,6,9], such situation will be recognized before controller execution and corrective measures may be taken, such as improving the engineering of the plant. Another advantage of using (26) only for design, as opposed for control input computation[5,6,9], is that online we will only solve QPs, which makes our method feasible also for applications with fast dynamics and low-cost microcontrollers[2,3].

Next, we analyze the properties of IAMPC designed using the constructive procedure from this section. Thus, we first focus on the performance properties in the unconstrained case, i.e., ISS, which are achieved by designing the parameter-dependent terminal cost, and then we focus on the safety properties in the constrained case, i.e, robust constraint satisfaction, which are achieved by additionally designing the constraint set and the terminal set.

4.1 | Unconstrained IAMPC: Stability

First, we focus the performance, i.e., input to state stability, by considering the unconstrained case, $\mathcal{X} = \mathbb{R}^n$, $\mathcal{U} = \mathbb{R}^m$. Once again, we split the analysis into two steps. First we assess the nominal case when $\xi_{k|t} = \xi(t+k)$, $k \in \mathbb{Z}_{[0,N]}$, and possibly $\xi_{k_1|t} \neq \xi_{k_2|t}$, for $k_1, k_2 \in \mathbb{Z}_{[0,N]}$. This amounts to controlling an LPV system with preview on the parameters for $N-1$ steps in the future, but no information afterwards. Then, we assess the impact of the uncertainty and the estimation of the parameter vector.

Lemma 2. Let Assumption 1 hold and consider (21a) and the nominal MPC that solves (23) where $\mathcal{X}_N = \mathbb{R}^n$, $C_{xu} = \mathcal{X}\times\mathcal{U} = \mathbb{R}^{n+m}$, for any $t \in \mathbb{Z}_{0+}$, $\xi_{k|t} = \xi(t+k)$, $\xi(t+k) \in \Xi$, for all $k \in \mathbb{Z}_{[0,N]}$, and $\mathcal{P}(\xi)$, $\kappa(\xi)$ are from (26). Then in the PI set $\mathcal{X}_\Xi^F = \{(x,\xi_N) : \ \mathcal{V}_{\xi_N}(x) < \infty\}$ there exists $\alpha_1,\alpha_2,\alpha_\Delta \in \mathcal{K}_\infty$ such if $(x(\bar t),\xi_N(\bar t)) \in \mathcal{X}_\Xi^F$, for some $\bar t \in t$, then for all $t > \bar t$ and all $\{\xi_N(t)\}_t$,

$$\alpha_1(\|x\|) \le \mathcal{V}_{\xi_N(t)}(x(t)) \le \ \alpha_2(\|x\|) \tag{28a}$$

$$\Delta\mathcal{V}_{\xi_N(t)}(x(t)) \le -\,\alpha_\Delta(\|x\|), \tag{28b}$$

and hence the closed loop is AS. □

The proof of Lemma 2 is in Appendix A.

By Lemma 2, the MPC based on (23) with perfect preview along the horizon is stabilizing. Next, we exploit this to address the case of Problem 1 where $\xi(t)$ is constant, i.e., $\xi(t) = \bar\xi$ for all $t \in \mathbb{Z}_{0+}$ as in (21c), unknown, and being estimated. Thus,

$\tilde{\xi}_{0|t} = \bar{\xi} - \xi_{0|t}$ is the error in the parameter estimate, which may be time-varying, and $\tilde{\xi}_{0|t} \in \Xi(\xi_{0|t})$. The parameter estimation error induces a state prediction error

$$\varepsilon_x = \sum_{i=1}^{\ell} [\bar{\xi}]_i A_i x - \sum_{i=1}^{\ell} [\xi_{0|t}]_i A_i x = \sum_{i=1}^{\ell} [\tilde{\xi}_{0|t}]_i A_i x, \tag{29}$$

which is bounded by the estimation error

$$\|\varepsilon_x\| = \left\| \sum_{i=1}^{\ell} [\tilde{\xi}_{0|t}]_i A_i x \right\| \leq \left\| \sum_{i=1}^{\ell} [\tilde{\xi}_{0|t}]_i A_i \right\| \cdot \|x\| \leq \left(\sum_{i=1}^{\ell} |[\tilde{\xi}_{0|t}]_i| \, \|A_i\| \right) \|x\| \leq \gamma_A \|\tilde{\xi}_{0|t}\|_1 \|x\| = \gamma \|\tilde{\xi}_{0|t}\| \|x\|, \tag{30}$$

where $\gamma_A = \max_{i=1,\ldots\ell} \|A_i\|$ and the last equality is due to the norm equivalence in finite dimensional spaces.

Consider the value function $\mathcal{V}_{\xi_N}$ of (23). The following result is straightforward from [1].

Result 3. For every compact $\mathcal{X}_L \subset \mathbb{R}^n$, the value function of (23), where $\mathcal{P}(\xi)$ is designed according to (26), is Lipschitz-continuous in $x \in \mathcal{X}_L$, that is, there exists $L \in \mathbb{R}_+$ such that for every $x_1, x_2 \in \mathcal{X}_L$, $\|\mathcal{V}_{\xi_N}(x_1) - \mathcal{V}_{\xi_N} C(x_2)\| \leq L\|x_1 - x_2\|$, for every $\xi_N \in \Xi^{N+1}$. □

Result 3 follows directly from the fact that for every $\xi_N \in \Xi^{N+1}$, $\mathcal{V}_{\xi_N}^{\mathrm{MPC}}$ is piecewise quadratic [1] and hence Lipschitz continuous in any compact set $\mathcal{X}_L$. Thus, for any $\mathcal{X}_L \subseteq \mathbb{R}^n$ and $\xi_N \in \Xi^{N+1}$, there exists a Lipschitz parameter $L_{\xi_N} \in \mathbb{R}_+$. Since Ξ^{N+1} is closed and bounded, there exists a maximum of $L_{\xi_N} \in \mathbb{R}_+$ for $\xi_N \in \Xi^{N+1}$, which gives a Lipschitz constant L for $\mathcal{V}_{\xi_N}$.

Theorem 2. Let Condition U1 and Assumption 1 hold. For the MPC that at every step solves (23) where $\mathcal{P}(\xi)$ is designed according to (26), $\mathcal{X}_N = \mathbb{R}^n$, $\mathcal{C}_{xu} = \mathcal{X} \times \mathcal{U} = \mathbb{R}^{n+m}$, in any compact set $\mathcal{X}_\eta$ that is RPI with respect to $\tilde{\xi}_{0|t}$ for the closed loop,

$$\alpha_1(\|x\|) \leq \mathcal{V}_{\xi_N(t)}(x(t)) \leq \alpha_2(\|x\|) \tag{31a}$$

$$\Delta\mathcal{V}_{\xi_N(t)}(x(t)) \leq -c_\Delta \alpha_\Delta(\|x\|) + \tilde{\gamma}(\|\tilde{\xi}_{0|t}\|), \tag{31b}$$

where $c_\Delta \in (0,1)$, i.e., the closed-loop is ISS with respect to $\tilde{\xi}_{0|t} = \bar{\xi} - \xi_{0|t} \in \tilde{\Xi}(\xi_{0|t})$.

Proof. Conditions R1, R2 are trivially satisfied because $\mathcal{X}_N = \mathbb{R}^n$, $\mathcal{C}_{xu} = \mathcal{X} \times \mathcal{U} = \mathbb{R}^{n+m}$.

Condition R3 is satisfied because for every compact $\mathcal{X}_L \subseteq \mathbb{R}^n$, the value function of (23), where $\mathcal{P}(\xi)$ is designed according to (26), is Lipschitz-continuous in $x \in \mathcal{X}_L$, i.e., there exists $L \in \mathbb{R}_+$ such that for every $x_1, x_2 \in \mathcal{X}_L$, $\|\mathcal{V}_{\xi_N}(x_1) - \mathcal{V}_{\xi_N}(x_2)\| \leq L\|x_1 - x_2\|$, for every $\xi_N \in \Xi^{N+1}$. This follows directly from $\mathcal{V}_{\xi_N}$ being piecewise quadratic [1] and hence it is Lipschitz continuous in any compact set $\mathcal{X}_L$ for every $\xi_N \in \Xi^{N+1}$.

As for R4, from (30) and the Lipschitz property, $\bar{\gamma}(\|\tilde{\xi}_{0|t}\|\|x\|) = \gamma_L \gamma \|\tilde{\xi}_{0|t}\|\|x\|$, and by Young's (Peter & Paul) Theorem, for any $\varepsilon > 0$

$$\|\tilde{\xi}_{0|t}\|\|x\| \leq \frac{\varepsilon}{2}\|\tilde{\xi}_{0|t}\|^2 + \frac{1}{2\varepsilon}\|x\|^2,$$

so that $\bar{\gamma}_2(\xi_{0|t}) = \gamma_L \gamma \frac{1}{2\varepsilon}\|\tilde{\xi}_{0|t}\|^2$ and $\bar{\gamma}_1(x) = \gamma_L \gamma \frac{\varepsilon}{2}\|x\|^2$. Hence, for every c there exists ε such that $\gamma_L \gamma \frac{\varepsilon}{2}\|x\|^2 \leq c\lambda_{\min}(Q)\|x\|^2 = \alpha_\Delta(||x(t)||)$. Thus, the conditions of Theorem 1 hold and ISS follows.

Finally note that the existence of the RPI set $\mathcal{X}_\eta$ is guaranteed by $\|\tilde{\xi}\|$ being bounded, since $\xi, \bar{\xi} \in \Xi$ and Ξ is bounded, and the nominal closed-loop system being AS due to Lemma 2. □

Remark 4. A proof for Theorem 2 can be obtained based on uniform continuity, instead of Lipschitz continuity from Result 3, following [33, Th.2], where it is pointed out [33, A.4.3] that the two assumptions are equivalent for (23), given $\xi_N(t) \in \Xi^{N+1}$.

Next we show that the closed-loop system is AS for a sufficiently small, yet non-zero, error.

Corollary 2. Under the same assumptions of Theorem 2, there exists $\hat{\mathcal{X}}_0 \subseteq \mathcal{X}$, $\hat{c}_2 \in \mathbb{R}_+$ and $\bar{c}_\Delta \in (0,1)$, such that, if $\|\tilde{\xi}\| \leq \hat{c}_2$,

$$\Delta\mathcal{V}_{\xi_N(t)}(x(t)) \leq -\bar{c}_\Delta \alpha_\Delta(\|x\|),\ \forall x \in \hat{\mathcal{X}}_0, \tag{32}$$

that is, the closed-loop is asymptotically stable for a sufficiently small, yet non-zero, error. □

The proof of Corollary 2 is in Appendix A.

