

## REGULAR ISSUE ARTICLE

# A Robust Nonlinear Tracking MPC using qLPV Embedding and Zonotopic Uncertainty Propagation

Marcelo M. Morato<sup>\*1,3</sup> | Victor M. Cunha<sup>2</sup> | Tito L. M. Santos<sup>2</sup> | Julio E. Normey-Rico<sup>1</sup> | Olivier Sename<sup>3</sup>

<sup>1</sup>Departamento de Automação e Sistemas,  
Universidade Federal de Santa Catarina,  
Florianópolis, Brazil

<sup>2</sup>Dept. de Eng. Elétrica e Computação,  
Universidade Federal da Bahia, BA,  
Brazil

<sup>3</sup>Univ. Grenoble-Alpes, CNRS, Grenoble  
INP<sup>T</sup>, GIPSA-Lab, 38000 Grenoble,  
France.

T Institute of Engineering, Univ.  
Grenoble-Alpes.

## Correspondence

\*Marcelo Menezes Morato,  
DAS/CTC/UFSC, Trindade, Caixa Postal  
476, Florianópolis-SC, 88040-900, Brazil  
Email: marcelomnzm@gmail.com

## Summary

In this paper, we propose a novel Nonlinear Model Predictive Control (NMPC) framework for tracking for piece-wise constant reference signals. The main novelty is the use quasi-Linear Parameter Varying (qLPV) embeddings in order to describe the nonlinear dynamics. Furthermore, these embeddings are exploited by an extrapolation mechanism, which provides the future behaviour of the scheduling parameters with bounded estimation error. Therefore, the resulting NMPC becomes computationally efficient (comparable to a Quadratic Programming algorithm), since, at each sampling period, the predictions are linear. Benefiting from artificial target variables, the method is also able to avoid feasibility losses due to large set-point variations. Robust constraint satisfaction, closed-loop stability, and recursive feasibility certificates are provided, thanks to uncertainty propagation zonotopes and parameter-dependent terminal ingredients. A benchmark example is used to illustrate the effectiveness of the method, which is compared to state-of-the-art techniques.

## KEYWORDS:

Model Predictive Control, Linear Parameter Varying Systems, Nonlinear Systems, Tracking, Zonotopes.

## 1 | INTRODUCTION

Model Predictive Control (MPC) is indisputably well established. Yet, the application of robust Nonlinear MPC (NMPC) is not trivial and is enabled at the expense of increased numerical burden, which is an impediment for real-time applications. The majority of stabilising NMPC schemes ensures regulation of the closed-loop dynamics to a fixed target<sup>1</sup>. Then, asymptotic stability and constraints satisfaction are usually guaranteed by the means of terminal ingredients (which verify invariance conditions in the neighborhood of a terminal target). Nevertheless, this standard design method is not valid for set-point changes, since feasibility may be lost.

Accordingly, there has been an increasing focus on NMPC schemes “for Tracking” time-varying piece-wise constant references (set-points). Specifically, we highlight two of the main frameworks towards this matter, as debated in recent years: (i) artificial reference variables, as proposed in<sup>2</sup>, which allow for less conservative terminal constraints and thus ensure feasibility is not lost for sudden set-point changes; and (ii) terminal equality constraints and optimised terminal sets, for the case of periodic reference signals, as developed in<sup>3</sup>.

In this paper, we deal with the issue of tracking possibly unreachable output target signals using state-feedback NMPC, closely building upon these previous papers. We focus on addressing the following disadvantages of the prior: (i) the use of online

<sup>0</sup>Abbreviations: NMPC, Nonlinear Model Predictive Control; qLPV, quasi-Linear Parameter Varying; QP, Quadratic Program; NLP, Nonlinear Program

Nonlinear Programming Algorithms (NLPs), which are numerically expensive and not viable for time-critical applications<sup>1</sup>; and (ii) no model uncertainties nor disturbances are considered, which should be included for any realistic application.

In parallel to the theoretical establishment of NMPC, the Linear Parameter Varying (LPV) toolkit has been brought to focus<sup>4</sup>. LPV models can be used to represent nonlinear and time-varying dynamics, which are encapsulated within known, bounded scheduling parameters  $\rho$ <sup>5</sup>. Accordingly, recent advances on NMPC have been presented by exploiting the use of quasi-LPV (qLPV) embeddings, e.g.<sup>6</sup> and references therein. The elegance of the qLPV approach is that the nonlinear predictions are replaced by linear laws, as shows<sup>7</sup>. In this paper, we benefit from such qLPV embeddings for the purpose of Tracking NMPC.

## 1.1 | Contributions

With regard to the detailed context<sup>2</sup>, we provide a novel Tracking NMPC algorithm based on nominal predictions generated via qLPV embedding. Following the final suggestions from<sup>3</sup>, the proposed solution also incorporates robustness features against model uncertainties and additive load disturbances (with known bounds), using a constraint tightening framework<sup>11,12,13,14</sup>. Based on the one-step-ahead disturbance propagation, we enforce the satisfaction of the performance requirements by binding the prediction error within zonotope extensions. Additionally, we benefit from the recent recursive extrapolation method from<sup>15</sup> in order to estimate the future trajectory of the qLPV scheduling parameters. The extrapolation helps by providing accurate (linear) predictions in the context of MPC, at each sampling instant, with bounded prediction errors.

The main novelties of this work are summarised next:

1. The proposed Tracking NMPC is based on qLPV embeddings, which enable linear model predictions. The model is exploited by an extrapolation mechanism that provides the complete sequence of future scheduling parameters, at each sampling instant. Accordingly, we compute simple bounds on the prediction error from these qLPV realisations. Furthermore, we propose zonotopes that bound the corresponding uncertainty propagation<sup>3</sup>, which are then used for robust constraints satisfaction.
2. We offer robust parameter-dependent terminal ingredients for the proposed NMPC. These tools ensure recursive feasibility of the optimisation procedure and stability of the tracking error dynamics, considering any set-point value within a predefined set. Furthermore, we propose an additional optimisation for the choice of the artificial reference variable, with relieved complexity.
3. Finally, using a benchmark example, we demonstrate that the numerical complexity of the proposed online algorithm is, on average, that of a semi-definite programming algorithm, comparable to a Quadratic Program (QP), being much faster than the NLPs from<sup>2,3</sup>.

*Remark 1.* With respect to the Authors' previous works, this paper enhances and generalises the robust regulation algorithm from<sup>16</sup> for the case of time-varying reference signals. In the prior, the framework was initially sketched for regulation only. Furthermore, the zonotope uncertainty propagation sets did not take into account the estimates for the future scheduling variables, resulting in more conservatism. **Moreover, the robust terminal ingredients provided in this paper are enhanced versions of those presented in<sup>16,9</sup>, now developed in the context of *tracking* robust positive invariant sets.**

**Organisation.** Sec. 2 provides preliminary discussions and the overall problem setup. Sec. 3 briefly details the recursive extrapolation algorithm from<sup>15</sup>; also, resulting prediction error bounds are provided. In Sec. 4, we propose novel zonotopes that bound the corresponding disturbance propagation. Sec. 5 presents the main contribution of this work: the robust NMPC algorithm for Tracking. Recursive feasibility and closed-loop stability are demonstrated based on novel robust parameter-dependent terminal ingredients (second contribution of this paper). Discussions regarding synthesis requirements and the choice of the artificial reference are also provided. Finally, a numerical benchmark is considered in Sec. 6, which illustrates the application of the proposed method, together with comparisons to state-of-the-art NMPC algorithms (third contribution). General conclusions are drawn in Sec. 7.

**Notation.** The index set  $\mathbb{N}_{[a,b]}$  represents  $\{i \in \mathbb{N} \mid a \leq i \leq b\}$ , with  $0 \leq a \leq b$ . The identity matrix of size  $j$  is denoted as  $I_j$ ;  $I_{j,\{i\}}$  denotes the  $i$ -th row of  $I_j$ ;  $\text{col}\{\cdot\}$  denotes the vectorisation of the entries and  $\text{diag}\{v\}$  denotes the diagonal matrix

<sup>1</sup>We stress that the application results provided in<sup>3</sup> solve such NLPs only very fast, but relying on a solver-based solution (CaSaDi), which internally approximates the solution of the problem.

<sup>2</sup>Up to our best knowledge, NMPC algorithms based on qLPV embeddings have only been formalised for regulation purposes, refer to the survey<sup>6</sup> and also to<sup>8,9,10,7</sup>.

<sup>3</sup>The proposed method also considers bounded process disturbances. The uncertainty propagation zonotopes offer a direct extension for such case.

generated with the line vector  $v$ .  $\mathbf{1}_{n \times m}$  stands for the  $n \times m$  vector of unit entries. The predicted value of a given variable  $v(k)$  at time instant  $k+i$ , computed based on the information available at instant  $k$ , is denoted as  $v(k+i|k)$ . In particular,  $v(k|k) = v(k)$ .  $\mathcal{K}$  refers to the class of continuous, positive and strictly increasing scalar functions that pass through the origin. A  $C^1$  function  $f: \mathbb{R}^m \rightarrow \mathbb{R}^n$  is such that it is differentiable with continuous derivatives. In this case,  $\nabla^T f: \mathbb{R}^m \rightarrow \mathbb{R}^{n \times m}$  denotes its Jacobian matrix. Consider sets  $\mathcal{A}, \mathcal{B} \subset \mathbb{R}^n$ ,  $\mathcal{C} \subset \mathbb{R}^m$  and a matrix  $R \in \mathbb{R}^{n \times m}$ . The Minkowski set addition is defined by  $A \oplus B := \{a + b | a \in A, b \in B\}$ , while the Pontryagin set difference is defined by  $A \ominus B := \{a | a \oplus B \subseteq A\}$ . A linear mapping is  $\mathcal{R}\mathcal{A} = \{y \in \mathbb{R}^n : y = Ra, a \in \mathcal{A}\}$ , while the Cartesian product holds as  $\mathcal{A} \times \mathcal{C} = \{z \in \mathbb{R}^{n+m} : z = (a^T c^T)^T, a \in \mathcal{A}, c \in \mathcal{C}\}$ . The unitary  $m$ -dimensional box is denoted  $\mathcal{B}_\infty^m = \{\xi \in \mathbb{R}^m : \|\xi\| \leq 1\}$ . The set of real compact intervals is given by  $\mathbb{I} = \{[a, b], a, b \in \mathbb{R}, a \leq b\}$ . An interval matrix is represented by  $\mathbf{J} \in \mathbb{I}^{n \times m}$ , with  $\text{mid}(\mathbf{J})$  and  $\text{rad}(\mathbf{J})$  denoting its middle point and radius, respectively.  $\|\cdot\|$  denotes the 2-norm. In matrix inequalities,  $(\star)$  denotes the corresponding symmetrical transpose.

## 2 | PRELIMINARIES

In this Section, we present the overall problem setup, detailing the nonlinear system model, its corresponding qLPV embedding, the considered control objective (output tracking of piece-wise constant reference), the used control formulation and the resulting disturbance that propagates along a fixed prediction horizon.

### 2.1 | System model and qLPV embedding

The focus of this paper is the application of a model-based predictive controller for tracking. As previously stated, we develop an algorithm for the class of nonlinear systems that can be represented by qLPV-embeddings. For a start, consider the following discrete-time nonlinear system:

$$\begin{cases} x(k+1) = f(x(k), u(k)) + w(k), \\ y(k) = h(x(k), u(k)), \end{cases} \quad (1)$$

where  $x \in \mathbb{R}^{n_x}$  represent the states,  $u \in \mathbb{R}^{n_u}$  the inputs, and  $y \in \mathbb{R}^{n_y}$  the outputs. The additive disturbance  $w \in \mathbb{R}^{n_x}$  is bounded to a compact set with the origin at its interior, in such a way that  $w(k) \in \mathcal{W} \subseteq \mathbb{R}^{n_x} \Rightarrow \|w(k)\| \leq \bar{w}$ . Throughout the sequel, we assume that the system in Eq. (1) satisfies Assumption 1; thus, we formulate a state-feedback NMPC algorithm.

**Assumption 1.** The states are measurable for all sampling instants  $k \geq 0$ .

**Definition 1.** The system in Eq. (1) is *admissibly operated* if the following hard constraints are continuously satisfied:  $(x(k), u(k)) \in \mathcal{Z} = \mathcal{X} \times \mathcal{U}$ , where:

$$\mathcal{X} := \{x \in \mathbb{R}^{n_x} : H_x x \leq h_x\}, \text{ and } \mathcal{U} := \{u \in \mathbb{R}^{n_u} : H_u u \leq h_u\}. \quad (2)$$

An admissible operation can be analogously represented<sup>4</sup> by:  $|x_j| \leq \bar{x}_j, \forall j \in \mathbb{N}_{[1, n_x]}$  and  $|u_j| \leq \bar{u}_j, \forall j \in \mathbb{N}_{[1, n_u]}$ . The set  $\mathcal{Y} := \{y \in \mathbb{R}^{n_y} | H_y y \leq h_y\} = h(\mathcal{X}, \mathcal{U})$  is compact and convex, defining the possible *admissible* outputs  $y$  (mapped by admissible state and input variables).

Next, the nonlinear model from Eq. (1) is re-written under the following **exact** qLPV representation:

$$\begin{cases} x(k+1) = A(\rho(k))x(k) + B(\rho(k))u(k) + w(k), \\ y(k) = C(\rho(k))x(k) + D(\rho(k))u(k), \\ \rho(k) = f_\rho(x(k)) \in \mathcal{P}. \end{cases} \quad (3)$$

<sup>4</sup>The outer box-type bounds  $\bar{x}_j$  and  $\bar{u}_j$  can be found from Eqs. (1)-(2) using linear programming.

This qLPV model is scheduled by the state-dependent time-varying parameter  $\rho(k) = f_\rho(x(k)) \in \mathcal{P} \subseteq \mathbb{R}^{n_\rho}$ , which is assumed to be bounded for all  $k \geq 0$  (Assumption 2). We stress that, as defined, since  $x(k)$  is measured and  $f_\rho(\cdot)$  is a **known** function, this parameter is known. Yet, we stress that the future scheduling trajectory  $\rho(k+j)$ ,  $\forall j \in \mathbb{N}_{[1,\infty]}$  is unknown.

**Assumption 2.** The following statements are valid:

1. The nonlinear scheduling map  $f_\rho : \mathcal{X} \rightarrow \mathcal{P}$  is **known**, algebraic and class  $C^1$  for all  $x \in \mathcal{X}$ . Moreover, it agrees to a local Lipschitz condition around any arbitrary point  $x \in \mathcal{X}$ , this is:

$$\|f_\rho(x) - f_\rho(\hat{x})\| \leq \gamma_\rho \|x - \hat{x}\|, \forall x, \hat{x} \in \mathcal{X}, \quad (4)$$

where the smallest constant  $\gamma_\rho$  that satisfies Eq. (4) is called its Lipschitz constant.

2. Under an admissible operation of the system (Definition 1), the scheduling parameters are assumed to be bounded to the following compact and convex scheduling set  $\mathcal{P} := \{\rho \in \mathbb{R}^{n_\rho} : \underline{\rho}_j \leq \rho_j \leq \bar{\rho}_j, j \in \mathbb{N}_{[1,n_\rho]}\}$ .
3. The scheduling variables exhibit a bounded rate of variation over samples, this is:  $\delta\rho(k+1) = (\rho(k+1) - \rho(k)) \in \delta\mathcal{P} \forall k \geq 0$ , where:

$$\delta\mathcal{P} := \left\{ \delta\rho_j \in \mathbb{R} : \underline{\delta\rho}_j \leq \delta\rho_j \leq \bar{\delta\rho}_j, \forall j \in \mathbb{N}_{[1,n_\rho]} \right\}. \quad (5)$$

*Remark 2.* We stress that nonlinear systems can be embedded to qLPV realisations, as long as a (exact, linear, convex, convex-concave) differential inclusion property is satisfied. This topic is extensively detailed in<sup>17,6,18,19</sup>, with experimental validations provided in<sup>20</sup>. Note that the boundedness of the scheduling variables relates to the existence of a bounded system state evolution. Assuming an admissible operation (Definition 1), it follow directly that  $\rho(k) = f_\rho(x(k)) \in \mathcal{P}$ ,  $\forall k \geq 0$ , since  $x(k) \in \mathcal{X}$ ,  $\forall k \geq 0$ .

*Remark 3.* The requirement of Lipschitz continuity for  $f_\rho(x)$  is not restrictive, since the selection of this function is a design choice. The qLPV realisation from Eq. (3) only requires that a corresponding differential inclusion exists and that  $f_\rho(x)$  is bounded for all states  $x \in \mathcal{X}$ . Note, anyhow, that the bounds on scheduling parameters' variations is **not** a design choice. These bounds, which describe the scheduling variation set  $\delta\mathcal{P}$ , naturally appear due to the discrete-time characteristic of the system. Note that  $\delta\rho(k+1) = f_\rho(x(k+1)) - f_\rho(x(k))$  and, since  $x(k+1), x(k) \in \mathcal{X}, \forall k \geq 0$ ,  $\delta\rho(k+1) \in \delta\mathcal{P}$ . Considering the nonlinear system to be ultimately bounded, we can obtain the scheduling variation bounds by minimising<sup>5</sup>  $\underline{\delta\rho}$  and  $-\bar{\delta\rho}$  such that  $\underline{\delta\rho} \leq f_\rho(A(f_\rho(x))x + B(f_\rho(x))u + w) - f_\rho(x) \leq \bar{\delta\rho}$  with  $(x, u) \in \mathcal{Z}$ ,  $(A(f_\rho(x))x + B(f_\rho(x))u + w, u) \in \mathcal{Z}$  and  $w \in \mathcal{W}$ .

## 2.2 | Tracking Objective

Taking into account the nonlinear system model shown in the prior, we discuss, next, the objective of the proposed MPC: tracking piece-wise constant output reference signals (set-points). For such, since state measurements are available, we first determine the state-input pairs that enable possible output targets. That is, we seek the admissible steady-state pairs  $(x_r, u_r)$  which imply in the stabilisation of the system in a given output target coordinate  $y_r$ . In practice, the proposed MPC is designed steer the system states  $x$  to the steady-state regime  $x_r$  such that the outputs  $y$  reach the tracking objective  $y_r$ .