4.2 | Constrained IAMPC: Terminal set and Robust Constraints

Next we include in (21) non-trivial constraints $\mathcal{X} \times \mathcal{U} \subset \mathbb{R}^n \times \mathbb{R}^m$, which requires expanding the design to satisfy conditions N1, R1, R2. Once again, we consider first the nominal case, i.e., when the parameter vector estimate is correct, to design the terminal constraint set $\mathcal{X}_N$ in (23d) to satisfy the conditions of Lemma 1, which ensures recursive constraint satisfaction and AS of the nominal system (21a) with perfect preview of the virtual parameter along the prediction horizon, i.e., $\xi_{k|t} = \xi(t+k)$, for all $k \in \mathbb{Z}_{[0,N]}$. Then, we account for the effect of the estimation error by designing the set $\mathcal{C}_{xu}$ to ensure constraint satisfaction despite uncertainty.

Consider (21a) where $\xi(t)$ is known at $t \in \mathbb{Z}_{0+}$ and the control law (24) resulting in the closed-loop LPV system

$$x(t+1) = \sum_{i=1}^{\ell} [\xi(t)]_i (A_i + BK_i) x(t). \tag{33}$$

The trajectories of (33) are contained in those of the pLDI

$$x(t+1) \in \mathrm{co}\{(A_i + BK_i)x(t)\}_{i=1}^{\ell}. \tag{34}$$

For (34) where K_i, $i \in \mathbb{Z}_{[1,\ell]}$ is designed by (26) subject to (21b), one can compute a PI set as the maximum constraint admissible set[35] $\mathcal{X}^\infty \subseteq \bar{\mathcal{X}}$, for all $\xi \in \Xi$. Let $\mathcal{X}_{xu}$ be a given set of feasible states and inputs, $\mathcal{X}_{xu} \subseteq \mathcal{X} \times \mathcal{U}$, $0 \in \mathrm{int}(\mathcal{X}_{xu})$, and construct the set sequence

$$\begin{aligned}
\mathcal{X}^{(0)} &= \{x : \ (x, K_i x) \in \mathcal{X}_{xu}, \ \forall i \in \mathbb{Z}_{[1,\ell]}\} \\
\mathcal{X}^{(h+1)} &= \{x : \ (A_i + BK_i)x \in \mathcal{X}^{(h)}, \ \forall i \in \mathbb{Z}_{[1,\ell]}\} \cap \mathcal{X}^{(h)} \\
\mathcal{X}^{\infty} &= \lim_{h\to\infty} \mathcal{X}^{(h)}.
\end{aligned} \tag{35}$$

Each iteration of (35) requires only matrix manipulations, to construct the new polyhedral set, and the solution of linear programs to remove redundant hyperplanes[36].

Result 4 ([28,35]). There exists a finite $\bar{h} \in \mathbb{Z}_{0+}$ such that $\mathcal{X}^{(\bar{h}+1)} = \mathcal{X}^{(\bar{h})} = \mathcal{X}^\infty$, i.e., the limit in (35) is reached in a finite number of iterations. The resulting $\mathcal{X}^{(\bar{h})}$ has non-empty interior, is polyhedral and maximal. □

The proof of Result 4 follows from (24) being a stabilizing controller for (21a) under arbitrary known changes of $\xi \in \Xi$, with (25) being its Lyapunov function. Since $\mathcal{X}_{xu} \subseteq \mathcal{X} \times \mathcal{U}$, $0 \in \mathrm{int}(\mathcal{X}_{xu})$ is compact, there exists a sublevel set of the pLF (25) that is entirely contained in $\mathcal{X}^{0)}$, such compact set containing the origin in its interior is reached in finite time $h(x)$ starting from any $x \in \mathcal{X}_{xu}$, and $\bar{h} = \max_{x\in\mathcal{X}_{xu}} h(x)$ exists, finite. Thus, the constraints up to $\bar{h}$ determine the initial conditions for which the dynamics satisfies the constraints at any future time instant, and maximality follows from the removal of any non-redundant constraints causing inclusion of initial states for which the constraints are violated, while including additional non-redundant constraints will eliminate initial states for which constraints are satisfied.

Lemma 3. Let Assumption 1 hold. Consider (21a) and the nominal MPC that solves (23) where $\mathcal{P}(\xi)$, $\kappa(\xi)$ are from (26), $\mathcal{C}_{xu} \subseteq \mathcal{X} \times \mathcal{U}$, $\mathcal{X}_N = \mathcal{X}^\infty$ is designed by (35), and for any $t \in \mathbb{Z}_{0+}$, $\xi_{k|t} = \xi(t+k)$, $\xi(t+k) \in \Xi$, for all $k \in \mathbb{Z}_{[0,N]}$. At a given $t \in \mathbb{Z}_{0+}$, let $x(t) \in \mathcal{X}$, $\xi_N(t) \in \Xi^{N+1}$ be such that (23) is feasible. Then, in the PI set $\mathcal{X}_\Xi^F = \{(x, \xi_N) : \ \mathcal{V}_{\xi_N}(x) < \infty\}$ there exists $\alpha_1, \alpha_2, \alpha_\Delta \in \mathcal{K}_\infty$ such if $(x(\bar{t}), \xi_N(\bar{t})) \in \mathcal{X}_\Xi^F$, for some $\bar{t} \in t$, then for all $t > \bar{t}$ and all $\{\xi_N(t)\}_t$

$$\alpha_1(\|x\|) \leq \mathcal{V}_{\xi_N(t)}(x(t)) \leq \ \alpha_2(\|x\|) \tag{36a}$$

$$\Delta\mathcal{V}_{\xi_N(t)}(x(t)) \leq -\,\alpha_\Delta(\|x\|), \tag{36b}$$

and hence the closed loop is AS. □

The proof of Lemma 3 is in Appendix A.

Next, we ensure robust satisfaction of (23c), (23d), in presence of uncertainty, i.e., when there is estimation error $\tilde{\xi}_{0|t} \neq 0$. In this case we need to satisfy conditions R1, R2 in order to ensure robust constraint satisfaction.

To satisfy R1 we design (23c) from a RCI set for the pLDI (22), whose trajectories include those of (21). Based on Definition 2, let $\mathcal{C}_x \subseteq \mathcal{X}$ be a polyhedron such that for any $x \in \mathcal{C}_x$ there exists $u \in \mathcal{U}$ such that $A_i x + Bu \in \mathcal{C}_x$ for all $i \in \mathbb{Z}_{[1,\ell]}$. Given $\mathcal{C}_x$, we design $\mathcal{C}_{xu}$ in (23c) as

$$\mathcal{C}_{xu} = \{(x, u) \in \mathcal{C}_x \times \mathcal{U}, \ A_i x + Bu \in \mathcal{C}_x, \ \forall i \in \mathbb{Z}_{[1,\ell]}\}, \tag{37}$$

that is, the state-input pairs that result in states within the RCI set for any vertex system of the pLDI (22).

$\mathcal{C}_x$ can be computed[28] as the maximal RCI set for (22) from the sequence,

$$\mathcal{C}^{(0)} = \mathcal{X}, \tag{38a}$$

$$\mathcal{C}^{(h+1)} = \{x : \exists u \in \mathcal{U},\ A_i x + Bu \in \mathcal{C}^{(h)},\ \forall i \in \mathbb{Z}_{[1,\ell]}\} \cap \mathcal{C}^{(h)}. \tag{38b}$$

The maximal RCI set in $\mathcal{X}$ is the fixpoint of (38), i.e., $\mathcal{C}_x = \mathcal{C}^\infty = \mathcal{C}^{(\bar{h})}$ such that $\mathcal{C}^{(\bar{h}+1)} = \mathcal{C}^{(\bar{h})}$, and is the largest set within $\mathcal{X}$ that can be made invariant for (22) with inputs in $\mathcal{U}$. Conditions for existence of $\mathcal{C}^\infty$ are related to the existence of a nonlinear set-stabilizing law for (22), and are discussed in detail in[28]. Besides the operations required in (35), each iteration of (38) requires the projection of a polyhedral subset of $\mathcal{X} \times \mathcal{U}$ onto the subspace of $\mathcal{X}$[36]. The projection of a polyhedron may be a computationally expensive operation, but here it is executed only at design time. Furthermore, several efficient algorithms exists[37], which are implemented in common toolboxes, e.g.,[38].

To satisfy R2, the horizon N is selected such that for every $x \in \mathcal{C}_x$ and $\xi_N(0) \in \Xi^{N+1}$, there exists $[u(0) \dots u(N-1)]$ such that for (21a) with $x(0) = x$, $\xi(k) = \xi_{k|0} \in \Xi$ for all $k \in \mathbb{Z}_{[0,N]}$, $(x(k), u(k)) \in \mathcal{C}_{xu}$ for all $k \in \mathbb{Z}_{[0,N-1]}$, and $x(N) \in \mathcal{X}_N$. To this end, construct the set sequence

$$\mathcal{S}^{(0)} = \mathcal{X}_N, \tag{39a}$$

$$\mathcal{S}_i^{(h+1)} = \{x \in \mathcal{X} : \exists u \in \mathcal{U},\ A_i x + Bu \in \mathcal{S}^{(h)}\},\ i \in \mathbb{Z}_{[1,\ell]} \tag{39b}$$

$$\mathcal{S}^{(h+1)} = \bigcap_{i=1}^{\ell} \mathcal{S}_i^{(h+1)}. \tag{39c}$$

The set $\mathcal{S}^{(h)}$ satisfies: for any $x(0) \in \mathcal{S}^{(h)}$, given any $\xi_{h-1}(0) \in \Xi^h$, there exists a sequence $[u(0) \dots u(h-1)]$ such that for (21a) with $x(0) = x$ and $\xi(k) = \xi_{k|0}$ for all $k \in \mathbb{Z}_{[0,N]}$, $(x(k), u(k)) \in \mathcal{C}_{xu}$ and $x(h) \in \mathcal{X}_N$. Thus, N is chosen such that $N \geq \bar{h}$, where $\bar{h} \in \mathbb{Z}_+$, $\mathcal{S}^{(\bar{h})} \supseteq \mathcal{C}_x$, and hence $\mathcal{S}^{(\bar{N})} \supseteq \mathcal{C}_x$, by the invariance of $\mathcal{X}_N$.

As opposed to (38), in computing (39) the parameter sequence ξ_h is known. This is due to enforcing the terminal set only with respect to the nominal dynamics, while the robust invariance of $\mathcal{C}_x$ and choosing N so that $\mathcal{S}^{(N)} \supseteq \mathcal{C}_x$ guarantee that at a successive step, even in presence of a parameter estimation error which causes a prediction error, the terminal set will still be reachable in N steps. This is reflected in (39) where the predecessor set is computed for each vertex model separately, and hence with different input values, in (39b), and then taking the intersection in (39c), as opposed to (38) where a single predecessor set is computed with the same input for all the vertex models. In other words, the order of set update and intersection is reversed in (38) and (39), due to their different reliance on the parameter vector estimate.

Theorem 3. Let Condition U1 and Assumption 1 hold. Consider (23), let $\mathcal{P}(\xi)$, $\kappa(\xi)$ be designed according to (26) $\bar{h} \in \mathbb{Z}_{0+}$ be such that $\mathcal{C}^{(\bar{h}+1)} = \mathcal{C}^{(\bar{h})} = \mathcal{C}_x$ in (38), and let $\mathcal{C}_{xu}$ be defined by (37). Let $\mathcal{X}_N = \mathcal{X}^\infty$ from (35), where $\mathcal{X}_{xu} = \mathcal{C}_{xu}$, and $N \in \mathbb{Z}_{0+}$ be such that $\mathcal{S}^{(N)} \supseteq \mathcal{C}_x$. Then, (23) is always feasible in $\mathcal{C}_x$, which is RPI for the closed-loop, and

$$\alpha_1(\|x\|) \leq \mathcal{V}_{\xi_N(t)}(x(t)) \leq \alpha_2(\|x\|) \tag{40a}$$

$$\Delta\mathcal{V}_{\xi_N(t)}(x(t)) \leq -c_\Delta \alpha_\Delta(\|x\|) + \tilde{\gamma}(\|\tilde{\bar{\xi}}_{0|t}\|), \tag{40b}$$

where $c_\Delta \in (0,1)$, i.e., the closed-loop is ISS with respect to $\tilde{\bar{\xi}}_{0|t} = \bar{\xi} - \xi_{0|t} \in \tilde{\Xi}(\xi_{0|t})$.

Proof. We demonstrate that N1-N2, R1-R4, U1 hold, so that the results follows from Theorem 1. The design of $\mathcal{X}_N$ and $\mathcal{P}(\xi)$ ensures satisfaction of N1, N2, R3, R4, as in Theorem 2, and U1 holds by the assumptions.

Due to the construction of $\mathcal{C}_{xu}$, for all $x \in \mathcal{C}_x$ there exists $u \in \mathcal{U}$ such that $(x,u) \in \mathcal{C}_{xu}$, and if $(x(t), u(t)) \in \mathcal{C}_{xu}$, by convexity $x(t+1) = \sum_{i=1}^{\ell} [\bar{\xi}]_i A_i x(t) + Bu(t) \in \mathcal{C}_x \subseteq \mathcal{X}$, for all $\xi_{0|t} + \tilde{\bar{\xi}}_{0|t} = \bar{\xi} \in \Xi$. Thus, R1 is satisfied. Since $\mathcal{C}_x \subseteq \mathcal{S}^{(N)}$ for every $x(t) \in \mathcal{C}_x$ and $\xi_N(t) \in \Xi^{N+1}$, there exists an input sequence of length N such that $(x_{k|t}, u_{k|t}) \in \mathcal{C}_{xu}$ for all $k \in \mathbb{Z}_{[0,N-1]}$, and $x_{N|t} \in \mathcal{X}_N$. For $u_{0|t}$ such that $(x_{0|t}, u_{0|t}) \in \mathcal{C}_{xu}$, $\sum_{i=1}^{\ell} [\xi_{0|t} + \tilde{\bar{\xi}}_{0|t}]_i A_i x_{0|t} + Bu_{0|t} \in \mathcal{C}_x$, for every $\tilde{\bar{\xi}}_{0|t} \in \tilde{\Xi}(\xi_{0|t})$. Thus, R2 is satisfied and as said, the result follows from Theorem 1. □

Remark 5. $\mathcal{X}^\infty$ computed for (33) is PI for any known and time varying $\xi \in \Xi$, when system (21a) and controller (24) use the same ξ. This is not the case when the estimate ξ is different from the real parameter $\bar{\xi}$. Hence, even if $x \in \mathcal{X}^\infty$ this does not imply that $\sum_{i=1}^{\ell} [\xi_{0|t} + \tilde{\bar{\xi}}_{0|t}]_i A_i x + B\kappa(x) \in \mathcal{X}^\infty$ due to $\tilde{\bar{\xi}}_{0|t} \neq 0$. Instead, ensuring $\mathcal{S}^{(N)} \supseteq \mathcal{C}_x$ guarantees existence of an admissible input sequence from any $x \in \mathcal{C}_x$ that reaches $\mathcal{X}^\infty$ in at most N steps satisfying (23d). □

An alternative construction of $\mathcal{C}_x$ is an RCI set that is smaller than $\mathcal{C}^\infty$, that: (i) is computed for a given $N \in \mathbb{Z}_+$, i.e., it does not impose restrictions on N, and (ii) is computed in a predetermined number of iterations, and described by a finite number of constraints. This alternative method requires the modification of (24) such that $\mathcal{X}^\infty$ is RPI for (21) in closed-loop with $\kappa(x)$ for any $\tilde{\xi}_{0|t} \in \Xi(\xi_{0|t})$.

In order to obtain $\mathcal{X}_N$ which is RPI when $\tilde{\xi}_{0|t} \neq 0$, we compute a more conservative control law (24), which is obtained from (26) with the additional constraints $E_i = E_j$, $G_i = G_j$, for all $i, j \in \mathbb{Z}_{[1,\ell]}$ that guarantee that $K_i = K$, for all $i \in \mathbb{Z}_{[1,\ell]}$, while still, in general, $P_i \neq P_j$ for all $i, j \in \mathbb{Z}_{[1,\ell]}$. This results in a control law (24) that is independent of ξ as in[6], although here the LMI is solved only at design time to avoid excessive computational burden.

Lemma 4. Consider (21) where (24) is computed from (26) with the additional constraints $E_i = E$, $G_i = G$, for all $i \in \mathbb{Z}_{[1,\ell]}$, and $\mathcal{X}^\infty$ is computed from (35) with $\mathcal{X}_{xu} = \mathcal{X} \times \mathcal{U}$. Then, $\mathcal{X}_N$ is RPI for (21) in closed loop with (24) for every $\tilde{\xi}_{0|t} \in \Xi(\xi_{0|t})$. □

The proof of Lemma 4 is in Appendix A.

Given an RPI set $\mathcal{X}^\infty$ for (21), (24) with respect to $\tilde{\xi} \in \Xi(\xi)$, $\xi \in \Xi$, consider the set sequence

$$\begin{aligned} \mathcal{R}^{(0)} &= \mathcal{X}^\infty, \\ \mathcal{R}^{(h+1)} &= \{x \in \mathcal{X} : \exists u \in \mathcal{U}, A_i x + Bu \in \mathcal{R}^{(h)}, \forall i \in \mathbb{Z}_{[1,\ell]}\}. \end{aligned} \tag{41}$$

$\mathcal{R}^{(h)}$ is the set of states that can be steered to $\mathcal{R}^{(0)}$ in h steps, for any unknown $\xi_h \in \Xi^h$ and any $\tilde{\xi}_h \in \tilde{\Xi}^h(\xi_h)$, while satisfying state and input constraints by using state feedback. Due to $\mathcal{R}^{(0)}$ being RPI, $\mathcal{R}^{(h+1)} \supseteq \mathcal{R}^{(h)}$ and $\mathcal{R}^{(h)}$ is RCI for every $h \in \mathbb{Z}_{0+}$. The operations in (41) are similar to those in (38), except for removing the set intersection due to $\mathcal{X}^\infty$ being RPI.

Theorem 4. Let Condition U1 hold and let $\mathcal{P}(\xi)$, $\kappa(\xi)$ be designed according to (26), with the additional constraints $E_i = E_j$, $G_i = G_j$, for all $i, j \in \mathbb{Z}_{[1,\ell]}$, where and Assumption 1 hold with such additional constraints. Let $\mathcal{X}_N = \mathcal{X}^\infty$ be constructed from (35) with $\mathcal{X}_{xu} = \mathcal{X} \times \mathcal{U}$. Given $N \in \mathbb{Z}_+$, let $\mathcal{C}_{xu}$ in (23c) be constructed by (37) with $\mathcal{C} = \mathcal{R}^{(N)}$. Then (23) is always feasible in $\mathcal{C}_x$, which is RPI for the closed-loop, and

$$\alpha_1(\|x\|) \leq \mathcal{V}_{\xi_N(t)}(x(t)) \leq \alpha_2(\|x\| \tag{42a}$$

$$\Delta\mathcal{V}_{\xi_N(t)}(x(t)) \leq -c_\Delta \alpha_\Delta(\|x\|) + \tilde{\gamma}(\|\tilde{\xi}_{0|t}\|) \tag{42b}$$

where $\bar{c}_\Delta \in (0, 1)$, i.e., the closed-loop is ISS with respect to $\tilde{\xi}_{0|t} = \bar{\xi} - \xi_{0|t} \in \tilde{\Xi}(\xi_{0|t})$.

Proof. The proof is similar to that of Theorem 3, where the set $\mathcal{C}_{xu}$ is (37) constructed from $\mathcal{C}_x$, i.e., the N-steps robust backward reachable set of $\mathcal{X}_N$ which is now RPI for (22) (24) with respect to $\tilde{\xi}$, because $K_i = K$ for all $i \in \mathbb{Z}_{[1,\ell]}$. Thus, by choosing the prediction horizon equal to N, R2 holds. □

Finally, it is worth noting that the results of Corollary 2 apply directly to the constrained case $\mathcal{X} \times \mathcal{U} \subset \mathbb{R}^N$ since they only affect a neighborhood of the origin where the constraints are inactive. Thus, the only remaining design step is the choice of $\mathcal{G}$, $\mathcal{G}_N$ in (7), (8) to satisfy Condition U1, which ensure integration with a general parameter estimator.

4.3 | Interface to the Parameter Estimator And Overall Result

The final step is the design of (7) and (8) such that Condition U1 holds, as required by Theorems 2, 3, 4.