Following the lines of previous works on Tracking MPC<sup>2,21</sup>, we consider that there exists a unique combination of the states and inputs which ensures that  $\lim_{k \rightarrow +\infty} y(k) = y_r$  (stabilisation at the set-point target, considering that  $\lim_{k \rightarrow +\infty} w(k) \rightarrow 0$ ). For such, the following additional hypothesis is required:

<sup>5</sup>The solution to this minimisation problem can be found either by interval arithmetic or optimisation.

**Assumption 3.** For all admissible steady-state targets  $y_r \in \mathcal{Y}$ , there exists a unique admissible steady-state state-input pair  $z_r = (x_r^T, u_r^T)^T \in \mathcal{Z}$  such that the following inequality holds:

$$\begin{bmatrix} x_r - (A(f_\rho(x_r))x_r - B(f_\rho(x_r))u_r) \\ C(f_\rho(x_r))x_r + D(f_\rho(x_r))u_r \end{bmatrix} = \begin{bmatrix} 0_{n_x} \\ y_r \end{bmatrix}. \quad (6)$$

From Assumption 3, we denote  $\mathcal{Y}_T \subset \mathcal{Y}$  as the set of admissible steady-state tracking outputs, i.e. those generated by an admissible state-input pair  $z_r \in \mathcal{Z}$  which satisfies Eq. (6). Specifically, we define this set as follows:  $\mathcal{Y}_T := \{y \in \mathcal{Y} \mid y = C(f_\rho(x_r))x_r + D(f_\rho(x_r))u_r \mid \left( (A(f_\rho(x_r))x_r + B(f_\rho(x_r))u_r)^T, u_r^T \right)^T + e \in \mathcal{Z}, \forall e \in \epsilon \mathcal{B}_\infty\}$ . Note that, in this definition,  $\epsilon$  is an arbitrarily small constant included so that the frontier of  $\mathcal{Z}$  is excluded.

Next, we exploit the unique steady-state state-input pair to tracking output set-point correspondence in terms of locally Lipschitz continuous functions. That is: Assumption 3 implies that there exists a **unique** state-input pair  $z_r$  that matches each admissible target  $y_r \in \mathcal{Y}$ . Thus, as demonstrated in<sup>2</sup>, Assumption 1 and Remark 1 using the implicit function theorem, we assume that there exists locally Lipschitz continuous maps  $g_x : \mathcal{Y} \rightarrow \mathcal{X}$  and  $g_u : \mathcal{Y} \rightarrow \mathcal{U}$  such that  $x_r = g_x(y_r)$  and  $u_r = g_u(y_r)$  for all  $y_r \in \mathcal{Y}_T$ . As debated in<sup>2</sup>, the Jacobian matrix of the left hand-side in Eq. (6) must be square and non-singular for all  $z_r \in \mathcal{Z}$ . Complementary, we also consider a composed locally Lipschitz<sup>6</sup> continuous function  $g_\rho : \mathcal{Y} \rightarrow \mathcal{P}$  that maps the equilibrium scheduling parameter, i.e.  $\rho_r = g_\rho(y_r) = f_\rho(g_x(y_r), g_u(y_r))$ .

### 2.3 | Control Scheme and Disturbance Propagation

Taking into account the qLPV model representation and the output tracking objective, detailed in the prequel, we proceed by further detailed the proposed control scheme. Since state measurements are available, and the developed controller has the objective of steering the states to steady-state conditions which imply an output tracking goal, we use the following law:

$$u(k) = v(k) + K_\pi x(k), \quad (7)$$

where  $v(k)$  is a virtual input, determined by the MPC, and  $K_\pi$  a feedback gain, used to locally stabilise the process and attenuate the propagation of disturbances. Accordingly, from Eqs. (3)-(7), we obtain the following closed-loop dynamics:

$$x(k+1) = \overbrace{(A(\rho(k)) + B(\rho(k))K_\pi)}^{:= A_\pi(\rho(k))} x(k) + B(\rho(k))v(k) + w(k). \quad (8)$$

Then, from Eqs. (2)-(8), we re-write the admissibility constraints  $(x(k), u(k)) \in \mathcal{Z}$  in terms of the virtual input as follows:  $(x(k), v(k)) \in \mathcal{Z}_\pi$ , where:

$$\mathcal{Z}_\pi := \left\{ z \in \mathbb{R}^{n_x+n_u} : \begin{pmatrix} H_x & 0 \\ H_u K_\pi & H_u \end{pmatrix} z \leq \begin{pmatrix} h_x \\ h_u \end{pmatrix} \right\}. \quad (9)$$

For the application of the Tracking MPC policy, we distinguish *nominal predictions* of the system dynamics from the *real system trajectories*. Note that the MPC has no knowledge of the disturbance variables  $w(k+j)$ ,  $\forall j \geq 0$ , and also does not know the future scheduling trajectory  $\rho(k+j) \forall j \geq 1$ , which thus makes the nominal predictions differ from the real system trajectories. In terms of the scheduling trajectory, we assume that there exists an estimate for its behaviour:

**Assumption 4.** Consider a prediction horizon of  $N_p$  samples. Then, at each sample  $k$ , there exists a scheduling trajectory guess  $\hat{P}_k = [\hat{\rho}^T(k|k) \dots \hat{\rho}^T(k+N_p-1|k)]^T$  with bounded mismatch towards the real scheduling trajectory  $P_k = [\rho^T(k) \dots \rho^T(k+N_p-1)]^T$ . That is:  $\|\xi_\rho(k+j|k)\| = \|(\rho(k+j) - \hat{\rho}(k+j|k))\| \leq \bar{\xi}_\rho < +\infty$ .

<sup>6</sup>Note that  $g_\rho(\cdot)$  is locally Lipschitz continuous since it is a composition of locally Lipschitz continuous functions, due to Assumption 2.

Then, by expanding Eq. (8) forward, along the following  $N_p$  steps, the **real system trajectories** are given, from an initial condition  $x(k) \in \mathbb{R}^{n_x}$ , by:

$$X_k = \begin{bmatrix} x(k+1) \\ x(k+2) \\ \vdots \\ x(k+N_p) \end{bmatrix} := \phi_{N_p}(x(k), V_k, W_k, P_k), \quad (10)$$

where  $V_k := [v^T(k) \dots v^T(k+N_p-1)]^T$  is the vector of future inputs and  $W_k := [w^T(k) \dots w^T(k+N_p-1)]^T$  the vector of future disturbances.

From Assumption 4, the **nominal predictions** are then given, from an initial condition  $x(k) \in \mathbb{R}^{n_x}$ , by:

$$\hat{X}_k = \begin{bmatrix} x(k+1|k) \\ x(k+2|k) \\ \vdots \\ x(k+N_p|k) \end{bmatrix} := \phi_{N_p}(x(k), V_k, \mathbf{0}, \hat{P}_k). \quad (11)$$

Note that, in these predictions, the future disturbances are presumably nil, i.e. Eq. (10) is equivalent to using:  $x(k+1|k) = A_\pi(\hat{\rho}(k|k))x(k) + B(\hat{\rho}(k|k))v(k)$ ,  $x(k+2|k) = A_\pi(\hat{\rho}(k+1|k))x(k+1|k) + B(\hat{\rho}(k+1|k))v(k+1)$ , and so forth. The scheduling trajectory estimates  $\hat{P}_k$  which verify Assumption 4 are provided by an extrapolation algorithm, as detailed in Sec. 3.

### 3 | THE RECURSIVE EXTRAPOLATION MECHANISM

The scheduling parameter trajectories  $P_k$ , at each discrete-time sample  $k$ , are estimated by an extrapolation method based on first-order Taylor expansions of the scheduling proxy  $f_\rho(\cdot)$ . This scheme was originally presented in<sup>15</sup> and herein briefly recalled. Application results of this approach have also been presented in<sup>22,23</sup>.

Before any detail is given, we stress that this extrapolation method has a significant advantage over competing techniques: it is able to operate very fast. Unlike the Hessian-based estimation approach from<sup>7</sup> and the iteratively refined estimated from<sup>24</sup>, it does not resort to optimisation-related tools, i.e. the detailed method consists only of **linear operators**. Furthermore, we are also able to compute concrete (reduced) bounds on the extrapolation error, which is not possible using<sup>24</sup>. Thereof, the corresponding prediction uncertainties that arise when applying MPC became less significant, especially in comparison to *frozen*-based scheduling trajectory estimates, e.g.<sup>25,26</sup>, thus enabling less conservative control. Further comparisons between the algorithms from<sup>15</sup> (Taylor scheme) and<sup>7</sup> (Hessian-based approach) are still lacking, and topic of future studies.

First, denote  $\delta x(k+1) = x(k+1) - x(k)$  as the incremental state deviations over samples,  $\forall k \geq 0$ . These variables are naturally bounded<sup>7</sup> to a compact and convex box-type set  $\delta\mathcal{X} := \left\{ \delta x \in \mathbb{R}^{n_x} \mid \|\delta x_j\| \leq \overline{\delta x_j}, \forall j \in \mathbb{N}_{[1, n_x]} \right\}$ . In vector notation, we use  $\delta X_k = [\delta x^T(k) \dots \delta x^T(k+N_p-1)]^T$  (real state deviations over the prediction horizon) and  $\delta \hat{X}_k = [\delta x^T(k|k) \dots \delta x^T(k+N_p-1|k)]^T$  (nominal state deviations over the prediction horizon).

Then, the extrapolation mechanism from<sup>15</sup> consists in the following steps: consider a first-order Taylor description of the static scheduling map  $f_\rho(x)$  holds, i.e.:

$$f_\rho(x) = f_\rho(x)|_{\check{x}} + \left. \frac{\partial f_\rho(x)}{\partial x} \right|_{\check{x}} (x - \check{x}) + \xi_\rho, \quad (12)$$

being  $\check{x}$  the expansion point and  $\xi_\rho$  a residual that inherits the discrepancy between the real function and its approximate. Recall that the map  $f_\rho(x)$  is of class  $C^1$  in  $\mathcal{X}$  (Assumption 2) and  $\left. \frac{\partial f_\rho(x)}{\partial x} \right|_{\check{x}}$  is ultimately bounded for all  $\check{x} \in \mathcal{X}$  (due to the boundedness of  $f_\rho(x)$  and  $x$ ). Thus, from Eq. (12), we obtain,  $\forall j \geq 1$ :

$$f_\rho(x(k+j|k)) = f_\rho(x(k+j-1|k)) + \underbrace{\left. \frac{\partial f_\rho}{\partial x(k+j)} \right|_{x(k+j-1|k)}}_{f_\rho^\partial(k+j-1|k)} \delta x(k+j-1|k) + \xi_\rho(k+j-1|k). \quad (13)$$

<sup>7</sup>Bounds can be found from Eqs. (3)-(2) via interval arithmetic.

Next, by expanding Eq. (13) along a horizon of  $N_p$  steps, we obtain:

$$\begin{aligned}\rho(k+1|k) &= \rho(k|k) + f_\rho^\partial(k|k)\delta x(k|k) + \xi_\rho(k|k), \\ \rho(k+2|k) &= \rho(k+1|k) + f_\rho^\partial(k+1|k)\delta x(k+1|k) + \xi_\rho(k+1|k), \\ &\vdots \\ \rho(k+N_p-1|k) &= \rho(k+N_p-2|k) + \xi_\rho(k+N_p-2|k) + f_\rho^\partial(k+N_p-2|k)\delta x(k+N_p-2|k).\end{aligned}$$

Notice that  $\rho(k|k) = \rho(k)$  and  $\delta x(k|k) = \delta x(k)$  are known variables at each instant  $k$ , whereas the partial derivative term  $f_\rho^\partial(k|k) = \left. \frac{\partial f_\rho(x)}{\partial x} \right|_{x(k)}$  can be numerically evaluated. In practice,  $f_\rho^\partial(k+j|k)$  is unknown for all  $j \in \mathbb{N}_{[1, N-2]}$ . Then, in order to construct the estimation mechanism on the basis of the previous Taylor expansions, we consider that  $f_\rho^\partial(k+j|k) \approx f_\rho^\partial(k)$ ,  $\forall j \in \mathbb{N}_{[1, N-2]}$ . By doing so, and by disregarding the residual terms, we obtain the following recursive law:  $\hat{\rho}(k+j|k) = \hat{\rho}(k+j-1|k-1) + f_\rho^\partial(k)\delta x(k+j-1|k)$ . That is: the estimate for the current scheduling trajectory  $\hat{P}_k$  can be written as the sum of the estimate from the previous sample  $\hat{P}_{k-1}$  with an adjustment term  $f_\rho^\partial(k)\delta \hat{X}_k$ . Accordingly, we are able to write the extrapolation in a recursive vector fashion:

$$\hat{P}_k = \hat{P}_{k-1}^\dagger + f_\rho^\partial(k)\delta \hat{X}_k^\dagger, \quad (14)$$

where the  $(\cdot)^\dagger$  operator indicates a correction of the vectors with the available data (i.e. known terms): note that at, instant  $k$ , the scheduling variable  $\rho(k)$  and the state deviation  $\delta x(k)$  are known. Thus,  $\hat{P}_{k-1}^\dagger$  stands for  $\hat{P}_{k-1}$  with its first entry  $\hat{\rho}(k|k-1)$  replaced by the available data  $\rho(k)$ , this is:  $\hat{P}_{k-1} = [\rho^T(k-1) \ \rho^T(k) \ \rho^T(k+1|k-1) \ \dots \ \rho^T(k+N_p-2|k-1)]^T$ . Similarly,  $\delta \hat{X}_k^\dagger$  stands for  $\delta \hat{X}_k$  with the term  $\delta x(k|k-1)$  replaced by  $\delta x(k)$ , this is:  $\delta \hat{X}_k^\dagger = [\delta^T x(k) \ \delta x^T(k+1|k) \ \dots \ \delta x^T(k+N_p-1|k)]^T$ . We stress that  $\delta \hat{X}_k$  can be computed thanks to the nominal prediction model from Eq. (11), using  $\hat{P}_{k-1}^\dagger$  and  $V_k$  as the solution of the previous MPC iteration.

In general, the selection of the scheduling trajectory estimate  $\hat{P}_k$  is done as follows:

- At the first sample,  $k = 0$ , no previous estimate is available, and thus Eq. (14) cannot be evaluated. Therefore, the algorithm initialises with  $\hat{P}_0 = [\rho^T(0) \ \rho^T(0) \ \dots \ \rho^T(0)]^T$ , i.e. a vector of repeated  $\rho(0)$  entries. Note that  $\rho(0)$  is known by definition.
- At the following samples:
  1. The previous estimate  $\hat{P}_{k-1}$  is corrected with the new available data  $\rho(k)$ , thus generating  $\hat{P}_{k-1}^\dagger$ ;
  2. The previous state deviation prediction  $\delta \hat{X}_{k-1}$  is also corrected with the new available state information  $\delta x(k)$ , thus generating  $\delta \hat{X}_{k-1}^\dagger$ ;
  3. The current derivative  $f_\rho^\partial(k)$  is evaluated at the current state value, i.e. as  $\left. \frac{\partial f_\rho(x)}{\partial x} \right|_{x(k)}$ ;
  4. Finally, the admissible solution for the scheduling trajectory prediction is generated via Eq. (14).

### 3.1 | Convergence and Error Boundedness

Next, we develop two separate Lemmas, which provide the convergence and error boundedness certificates for the generated scheduling trajectory guesses, respectively. The convergence property (Lemma 1) is essential, since it enables the scheduling extrapolation method from Eq. (14) to be used in the context of MPC. Also, the boundedness of the estimation error (Lemma 2) is also of utter interest, since it is later used to compute the uncertainty propagation when using the nominal predictions from Eq. (11).

**Lemma 1.** Assume that  $f_\rho(\cdot)$  is class  $C^1$  and that  $f_\rho^\partial(k)$  is ultimately bounded. Assume that the system (3) is stable under the feedback defined by Eq. (7). Then, the recursive extrapolation algorithm in Eq. (14) converges<sup>8</sup> if the closed-loop is stable.

<sup>8</sup>The convergence of Eq. (14) implies that the residual errors  $\xi_\rho(\cdot|k)$  turn null as  $k$  increases, even if consecutive biases within samples are increasing, i.e.  $\lim_{k \rightarrow +\infty} \xi_\rho(\cdot|k) \rightarrow 0$  even if  $\xi_\rho(k_1+j|k_1) \geq \xi_\rho(k_1+j+1|k_1)$  for some  $k_1 \geq 0$ .



*Proof.* The convergence property can be demonstrated with the aid of the residual term  $\xi_\rho(k+j|k)$ , which should turn null. This proof is reduced for brevity, full details are given in<sup>15</sup>. Consider  $\lim_{k \rightarrow +\infty} x(k) = x_a$  holds (stability) and take  $\xi_\rho(k+j|k) = f_\rho(x(k+j|k)) - f_\rho(x(k+j-1|k)) - f_\rho^\partial(k)\delta x(k+j|k)$ . Due to the stabilisation<sup>9</sup>, it directly follows that  $\lim_{k \rightarrow +\infty} f_\rho(x(k+j|k)) = \lim_{k \rightarrow +\infty} f_\rho(x_a)$  and  $\lim_{k \rightarrow +\infty} \delta x(k|k) \rightarrow 0$ , which implies that  $\lim_{k \rightarrow +\infty} \xi_\rho(\cdot|k) = -\lim_{k \rightarrow +\infty} f_\rho^\partial \delta x(\cdot|k) \rightarrow 0$ . This concludes the proof.  $\square$

*Remark 4.* Note that the convergence of Eq. (14) implies that the residual errors  $\xi_\rho(\cdot|k)$  turn null as  $k$  increases, even if consecutive biases within samples are increasing, i.e.  $\lim_{k \rightarrow +\infty} \xi_\rho(\cdot|k)$  even if  $\xi_\rho(k_1+j|k_1) \geq \xi_\rho(k_1+j+1|k_1)$  for some  $k_1 \geq 0$ .