The virtual parameter estimate $\xi(t)$ is computed at time t by $\mathcal{G}$ from the parameter estimate $\vartheta(t)$, which ensures that $\xi \in \Xi$. Thus, $\mathcal{G}$ operates as a projection of ϑ onto the set Ξ

$$\mathcal{G}(\vartheta) = \arg\min_{\xi} J\left(\sum_i [\xi]_i A_i, \bar{A}(\vartheta)\right) \tag{43a}$$

$$\text{s.t. } \xi \in \Xi \tag{43b}$$

where J denotes the projection metrics, which is usually aimed at minimizing some distance between the models based on the estimated actual and virtual parameters, possibly with additional regularization terms, and can be easily defined to be a quadratic function of ξ.

To satisfy the Condition U1, the predicted virtual parameter sequence must satisfy for all $t \in \mathbb{R}_+$, $\xi_{k|t+1} = \xi_{k+1|t}$, $k \in \mathbb{Z}_{[0,N-1]}$, and for all $\xi_N(t) \in \Xi^{N+1}$. Since $\bar{\xi}$ in (21) is assumed to be constant or slowly varying, an obvious choice would be $\xi_{k|t} = \xi(t)$, for all $k \in \mathbb{Z}_{[0,N]}$, for all $t \in \mathbb{Z}_{0+}$. However, this choice violates Condition U1. In fact, if the entire parameter prediction vector $\xi_N(t)$ suddenly changes, the value function $\mathcal{V}_{\xi_N}$ may jump and may not decrease. This is due to using the pLF as terminal cost, which in turn allows real-time operation by only solving QPs, as opposed to bounding the infinite-horizon cost function[5,6,9], which requires solving LMIs in real-time.

Instead, Condition U1 can be achieved by defining the virtual parameter sequence update function $\mathcal{G}_N$ in (8) as the delay buffer $\xi_{k|t} = \xi(t-N+k)$, for all $k \in \mathbb{Z}_{[0,N]}$, which, arranging $\xi_N(t)$ as the vector $\hat{\xi}_N(t) = (\xi_{0|t}, \dots, \xi_{N|t}) \in \mathbb{R}^{\ell N}$, can be implemented as

$$\xi_N(t) = \begin{bmatrix} 0 & I & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & I \\ 0 & \cdots & 0 & 0 \end{bmatrix} \xi_N(t-1) + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ I \end{bmatrix} \xi(t) \tag{44}$$

It may be of interest noting the parallel between the low pass filtering of the adaptation signal in classical adaptive control and the time delay in the prediction parameter update for IAMPC, as both "ease the adjustments into" the controller by slowing the updates on the parameter estimate in frequency and time, respectively.

We can now summarize the result for the IAMPC strategy for uncertain polytopic systems.

Result 5. Given (4), the IAMPC controller where $\mathcal{G}$ and $\mathcal{G}_N$ in (7) and (8) are designed according to (43) and (44), respectively, and the control law (9) solves (23) at every $t \in \mathbb{Z}_{0+}$, where $\mathcal{P}(\xi)$ and $\kappa(\xi)$ are constructed from (26), N, $\mathcal{C}_x$ and $\mathcal{X}_N$ are designed according to Theorem 3 or Theorem 4, the closed-loop solves Problem 1, since it recursively satisfies (21b), is ISS in the RPI set $\mathcal{C}_x$ with respect to $\tilde{\xi}_{0|t} = \bar{\xi} - \xi_{0|t}$, i.e., the N-steps delayed estimation error $\tilde{\xi}_{0|t} = \bar{\xi} - \xi(t-N)$, and is AS for a sufficiently small estimation error $\tilde{\xi}_{0|t}$. Furthermore, if J in (7) is quadratic and convex in ξ, both (23) and (7) are quadratic programs with guaranteed convergence. □

Solving only QPs during execution results in a significant reduction of computational load and code complexity with respect to robust MPC based on LMIs[5,6,9]. This makes the method feasible in applications with fast dynamics and microprocessors with limited capabilities, such as those in[2,32]. On the other hand, the method proposed here guarantees ISS and local AS for sufficiently small error versus the robust stability result in[5,6,9]. It should be noted that the method proposed here still ensures robust constraint satisfaction. This is aligned with the philosophy proposed in this and other works, see, e.g.,[11,12,16] where models are used to ensure robust constraint satisfaction and data is used to improve performance, i.e., the ultimate bound due to ISS and eventually AS, here.

While still using robust invariant sets, the approach proposed here is different from tube MPC. In the latter, the uncertainty is dealt with by a disturbance rejection controller and the states and input constraints are tightened accordingly, while IAMPC does not require constraints tightening, thus leaving more degrees of freedom to the controller for optimizing performance. Considering the standard, i.e., rigid, tube MPC[7], the difference is noticeable as that exploits the RPI set obtained based on the disturbance rejection controller, while IAMPC uese the RCI set that can be chosen to be the provide the largest feasible domain, i.e., the maximal RCI set. The reduced conservativeness of IAMPC with respect to tube MPC is due to the different assumptions on the uncertainty, that translate into different objectives, namely, robust stabilization in tube MPC as opposed to ISS in IAMPC, which relies on the estimator to reduce the estimation error to eventually achieve AS. While more flexible tube MPC, e.g.,[39,14], allow for dynamically allocating the control range among disturbance rejection and nominal control, they may still induce some conservativeness due to the parametrization of sets, and the usually fixed disturbance rejection control law.

5 | EXTENSIONS: UNCERTAINTY IN INPUT GAIN, OUTPUT TRACKING

Next we briefly discuss some modifications that allow IAMPC to handle different classes of uncertain polytopic systems that are relevant to some practical applications. In particular, the cases with additional uncertainty in the input-to-state matrix B, and the cases of reference tracking. These modifications require additional constraints in the design of the optimization problem (23).

When the input-to-state matrix B is also dependent on uncertain parameters,

$$x(t+1) = A(\bar{\vartheta})x(t) + B(\bar{\vartheta})u(t) \tag{45}$$

(21a) is reformulated as

$$x(t+1) = \sum_{i=1}^{\ell} [\xi]_i \left(A_i x(t) + B_i u(t)\right), \tag{46}$$

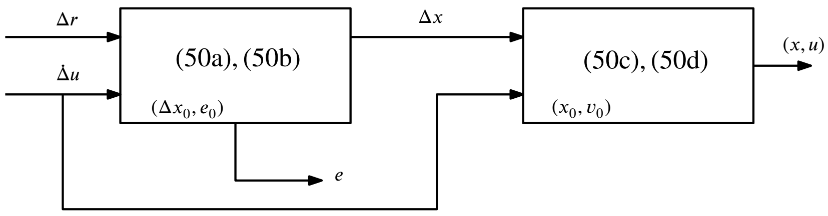


FIGURE 2 Cascaded structure of the constrained velocity model used for reference tracking in IAMPC.

where the convex combination vector ξ now determines $(A, B) = \left(\sum_{i=1}^{\ell}[\xi]_i A_i, \sum_{i=1}^{\ell}[\xi]_i B_i\right)$. The proposed approach can model (46) with only few modifications in the design and proving steps of Section 4.

In terms of design, to retain convexity of the LMI (26), we impose the additional constraints $E_i = E$, $G_i = G$ for all $i \in \mathbb{Z}_{[1,\ell]}$ so that we compute a pLF for (46) in closed loop with a fixed linear controller $u = Kx$ as in[6].

According to Lemma 4, the set $\mathcal{X}^\infty$ computed according to (35) with $K_i = K$, for all $i \in \mathbb{Z}_{[1,\ell]}$, is RPI. Thus, for this class of systems, the computation of C_x as the N-steps backward reachable set of the RPI $\mathcal{X}^\infty$ by (41) does not impose additional conservativeness in the terminal controller with respect to using instead the maximal RCI (38), since in both cases it is the linear feedback $u = Kx$, i.e., $K_i = K$, for all $i \in \mathbb{Z}_{[1,\ell]}$, in (24).

In terms of the proofs, the only change necessary to Theorem 2 is to prove that (17a) in condition R4 is satisfied. For the class of systems (45) the error is

$$\varepsilon_x = \left(\sum_{i=1}^{\ell}[\tilde{\xi}]_i A_i\right)x + \left(\sum_{i=1}^{\ell}[\xi]_i B_i\right)u \leq c_A\|\tilde{\xi}\|\|x\| + c_B\|\tilde{\xi}\|\|u\| \tag{47}$$

However, under Conditions N1, N2,[1, Prop. 2.18] shows that $L(x, u) \leq c_v F(x)$. Then, $u'Ru + x'Qx \leq c_v x'P(\xi)x \leq c_v \max_i x'P_i x \leq \max_i \lambda_{\max}(P_i)\|x\|^2$ and hence

$$\|u\|^2 \leq \frac{c_v \max_i \lambda_{\max}(P_i) - \lambda_{\min}(Q)}{\lambda_{\min}(R)}\|x\|^2. \tag{48}$$

Combining (48) with (47) we obtain $\varepsilon_x \leq c_{A,B}\|\tilde{\xi}\|\|x\|$ which satisfies (17a) in Condition R4.

Another useful extension, that is relevant to real applications, is to consider output reference tracking problems. While often output reference tracking is transformed into a regulation problem by "shifting" the state and input vectors by their setpoints, $x^s \in \mathbb{R}^n$, $u^s \in \mathbb{R}^m$, in the cost function

$$F(x_{N|t} - x^s_{N|t}) + \sum_{k=0}^{N-1} L(x_{k|t} - x^s_{k|t}, u_{k|t} - u^s_{k|t}). \tag{49}$$

this is harder in the context of uncertain systems because the generation of setpoints for state and input from an output reference is based on the model. As the model is updated, the setpoint will change from step to step in an unpredictable way, and the terminal cost condition $F(\delta x_{N|t+1}) + L(\delta x_{N-1|t+1}, \delta u_{N-1|t+1}) - F(\delta x_{N|t}) \leq 0$, where $\delta a = a - a^s$, may not even guarantee the decrease of the nominal optimal cost, due to the setpoint changes in the different terms.