**Lemma 2.** The sampled estimation error (residual)  $\xi_\rho(k+j|k)$  is ultimately bounded,  $\forall j \in \mathbb{N}_{[1, N_p-1]}$ ,  $k \geq 0$  to a convex set  $\mathcal{Q} := \{\xi_\rho \in \mathbb{R}^{n_\rho} \mid \|\xi_\rho\|_\infty \leq (\gamma_\rho + \overline{f_\rho^\partial}) \overline{\delta x}\}$ .

*Proof.* The residual term from Eq. (13) is:

$$\xi_\rho(k+j+1|k) = f_\rho(x(k+j+1|k)) - f_\rho(x(k+j|k)) - f_\rho^\partial(k)\delta x(k+j|k). \quad (15)$$

Using a triangular inequality, we obtain:

$$\|\xi_\rho(k+j+1|k)\| \leq \|f_\rho(x(k+j+1|k)) - f_\rho(x(k+j|k))\| + \|f_\rho^\partial(k)\delta x(k+j|k)\|. \quad (16)$$

Assumption 2 implies that:

$$\|\xi_\rho(k+j+1|k)\| \leq \gamma_\rho \|\delta x(k+j|k)\| + \|f_\rho^\partial(k)\delta x(k+j|k)\|.$$

Since  $f_\rho^\partial(k)$  is ultimately bounded, i.e.  $\|f_\rho^\partial(k)\| \leq \overline{f_\rho^\partial}$ , it follows that:  $\|\xi_\rho\| \leq \underbrace{(\gamma_\rho + \overline{f_\rho^\partial})}_{\overline{\xi_\rho}} \overline{\delta x}$ . Note that as long as  $\|\xi_\rho(k+j|k)\| < \overline{\delta \rho}$ , it follows that  $\mathcal{Q} \subset \delta \mathcal{P}$ . This concludes the proof.  $\square$

As a final comment, we highlight that Lemma 2 stands for a major advantage of the extrapolation method. In general, many works use the so-called *frozen* scheduling estimates, as seen in<sup>25,26</sup>, i.e. taking  $\hat{P}_k = [\rho^T(k) \dots \rho^T(k)]^T$  and, thus,  $\hat{\rho}(k+j|k) = \rho(k)$ ,  $\forall j \in \mathbb{N}_{[1, N_p-1]}$ . Under such conditions, the resulting scheduling estimation error is bounded as follows:  $\|\xi_\rho(k+j|k)\| \leq (N_p-1)\overline{\delta \rho}$ . Therefore, more conservative control laws would have been generated, since the corresponding uncertainty propagation is far larger. We stress that the error obtained with the proposed mechanism does not depend on the horizon size, but only on characteristics of the scheduling proxy and the bounds on the state deviations.

## 4 | DISTURBANCE PROPAGATION USING ZONOTOPES

Next, we detail how to conceive zonotopes that bound the disturbance propagation that arises between the nominal qLPV prediction model from Eq. (11) and the real system trajectories from Eq. (10). This topic is one of the contributions of this paper, as previously detailed.

Consider a prediction horizon of  $N_p$  steps and a group of generic compact sets  $\mathcal{E}(j)$ ,  $j \in \mathbb{N}_{[0, N_p-1]}$ . These sets bound the difference between the scheduling parameter estimates generated at samples  $k$  and  $k+1$ , using the Taylor-based extrapolation approach detailed in the previous section (i.e. via Eq. (14)). That is, we consider that the scheduling parameter estimation error between samples  $(\rho(k+j+1|k+1) - \rho(k+j+1|k))$  belongs to the compact set  $\mathcal{E}(j)$ ,  $\forall j \in \mathbb{N}_{[0, N_p-1]}$ .

By leveraging from Lemma 2, we can rapidly conclude that  $\|(\rho(k+j+1|k+1) - \rho(k+j+1|k))\| \leq \|\xi_\rho(k+j+1|k+1)\| \leq \overline{\xi_\rho}$ , i.e.  $(\rho(k+j+1|k+1) - \rho(k+j+1|k)) \in \mathcal{Q}$ . Seeking simplicity, we use  $\mathcal{E}(j) = \mathcal{Q}$ ,  $\forall j$ . Usually, this is not at all conservative since these residuals have considerably small bounds in many applications (as shown in the validation results

<sup>9</sup>The proposed MPC will be verified to stabilise the process in Sec. 5.



provided in Sec. 6). In any case, if one considers a known decay rate of these residuals  $\xi_\rho(k+j|k)$  as the predictions span along the horizon  $j \in \mathbb{N}_{[0, N_\rho-1]}$ ,  $j$ -decaying sets could replace  $\mathcal{Q}$  over the prediction samples  $j \in \mathbb{N}_{[1, N_\rho]}$ .

**Remark 5.** We stress that if the proposed Taylor-based extrapolation mechanism was not used, as trivial choice for  $\hat{P}_k$  would be  $[\rho^T(k), \dots, \rho^T(k)]^T$ , amounting to a *frozen* scheduling trajectory estimate. Thereby, we could consider the following compact sets  $\mathcal{E}(j) = \delta\mathcal{P}, \forall j$ . Nonetheless, as previously detailed, we assume that  $\mathcal{Q} \subset \delta\mathcal{P}$ . Therefore, we implicitly consider smaller deviations between the entries of  $\hat{P}_{k+1}$  and  $\hat{P}_k$ , thus leading to smaller sets  $\mathcal{E}(j)$ , which reduces the conservatism of the corresponding disturbance propagation zonotopes (detailed in the sequel).

In order to bound the deviance from the nominal state trajectories  $x(k+j|k)$  and the real ones  $x(k+j)$ , we consider **one-step-ahead** propagation sets, which will be later used provide performance certificates for our proposed MPC (in Sec. 5):

**Definition 2.** One-step-ahead disturbance propagation sets  $S(j), j \in \mathbb{N}_{[0, N_\rho]}$  are compact sets that satisfy the following conditions:

1. The initial set  $S(0)$  bounds the load disturbances  $w(k)$ , i.e.  $\mathcal{W} \subseteq S(0)$ ;
2. Consider states  $x(k+j|k), x(k+j) \in \mathbb{R}^{n_x}$  (nominal prediction, real value), control input  $v(k+j) \in \mathbb{R}^{n_u}$ , and scheduling parameters  $\rho(k+j|k), \rho(k+j) \in \mathbb{R}^{n_\rho}$  (extrapolated estimate, real value). Then, if following tightened bounds are satisfied:  $(x(k+j|k), v(k+j)) \in \mathcal{Z}_\pi \ominus (S(j-1) \times \{0\})$ ,  $(x(k+j) - x(k+j|k)) \in S(j-1)$ , and  $(\rho(k+j) - \rho(k+j|k)) \in \mathcal{E}(j-1)$ , it follows that  $\underbrace{(A_\pi(\rho(k+j))x(k+j) + B(\rho(k+j))v(k+j))}_{x(k+j+1)} - \underbrace{(A_\pi(\rho(k+j|k))x(k+j|k) + B(\rho(k+j|k))v(k+j))}_{x(k+j+1|k)} \in S(j)$ , for all  $j \in \mathbb{N}_{[1, N_\rho]}$ , i.e. the one-step-ahead deviation. from the nominal state prediction and the real state value is bounded.

In sum, Definition 2 implies that  $x(k+j|k+1) \in x(k+j|k) \oplus S(j-1), \forall j \in \mathbb{N}_{[1, N_\rho+1]}$ , for any admissible sequence of inputs and scheduling parameter predictions. These one-step-ahead disturbance propagation sets  $S(j)$ , thus, bound the difference between the predictions made in  $k$  and  $k+1$ , and therefore can be used to guarantee recursive feasibility and constraint satisfaction of the MPC based on nominal predictions (Sec. 5).

Next, we detail an exact structure in order to compute such disturbance propagation sets. Specifically, in the following Theorem, we provide zonotope reachable sets that satisfy Definition 2, based on Lemma 3. The main idea behind Theorem 1 is that it offers a direct and rather simple way on how to compute one-step-ahead disturbance propagation sets (i.e. satisfying Definition 2). In order to generate these zonotopes, one only needs the interval matrices  $\mathbf{A}$ ,  $\Delta_A$ , and  $\Delta_B$ , which can be found using interval algebra. Then, by using the zonotopic extensions enabled by Lemma 3, the computation of each  $S(j)$  is direct from Eq. (19), using Minkowski set addition operators. The main interest is that these zonotopes are numerically cheap to compute, and can be directly used in the design of robust MPC algorithms.

**Lemma 3** (Zonotopic extensions, adapted from<sup>27,14</sup>). Consider a centered zonotope  $X = MB_\infty^{n_g} \subseteq \mathbb{R}^m$  and an interval matrix  $\mathbf{J} \in \mathbb{I}^{n \times m}$ . The columns of  $M \in \mathbb{R}^{m \times n_g}$  are named the generators of the zonotope  $X$ . Then, the zonotopic inclusion of the product of the zonotope and interval matrix is defined by:

$$\diamond(\mathbf{J}X) := \text{mid}(\mathbf{J})X \oplus TB_\infty^n, \quad (17)$$

where  $T$  is a diagonal matrix with the following entries along its diagonal:

$$T_{ii} = \sum_{j=1}^{n_g} \sum_{k=1}^m \text{rad}(\mathbf{J})_{ik} |M_{kj}|, \quad \forall i \in \mathbb{N}_{[1, n]}. \quad (18)$$

It holds that  $\mathbf{J}X \subseteq \diamond(\mathbf{J}X)$ , for all  $\mathbf{J} \in \mathbf{J}$ .

*Proof.* Follows directly from Theorem 3 of<sup>27</sup>, by taking  $p = 0$  and  $\mathbf{M} = \mathbf{J}M$ .  $\square$

**Theorem 1.** Consider two zonotopes  $Z \in \mathbb{R}^{n_x+n_u}$  and  $S_0 \in \mathbb{R}^{n_x}$ , an interval matrix  $\mathbf{A} \in \mathbb{I}^{n_x \times n_x}$ , and two sets of interval matrices  $\Delta_A(j) \in \mathbb{I}^{n_x \times n_x}$  and  $\Delta_B(j) \in \mathbb{I}^{n_x \times n_u}, \forall j \in \mathbb{N}_{[0, N_\rho]}$ . Furthermore, consider that  $S_0$  bounds the load disturbances, i.e.

$\mathcal{W} \subseteq \mathcal{S}_0$ , and that  $\mathcal{Z}$  contains the admissibility bounds of the nonlinear system in Eq. (1), i.e.  $\mathcal{Z}_\pi \subseteq \mathcal{Z}$ . Furthermore, consider that:

1. The state transition matrix is bounded, i.e.  $A_\pi(\rho) \in \mathbf{A}$ ,  $\forall \rho \in \mathcal{P}$ ;
2. The difference between the real state transition matrix (computed with the correct scheduling parameter) and a biased one (computed based on a scheduling parameter estimate) is bounded, i.e.  $A_\pi(\rho(k+j)) - A_\pi(\rho(k+j|k)) \in \Delta_A(j)$ ,  $\forall \rho(k+j), \rho(k+j|k) \in \mathcal{P}$ , under  $(\rho(k+j) - \rho(k+j|k)) \in \mathcal{E}(j)$ ,  $\forall j \in \mathbb{N}_{[0, N_p-1]}$ ;
3. The difference between the real input transition matrix (computed with the correct scheduling parameter) and a biased one (computed based on a scheduling parameter estimate) is bounded, i.e.  $B(\rho(k+j)) - B(\rho(k+j|k)) \in \Delta_B(j)$ ,  $\forall \rho(k+j), \rho(k+j|k) \in \mathcal{P}$ , under  $(\rho(k+j) - \rho(k+j|k)) \in \mathcal{E}(j)$ ,  $\forall j \in \mathbb{N}_{[0, N_p-1]}$ .

Then, for  $\mathcal{V}(j) := \diamond((\Delta_A(j) \ \Delta_B(j)) \ \mathcal{Z})$ , the following set of zonotopes  $\mathcal{S}(j)$ ,  $\forall j \in \mathbb{N}_{[0, N_p]}$ , satisfy Definition 2:

$$\mathcal{S}(j) := \begin{cases} \mathcal{S}_0, & j = 0, \\ \mathcal{V}(j) \oplus \diamond(\mathbf{A}\mathcal{S}(j-1)), & j \in \mathbb{N}_{[1, N_p]}. \end{cases} \quad (19)$$

*Proof.* Use  $\mathcal{S}_0 = \mathcal{W}$ . Then, consider  $x(k+j|k), x(k+j) \in \mathbb{R}^{n_x}$ ,  $v(k+j) \in \mathbb{R}^{n_u}$ , and  $\rho(k+j|k), \rho(k+j) \in \mathbb{R}^{n_\rho}$ , for all  $j \in \mathbb{N}_{[1, N_p-1]}$ . Take  $\Delta(k+j|k) = (A_\pi(\rho(k+j))x(k+j) + B(\rho(k+j))v(k+j)) - (A_\pi(\rho(k+j|k))x(k+j|k) + B(\rho(k+j|k))v(k+j|k))$ . Thus, we obtain:

$$\Delta(k+j|k) = (A_\pi(\rho(k+j)) - A_\pi(\rho(k+j|k)))x(k+j|k) \quad (20)$$

$$\begin{aligned} &+ A_\pi(\rho(k+j))(x(k+j) - x(k+j|k)) + (B(\rho(k+j)) - B(\rho(k+j|k)))v(k+j) \\ &\in (\Delta_A(j) \ \Delta_B(j)) \ \mathcal{Z} \oplus \mathbf{A}\mathcal{S}(j-1) \subseteq \mathcal{V}(j) \oplus \diamond(\mathbf{A}\mathcal{S}(j-1)) = \mathcal{S}(j). \end{aligned} \quad (21)$$

Therefore, the sets  $\mathcal{S}(j)$  satisfy Definition 2, which concludes this proof.  $\square$

The zonotopes  $\mathcal{S}(j)$  obtained via Theorem 1 grow with regard to the disturbance propagation term  $\Delta(k+j|k)$  from Eq. (20), which measures the one-step deviance from the real system trajectories and the nominal ones, which consider the estimated scheduling parameters from Eq. (14). Accordingly, these sets depend on the original admissibility bounds of the system  $\mathcal{Z}_\pi$  and also on the estimation errors  $\xi_\rho(k+j|k)$ , which appears as a multiplicative uncertainty. Thanks to Lemma 2, we are able to replace these errors by their worst-case bounds  $\overline{\xi}_\rho \geq \|\xi_\rho(k+j|k)\|$ .

Regarding the conservatism implied with zonotopes, we note that the worst-case bounds  $\overline{\xi}_\rho$  from Lemma 2 are quite reduced, which thus make these sets not so large with respect to the original constraints  $\mathcal{Z}$  (as shown in practice, in Sec. 6). The use of zonotopes that bound disturbance propagation has been recurrently seen in recent robust MPC literature, e.g.<sup>13,14</sup>, even for the LPV case, e.g.<sup>8,26</sup>. An alternative approach would be to compute disturbance propagation tubes, as done in<sup>28</sup>.

We note that the proposed zonotopes tend to be slightly larger than the ones proposed in<sup>14</sup>, due to the addition of the term  $\mathcal{V}(j)$ . However, this is due to the use of linear estimations to describe nonlinear dynamics, since the online update of the qLPV model must be taken into account in the disturbance propagation. Nonlinear predictions can result in less conservative sets  $\mathcal{S}(j)$ , but also result in a higher online computational burden. This represents a trade-off between larger uncertainty propagation and lower online computational cost brought by the linear predictions.

*Remark 6.* In the case of qLPV systems with  $A_\pi(\rho)$  and  $B(\rho)$  affine<sup>10</sup> on  $\rho$ , it follows that  $A_\pi(\rho(k+j)) - A_\pi(\rho(k+j|k)) = \overline{A}_\pi(\rho(k+j) - \rho(k+j|k))$  and  $B(\rho(k+j)) - B(\rho(k+j|k)) = \overline{B}(\rho(k+j) - \rho(k+j|k))$ , with  $\overline{A}_\pi(\cdot)$  and  $\overline{B}(\cdot)$  being linear mappings. Then, the interval matrices  $\Delta_A(j)$  and  $\Delta_B(j)$  can be computed directly from  $\mathcal{E}(j)$ . In the case of non-affine models, interval arithmetic or optimisation can be used to obtain the interval matrices  $\Delta_A(j)$  and  $\Delta_B(j)$  from  $\mathcal{P}$  and  $\mathcal{E}(j)$ .

*Remark 7.* Being zonotopes a symmetric class of sets, the disturbance propagation given by Theorem 1 may be conservative if the uncertainty distribution is excessively asymmetrical. This problem can be mitigated by considering constrained zonotopes, which do not suffer from this source of conservatism. To this end, Theorem 1 can possibly be adapted based on the mean-value extension of constrained zonotopes proposed in<sup>29</sup>. This discussion is out of the scope of this paper and thus not extended herein.

<sup>10</sup>In this case, the feedback gain  $K_\pi$  must be parameter-independent such that  $A_\pi(\rho) = A(\rho) + B(\rho)K_\pi$  becomes affine.

## 5 | A NOVEL ROBUST NMPC ALGORITHM FOR TRACKING

In this Section, we present the main contribution of this paper, which is the novel robust NMPC scheme for Tracking. Although the proposed strategy holds similarities to the tracking NMPC algorithms from<sup>2,3</sup>, it differs significantly due to the fact that it is based on a qLPV prediction model of the system, i.e. Eq. (11). By using such qLPV nominal predictions, the proposed method is able to operate much faster than the prior, since the resulting numerical complexity becomes much closer to that of a QP (rather than an NLP). Complementary, we employ constraint tightening, terminal cost and terminal constraints in order to ensure recursive feasibility and stability properties of the resulting closed-loop, as done in<sup>13</sup>.

### 5.1 | Motivation: Generic NMPC for Tracking

Firstly, we motivate the debate by detailing how NMPC algorithms can be tuned for the case of possibly unreachable output reference signals, as shown in the literature<sup>2,12</sup>. In general, in order to potentially increase the closed-loop domain of attraction (and avoid feasibility losses due to set-point changes), artificial (virtual) reference variables are included to the optimisation, e.g.<sup>2,30</sup>.

The concept is as follows: instead of ensuring that the output variable tracks the time-varying output set-point  $y_r$ , the MPC is tuned so that the output alternatively tracks a new decision variable  $y_a$ . Furthermore, an additional offset cost  $V_O(y_a - y_r)$  is included to the optimisation, which ensured that the deviation between artificial reference  $y_a$  and the real set-point  $y_r$  is minimised, while  $y_a$  stays within the set of admissible output targets reached within  $N_p$  steps (horizon of the MPC).

For correct implementation, this artificial variable tool must also be converted into related steady-state state and input variables. That is, the MPC must choose a reachable and admissible artificial reference  $y_a \in \mathcal{Y}_a$  and, then, convert it into state and input coordinates through  $x_a = g_x(y_a)$  and  $u_a = g_u(y_a)$ . These (nonlinear) constraints are, thus, included to the optimisation problem.

Then, in the general nonlinear case, for time-varying piece-wise constant output reference signal  $y_r \in \mathcal{Y}_T$ , the NMPC for Tracking algorithm (from<sup>2</sup>) is as given by the solution of the following NLP at each sampling instant:

**NMPC Optimisation Problem (from<sup>2</sup>)**

$$\begin{aligned} \min_{V_k, y_a} \quad & \sum_{j=0}^{N_p-1} \overbrace{\ell(x(k+j|k) - x_a, u(k+j|k) - u_a) + V(x(k+N_p|k) - x_a) + V_O(y_a - y_r)}^J, \\ \text{s.t. :} \quad & \begin{cases} x(k+j+1|k) = f(x(k+j|k), v(k+j|k) + K_\pi x(k+j|k)), j \in \mathbb{N}_{[0, N_p-1]}, \\ u(k+j|k) = v(k+j|k) + K_\pi x(k+j|k), j \in \mathbb{N}_{[0, N_p-1]}, \\ (x(k+j|k), v(k+j|k)) \in \mathcal{Z}_\pi, j \in \mathbb{N}_{[0, N_p-1]}, \\ x_a = g_x(y_a), u_a = g_u(y_a) \\ (x(k+N_p|k), y_a) \in \Gamma \\ y_a \in \mathcal{Y}_a \end{cases} \end{aligned} \quad (22)$$

where  $\ell(\cdot, \cdot)$  is a quadratic stage cost,  $V(\cdot)$  is a terminal cost, and  $\Gamma$  is Tracking Positive Invariant set. From the optimal solution of this NLP, i.e.  $V_k^*$ , the first entry  $v^*(k|k)$  is applied to the process according to Eq. (7), which implicitly defines  $u(k)$ .

### 5.2 | The Proposed qLPV NMPC Optimisation Problem

In opposition to the NLP in Eq. (22), we propose herein a novel robust formulation, which makes use of the qLPV model and removes the nonlinear constraints from the prior.

We make reference to Remark 9 from<sup>3</sup>: the NMPC propositions in previous references<sup>2,3</sup> provide exponential closed-loop stability and recursive feasibility guarantees in the case of no model mismatch. Nevertheless, this is rarely the case in any practical application. Thus, one of the main features of the proposed mechanism is that it includes constraint tightening tools in order to robustly tackled the issue of disturbances and model mismatches.

When the qLPV predictions from Eq. (11) are used, considering the scheduling trajectory estimates from Eq. (14), model uncertainties inherently emerge, and thus should be accounted for. For such, as previously debated, the synthesis of our NMPC

benefits from the zonotopes detailed in Sec. 4. Through the sequel, we assume that robust constraint satisfaction is guaranteed thanks to the constraint tightening and the corresponding disturbance propagation zonotopes.

The contracted constraints that ensure robustness are detailed: consider an initial constraint set  $\mathcal{Z}_\pi(0) = \mathcal{Z}_\pi$ ; then, the following sets, along the prediction horizon, for  $j \in \mathbb{N}_{[1, N_p]}$ , are iteratively given by:

$$\mathcal{Z}_\pi(j+1) = \mathcal{Z}_\pi(j) \ominus (S(j) \times \{0\}), \quad (23)$$

where  $S(j)$  stands for the zonotopes derived by the means of Theorem 1, which propagate the uncertainty along the prediction horizon.

*Remark 8.* Note that the disturbance propagation sets  $S(j)$  increase along the horizon and thus the sets  $\mathcal{Z}_\pi(j)$  from Eq. (23) shrink as  $j$  increases. For correctness of the NMPC application, these sets must be non-empty for all steps within the horizon  $j \in \mathbb{N}_{[1, N_p]}$ .

Furthermore, the terminal constraint in Eq. (22) is also adjusted. Specifically, we consider a parameter-dependent Tracking Robust Positive Invariant<sup>11</sup> (TRPI) set, defined as follows:

**Definition 3** (Parameter-dependent TRPI Set). Consider a set  $\Gamma(\rho) \subseteq \mathbb{R}^{n_x+n_y}$  and a terminal control law  $u_t = \kappa_t(x, y_r) - K_\pi x$ . The set  $\Gamma(\rho)$  is Tracking Robust Positive Invariant for the qLPV system in Eq. (3) if, for all  $(x, y_r) \in \Gamma(\rho)$ ,  $\rho, (\rho + \delta\rho) \in \mathcal{P}$ , and  $w \in S(N_p)$ , it follows that  $(A(\rho)x + B(\rho)\kappa_t(x, y_r)) + w, y_r) \in \Gamma(\rho + \delta\rho)$ .

*Remark 9.* TRPI sets differ from the Tracking Positive Invariant (TPI) sets from<sup>2</sup> due to the robustness properties. The synthesis of TPI sets implicitly considers that the real system trajectories and the nominal predictions are identical.

*Remark 10.* The definition of a TRPI set implies that once the states  $x$  and the virtual reference  $y_a$  are found inside such set, terminal control law  $\kappa_t : \mathbb{R}^{n_x+n_y} \rightarrow \mathbb{R}^{n_u}$  ensures that any subsequent state, for the same output reference, also stays inside this set, regardless of the bounded load disturbance  $w \in \mathcal{W}$ .

Therefore, our method is as follows: at each sampling instant  $k$ , we measure the state  $x(k)$ , compute the scheduling parameter  $\rho(k)$ , estimate the scheduling sequence  $\hat{P}_k$  using Eq. (14), and solve the following optimisation problem:

#### Proposed NMPC Optimisation Problem

$$\begin{aligned} \min_{V_k, y_a} \quad & \sum_{j=0}^{N_p-1} \overbrace{\ell(x(k+j|k) - x_a, u(k+j|k) - u_a) + V(x(k+N_p|k) - x_a, \hat{\rho}(k+N_p-1|k)) + V_O(y_a - y_r)}^J, \\ \text{s.t. :} \quad & \begin{cases} x(k+j+1|k) = A_\pi(\hat{\rho}(k+j|k))x(k+j|k) + B(\hat{\rho}(k+j|k))v(k+j|k), j \in \mathbb{N}_{[0, N_p-1]}, \\ u(k+j|k) = v(k+j|k) + K_\pi x(k+j|k), j \in \mathbb{N}_{[0, N_p-1]}, \\ (x(k+j|k), v(k+j|k)) \in \mathcal{Z}_\pi(j), j \in \mathbb{N}_{[0, N_p-1]}, \\ x_a = g_x(y_a), u_a = g_u(y_a) \\ (x(k+N_p|k), y_a) \in \Gamma(\hat{\rho}(k+N_p-1|k)) \\ y_a \in \mathcal{Y}_a \end{cases} \end{aligned} \quad (24)$$

The main changes from the NLP in Eq. 22 (NMPC for Tracking from<sup>2</sup>) to the proposed approach in Eq. (24) are:

- The qLPV nominal predictions replace the nominal nonlinear predictions;
- Tightened constraints sets  $\mathcal{Z}_\pi(j)$  are used in order to ensure robustness;
- A parameter-dependent TRPI set  $\Gamma(\rho)$  is used as a terminal set.

<sup>11</sup>The requirement of a tracking positive invariant set is not at all restrictive for piece-wise constant reference signals, and equally used in other references, e.g.<sup>2</sup>. Note that this set can be partitioned in multiple regions, as shown in Sec. 6.

The set of states  $x(k) \in \mathcal{X}_a(N_p)$  such that Eq. (24) has a feasible solution is called the domain of attraction of the proposed controller<sup>12</sup>.

### 5.3 | Artificial Reference Computation

We note that the pair of constraints  $x_a = g_x(y_a)$  and  $u_a = g_u(y_a)$  in Eq. (24) still make it an NLP. Thus, we proceed by providing a final adjustment to the proposed MPC, in order to further alleviate its resulting numerical burden. Note that the presence of the artificial reference variable  $y_a \in \mathcal{Y}_a$  in the NMPC optimisation serves to prevent the possible losses of feasibility due to abrupt changes in the set-point (or non-admissible set-point value). Nevertheless, the constraints related to this variable (i.e.  $x_a = g_x(y_a)$  and  $u_a = g_u(y_a)$ ) significantly increase the computational complexity of the optimisation, since they are associated to (most possibly) nonlinear functions  $g_x(\cdot)$  and  $g_u(\cdot)$ .

Therefore, we replace these constraints by solving the optimisation in two steps: (1) first, we determine the artificial reference variable  $y_a$  via a separate optimisation (typically named "reference governor" schemes, e.g. <sup>11,21</sup>), and (2) then, we determine the control policy. By doing so, we are able to make the proposed NMPC optimisation just as complex as only two consecutive QPs.

First, for any new output reference target value  $y_r$ , we consider the optimal admissible target as given by:

$$y_a^o = \arg \min_{y_a \in \mathcal{Y}_a} V_O(y_a - y_r). \quad (25)$$

Note that, since  $V_O(\cdot)$  is a quadratic offset that weights the deviation from the artificial set-point to the real one, we can understand this optimal target as the closes one to the new set-point value  $y_r$  within the set of admissible targets.

Next, consider a *feasible* candidate artificial target  $y_a^c$ . In practice, this candidate variable is simply taken as the last artificial reference variable  $y_a^*$ , which is ensured to be feasible due to the recursive feasibility property (Theorem 1, presented in the sequel) of the optimisation.

*Remark 11.* We stress that  $y_a^c$  can only be selected from the second piece-wise constant reference change moment onward, since, at the first iteration of the MPC, the full nonlinear optimisation from Eq. (24) should be solved such that a recursively feasible candidate exists for the following samples.

Next, let  $y_a^\alpha$  be a convex combination of the candidate and optimal artificial reference targets, that is:

$$y_a^\alpha = (1 - \alpha) y_a^c + \alpha y_a^o, \quad (26)$$

considering a selection scalar  $\alpha \in [0, 1]$ . Note that, from the convexity of  $\mathcal{Y}_a$ , it is implied that  $y_a^\alpha \in \mathcal{Y}_a, \forall \alpha \in [0, 1]$ .

Taking into consideration these new variables, we propose the following auxiliary optimisation problem to select the artificial reference when a set-point change occurs:

#### Proposed Artificial Reference Choice Optimisation Problem

$$\begin{aligned} \max_{V_k, \alpha} \quad & \alpha \\ \text{s.t.:} \quad & \begin{cases} x(k+j+1|k) = A_\pi(\hat{\rho}(k+j|k))x(k+j|k) + B(\hat{\rho}(k+j|k))v(k+j|k), j \in \mathbb{N}_{[0, N_p-1]}, \\ (x(k+j|k), v(k+j|k)) \in \mathcal{Z}_\pi(j), j \in \mathbb{N}_{[0, N_p-1]}, \\ \alpha \in [0, 1], \\ (x(k+N_p|k), y_a^\alpha) \in \Gamma(g_\rho(y_a^o)). \end{cases} \end{aligned} \quad (27)$$

From the solution of the quadratic optimisation program in Eq. (27), we obtain the optimal value  $\alpha^*$ , which is thus used to select the new artificial reference value  $y_a = (1 - \alpha^*)y_a^c + \alpha^*y_a^o$ .

Then, the proposed NMPC in Eq. (24) is solved without the constraints  $x_a = g_x(y_a)$ ,  $u_a = g_u(y_a)$ , and  $y_a \in \mathcal{Y}_a$  and without the offset cost  $V_O(y_a - y_r)$ , since  $y_a$  is no longer a decision variable, but an input to the optimisation. From it, the optimal control sequence  $V_k^*$  (solution to the optimisation in Eq. (24)). We stress that this two-step optimisation procedure is analogous to solving the NLP in Eq. (24).

<sup>12</sup>Due to the freedom provided by the artificial reference  $y_a$ , the feasibility property becomes independent of  $y_r$ <sup>2</sup>.

*Remark 12.* Note that the maximisation in Eq. (27) is convex, quadratic and has a real positive scalar  $\alpha \in [0, 1]$  as decision variable. Thus, an equivalent approach to solving is to perform a bisection search over the unit simplex. By doing so, we can test the feasibility of (24) for a given  $\alpha$ . If the problem is feasible, we use the corresponding artificial variable  $y_a$  and proceed to the MPC solution; otherwise, we pursue with the bisection. We emphasise that the bisection search should also satisfy preserve the Lyapunov decay<sup>13</sup> condition in order to maintain the tracking error convergence property, that is, the new artificial target should satisfy:  $V(x(k+1|k) - g_x(y_a^\alpha)) - V(x(k|k) - g_x(y_a^c)) \leq -\ell(x(k|k) - g_x(y_a^c), \kappa_t(x(k), y_a^c) - g_u(y_a^c))$ .

*Remark 13.* Again, recall that the at the initial sample, i.e.  $k = 0$ , the complete NLP from Eq. (24) must be solved. Anyhow, if for any consecutive time sample  $k \geq 1$  (when a reference change occurs), the solution of optimisation in Eq. (27) leads to  $\alpha^* = 1$ , it is implied that  $y_a = y_a^o$ . In this case, the auxiliary reference governor optimisation becomes irrelevant until there is another variation of the set-point.

## 5.4 | Synthesis Requirements

Next, we give some specific hypothesis on the form of the quadratic penalties costs  $\ell(\cdot, \cdot)$ ,  $V(\cdot)$ , and the terminal TRPI set  $\Gamma(\cdot)$ , which are required to construct Eq. (24) with performance certificated. We note that the following requirements are similar to those presented in<sup>13</sup> and<sup>2</sup>, which develop predictive control strategies for robust regulation and nominal tracking, respectively.

**Assumption 5.** The MPC cost  $J = \sum_{j=0}^{N_p-1} \ell(x(k+j|k) - x_a, (v(k+j|k) + K_\pi x(k+j|k)) - u_a) + V(x(k+N_p|k) - x_a, \hat{p}(k+N_p-1|k))$  satisfies the requirements:

1. Its stage cost  $\ell(x, u)$  quadratic, positive definite and uniformly continuous. Therefore, it follows that  $\ell(x, u) \geq \alpha_\ell(\|x\|)$  and  $|\ell(x_1, u_1) - \ell(x_2, u_2)| \leq \lambda_x(\|x_1 - x_2\|) + \lambda_u(\|u_1 - u_2\|)$ , where  $\alpha_\ell$ ,  $\lambda_x$  and  $\lambda_u$  are  $\mathcal{K}$ -functions.
2. The set of **reachable** admissible artificial references  $\mathcal{Y}_a = \{y_a \in \mathbb{R}^{n_y} : (g_x(y_a), y_a) \in \Gamma(g_\rho(y_a))\}$  is a convex subset of the admissible tracking outputs reached within  $N_p$  steps, i.e.  $\{y_a \in \mathcal{Y}_T : (g_x(y_a), g_u(y_a)) \in \mathcal{Z}_\pi(N_p)\}$ .
3. Its output offset cost  $V_O(\cdot)$  is quadratic, positive definite, uniformly continuous and convex, thus assuring that the minimiser  $y_a^o = \arg \min_{y_a \in \mathcal{Y}_a} V_O(y_a - y_r)$  is unique. Furthermore, for any  $y_r \in \mathbb{R}^{n_y}$  and  $y_a \in \mathcal{Y}_a$ , we have  $V_O(y_a - y_r) - V_O(y_a^o - y_r) \geq \alpha_O(\|y_a - y_a^o\|)$ , where  $\alpha_O$  is a  $\mathcal{K}$ -function<sup>14</sup>.
4. Its terminal cost  $V(\cdot)$  is a control Lyapunov function for the unconstrained qLPV system in Eq. (3), such that for all  $(x, y_a) \in \Gamma(\rho)$  there exist constants  $b > 0$  and  $\sigma > 1$  such that  $V(x - x_a, \rho) \leq b\|x - x_a\|^\sigma$ . Also, due to the uniform continuity of  $V(x, \rho)$  w.r.t.  $x$ , we have:  $V(A(\rho)(x - x_a) + B(\rho)\kappa_t(x - x_a, y), \rho^+) - V(x - x_a, \rho) \leq -\ell(x - x_a, \kappa_t(x, y)x - u_a)$ , where  $x_a = g_x(y_a)$  and  $u_a = g_u(y_a)$ , and  $V(x_1 - x_a, \rho_1) - V(x_2 - x_a, \rho_2) \leq \lambda_r(\|x_1 - x_2\|)$ .