Instead, to avoid the explicit setpoint calculation we use a velocity model[40,41], which considers an incremental input formulation $v(t+1) = v(t) + \Delta u(t)$, where $v(t) = u(t-1)$, and the predicted tracking error e, $e(t+1) = y(t+1) - r(t+1) = e(t) + C\Delta x(t+1) - \Delta r(t)$, where $\Delta r(t) = r(t+1) - r(t)$ is the reference change, resulting in

$$\varphi(t+1) = \sum_{i=1}^{\ell}[\xi]_i A^v_i \varphi(t) + B^v \Delta u(t) + G^v \Delta r(t) \tag{50a}$$

$$e(t) = C^v \varphi(t). \tag{50b}$$

with state $\varphi(t) = (\Delta x(t), e(t))$. As opposed to[41], here we cannot exploit the exact model knowledge to enforce the constraints by the exact knowledge of the model, hence we formulate an auxiliary system

$$\omega(t+1) = \omega(t) + \sum_{i=1}^{\ell}[\xi]_i E^v_i \varphi(t) + F^v \Delta u(t), \tag{50c}$$

$$z(t) = \omega(t) + H^v \Delta u(t), \tag{50d}$$

with state $\omega(t) = (x(t), v(t))$, which is a discrete-time integrator for $(\Delta x, \Delta u)$. The complete model (50) has state $\chi(t) = (\varphi(t), \omega(t))$, performance output $e(t)$, and constrained output $z(t) = (x(t), u(t))$, and has the structure shown in Fig. 2. Because of the cascade coupling and since (50c), (50d) model the constrained variables, (50a), (50b) need to be controlled such that the trajectory of $(\Delta x(t), \Delta u(t))$ ensures that $(x(t), u(t))$ satisfy the constraints. However, for achieving output tracking only the tracking error e needs to vanish. Under the assumption of a unique steady state (x^s, u^s) per each constant reference value r, tracking a constant reference implies convergence of (x, u) to a unique equilibrium. Thus, when tracking a constant reference, at steady state Δu and Δx asymptotically vanish, and the controller asymptotically stabilizes the state $\varphi(t) = (\Delta x(t), e(t))$ of (50a), (50b).

Based on (50) with cost function (2a) where $L(\chi, \Delta u) = \varphi' Q \varphi + \Delta u' R \Delta u$, $F_\xi(\chi) = \varphi' P(\xi) \varphi$ the unconstrained design of Section 4 , guarantees that

$$\mathcal{V}_{\xi_N(t+1)}(\chi(t+1)) - \mathcal{V}_{\xi_N(t)}(\chi(t)) \leq -\alpha|\varphi(t)|^2 + \gamma|\tilde{\xi}_{0|t}|^2 + \rho|\Delta r(t)|, \tag{51}$$

is an ISS-LF with respect to the parameter estimation error and the reference step change. For the constrained case, the only notable difference is that for determining the length of the horizon that satisfies Condition R2 one needs to consider the augmented set

$$\begin{aligned}\hat{\mathcal{D}}_\chi = \big\{&(\Delta x, r, x, v) \in \mathbb{R}^n \times \Omega_r \times \mathcal{C}_x \times \mathcal{V} \,:\, \exists \Delta u, \\ &(x, v + \Delta u) \in \mathcal{C}_{xu},\; x + A_i \Delta x + B \Delta u \in \mathcal{C}_x, \forall i \in \mathbb{Z}_{[1,\ell]}\big\},\end{aligned} \tag{52}$$

that includes the reference r, for which the admissible set $r \in \Omega_r$ must be known. Then, $N \in \mathbb{R}_+$ is chosen such that $\hat{\mathcal{D}}_\chi \subseteq \mathcal{S}^{(N)}$, i.e., the augmented set is contained in the N-steps backward reachable set of $\mathcal{X}_N$.

6 | SIMULATION CASE STUDIES

In this section, we demonstrate the effect of the proposed IAMPC on two case studies: an illustrative numerical example of an uncertain second order system, and a more realistic example of compressor control of a variable refrigerant flow air-conditioning system. The simulations are executed in a 13" Macbook Pro 2016, with Intel i-7 3.3GHz dual-core CPU and 16GB RAM. For solving LMIs we use SDPT3 [42], and for polyhedral computations we use the linear programming solver in CDD [43] and the projection algorithms in MPT3 [38]. For solving online the QPs of the IAMPC, we use the ADMM algorithm [44] code in C without any external library in a Matlab a mex function.

6.1 | Numerical Example

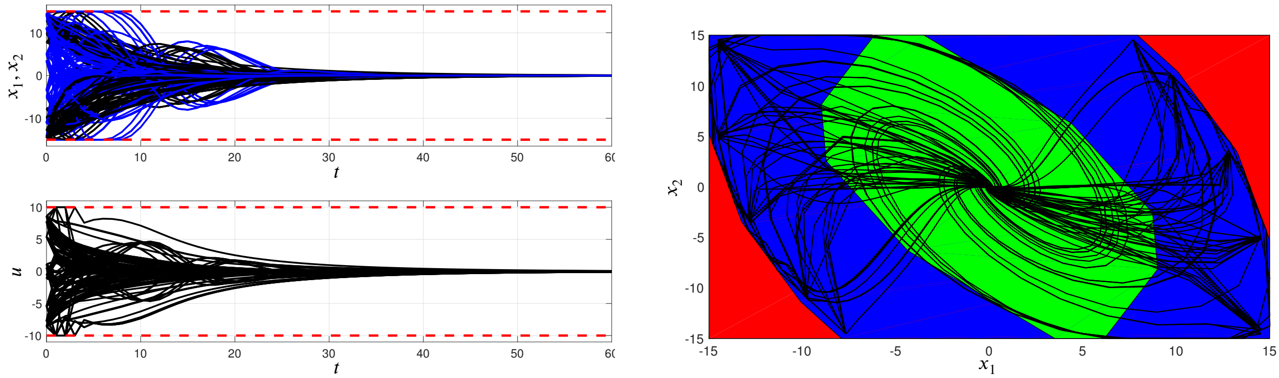
We consider a system already formulated as the uncertain polytopic system (21), i.e., $\xi = \vartheta$, where $x = [x_1\ x_2]'$, $\ell = 5$, and $A_1 = \left[\begin{smallmatrix}1 & 0.2\\ 0 & 1\end{smallmatrix}\right]$, $A_2 = 1.1 \cdot A_1$, $A_3 = 0.6 \cdot A_1$, $A_4 = \left[\begin{smallmatrix}0.9 & 0.3\\ 0.4 & 0.6\end{smallmatrix}\right]$ $A_5 = \left[\begin{smallmatrix}0.95 & 0\\ 0.8 & 1.02\end{smallmatrix}\right]$, and $B = [\,-0.035\ -0.905\,]'$. Some of the vertex systems are AS and some are unstable. The constraint sets in (21b) are $\mathcal{X} = \{x \in \mathbb{R}^2 \,:\, |x_i| \leq 15,\ i = 1, 2\}$, $\mathcal{U} = \{u \in \mathbb{R} \,:\, |u| \leq 10\}$. As shown in [35] for a similar system, without proper terminal cost adaptation, the closed-loop with MPC may not be AS even if the perfect prediction model is estimated, unless other components of the MPC optimal control problem are adjusted as well.

The parameter estimation algorithm (6) computes a moving horizon least squares (MHLS) estimate based on a past data window of N_m steps, applies a first order filter and projects the results onto Ξ. By defining $z = [z_1'\ z_2]'$, where $z_1 \in \mathbb{R}^{(n+m)N_m}$ is a buffer of past state-input pairs and $z_2 \in \mathbb{R}^\ell$ is the filtered parameter vector

$$\begin{aligned}\zeta(z, x, u) &= \begin{bmatrix}\texttt{shift}(z_1, [x'u']') \\ (1 - \varsigma)z_2(t) + \varsigma\,\texttt{MHLS}(z_1, x, u),\end{bmatrix} \\ \psi(z, x) &= z_2,\end{aligned}$$

where `shift` denotes the operator that shifts the data buffer and adds the new data to the buffer and `MHLS` denotes the moving horizon least square computation. We implement $\mathcal{G}(\vartheta)$ as in (43), using the 2-norm projection onto Ξ, and $\mathcal{G}_N(\xi_N(t))$ as in (44).

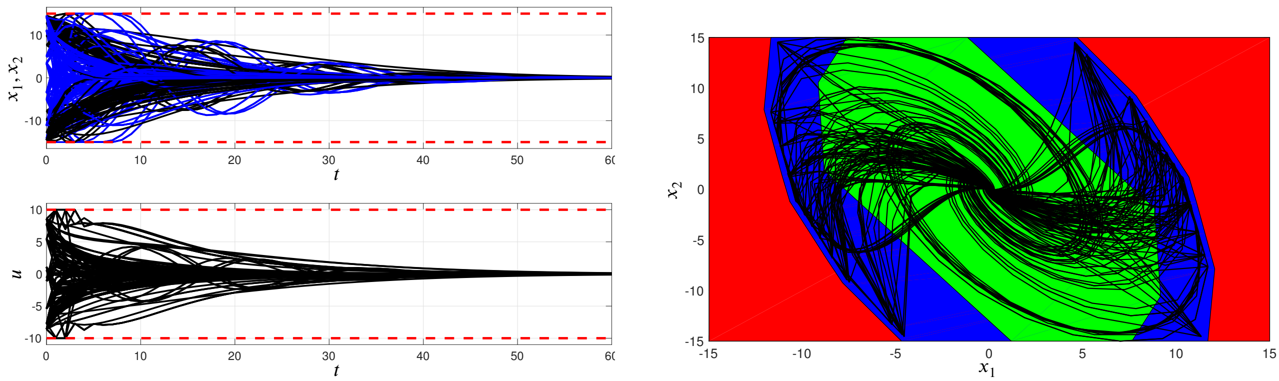
The case where we design the controller according to Theorem 3, where $\mathcal{C}_x = \mathcal{C}^\infty$, and $N = 8$, which is the smallest value such that $\mathcal{S}^{(N)} \supseteq \mathcal{C}^\infty$ by (39), and $\varsigma = 1/8$ is shown in Figure 3. The initial states in the simulations were obtained by moving each of the 18 vertices of $\mathcal{C}_x$ slightly into its interior (to avoid numerical ill-conditioning). For each initial state, 8 different simulations were executed with $\bar{\xi} \in \Xi$ being, either randomly generated with a uniform distribution over Ξ or with $\bar{\xi} \in \Xi$ being a standard unit vector so that the simulated system is one of the vertex systems. These two possibilities were chosen randomly



(a) Top plot: state trajectories, x_1 (black), x_2 (blue). Bottom plot: input trajectories, u (black). Constraints are shown in red.

(b) Phase plane trajectories (black), $\mathcal{X}^\infty$ (green), $\mathcal{C}_x$ (blue), $\mathcal{X}$ (red).

FIGURE 3 Simulation of the numerical example system in closed-loop with IAMPC for $\mathcal{C}_x = \mathcal{C}^\infty$ and $N = 8$.



(a) Time trajectories of state (upper plot, $[x]_1$ blue, $[x]_2$ black), and input (lower plot) for the case of a slower estimator $\varsigma = 1/15$.