Furthermore, the terminal control law  $\kappa_t(x_a, y_a)$  and terminal set  $\Gamma$  satisfy:

1. The terminal control law implies that  $\kappa_t(g_x(y_a), y_a) = g_u(y_a)$  for all admissible equilibrium points<sup>15</sup>.
2. The terminal set  $\Gamma(\rho)$  is an admissible TRPI set. That is,  $\Gamma(\rho)$  is a subset<sup>16</sup> of  $\Lambda(N_p) = \{(x, y) \in \mathbb{R}^{n_x} \times \mathcal{Y}_a : (x, \kappa_t(x, y) - K_\pi x) \in \mathcal{Z}(N_p)\}$ , thus satisfying Definition 3 for the terminal control law and disturbances  $w \in \mathcal{S}(N_p)$ . Equivalently, it is implied that, for  $\rho, (\rho + \delta\rho) \in \mathcal{P}$ ,  $(x, y_s) \in \Gamma(\rho) \Rightarrow (x^+, y_s) \in \Gamma(\rho + \delta\rho)$ ,  $x^+ = A(f_\rho(x))x + B(f_\rho(x))\kappa_t(x, y_s) + w$ ,  $w \in \mathcal{S}(N_p)$ ,  $(x, \kappa_t(x, y_s)) \in \mathcal{Z}$ .

## 5.5 | Terminal Ingredients

Next, we provide a computationally elegant solution that can be used to compute parameter-dependent qLPV terminal ingredients, through the solution of matrix inequalities. The following Theorems provide recursive feasibility and exponential stability

<sup>13</sup>This condition is further detailed in the sequel, e.g. Theorem 2.

<sup>14</sup>Note that if the target output is admissible and reachable (i.e.  $y_r \in \mathcal{Y}_a$ ), the minimiser is  $y_a^o = y_r$  and the previous inequality is reduced to  $V_O(y) \geq \alpha_O(\|y\|)$ .

<sup>15</sup>An evident explicit alternative for this terminal law is  $\kappa_t(x, y_a) = K_t(f_\rho(x))(x - g_x(y_a)) + g_u(y_a)$ .

<sup>16</sup>Note that  $\mathcal{Z}(N_p)$  is the set of points  $(x, u)$  such that  $u = v + K_\pi x$  with  $(x, v) \in \mathcal{Z}_\pi(N_p)$ .

guarantees for the qLPV system in Eq. (3) subject to the proposed MPC control law from Eq. (7). For the sake of presentation clarity, the longer proofs are provided in Appendix A.

In this paper, we synthesise  $\rho$ -dependent terminal ingredients, based on a positive-definite matrix  $P(\rho)$ . We use an ellipsoidal invariant  $\mathbf{X}_f := \{x \mid x^T P(\rho)x \leq 1\}$ , which is robust positively invariant regarding the closed-loop dynamics (Eq (8)), in such way that  $\Gamma(\rho) := \mathbf{X}_f(\rho) \times \mathcal{Y}_a$  is a TRPI set. Complementary, we consider a sub-level terminal cost  $V(e, \rho) = e^T P(\rho)e$  and a parameter-dependent terminal feedback  $\kappa_t(e, y_a) = K_t(\rho)(e) + g_u(y_a)$ , being  $e = x - g_x(y_a)$ . Note that the this TRPI set is related to the dynamics at the end of the horizon, inherently linked to the corresponding last scheduling parameter. Thus, we use the following constraint  $x(k + N_p | k, y_a) \in \Gamma(\hat{\rho}(k + N_p - 1 | k))$  in the proposed optimisation.

Consider the tracking error dynamics  $e(k + j | k) = x(k + j | k) - g_x(y_a)$ , being  $x(k + j | k)$  being the nominal qLPV predictions from Eq. (11). We define  $\theta(k + j | k) := w(k + j | k) + \Delta(k + j | k) \in S(j)$  as the (bounded) uncertainties: result of the disturbance and the model-process mismatch due to the differences between the real scheduling variables  $\rho(k + j)$  and  $\hat{\rho}(k + j | k)$  (scheduling trajectory estimates from Eq. (14)).

From the nominal control law and the terminal condition ( $u(k + N_p | k) = \kappa_t(x(k + N_p | k), y_a) - K_\pi x(k + N_p | k)$ , with  $\kappa_t(x(k + N_p | k), y_a) = K_t(\hat{\rho}(k + N_p - 1 | k))e(k + N_p | k) + g_u(y_a)$ ), we obtain:

$$e(k + N_p | k) = \overbrace{(A(\hat{\rho}(k + N_p - 1 | k)) + B(\hat{\rho}(k + N_p - 1 | k))K_t(\hat{\rho}(k + N_p - 1 | k)))}^{A_t(\hat{\rho}(k + N_p - 1 | k))} e(k + N_p - 1 | k) + \theta(k + N_p - 1 | k) \quad (28)$$

$$\theta(k + N_p - 1 | k) = (A(\hat{\rho}(k + N_p - 1 | k)) - I_{n_x})g_x(y_a) + B(\hat{\rho}(k + N_p - 1 | k))g_u(y_a) + w(k + N_p | k) \in S(N_p). \quad (29)$$

Then, the following Theorems ensure that the error dynamics from Eq. (28) converge to the origin as  $\lim_{k \rightarrow +\infty} w(k) \rightarrow 0$  (i.e. tracking is ensured). Again, recall that the total uncertainty set  $S(j)$  encompasses the disturbances and the model-mismatches (refer to Definition 2).

**Theorem 2.** Stability and Recursive Feasibility, adapted from<sup>31</sup>

Consider Assumption 5 holds. Suppose there exists a terminal control law  $u_t = (K_t(\rho)(x - g_x(y_a)) + g_u(y_a))$ . Consider that the MPC is given by Eq. (24), with a terminal state set given by  $\mathbf{X}_f(\rho)$  and a terminal cost  $V(x, \rho)$ . Then, input-to-state stability is ensured if the following conditions hold  $\forall \rho \in \mathcal{P}$ :

- (C1) The origin  $e = 0$  lies in the interior of  $\mathbf{X}_f(\rho)$ ;
- (C2) Any consecutive error to  $e \in \mathbf{X}_f(\rho)$ , given in closed-loop by  $e^+ = A_t(\rho)e + \theta$ , lies within  $\mathbf{X}_f(\rho + \delta\rho)$ , i.e.  $e \in \mathbf{X}_f(\rho) \Rightarrow e^+ \in \mathbf{X}_f(\rho + \delta\rho)$ ,  $\forall \theta \in S(N_p)$ ,  $\forall \delta\rho \in \delta\mathcal{P}$ ;
- (C3) The discrete-time Lyapunov inequality is verified within this invariant set, this is,  $\forall x \in \mathbf{X}_f(\rho)$ ,  $\forall \rho \in \mathcal{P}$ , and  $\forall \delta\rho \in \delta\mathcal{P}$ :  $V(e^+, \rho + \delta\rho) - V(e, \rho) \leq -\ell(e, \kappa_t(x, y) - g_u(y))$ .
- (C4) The image of the terminal feedback lies within the admissible control domain:  $K_t(\rho)(x - g_x(y)) + g_u(y) \in \mathcal{U}$ ,  $\forall \rho \in \mathcal{P}$ .
- (C5) The terminal set  $\mathbf{X}_f(\rho)$  is a subset of  $\mathcal{X}$ ,  $\forall \rho \in \mathcal{P}$ .

Assuming that the initial solution of the MPC problem  $V_0^*$  is feasible, then, the MPC is recursively feasible, steering  $e = (x - g_x(y))$  to the origin.

*Proof.* This proof is standard and basically gathers the sufficient conditions from Assumption 5. Refer to<sup>31</sup> for the full proof, taking into account the corresponding changes in the model.  $\square$

**Theorem 3.** Tracking Robust Positive Invariant Set

Assume that there exists an ellipsoidal terminal set  $\mathbf{X}_f(\rho)$ .  $\mathbf{X}_f$  is a robust positively invariant set iff, for any  $e \in \mathbf{X}_f$  and  $\rho \in \mathcal{P}$ , i.e.  $e^T P(\rho)e \leq 1$ , it follows that  $(e^+)^T P(\rho + \delta\rho)e^+ \leq 1$ , i.e. the successor state  $e^+$  is also inside  $\mathbf{X}_f$ , which implies in:

$$(A_t(\rho)e + \theta)^T P(\rho + \delta\rho)(A_t(\rho)e + \theta) \leq 1. \quad (30)$$

Then,  $\Gamma(\rho)$  is a TRPI for system (3) as follows:

$$\Gamma(\rho) := \{(x, y) \in \mathbb{R}^{n_x \times n_y} \mid (x - g_x(y)) \in \mathbf{X}_f(\rho), h(x, K_t(\rho)(x - g_x(y)) + g_u(y)) \in \mathcal{Y}_a\}. \quad (31)$$

*Proof.* The validity of Eq. (30) follows directly from the following argument:  $x(k), x(k+1) \in \mathbf{X}_f(\rho)$ ,  $\forall k \geq 0$  and  $y = h(x, K_\pi x + K_t(\rho)(x - g_x(y)) + g_u(y)) \in \mathcal{Y}_a$ .  $\square$



$$\begin{aligned}
& \left[ \begin{array}{cccc|cccc} Y(\rho) & \star & \star & \star & 0 & \star & 0 & 0 \\ (A(\rho)Y(\rho) + B(\rho)W(\rho)) & Y(\rho^+) & 0 & 0 & \theta & \star & 0 & 0 \\ Y(\rho) & 0 & Q^{-1} & 0 & 0 & 0 & I_{n_x} & 0 \\ W(\rho) & 0 & 0 & R^{-1} & 0 & 0 & 0 & I_{n_x} \\ \hline 0 & \theta^T & 0 & 0 & -\zeta & \theta^T & 0 & 0 \\ (A(\rho)Y(\rho) + B(\rho)W(\rho)) & Y(\rho^+) & 0 & 0 & \theta & Y(\rho^+) & 0 & 0 \\ 0 & 0 & I_{n_x} & 0 & 0 & 0 & I_{n_x} & 0 \\ 0 & 0 & 0 & I_{n_x} & 0 & 0 & 0 & I_{n_x} \end{array} \right] \geq 0, \quad (32) \\
& \text{(a)} \left[ \begin{array}{cc} (\bar{u}_i - I_{n_u, \{i\}} u_r)^2 & \star \\ (I_{n_u, \{i\}} W(\rho))^T & Y(\rho) \end{array} \right] \geq 0, \forall i \in \mathbb{N}_{[1, n_u]}, \text{ (b)} \left[ \begin{array}{cc} (\bar{x}_j - I_{n_x, \{j\}} x_r)^2 & \star \\ (I_{n_x, \{j\}} Y(\rho))^T & Y(\rho) \end{array} \right] \geq 0, \forall j \in \mathbb{N}_{[1, n_x]}, \quad (33) \\
& \left[ \begin{array}{cc|cc} \lambda Y(\rho) & \star & 0 & \star \\ (A(\rho)Y(\rho) + B(\rho)W(\rho)) & Y(\rho^+) & \star & \star \\ \hline 0 & \theta^T & (1 - \lambda) & \star \\ (A(\rho)Y(\rho) + B(\rho)W(\rho)) & Y(\rho^+) & \theta & Y(\rho^+) \end{array} \right] > 0. \quad (34)
\end{aligned}$$

#### Theorem 4. Terminal Ingredients

Conditions (C1)-(C5) of Theorem 2 and the inequality of Theorem 3 are satisfied if there exist a symmetric parameter-dependent positive definite matrix  $P(\rho) : \mathbb{R}^{n_p} \rightarrow \mathbb{R}^{n_x \times n_x}$ , a parameter-dependent rectangular matrix  $W(\rho) : \mathbb{R}^{n_p} \rightarrow \mathbb{R}^{n_u \times n_x}$ , a scalar  $\zeta > 1$  and a scalar  $\lambda \in ]0, 1]$  such that  $Y(\rho) = (P(\rho))^{-1} > 0$ ,  $W(\rho) = K_i(\rho)Y(\rho)$  and that LMIs (32)-(33) and the BMI (34) hold under the minimisation of  $\log \det\{Y(\rho)\}$  using  $\rho^+ = \rho + \delta\rho$  for all  $\rho \in \mathcal{P}$  and  $\delta\rho \in \delta\mathcal{P}$ , considering  $\theta$  as the vertices of  $S(N_p)$ .

*Remark 14.* The terminal ingredients provided by Theorem 4 ensure recursive feasibility and convergence of the error trajectories (as verified in Propositions 1-2). Note that these ingredients are **robust** w.r.t. the mismatches between the nominal model from Eq. (11) and the real system trajectories from Eq. (10). The robustness is implied thanks to the bounds on uncertainty, which are derived from the bounds on the scheduling trajectory estimation error (Lemma 2). Moreover, constraint satisfaction is also enabled by robust design, when tightening the constraints along the horizon with the uncertainty propagation zonotopes  $S(j)$ .

*Remark 15.* The BMI in Theorem 4 can be solved through simple bisection search over the optimisation plane since  $0 < \lambda \leq 1$ , by construction, as argues<sup>32</sup>.

*Remark 16.* Theorem 4 provides infinite-dimensional inequalities, which must hold  $\forall \rho \in \mathcal{P}$  and  $\forall \delta\rho \in \delta\mathcal{P}$ . In practice, the solution can be found by enforcing the inequalities over a sufficiently dense grid of points  $(\rho, \delta\rho)$  along the  $\mathcal{P} \times \delta\mathcal{P}$  plane. Then, the solution can be verified over a denser grid. The parameter-dependency of  $P$  may be dropped if the system is quadratically stabilisable, but this may result in quite conservative performances.

## 5.6 | Certificates

**Proposition 1** (Recursive Feasibility). *Let there exist a solution  $Y(\rho)$  to Theorem 4. Then, given any  $x \in \mathcal{X}_a(N_p)$ ,  $y_a \in \mathbb{R}^{n_y}$  and  $u = K_\pi x + v$ ,  $v = \kappa(x, y_a) - K_\pi x$ , we have  $x^+ = A(f_\rho(x))x + B(f_\rho(x))u + w \in \mathcal{X}_a(N_p)$ ,  $\forall w \in \mathcal{W}$ . Consider an optimal sequence  $V_k^* = [(v^*(k|k))^T, (v^*(k+1|k))^T, \dots, (v^*(k+N_p-1|k))^T]^T$  and an optimal artificial target  $y_a^*$ . Then  $V_k^c = [(v^*(k+1|k))^T, \dots, (v^*(k+N_p-1|k))^T, \kappa_t(x(k+N_p|k), y_a^*) - u_a]^T$  and  $y_a^c = y_a^*$  define a feasible (candidate) solution from the optimisation in Eq. (24) for any  $\bar{y}_a \in \mathbb{R}^{n_y}$  and  $w \in \mathcal{W}$ , which means that Eq. (24) is recursively feasible.*

**Proposition 2** (Tracking Error Convergence). *Let there exist a solution  $Y(\rho)$  to Theorem 4. Then, the qLPV system in Eq. (3) in closed-loop with the MPC input from Eq. (7) is stable and the tracking error dynamics (Eq. (28)) uniformly converge to a neighbourhood of the origin. That is, for any feasible initial condition  $x_0$  and constant set-point  $y_r \in \mathbb{R}^{n_y}$ , with  $w(k) \in \mathcal{W}$ , it is*

implied that:

$$\|x(k) - x_a(k)\| \leq \beta(\|x(0)\|, k) + \gamma(\bar{w}), \quad (35)$$

where  $\beta$  and  $\gamma$  are respectively a  $\mathcal{KL}$ -function and a  $\mathcal{K}$ -function and  $\bar{w}$  is such that  $\|w(k)\| \leq \bar{w}, \forall k$ .

*Remark 17.* Note that for an admissible equilibrium state  $x_a$ , the virtual control sequence  $V_k^a = [v_a^T, \dots, v_a^T]^T$ , where  $u_a = v_a + K_\pi x_a$ , is admissible since it maintains the system at  $x_a$ . Therefore, the set of corresponding admissible equilibrium states  $\{x \in \mathbb{R}^{n_x} : \exists u_a \in \mathbb{R}^{n_u}, (x_a, u_a) \in \mathcal{Z}_\pi(N_p), h(x_a, u_a) \in \mathcal{Y}_a\}$  is a subset of  $\mathcal{X}_a(N_p)$  and feasibility is not lost for any set-point change.

*Remark 18.* In practice, we note that the feasible candidate solution  $(V_k^c, y_a^c)$  can be used as a warm-start to the optimisation in Eq. (24).

## 5.7 | A summary

Next, we provide a brief summary of how the proposed method is implemented:

**Offline procedure:**

- Verify the baseline Assumption 2, required to apply the method;
- **Conceive a qLPV realisation of the nonlinear dynamics, in the form of Eq. (24);**
- **Determine a (locally) stabilising state-feedback in the form of Eq. (7) (within the admissibility set  $\mathcal{X}$ );**
- Map the admissible equilibrium points determined by the output target  $y_r$  through Eq. (6), and determine the related maps  $g_x(\cdot)$ ,  $g_u(\cdot)$  and  $g_p(\cdot)$ ;
- Compute the estimation error bounds through Lemma 2.
- Compute the uncertainty propagation sets  $S(j)$  through Theorem 1.
- **Determine the robust tracking terminal ingredients through Theorem 4.**

**Online procedure:**

At each sample  $k$ :

- **Estimate the future sequence of scheduling variables, using the recursive extrapolation method (Eq. (14)), obtaining  $\hat{P}_k$ ;**
- For every reference change in the piece-wise constant target signal  $y_r$ : solve the artificial reference choice (Eq. (27) together with the bisection selection), obtaining  $y_a^*$
- **Solve the MPC optimisation in Eq. (24), using  $y_a^*$  and removing the related constraints and offset cost<sup>17</sup>;**
- The resulting control signal is applied using  $u(k) = K_\pi x(k) + v^*(k|k)$ .

## 6 | RESULTS

In this Section, we provide a simple case study in order to illustrate the features of the proposed Tracking NMPC algorithm. We debate the advantages and disadvantages of our method, as well as comparing it to a state-of-the-art approach (i.e.<sup>2</sup>, summarized in Eq. 22). All the results presented in the sequel were obtained in a 2.4 GHz, 8 GB RAM Macintosh computer.