(b) Phase plane trajectories (black), $\mathcal{X}^\infty$ (green), $\mathcal{C}_x$ (blue), $\mathcal{X}$ (red) for the case when $\mathcal{C}_x = \mathcal{R}^{(N)}$ and $N = 6$

FIGURE 4 Additional simulations of the numerical example system in closed-loop with IAMPC with different parameters.

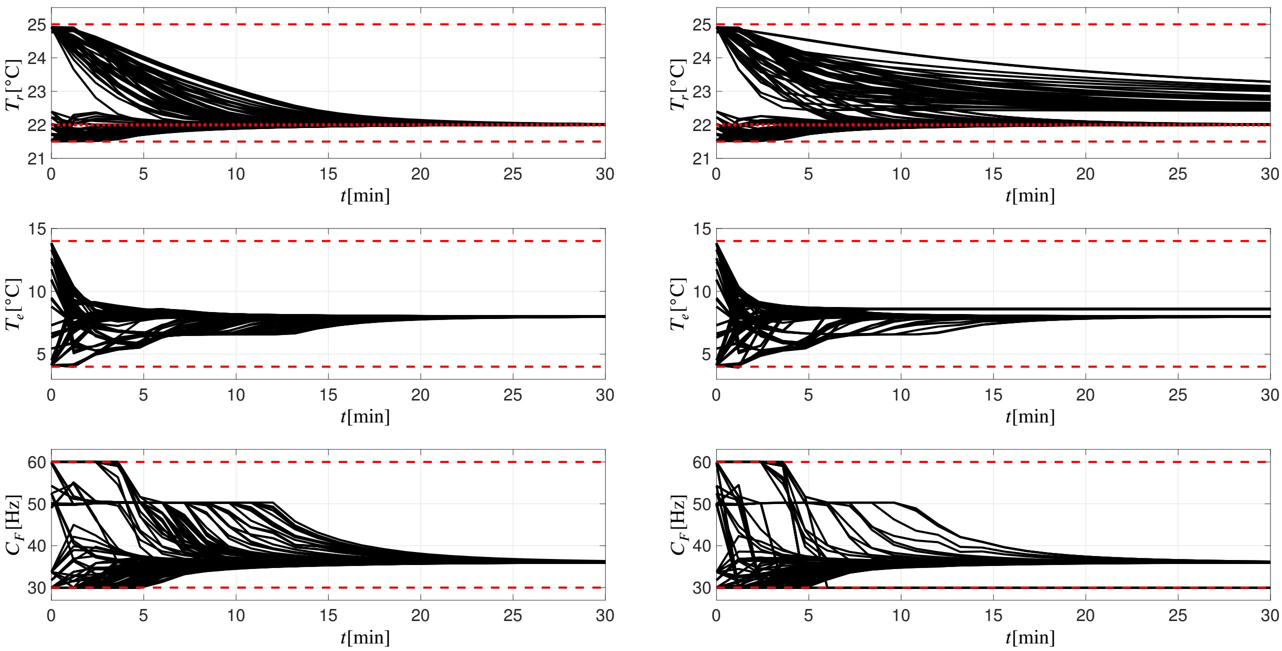
with equal probability. In all simulations, the estimator was initialized with $\xi(0) = 1/\ell \cdot [1 \ldots 1]'$. In all 96 simulations shown in Figure 3, the constraints are always satisfied and the closed-loop eventually converges to the origin, indicating both ISS and AS.

The worst case controller execution time is less than 4ms, including the function overhead. The procedure to designs the controller offline by solving the LMI (26), constructing the RPI set (35), and the maximal RCI set (38) and horizon N (39) or the N-steps RCI set (41) executes in less than 20s.

Figure 4 shows additional simulations when some of the IAMPC algorithm parameters are changed. Figure 4(a) highlights ISS and hence the dependency of the performance on the parameter estimator. Due to $\varsigma = 1/15$, convergence of the state is slower according to ISS results in Theorem 1, but constraint satisfaction is still guaranteed. Figure 4(b) shows another case, when the controller is designed according to Theorem 4, with $\mathcal{C}_x = \mathcal{R}^{(N)}$ and $N = 6$, and $\mathcal{R}^{(N)}$ is determined by (41). Indeed, $\mathcal{C}_x$ in Figure 4(b) is smaller than that in Figure 4(b), due to $\mathcal{R}^{(N)} \subset \mathcal{C}^\infty$, but in this case the controller can be designed with a shorter horizon, and hence lower computational burden.

6.2 | Control of Variable Refrigerant Flow Air Conditioner

The second case study is inspired by the compressor control of a variable refrigerant flow air conditioner (VRF-AC). The model is a simplification of that in[45], obtained by first principles and data acquired in experiments, where only one room is active, and the two flow valves and the fan speeds are kept constant at their setpoints, since these are either measured disturbances, or are used



(a) Using IAMPC for control.

(b) Using nominal MPC with external protection logic.

FIGURE 5 Simulation results of VRF-AC regulation by IAMPC. Time trajectories (solid) constraints (dash), setpoint (dot).

to achieve control objectives that are not considered here. The resulting 4^{th} order model is centered around the setpoint (x_{ss}, u_{ss}), where $x_{\rm ss} = [22\ 8\ 60\ 22]$, $u_{\rm ss} = 45$, and the states are chosen as $x = [T_r\ T_d\ T_e\ \tau]'$, T_r[deg] is the room temperature, T_e[deg] is the evaporator temperature, T_d[deg] is the compressor discharge temperature, τ is a state related to the internal conditions of the air conditioner, and the control input is the compressor frequency $u = C_F$[Hz]. We consider uncertainty in the thermal mass of the room by $\pm 35\%$ and in the efficiency of the energy transfer from the evaporator to the room by $\pm 20\%$, obtaining (4), which is then reformulated as (21) with $\ell = 4$. We first consider a regulation problem, where the controller must enforce upper and lower bounds on state, $(\overline{x} - x_{\rm ss}) = [3\ 6\ 10\ 20]$, $(\underline{x} - x_{\rm ss}) = -[0.5\ -4\ -10\ -20]$, and input $\overline{u} = 24$, $\underline{u} = -6$. We design the same parameter estimator as in the previous case study and the IAMPC with $N = 10$, $T_s = 1$min, $N_m = 3$. The simulation results for the same tests as in the numerical example in Section 6.1, i.e., initial condition of the estimator $\xi(0) = 1/\ell \cdot [1 \dots 1]'$, initial state just in the interior of C_x from each of the 42 vertexes, and, for each, 4 different realizations of the uncertainty, both random values and extremal values with equal probability, are shown in Figure (5(a)), where we show the temperatures with significant constraint activations. The constraints are enforced despite the uncertainty and the system is ISS which empirically verifies the theoretical results we presented in this paper. In fact, the system is eventually AS after the estimator converges close enough to a correct value. For comparison in Figure 5(b) we show the behavior of a nominal MPC where, if constraints are violated, the compressor is set to its minimal frequency to avoid damage, which may emulate the equipment protection logic. It can be noticed that, for the nominal MPC case, the room temperature converges more slowly, and that in a number of simulations there is a bias due to the constraint violation activating the protection logic.

As an additional investigation, we show the case where we modify the design as discussed in Section 5 to allow for piecewise constant setpoint tracking. The results are shown in Figure 6, where one can once again see constraint satisfaction, ISS and AS for piecewise constant references. For the segment where the reference ramps up, the closed-loop system is ISS with respect to the reference rate of change.

For this application where the computational resources are limited[45], IAMPC is appealing because it solves QP similar in nature to those of a nominal MPC, but provides guarantees in the presence of uncertainty which are not provided by nominal MPC. QP solvers such as[46,44] are feasible for implementation in the air conditioner microcontroller[45], as opposed to the LMI solvers required by robust MPC approaches such as[5,6,9].The worst case controller execution time is less than 7ms, including the function overhead, and the offline design procedure executes in less than 30s.

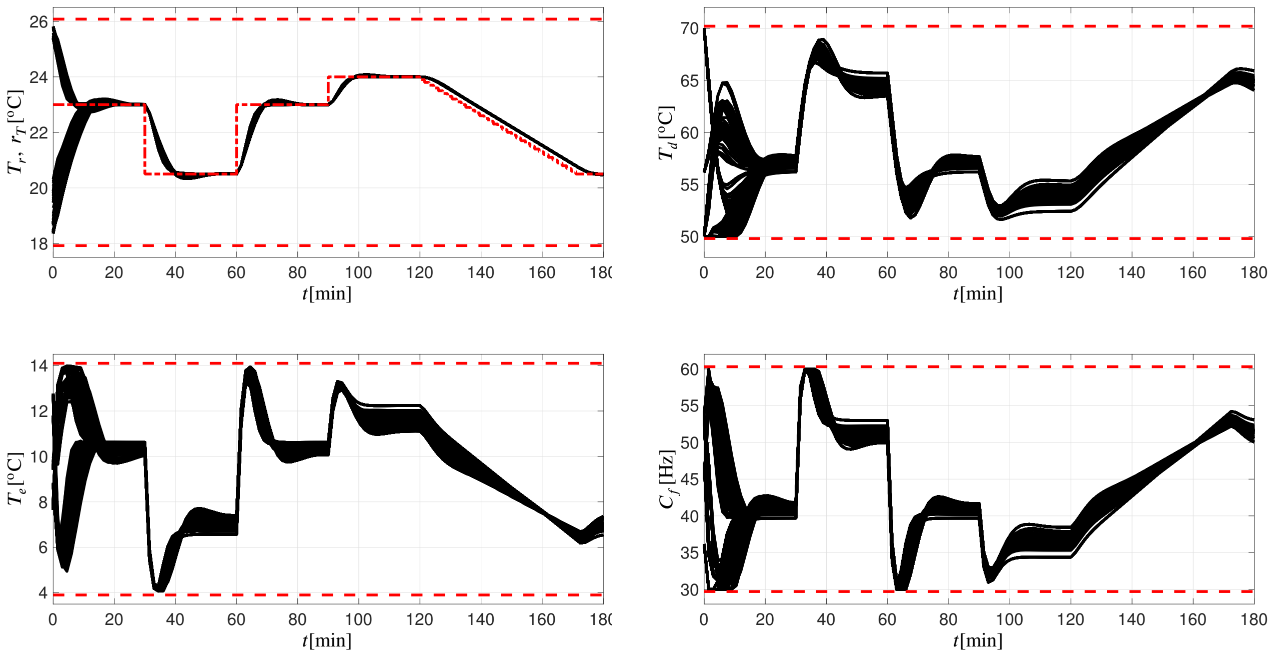


FIGURE 6 Simulation results of VRF-AC tracking by IAMPC. States T_r, T_e, T_d and input C_f (solid, black), room temperature reference r_T (dash-dot, red) and constraint bounds (dash, red).