<sup>17</sup>Remove constraints: (a)  $x_a = g_x(y_a)$ ,  $u_a = g_u(y_a)$ , and (b)  $y_a \in \mathcal{Y}_a$ ; and also remove the offset cost:  $V_O(y_a - y_r)$ .

## 6.1 | Nonlinear model and constraints

We consider the benchmark cascaded tank process from<sup>33</sup>, considering the interconnection of two tanks, with an open hole at the bottom of the first tank, which leaks fluid to the second. The latter has a pump at its end, regulated by a local proportional controller. The nonlinear level dynamics are:

$$\begin{cases} \frac{dh_1(t)}{dt} = -\frac{a\sqrt{2gh_1(t)}}{A_1} + \frac{\gamma}{A_1}u(t), \\ \frac{dh_2(t)}{dt} = \frac{a\sqrt{2gh_1(t)}}{A_2} - \frac{k_p}{A_2}h_2(t). \end{cases} \quad (36)$$

Each  $h_i(t)$  represents the water level at the  $i$ -th tank, measured in centimeters;  $u(t)$  represents the tension applied for the main pump in volts, for which the corresponding flow is  $\gamma u(t)$ . The tank cross sections  $A_i$  are of  $10 \text{ cm}^2$ , while the outlet hole cross section  $a$  is of  $0.05 \text{ cm}^2$ . The pump parameter  $\gamma$  is of  $1.4 \text{ cm}^3/(\text{Vs})$ . The proportional coefficient  $k_p$  is of  $1.1 \text{ cm}^2/\text{s}$ .

Both level signals are considered the system state variables, i.e.  $x(t) = [h_1(t) \ h_2(t)]^T$ , whereas  $y(t) = h_1(t)$  is the controlled output (the main variable of interest. The control input is the main pump tension signal  $u(t)$ .

This system has the following admissibility constraints:

- States:  $x_j \in [1, 10] \text{ cm}, \forall j \in \mathbb{N}_{[1,2]}$ , so that the water level does no overflow the tanks, while always staying over a given minimal threshold;
- Input:  $u \in [0, 5] \text{ V}$ , so that the tension signal does not saturate.

For coherence with the nonlinear model in Eq. (1), we assume that this process is also subject to bounded additive disturbances  $w(k) \in \mathbb{R}^2$  such that  $\|w(k)\| \leq 0.05 \text{ cm}$ . These disturbances perturb both level dynamics and could represent, for instance, unaccounted leaks or flows to each tank.

## 6.2 | qLPV Embedding

In order to obtain a discrete-time qLPV realisation of this nonlinear system, we first consider an Euler discretisation using  $T_s = 0.25 \text{ s}$ , which yields:

$$\begin{cases} h_1(k+1) = h_1(k) - T_s \frac{a\sqrt{2gh_1(k)}}{A_1} + T_s \frac{\gamma}{A_1}u(k), \\ h_2(k+1) = h_2(k) + T_s \frac{a\sqrt{2gh_1(k)}}{A_2} - T_s \frac{k_p}{A_2}h_2(k). \end{cases} \quad (37)$$

Then, we choose the following nonlinear scheduling proxy:

$$\rho(k) := f_\rho(x) := (x_1)^{-0.5}, \quad (38)$$

which satisfies the requirement for differential inclusion and boundedness (Assumption 2). Thus, we obtain a qLPV-embedded model in the form of Eq. (3) with matrices:

$$A(\rho) = \begin{bmatrix} (1 - \frac{T_s a \sqrt{2g}}{A_1} \rho) & 0 \\ \frac{T_s a \sqrt{2g}}{A_2} \rho & (1 - \frac{T_s k_p}{A_2}) \end{bmatrix}, \quad B(\rho) = \begin{bmatrix} \frac{T_s \gamma}{A_1} \\ 0 \end{bmatrix}.$$

From the bounds of the system states and the boundedness of the chosen scheduling proxy, we obtain the following scheduling parameter set:

$$\mathcal{P} := [0.3, 1] \text{ cm}^{-0.5}. \quad (39)$$

Furthermore, from the discrete-time model and the state bounds we obtain the following bounds for the scheduling parameters' variations:

$$\delta \mathcal{P} := \{\delta \rho \in \mathbb{R} \mid -0.034 \leq \delta \rho \leq 0.0052\}. \quad (40)$$

### 6.3 | Tracking

For tracking purposes, we consider the convex output tracking set of admissible references  $\mathcal{Y}_T := [1, 10]$  cm. Moreover, we stress that the output steady-state condition from Eq. (6) implicitly defines the following functions:

$$\begin{cases} g_x(y_r) := \left[ \frac{y_r}{a\sqrt{2gy_r}} \right] \\ g_u(y_r) := \frac{k_p}{\gamma} \end{cases}, \quad (41)$$

Note that controller that defines the input  $u(k)$  must ensure that the the output  $y(k) = x_1(k)$  tracks the pierce-wise reference target  $y_r$ , but also that the constraints on both states  $x(k) \in \mathcal{X}$  and input  $u(k) \in \mathcal{V}$  are respected. These constraints are always active.

### 6.4 | MPC Synthesis and Terminal Ingredients

Using this generated qLPV embedding model, we compare the proposed NMPC method with the "NMPC for Tracking" algorithm from<sup>2</sup>, i.e. Eq. (22). Note that this other algorithm operates on the basis of the original discrete-time nonlinear model of the system.

In order to synthesise the predictive controllers, we use a prediction horizon of  $N_p = 4$  steps and the quadratic stage cost  $\ell(x, u) = \|x\|_Q^2 + \|u\|_R^2$  with  $Q = I_{n_x}$  and  $R = 1$ . We note that the short size of the horizon is specifically chosen in order to emphasise the numerical capabilities of the proposed scheme which, even in these simpler cases, exhibits considerable decrease on the resulting time required to compute the control action, as shown in the sequel.

In order to synthesise the terminal ingredients via Theorem 4, we partition the output set  $\mathcal{Y}_T$  in ten different partitions, thus finding one parameter-dependent RPI set  $\Gamma(\rho)$  per partition, i.e. for  $y_r \in [1, 2]$  or  $(2, 3]$  or  $(3, 4]$ , and so on up to  $(9, 10]$ . For the NMPC algorithm, we use terminal ingredients synthesised through the procedure from<sup>2</sup>, Appendix B.

Figure 1 shows the parameter-dependent TRPI sets and the quadratic TPI sets used for the NMPC algorithm<sup>2</sup>. The parameter-dependent sets (proposed in this paper) are shown in bold black lines for frozen values of  $\rho \in \mathcal{P}$ , i.e. for  $\rho = 0.3, 0.37, 0.44, \dots, 1$ . All sets are translated from the error coordinates  $(x - x_r)$  to the state coordinates  $x$ , centered at the different state targets  $x_r$  of each partition. We note that, albeit the parameter-dependent sets being slightly smaller than the TPI sets (due to the robustness considerations), Theorem 4 generates sufficiently large terminal regions for each reference partition  $y_r \in \mathcal{Y}_T$ .

### 6.5 | Scheduling Sequence Estimation

Before presenting the actual control results, we provide the scheduling sequence extrapolation estimates obtained with the recursive method presented in Sec. 3. As detailed in Lemma 1, convergence is indeed verified. The obtained bounds for the estimation error, using Lemma 2, are  $\|\xi_\rho\| \leq 0.015 \text{ cm}^{-0.5}$ , as shown in Figure 2. The extrapolation mechanism offers very precise estimates  $\hat{P}_k$ , which means that the nominal qLPV predictions obtained through Eq. (11) are very close to the real system trajectories of Eq. (10) and thus the disturbance propagation along the horizon is reduced.

### 6.6 | Disturbance Propagation

The disturbance propagation reachable sets  $S(j), \forall j \in \mathbb{N}_{[0, N_p]}$  for the proposed algorithm were then obtained, considering the zonotopic disturbance propagation method and a closed-loop prediction paradigm defined by (7), with  $K_\pi = (-24.92 \ 0)$  calculated as proposed in<sup>14</sup>. Note that  $S(0)$  stands for the load disturbance set  $\mathcal{W}$ ; the following zonotopes comprise the propagation of the load disturbances and the model-process mismatches along the prediction horizon. These sets are computed according to Theorem 1.

In Figure 3, we show the collection of disturbance propagation sets  $S(j)$  over the  $x_1 \times x_2$  plane (Definition 2). In this Figure, we can also see the original state admissibility set  $\mathcal{X}$ . Since the zonotopes  $S(j)$  are much smaller in size than  $\mathcal{X}$ , we can infer that the conservatism of the proposed method is quite reduced. We recall that the vertices of  $S(N_p)$  were used to construct the terminal ingredients through Theorem 4.

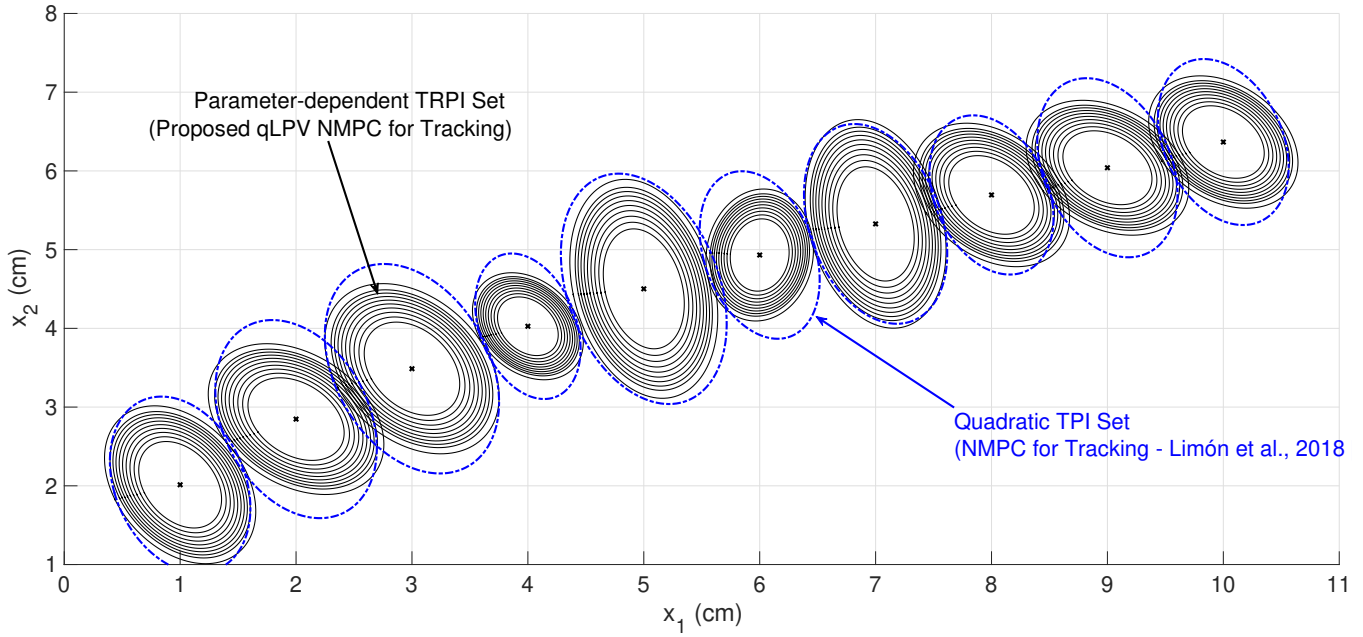


FIGURE 1 Synthesised TRPI set partitions  $\Gamma(\rho)$  and the quadratic TPI set partitions.

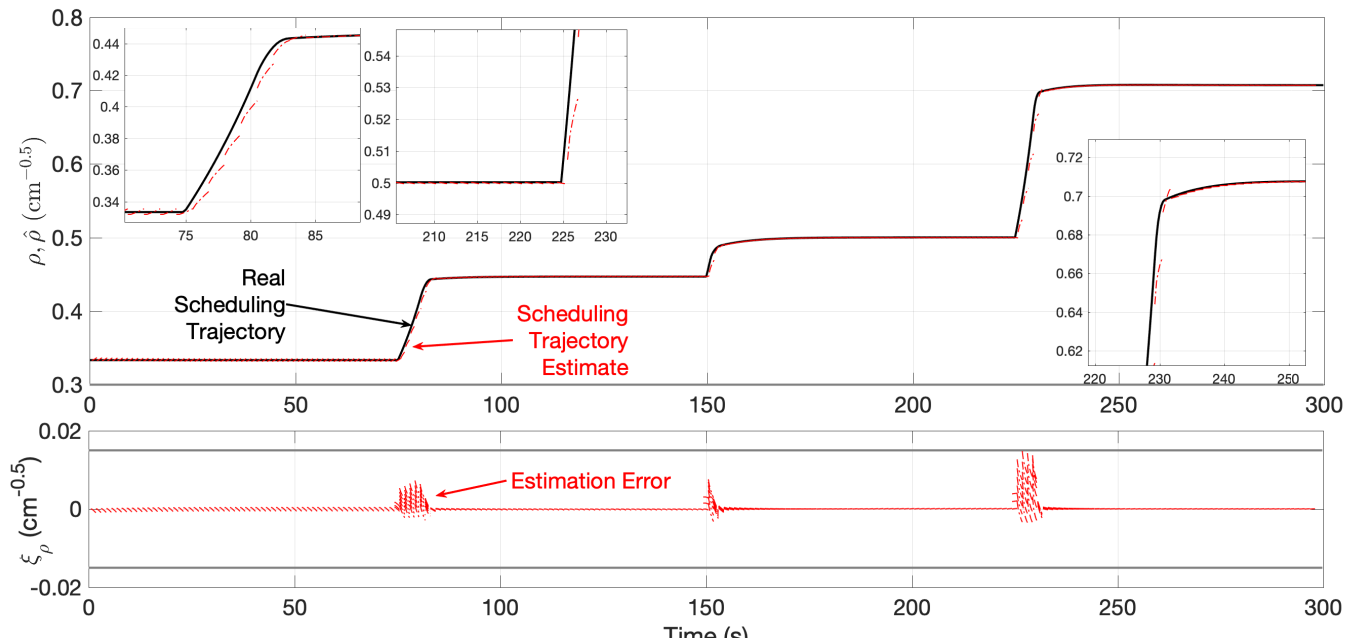
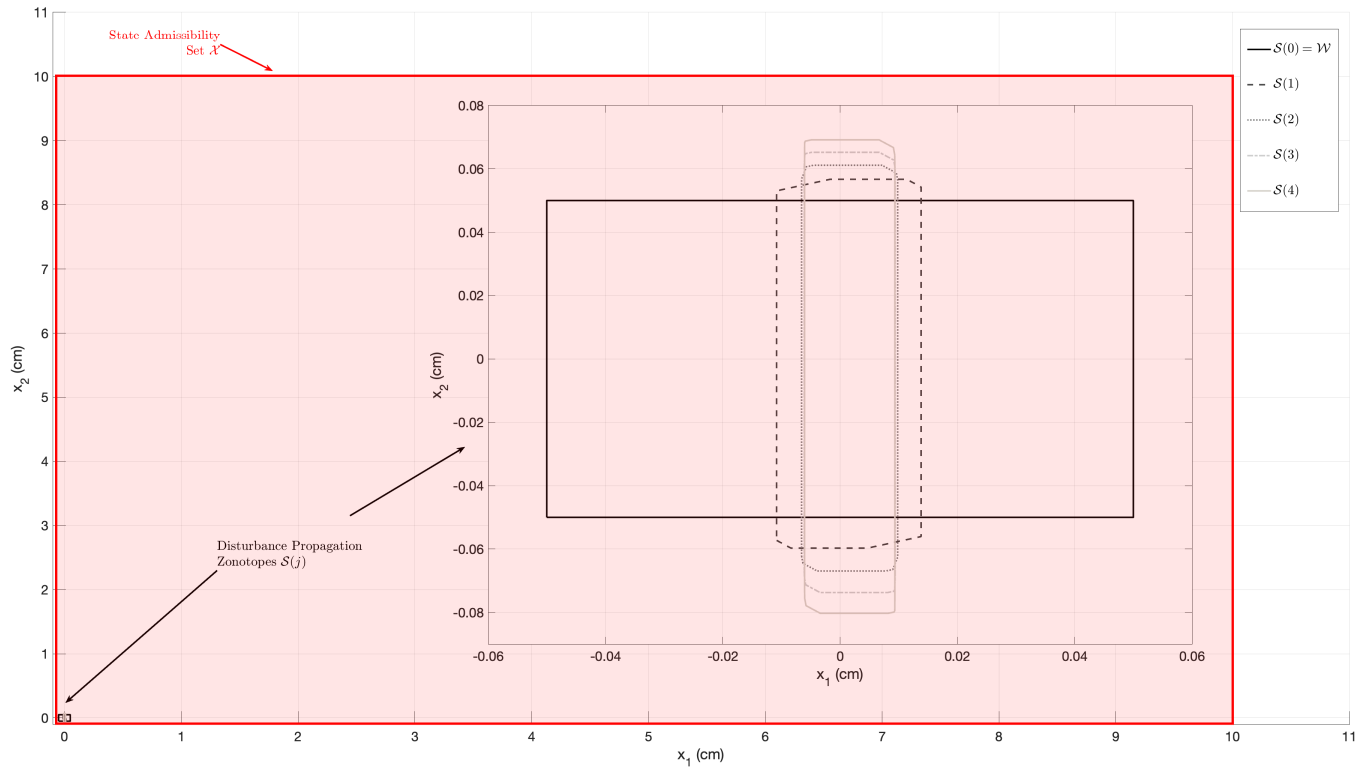


FIGURE 2 Scheduling Sequence Estimates  $\hat{P}_k$  at different samples  $k$  and estimation error  $\xi_{\rho}(k+j|k)$  (and bounds).

## 6.7 | Simulation Scenarios

We consider two different simulation scenarios. Since the NMPC algorithm from<sup>2</sup> is not robust by design, we first consider and compare the obtained tracking performances of both algorithms, without the presence of load disturbances (i.e. with  $w$  nil). Then, we consider the disturbance rejection robust performances solely of the proposed robust algorithm. **Note that both controllers consider the same system (and also the same MPC synthesis weights), just represent via different realisations (nonlinear and qLPV models). The simulation is obtained using the nonlinear model.**

FIGURE 3 Zonotopic sets  $S(j)$ .

### 6.7.1 | Nominal Tracking Performances

Considering a step-like piece-wise constant output target signal which passes through  $y_r = 2, 4, 5$  and  $9$  cm, the obtained tracking performances with both algorithms are shown in Figures 4 and 5.

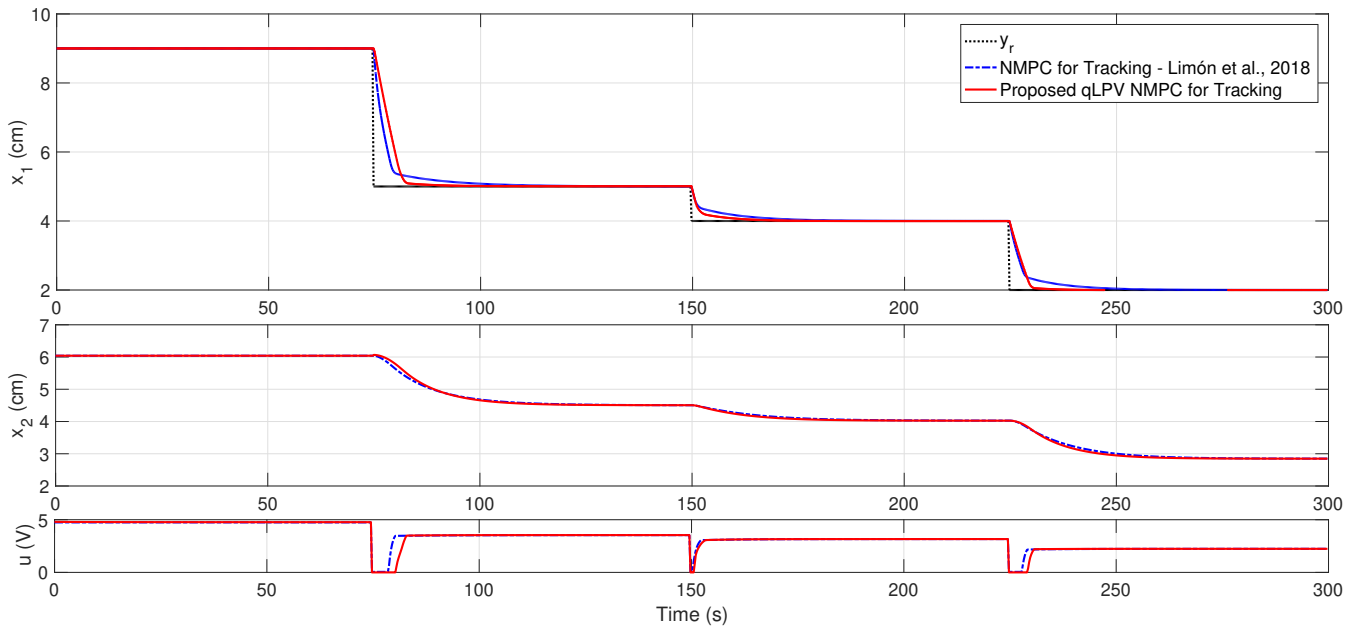
Figure 4 presents the resulting state, input and output trajectories, while Figure 5 shows the state phase plane and the terminal sets. Complementary, Figure 6 provides the values for the artificial reference tuning variable  $\alpha$  for the proposed mechanism.

As one can see, the obtained tracking performances with both methods are offset-free steady-state output points. The proposed method ensures slightly faster convergence than the original NMPC for Tracking scheme from<sup>2</sup>.

The main advantage resides in its simpler implementation, of QP-alike numerical burden, enable through the qLPV embedding. With the qLPV model, nonlinear constraints do not have to be solved internally by the optimisation procedure. The proposed qLPV NMPC mechanism requires only the operation of: one linear recursive law (Eq. (14)) and one QP problem (Eq. (24)). In the moments of reference changes, one bisection search (Remark 12) is also required, which increases the number of QPs to, at most, five iterations per sample. In contrast, the original NMPC for Tracking requires the solution of an NLP optimisation problem per sampling instant, which is numerical-wise much harder.

In order to better compare the two tracking controllers, we assess the obtained performance results with performance indexes, presented in Table 1. The results are debated:

- Firstly, we stress that there is an overall performance enhancement with the proposed method: there is a small decrease on the integral of absolute output tracking error (IAE) w.r.t.<sup>2</sup>, of roughly 12 %. This performance enhancement can also be quantified through the RMS index of the cost function  $J$  of the controllers, as well as in terms of the average tracking error (2.15 % with the proposed method, while 2.68 % using<sup>2</sup>). This performance enhancement is indeed an interesting feature of the proposed scheme, since the method from<sup>2</sup> has been exploited in many nonlinear applications presented in the literature.
- Complementary, we stress that the generated control input is smoother with the proposed method: we obtain a control signal with 43 % reduction on its total variance (TV).



**FIGURE 4** Nominal Performances: Proposed qLPV NMPC vs. NMPC for Tracking<sup>2</sup>. State and input trajectories.

- Furthermore, the main advantage of the proposed method is that the average computational time needed to solve the control problem ( $t_c$ ) is reduced over 44 %, as also exhibited in Figure 7. This is a strong and very significant feature, since the system order is small ( $n_x = 2$ ) and so is the chosen control horizon. The complexity of the NLP solution from<sup>2</sup> grows exponentially with  $(N_p \times n_x)$ , which can be a serious issue with time-critical systems. The proposed method has QP-alike burden, and thus  $t_c$  grows only linearly with  $(N_p \times n_x)$ . This means it is readily-conceived for embedded applications.
- Note that the model-process discrepancies (differences between Eqs. (10) and (11)) are very well-handled with the zonotope-based constraint-tightening approach, since the generated sets  $S(j)$  are arguably small (see Figure 3). This feature corroborates with prior discussions seen in the literature indicating this approach as a promising, e.g.<sup>3</sup>.
- Finally, we also stress that the terminal ingredients conceived with the proposed Theorem generate sufficiently large terminal sets, able to guarantee recursive feasibility for rather larger output-related sets.

### 6.7.2 | Robust Tracking Performances

In order to illustrate the robustness properties of the proposed algorithm, we provide a second brief simulation scenario<sup>18</sup>. Consider three uniformly random disturbance sequences with unitary seeds, multiplied by decaying exponential terms, as illustrated in Figure 8.

In Figure 9, we show the state and output behaviours w.r.t. a step-like output target goal  $y_r$ . Clearly, robust stability is ensured: as the load disturbance sequences dissipate, the error trajectories  $(x - x_r)$  converge to the origin; moreover, while  $w(k)$  is non-null, the states stabilize at constant steady-states regimes, as close as possible to  $x_r$ . **This is an additional nice feature of the proposed method, which is able to robustly tolerate bounded disturbances, which was not possible with competing NMPC techniques for tracking.**

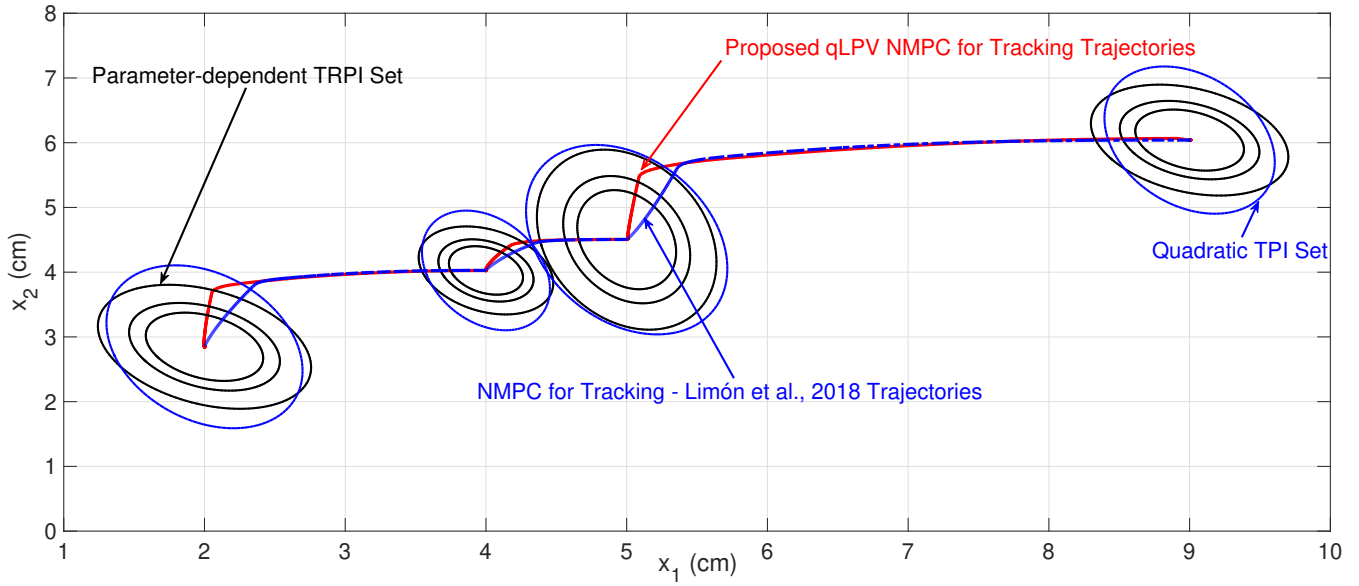
### 6.8 | Relevant concerns

As a summary of the previous results, we provide the following list of the assets and liabilities the proposed method:

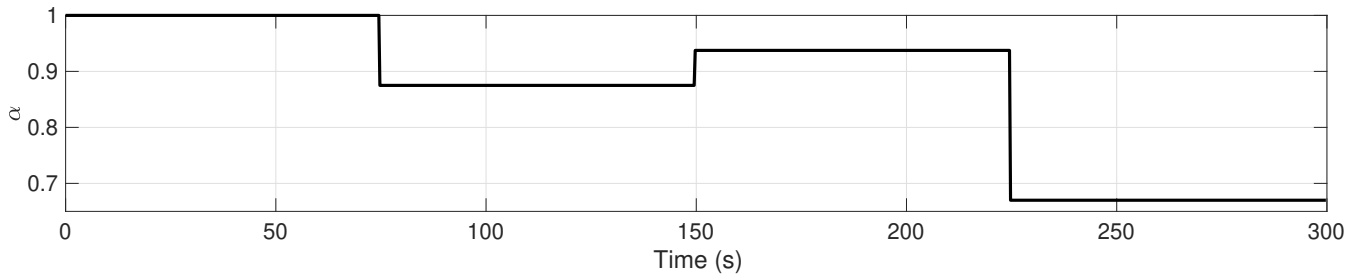
- **Advantages:**

<sup>18</sup>We opt not to test the method from<sup>2</sup> against load disturbances since it is not a robust algorithm, which would in turn result in an unfair comparison.





**FIGURE 5** Nominal Performances: Proposed qLPV NMPC vs. NMPC for Tracking<sup>2</sup>. State phase plane and TRPI sets.

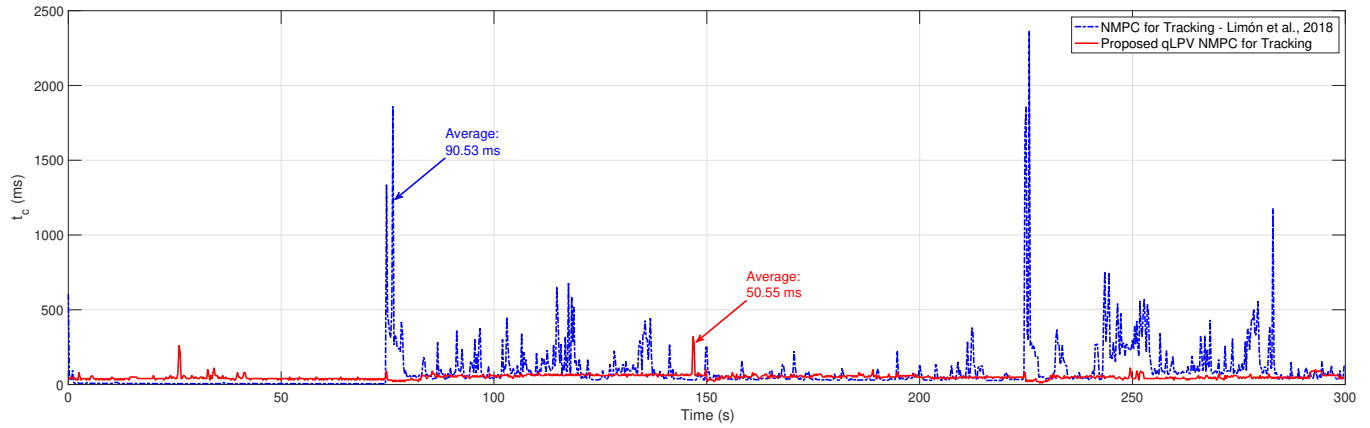


**FIGURE 6** Artificial reference choice variable.

1. It is able to operate much faster than state-of-the-art nonlinear MPCs for tracking, given that a reduced-complexity optimisation procedure is used (of QP-alike complexity).
2. It included artificial reference variables such that even unreachable reference goals are able to be directly taken into account by the controller.
3. It includes (easy-to-compute) robustness arguments, written in terms of known the bounds of the load disturbances.
4. Certificates of recursive feasibility and stability are available, which ensures an adequate behaviour of the resulting closed-loop.

• **Disadvantages:**

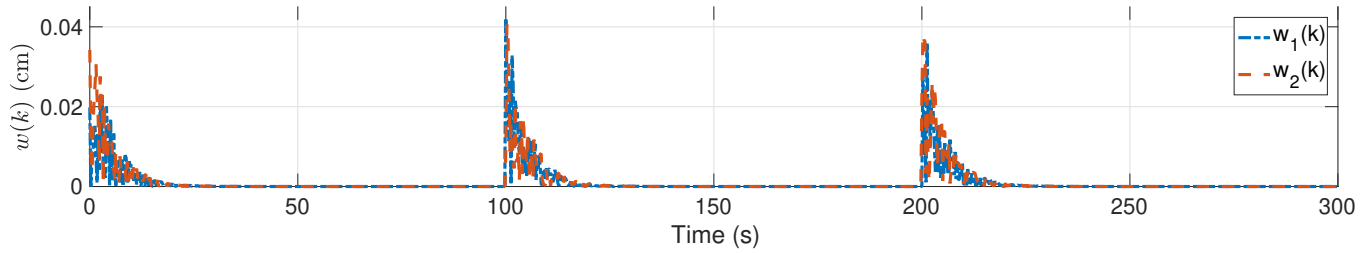
1. It requires a qLPV realisation of the nonlinear system, and thus the availability of a known proxy  $f_\rho(\cdot)$  that generates bounded scheduling variables (and also abides to the hypothesis given in Assumption 2).
2. As in many Tracking NMPC algorithms, the state variables should be measurable, since the controller guarantees output tracking by steering the states to given steady-state variables.
3. TRPI sets must be computed offline, before the online implementation of the controller, in order to ensure correct behaviours of the resulting closed-loop.



**FIGURE 7** Computation Time: Proposed qLPV NMPC vs. NMPC for Tracking<sup>2</sup>.

**TABLE 1** Performance Comparison.

Method	IAE := $\sum_k \ y_r(k) - y(k)\ $	RMS $\{J\}$	TV := $\sum_k \ u(k+1) - u(k)\ $	$t_c$
NMPC for Tracking <sup>2</sup>	108.23	80.18	35.93	90.53 ms
Proposed qLPV NMPC for Tracking	95.31	80.14	20.44	50.55 ms



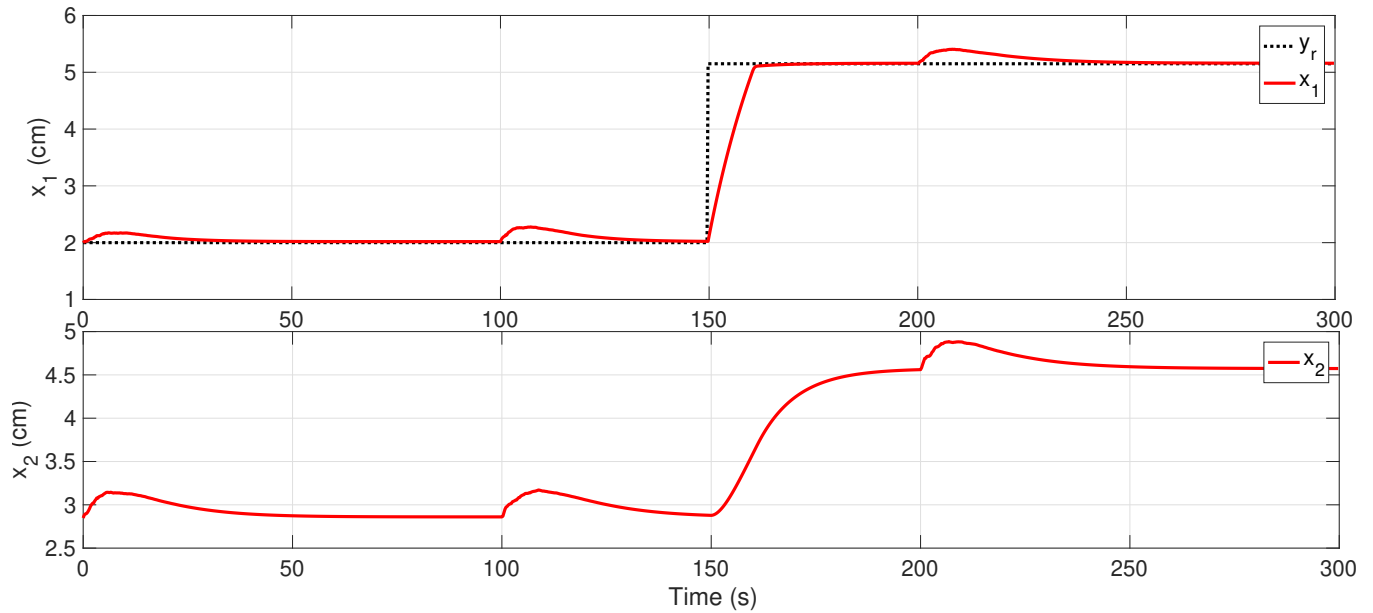
**FIGURE 8** Disturbance Scenario.