7 | CONCLUSIONS AND FUTURE WORK

We have proposed an indirect adaptive MPC that exploits a priori model knowledge to ensure system safety in terms of constraint satisfaction, and online data to improve closed-loop performance in terms of stabilization. The method guarantees robust constraint satisfaction, recursive feasibility, and ISS with respect to the parameter estimation error, and yet retains a computational load similar to nominal MPC. The proposed method achieves a separation principle between control and parameter estimation and hence allows for independent design of controller and estimator, without imposing restriction on the estimation algorithm to be used.

After deriving the design conditions that ensure the desired properties for the case of a general nonlinear system, we provided a constructive procedure for the case of uncertain constrained linear systems. Furthermore we showed several extensions to account for uncertainty in the input matrix, to exploit non-maximal yet simpler RCI sets, and to tracking problem. Including additive disturbances or uncertain offsets is straightforward, by combining the invariant set computations for pLDIs with the ones for additive disturbances see, e.g.,[23], and possibly learning also the additive disturbances, if they are constant or slowly varying.

Future work will involve adjusting the invariant sets in a computational feasible manner for our target embedded applications as the confidence in the parameter estimate increases, and possibly the integration with concepts from dual objective control to guarantee persistency of excitation, thus ensuring estimator convergence.

ACKNOWLEDGMENTS

The authors would like to thank Dr. D. Fan and Dr. J. Zhou for contributions to initial studies related to this work, while they were interns at Mitsubishi Electric Research Laboratories.

APPENDIX

A PROOFS OF LEMMAS AND COROLLARIES

Proof of Lemma 1. The lower and upper bounding $\mathcal{K}_\infty$ functions can be derived exactly as for a standard MPC[1], namely $\alpha_1(\|x\|) = \alpha_m(\|x\|) \leq L(x,u)$ and $\alpha_2(\|x\|) = \rho_\epsilon \alpha_F(\|x\|)$, for some $\rho_\epsilon > 0$ (see[1, Prop. 2.18]).

Since at any $t \in \mathbb{Z}_{0+}$, $\xi_N(t)$ is known, for all $t \in \mathbb{R}_{0+}$, $\xi_{k|t} = \xi(t+k)$, $k \in \mathbb{Z}_{[0,N]}$, and $x(t+1) = x_{1|t}$. As another consequence, at $\xi_{k|t+1} = \xi_{k+1|t}$, for all $k \in \mathbb{Z}_{[0,N]}$, $t \in \mathbb{R}_{0+}$. As discussed on Remark 1 the constraint satisfaction, i.e., recursive feasibility, is affected by $\xi_{0|t+1}, \ldots, \xi_{N-1|t}$, and the cost decrease is affected by $\xi_{N|t+1}$. Thus at time $t+1$ the terminal constraint is affected by the value of $\xi(t+N)$, which had no effect on the dynamics at the previous step, and the terminal cost is affected by the value of $\xi(t+1+N)$.

Consider an arbitrary $(x(t), \xi_N(t)) \in \mathcal{X}_\Xi^F$ for which (2) is feasible, and any $\xi(t+1+N) \in \Xi$. Let $U_N(t)^*$ be the optimal control sequence and $X_N(t)^* = [x^*_{0|t} \ldots x^*_{N|t}]$ be the optimal state trajectory at time t. Consider the input sequence $\tilde{U}_N(t+1) = [u^*_{1|t} \ldots u^*_{N-1|t}\ \kappa(x^*_{N|t}, \xi(t+N))]$, the corresponding state trajectory $\tilde{X}_N(t+1) = [x^*_{1|t} \ldots x^*_{N|t}\ f_{\xi(t+N)}(x^*_{N|t}, \kappa(x^*_{N|t}, \xi(t+N)))]$ and the resulting cost $\tilde{\mathcal{V}}_{\xi_N(t+1)}(x(t+1))$.

For $k \in \mathbb{Z}_{[0,N-2]}$, $(\tilde{x}_{k|t+1}, \tilde{u}_{k|t+1}) = (x^*_{k+1|t}, u^*_{k+1|t})$ and hence $(\tilde{x}_{k|t+1}, \tilde{u}_{k|t+1}) \in \mathcal{C}_{xu}$, $k \in \mathbb{Z}_{[0,N-2]}$ is satisfied. Since $\tilde{x}_{N-1|t+1} = x_{N|t} \in \mathcal{X}_N$, $(\tilde{x}_{N-1|t+1}, \kappa(x^*_{N-1|t+1}, \xi(t+N))) \in \mathcal{C}_{xu}$, and $x_{N|t+1} \in \mathcal{X}_N$ are satisfied based on Assumption N1. This proves recursive feasibility for any $\xi(t+N)$, i.e., invariance of $\mathcal{X}_\Xi^F$ for any update of $\xi_N(t-1)$ based on (7).

Because of Assumption N2, for any $\xi(t+1+N) \in \Xi$, which is known based on the assumptions,

$$\tilde{\mathcal{V}}_{\xi_N(t+1)}(x(t+1)) \leq \mathcal{V}_{\xi_N(t)}(x(t)) - \alpha_L(\|x(t)\|),$$

where $L(x,u) \leq \alpha_L(\|x(t)\|)$. This proves that (12) recursively holds for any update of $\xi_N(t)$ based on (7). □

Proof of Corollary 1. Since the closed-loop is ISS with respect to the estimation error $\|\tilde{\xi}\|$ which is bounded, the system has an ultimate bound, $\mathcal{X}_{ub}(\|\tilde{\xi}\|) = \{x : \mathcal{V}_{\xi_N}(x) \leq \rho(\|\tilde{\xi}\|)\}$, where $\rho \in \mathcal{K}_\infty$. If $\|\tilde{\xi}_{0|t}\| \leq \bar{c}_2$, there exists $\bar{c}_2 \in \mathbb{R}_+$ such that $\mathcal{X}_{ub}(\bar{c}_2) \subset \mathrm{int}(\bar{\mathcal{X}}_0)$ and hence there exists $\bar{t} \in \mathbb{Z}_{0+}$ such that for all $t \geq \bar{t}$, $x(t) \in \bar{\mathcal{X}}_0$. Since by the assumptions $\|\mathcal{V}_{\xi_N}(x(t+1) - \mathcal{V}_{\xi_N}(x_{1|t})\| \leq c_3\mathcal{V}_{\xi_N}(\varepsilon_x) + c_4\mathcal{V}_{\xi_N}(x_{1|t})$, taking $\hat{c}_2 = \min\{c_2, \bar{c}_2\}$, for $\|\tilde{\xi}\| \leq \bar{c}_2$, from (19) and $\mathcal{V}_{\xi_N}(x_{1|t}) \leq \mathcal{V}_{\xi_N}(x(t))$,

$$\begin{aligned}
\Delta\mathcal{V}_{\xi_N} &\leq -\alpha_\Delta(\|x\|) + c_3\mathcal{V}_{\xi_N}(\varepsilon_x) + c_4\mathcal{V}_{\xi_N}(x) \\
&\leq -\alpha_\Delta(\|x\|) + c_3\alpha_2\circ\gamma(\|\tilde{\xi}\|\|x\|) + c_4\alpha_2(x) \\
&\leq -\alpha_\Delta(\|x\|) + c_3\alpha_2\circ\gamma(\hat{c}_2\|x\|) + c_4\alpha_2(x) \\
&\leq -(1-c_1)\alpha_\Delta(\|x\|)
\end{aligned}$$

for all $x \in \bar{\mathcal{X}}_0$, which proves asymptotic stability for an estimation error that is sufficiently small, yet finite. □

Proof of Lemma 2. We prove that Conditions N1, N2 hold so that we can apply the result of Lemma 1. Since $\mathcal{X}_N = \mathbb{R}^n$, $\mathcal{C}_{xu} = \mathcal{X} \times \mathcal{U} = \mathbb{R}^{n+m}$ condition N1 is trivially satisfied. By designing $\mathcal{P}(\xi)$, $\kappa(\xi)$ using (26), condition N2 is also satisfied. Thus, all the assumptions of Lemma 1 hold, and hence the closed-loop is AS. Specifically, by the steps of Lemma 1 we obtain that for any $\xi_N(t)$ such that $\xi_{t|k} = \xi(t+k)$ for all $k \in \mathbb{Z}_{[0,N]}$, the Lyapunov function $\mathcal{V}^{\mathrm{MPC}}_{\xi_N(t)}(x(t))$ satisfies

$$\mathcal{V}_{\xi^N_{t+1}}(x(t+1)) - \mathcal{V}_{\xi^N_t}(x(t)) \leq -x(t)Qx(t) \leq -\lambda_{\min}(Q)\|x(t)\|^2 = \alpha_\Delta(\|x(t)\|) \in \mathcal{K}_\infty.$$

□

Proof of Corollary 2. We need to prove that the conditions of Corollary 1 hold. Since $0 \in \mathrm{int}(\mathcal{X})$, in a neighborhood of the origin $\mathcal{X}_0 \subseteq \mathcal{X}$, the constraints are inactive and the value function is quadratic $\mathcal{V}_{\xi_N} = x'Hx$ where we dropped the subscript ξ_N from H for the simplicity of notation, as here we only consider values in the same time interval, i.e., for a fixed ξ_N. Let $e = x_1 - x_2$, then for every $\varepsilon > 0$

$$\begin{aligned}
\|\mathcal{V}_{\xi_N}(x+e) - \mathcal{V}_{\xi_N}(x)\| &= \|x_2'Hx_2 - x_2'Hx_2\| = \|e'He + 2e'Hx\| \leq \|e'He + \frac{1}{2\varepsilon}x'Hx + \frac{\varepsilon}{2}e'He\| \\
&\leq \|(1+\frac{\varepsilon}{2})e'He + \frac{1}{2\varepsilon}x'Hx\| \leq c_3\mathcal{V}_{\xi_N}(e) + c_4\mathcal{V}_{\xi_N}(x)
\end{aligned}$$

where $c_3 = 1 + \frac{\varepsilon}{2}$, $c_4 = \frac{1}{2\varepsilon}$. Finally, $\alpha_2(\|x\|) = \lambda_{\max}(H)\|x\|^2$ end hence for any $c_1 \in \mathbb{R}_{(0,1)}$ the condition $(1 + \frac{\varepsilon}{2})\lambda_{\max}(H)(\gamma c_2\|x\|)^2 + \frac{1}{2\varepsilon}\lambda_{\max}(H)\|x\|^2 \leq c_1 x'Qx \leq \lambda_{\min}(Q)\|x\|^2$, can be satisfied by choosing ε large enough, and c_2 small enough, yet finite. Thus conditions of Corollary 1 hold. $\square$

Proof of Lemma 3. We need to show that N1, N2 are satisfied, so that the result follows from Lemma 1. Because of the chosen $\mathcal{P}(\xi)$, N2 is satisfied as in Lemma 2. Choosing $\mathcal{X}_N = \mathcal{X}^\infty$ which is PI for (34), ensures that given any $x \in \mathcal{X}_N$, $\xi \in \Xi$, we have $\sum_i [\xi]_i (A_i + BK_i)x \in \mathcal{X}_N$ and $(x, \kappa(\xi)x) \in \mathcal{C}_{xu} \subseteq \mathcal{X} \times \mathcal{U}$. Thus N1 is satisfied. AS follows by the same arguments of Lemma 2 with $\mathcal{V}_{\xi_N}$ as Lyapunov function, and with $\mathcal{X}_\Xi^F$ as domain of attraction. $\square$

Proof of Lemma 4. Due to (35), if $x \in \mathcal{X}^\infty$, $\sum_{i=1}^{\ell}[\xi]_i(A_i x + BK_i)x \in \mathcal{X}^\infty$, for all $\xi \in \Xi$. Due to the assumptions, $K_i = K$ for all $i \in \mathbb{Z}_{[1,\ell]}$, and hence $\kappa(\xi)x = Kx$. Thus, if $x \in \mathcal{X}^\infty$, for $\bar{\xi} = \xi_{0|t} + \tilde{\xi}_{0|t} \in \Xi$, we have that $(\sum_{i=1}^{\ell}[\xi_{0|t} + \tilde{\xi}_{0|t}]_i A_i x) + BKx = (\sum_{i=1}^{\ell}[\xi_{0|t} + \tilde{\xi}_{0|t}]_i A_i x) + (\sum_{i=1}^{\ell}[\xi_{0|t} + \tilde{\xi}_{0|t}]_i BKx) = \sum_{i=1}^{\ell}[\xi_{0|t} + \tilde{\xi}_{0|t}]_i (A_i x + BK_i)x \in \mathcal{X}^\infty$. $\square$

References

1. Rawlings JB, Mayne DQ. *Model Predictive Control: Theory and Design*. Nob Hill, Madison, Wisconsin . 2009.

2. Di Cairano S. An Industry Perspective on MPC in Large Volumes Applications: Potential Benefits and Open Challenges. In: Proc. 4th IFAC Nonlinear Model Predictive Control Conf. ; 2012: 52–59.

3. Di Cairano S, Kolmanovsky IV. Real-time optimization and model predictive control for aerospace and automotive applications. In: American Control Conf. ; 2018: 2392–2409.

4. Grimm G, Messina MJ, Tuna SE, Teel A. Examples of zero robustness in constrained model predictive control. In: Proc. 42nd IEEE Conf. on Decision and Control. ; 2003: 3724–3729.

5. Kothare MV, Balakrishnan V, Morari M. Robust constrained model predictive control using linear matrix inequalities. *Automatica* 1996; 32(10): 1361–1379.

6. Cuzzola FA, Geromel JC, Morari M. An improved approach for constrained robust model predictive control. *Automatica* 2002; 38(7): 1183–1189.

7. Langson W, Chryssochoos I, Raković S, Mayne DQ. Robust model predictive control using tubes. *Automatica* 2004; 40(1): 125–133.

8. Fukushima H, Bitmead RR. Robust constrained predictive control using comparison model. *Automatica* 2005; 41(1): 97–106.

9. Mao WJ. Robust stabilization of uncertain time-varying discrete systems and comments on "An improved approach for constrained robust model predictive control". *Automatica* 2003; 39(6): 1109–1112.

10. Kim TH, Sugie T. Adaptive receding horizon predictive control for constrained discrete-time linear systems with parameter uncertainties. *Int. J. Control* 2008; 81(1): 62–73.

11. Adetola V, DeHaan D, Guay M. Adaptive model predictive control for constrained nonlinear systems. *Systems & Control Letters* 2009; 58(5): 320–326.

12. Aswani A, Gonzalez H, Sastry SS, Tomlin C. Provably safe and robust learning-based model predictive control. *Automatica* 2013; 49(5): 1216–1226.

13. Tanaskovic M, Fagiano L, Smith R, Morari M. Adaptive receding horizon control for constrained MIMO systems. *Automatica* 2014; 50(12): 3019–3029.

14. Lorenzen M, Allgöwer F, Cannon M. Adaptive model predictive control with robust constraint satisfaction. *IFAC-PapersOnLine* 2017; 50(1): 3313–3318.

15. Ostafew CJ, Schoellig AP, Barfoot TD, Collier J. Learning-based nonlinear model predictive control to improve vision-based mobile robot path tracking. *Journal of Field Robotics* 2016; 33(1): 133–152.

16. Hewing L, Kabzan J, Zeilinger MN. Cautious model predictive control using Gaussian process regression. *IEEE Trans. Control Systems Technology* 2019.

17. Soloperto R, Müller MA, Trimpe S, Allgöwer F. Learning-Based Robust Model Predictive Control with State-Dependent Uncertainty. *IFAC-PapersOnLine* 2018; 51(20): 442–447.

18. Rosolia U, Borrelli F. Learning model predictive control for iterative tasks. a data-driven control framework. *IEEE Transactions on Automatic Control* 2018; 63(7): 1883–1896.

19. Marafioti G, Bitmead RR, Hovd M. Persistently exciting model predictive control. *Int. J. Adaptive Control and Signal Processing* 2013.

20. Rathouskỳ J, Havlena V. MPC-based approximate dual controller by information matrix maximization. *Int. J. Adaptive Control and Signal Processing* 2013; 27(11): 974–999.

21. Heirung TAN, Ydstie BE, Foss B. Dual adaptive model predictive control. *Automatica* 2017; 80: 340–348.

22. Weiss A, Di Cairano S. Robust dual control MPC with guaranteed constraint satisfaction. In: Proc. 53rd IEEE Conf. Decision and Control. ; 2014: 6713-6718.

23. Cheng Y, Haghighat S, Di Cairano S. Robust dual control MPC with application to soft-landing control. In: ; 2015: 3862–3867.

24. Mesbah A. Stochastic model predictive control with active uncertainty learning: a survey on dual control. *Annual Reviews in Control* 2018; 45: 107–117.

25. Sehr MA, Bitmead RR. Probing and Duality in Stochastic Model Predictive Control. In: Rakovic S, Levine W. , eds. *Handbook of Model Predictive Control*Springer. 2019 (pp. 125–144).

26. Adetola V, Guay M. Finite-time parameter estimation in adaptive control of nonlinear systems. *IEEE Trans. Automatic Control* 2008; 53(3): 807–811.

27. Daafouz J, Bernussou J. Parameter dependent Lyapunov functions for discrete time systems with time varying parametric uncertainties. *Systems & control letters* 2001; 43(5): 355–359.

28. Blanchini F, Miani S. *Set-theoretic methods in control*. Springer Science & Business Media . 2007.

29. Di Cairano S. Indirect adaptive model predictive control for linear systems with polytopic uncertainty. In: Proc. American Control Conf. ; 2016: 3570–3575.

30. Fan D, Di Cairano S. Further results and properties of indirect adaptive model predictive control for linear systems with polytopic uncertainty. In: Proc. American Control Conf. ; 2016: 2948–2953.

31. Zhou J, Di Cairano S, Danielson C. Indirect adaptive MPC for output tracking of uncertain linear polytopic systems. In: Proc. American Control Conf. ; 2017: 3054–3059.

32. Hrovat D, Di Cairano S, Tseng HE, Kolmanovsky IV. The development of model predictive control in automotive industry: A survey. In: IEEE Conf. Control Applications. ; 2012: 295–302.

33. Limon D, Alamo T, Raimondo D, et al. Input-to-state stability: a unifying framework for robust model predictive control. In: Nonlinear model predictive control. Springer. 2009 (pp. 1–26).

34. Sontag ED. Smooth stabilization implies coprime factorization. *IEEE Trans. Automatic Control* 1989; 34(4): 435–443.

35. Di Cairano S. Model Adjustable Predictive Control with Stability Guarantees. In: Proc. American Control Conf. ; 2015: 226–231.

36. Borrelli F, Bemporad A, Morari M. *Predictive control for linear and hybrid systems*. Cambridge University Press . 2017.

37. Huynh T, Lassez C, Lassez JL. Practical issues on the projection of polyhedral sets. *Annals of mathematics and artificial intelligence* 1992; 6(4): 295–315.

38. Herceg M, Kvasnica M, Jones C, Morari M. Multi-Parametric Toolbox 3.0. In: ; 2013: 502–510.

39. Fleming J, Kouvaritakis B, Cannon M. Robust tube MPC for linear systems with multiplicative uncertainty. *IEEE Trans. Automatic Control* 2014; 60(4): 1087–1092.

40. Pannocchia G, Rawlings JB. The velocity algorithm LQR: a survey. Tech. Rep. TWMCC-2001–004, Dep. Chemical Eng., Univ. Wisconsin, Madison; : 2001.

41. Betti G, Farina M, Scattolini R. A robust MPC algorithm for offset-free tracking of constant reference signals. *IEEE Trans. Automatic Control* 2013; 58(9): 2394–2400.

42. Toh K, Tutuncu R, Todd M. On the implementation of SDPT3 (version 3.1)-a MATLAB software package for semidefinite-quadratic-linear programming. In: ; 2004: 290–296.

43. Fukuda K. *cdd/cdd+ Reference Manual*. Institute for Operations Research, ETH-Zentrum; Zurich, Switzerland: 1997.

44. Raghunathan AU, Di Cairano S. Optimal Step-Size Selection in Alternating Direction Method of Multipliers for Convex Quadratic Programs and Model Predictive Control. In: Int. Symp. Mathematical Theory of Networks and Systems. ; 2014: 807–814.

45. Burns DJ, Danielson C, Zhou J, Di Cairano S. Reconfigurable model predictive control for multievaporator vapor compression systems. *IEEE Trans. Contr. Systems Technology* 2017; 26(3): 984–1000.

46. Di Cairano S, Brand M, Bortoff SA. Projection-free parallel quadratic programming for linear model predictive control. *Int. J. Control* 2013; 86(8): 1367–1385.