## 7 | CONCLUSION

In this work, we developed a novel Tracking NMPC algorithm, based on qLPV embeddings. The method uses the recursive estimation of the future qLPV scheduling trajectories, made available through a simple Taylor expansion. The propagation of the model mismatches along the NMPC horizon are addressed by the means zonotopes which bound the uncertainty propagation. Furthermore, we provide an LMI-solvable remedy for the case of bounded additive disturbances, which computes a robust LPV feedback gain and parameter-dependent terminal ingredients. The derived tracking robust positive invariant set ensures recursive feasibility of the optimisation procedure as well as input-to-state stability of the process.

Considering a benchmark cascaded tank system, we thoroughly compare the proposed method against the nominal tracking NMPC framework from<sup>2</sup>. We are able to demonstrate that the proposed scheme achieved very similar tracking performances, with much smaller computational stress, benefiting from the linear predictions enabled by the qLPV realisation. The method is ready for embedded applications (the online stress is similar to that of a QP) and offers robustness towards bounded load disturbances with reduced conservatism.

In future works, we plan on assessing how the bounds of the scheduling parameters can be selected to enforce further conservatism/aggressiveness to the control law.



**FIGURE 9** Robust Performances.

## ACKNOWLEDGEMENTS

The present work was carried out with the support of CAPES, an entity of the Brazilian Government dedicated to the training of human resources. Also, we received the support of CNPq (grants 304032/2019 – 0, 403949/2021 – 1, and 308741/2021 – 8) and Campus France.

## Author contributions

All authors have contributed equally for this paper.

## Financial disclosure

None reported.

## Conflict of interest

The authors declare no potential conflict of interests.

## References

1. Boccia A, Grüne L, Worthmann K. Stability and feasibility of state constrained MPC without stabilizing terminal constraints. *Systems & control letters* 2014; 72: 14–21.
2. Limon D, Ferramosca A, Alvarado I, Alamo T. Nonlinear MPC for tracking piece-wise constant reference signals. *IEEE Transactions on Automatic Control* 2018; 63(11): 3735–3750.
3. Köhler J, Müller MA, Allgöwer F. A nonlinear tracking model predictive control scheme for dynamic target signals. *Automatica* 2020; 118: 109030.

4. Sename O, Gaspar P, Bokor J. *Robust control and linear parameter varying approaches: application to vehicle dynamics*. 437. Springer . 2013.
5. Mohammadpour J, Scherer CW. *Control of linear parameter varying systems with applications*. Springer Science & Business Media . 2012.
6. Morato MM, Normey-Rico JE, Sename O. Model predictive control design for linear parameter varying systems: A survey. *Annual Reviews in Control* 2020; 49: 64 - 80.
7. Hanema J, Tóth R, Lazar M. Stabilizing non-linear model predictive control using linear parameter-varying embeddings and tubes. *IET Control Theory & Applications* 2021; 15(10): 1404–1421.
8. Morato MM, Normey-Rico JE, Sename O. Novel qLPV MPC Design with Least-Squares Scheduling Prediction. *IFAC-PapersOnLine* 2019; 52(28): 158–163.
9. González Cisneros PS, Werner H. Nonlinear model predictive control for models in quasi-linear parameter varying form. *International Journal of Robust and Nonlinear Control* 2020; 30(10): 3945–3959.
10. Morato MM, Normey-Rico JE, Sename O. Sub-optimal recursively feasible Linear Parameter-Varying predictive algorithm for semi-active suspension control. *IET Control Theory & Applications* 2020; 14(18): 2764–2775.
11. Chisci L, Falugi P, Zappa G. Gain-scheduling MPC of nonlinear systems. *International Journal of Robust and Nonlinear Control* 2003; 13(3-4): 295–308.
12. Köhler J, Müller MA, Allgöwer F. A novel constraint tightening approach for nonlinear robust model predictive control. In: *2018 Annual American Control Conference IEEE*. ; 2018: 728–734.
13. Santos TLM, Bonzanini AD, Heirung TAN, Mesbah A. A constraint-tightening approach to nonlinear model predictive control with chance constraints for stochastic systems. In: *American Control Conference IEEE*. ; 2019: 1641–1647.
14. Cunha VM, Santos TL. Robust Nonlinear Model Predictive Control with Bounded Disturbances Based on Zonotopic Constraint Tightening. *Journal of Control, Automation and Electrical Systems* 2021: 1–14.
15. Morato MM, Normey-Rico JE, Sename O. Sufficient conditions for convergent recursive extrapolation of qLPV scheduling parameters along a prediction horizon. *IEEE Transactions on Automatic Control* 2022. doi: 10.1109/TAC.2022.3183534
16. Morato MM, Cunha V, Santos TLM, Normey-Rico JE, Sename O. Robust Nonlinear Predictive Control through qLPV embedding and Zonotope Uncertainty Propagation. In: *Proceedings of the 4<sup>th</sup> IFAC Workshop on Linear Parameter Varying Systems IFAC*. ; 2021.
17. Shamma JS. An overview of LPV systems. In: *Control of linear parameter varying systems with applications* Springer. 2012 (pp. 3–26).
18. Boyd S, El Ghaoui L, Feron E, Balakrishnan V. *Linear matrix inequalities in system and control theory*. 15. Siam . 1994.
19. Bruzelius F, Pettersson S, Breitholtz C. Linear parameter-varying descriptions of nonlinear systems. In: *Proceedings of the 2004 American Control Conference*. 2. IEEE. ; 2004: 1374–1379.
20. Hoffmann C, Werner H. A survey of linear parameter-varying control applications validated by experiments or high-fidelity simulations. *IEEE Transactions on Control Systems Technology* 2014; 23(2): 416–433.
21. Köhler J, Müller MA, Allgöwer F. A nonlinear model predictive control framework using reference generic terminal ingredients. *IEEE Transactions on Automatic Control* 2019; 65(8): 3576–3583.
22. Morato MM, Normey-Rico JE, Sename O. A fast dissipative robust nonlinear model predictive control procedure via quasi-linear parameter varying embedding and parameter extrapolation. *International Journal of Robust and Nonlinear Control* 2021; 31(18): 9619–9651.

23. Morato MM, Naspolini A, Normey-Rico JE, Sename O. NMPC via qLPV models and Taylor-based Scheduling Parameter Extrapolation: A Cartesian Robot Case Study. In: *2022 30th Mediterranean Conference on Control and Automation (MED)*IEEE. ; 2022: 19–24.
24. Cisneros PG, Werner H. Fast nonlinear MPC for reference tracking subject to nonlinear constraints via quasi-LPV representations. *IFAC-PapersOnLine* 2017; 50(1): 11601–11606.
25. Morato MM, Sename O, Dugard L. LPV-MPC Fault Tolerant Control of Automotive Suspension Dampers. *IFAC-PapersOnLine* 2018; 51(26): 31–36.
26. Alcalá E, Puig V, Quevedo J. LPV-MPC Control for Autonomous Vehicles. *IFAC-PapersOnLine* 2019; 52(28): 106–113.
27. Alamo T, Bravo JM, Camacho EF. Guaranteed state estimation by zonotopes. *Automatica* 2005; 41(6): 1035–1043.
28. Hanema J, Lazar M, Tóth R. Stabilizing tube-based model predictive control: Terminal set and cost construction for LPV systems. *Automatica* 2017; 85: 137–144.
29. Rego BS, Raffo GV, Scott JK, Raimondo DM. Guaranteed methods based on constrained zonotopes for set-valued state estimation of nonlinear discrete-time systems. *Automatica* 2020; 111: 108614.
30. Skibik T, Liao-Mc Pherson D, Cunis T, Kolmanovsky IV, Nicotra MM. A Feasibility Governor for Enlarging the Region of Attraction of Linear Model Predictive Controllers. *IEEE Transactions on Automatic Control* 2021.
31. Mayne DQ, Rawlings JB, Rao CV, Scokaert PO. Constrained model predictive control: Stability and optimality. *Automatica* 2000; 36(6): 789–814.
32. Yang W, Gao J, Feng G, Zhang T. An optimal approach to output-feedback robust model predictive control of LPV systems with disturbances. *International Journal of Robust and Nonlinear Control* 2016; 26(15): 3253–3273.
33. Johansson KH. The Quadruple-tank Process: A Multivariable Laboratory Process with an Adjustable Zero. *IEEE Transactions on Control Systems Technology* 2000; 8(3): 456–465.

**How to cite this article:** M. M. Morato, V. M. Cunha, T. L. M. Santos, J. E. Normey-Rico, and O. Sename (2021), A Robust Nonlinear Tracking MPC using qLPV Embedding and Zonotopic Uncertainty Propagation, *International Journal of Robust and Nonlinear Control*, 202x;xx:x–x.

## APPENDIX

### A PROOFS OF THEOREMS AND PROPOSITIONS

#### A.1 Proof of Theorem 4

We begin by showing the positive invariance of the ellipsoid. Applying the  $\mathcal{S}$ -procedure, with  $\lambda > 0$  to the inequality in Eq. (30) and  $(1 - e^T P(\rho)e \leq 1)$ , we get:

$$1 - (A_t(\rho)e + \theta)^T P(\rho^+)(A_t(\rho)e + \theta) - \lambda (1 - e^T P(\rho)e) > 0,$$

which can be rewritten as:

$$\underbrace{(e^T P(\rho) \ I_{n_x}) \begin{bmatrix} N_{11} & N_{12} \\ N_{21} & N_{22} \end{bmatrix}}_N \begin{pmatrix} P(\rho)e \\ I_{n_x} \end{pmatrix} > 0, \quad (\text{A1})$$

with  $N > 0$  and:

$$N_{11} = \lambda Y(\rho) - Y^T(\rho) A_t^T(\rho) P(\rho^+) A_t(\rho) Y(\rho), \quad (\text{A2})$$

$$N_{12} = -Y^T(\rho) A_t^T(\rho) P(\rho^+) \theta, \quad (\text{A3})$$

$$N_{21} = -\theta^T P(\rho^+) A_t(\rho) Y(\rho), \quad (\text{A4})$$

$$N_{22} = (1 - \lambda) - \theta^T P(\rho^+) \theta. \quad (\text{A5})$$

Applying a Schur complement over  $P(\rho + \delta)$  for each  $N_{ij}$  leads to BMI (34), which ensures the requirements of Theorem 3.

Complementary, we proceed by demonstrating that the resulting  $P(\rho)$  satisfies all five conditions of Theorem 2. (C1) trivially holds due to the ellipsoidal form of  $\mathbf{X}_f$ . (C2) is verified due to the fact that  $\mathbf{X}_f$  is a sub-level set of the terminal cost  $V(\cdot)$ . Therefore, if condition (C3) is verified, (C2) is consequently ensured.

The discrete Ricatti condition (C3) is verified through the solution of BMI (32). Since  $Q^{-1} > 0$ ,  $R^{-1} > 0$  and  $Y(\rho + \delta\rho) > 0$ , we apply two consecutive Schur complements to this BMI, which leads to:

$$\left( e^T P(\rho) \ I_{n_x} \right) \underbrace{\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}}_M \begin{pmatrix} P(\rho)e \\ I_{n_x} \end{pmatrix} \geq 0, \quad (\text{A6})$$

with  $M \geq 0$  and:

$$M_{11} = Y(\rho) - Y^T(\rho) Q Y(\rho) - W^T(\rho) R W(\rho) \quad (\text{A7})$$

$$- Y^T(\rho) A_t^T(\rho) P(\rho^+) A_t(\rho) Y(\rho),$$

$$M_{12} = -Y(\rho) A_t^T(\rho) P(\rho^+) \theta, \quad (\text{A8})$$

$$M_{21} = -\theta^T P(\rho^+) A_t(\rho) Y(\rho), \quad (\text{A9})$$

$$M_{22} = -\zeta - \theta^T P(\rho^+) \theta. \quad (\text{A10})$$

Using  $e = eP(\rho)Y(\rho)$  and  $W(\rho) = K_t(\rho)Y(\rho)$  leads to:

$$(A_t(\rho)e + \theta)^T P(\rho^+) (A_t(\rho)e + \theta) - e^T P(\rho)e + e^T Q e + e^T K^T(\rho) R K_t(\rho)e \leq -\zeta < 0. \quad (\text{A11})$$

This inequality is a sufficient condition for (C3) with  $V(\cdot)$  as a sub-level of  $\mathbf{X}_f$ .

The fourth and fifth conditions (C4-C5) are verified by the direct application of the Schur complement to Eq. (33a) and Eq. (33b), respectively, using  $W(\rho) = K_t(\rho)Y(\rho)$ . These lead, respectively, to:

$$(\bar{u}_i - I_{n_u, \{i\}} u_r)^2 \geq (I_{n_u, \{i\}} K_t(\rho)) Y(\rho) P(\rho) Y(\rho) (I_{n_u, \{i\}} K_t(\rho))^T, \quad (\text{A12})$$

$$(\bar{x}_j - I_{n_x, \{j\}} x_r)^2 \geq (I_{n_x, \{j\}}) Y(\rho) P(\rho) Y(\rho) (I_{n_x, \{j\}})^T. \quad (\text{A13})$$

Since the maximum normed  $Fe$  of an  $e$  that belongs to some ellipsoid  $e^T P e \leq 1$  is given by  $\sqrt{F^T (P^{-1}) F}$ , it follows that Ineq. (A12) implies that the projection  $I_{n_u, \{i\}} K_t(\rho)e$  (i.e.  $i$ -th control signal) is upper-bounded, in norm, by  $\bar{u}_i - I_{n_u, \{i\}} u_r$ , which satisfies (C4). Analogously, Ineq. (A13) ensures that the projection  $I_{n_x, \{j\}} x$  (i.e.  $j$ -th state) is norm-bounded by  $\bar{x}_j - I_{n_x, \{j\}} x_r$ , which satisfies condition (C5). This concludes the proof.  $\square$

## A.2 Proof of Proposition 1 (Recursive Feasibility)

Let Assumption 5 hold. Consider there exists a solution  $Y(\rho)$ ,  $W(\rho)$  to Theorem 4 which generate a complete terminal state-feedback law  $u_t = K_t(\rho)(x - g_x(y)) + g_u(y)$  and a TRPI set  $\Gamma(\rho)$ . Furthermore, consider an initial state condition  $x(0) \in \mathcal{X}$ , with a corresponding initial scheduling variable  $\rho(0) \in \mathcal{P}$  and a scheduling trajectory estimate  $\hat{P}_0$ . Recall that  $\rho(N_p - 1|0)$  denotes the last entry of this vector.

Then, denote  $(V_0^*, y_a^*)$  as the optimal solution of the NMPC optimisation from Eq. (24), related to these initial conditions. Furthermore, consider  $\hat{X}_0^*$  as the corresponding optimal state sequence, from which the first entry is  $x^*(0|0) = x(0)$ . It is implied that  $(x^*(N_p|0), y_a^*) \in \Gamma(\rho(N_p - 1|0))$ .

Let the successor state be defined as  $x^+ = A(f_\rho(x))x + B(f_\rho(x))K_t(f_\rho(x))(x - g_x(y_a^*))$ . Since the scheduling parameters' deviations are bounded (Assumption 2) and the estimation error is also bounded (Lemma 2), we know that the following scheduling sequence estimate  $\hat{P}_1$  has its last entry  $\rho(N_p|1)$  close to  $\rho(N_p -$

$1|0)$ , that is:  $(\rho(N_p|1) - \rho(N_p - 1|0)) \in \mathcal{Q} \subset \delta\mathcal{P}$ . Complementary, consider  $\tilde{y}_a^+ = y_a^*$ , and  $V_1 = \left[ (v^*(1|0))^T, \dots, (v^*(N_p - 1|0))^T, (K_t(\rho(N_p|0))(x^*(N_p|0) - g_x(y_a^*)) + g_u(y_a^*) - K_\pi x^*(N_p|0))^T \right]^T$ .

Then, the predicted sequence of candidate states for the successor step is given by:  $\hat{X}_1 = \left[ (x^*(1|0))^T, \dots, (x^*(N_p|0))^T, x_t^T(N_p + 1|0) \right]^T$ , where the last entry  $x_t(N_p + 1|0) = A(\rho(N_p - 1|0)x^*(N_p|0) + B(\rho(N_p - 1|0))K_t(\rho(N_p - 1|0))(x^*(N_p|0) - g_x(y_a^*)) + B(\rho(N_p - 1|0))g_u(y_a^*))$ . This last successor state implies in  $x_t(N_p + 1|0) - g_x(y_a^*) = A_t(\rho(N_p - 1|0))(x^*(N_p|0) - g_x(y_a^*))$  and, since  $(x^*(N_p|0), y_a^*) \in \Gamma(\rho(N_p - 1|0))$  and  $(\rho(N_p|1) - \rho(N_p - 1|0)) \in \mathcal{Q}$ , it follows that  $(x_t(N_p + 1|0), y_a^+) \in \Gamma(\rho(N_p|1) - \xi_\rho)$ ,  $\forall \xi_\rho \in \mathcal{Q}$ . Therefore, it follows that  $x^*(1|0) = x^+$ , and thus that  $(V_1, \tilde{y}_a^+)$  is a feasible solution for the NMPC optimisation in Eq. (24) at the successor step, i.e.  $k = 1$ .

In order to conclude this recursive feasibility proof, we show that  $x(N_p + 1|1)$  is inside the invariant set, i.e. that  $(x(N_p + 1|1), y_a^*) \in \Gamma(\rho(N_p|1))$  and  $(\rho(N_p|1) - \rho(N_p - 1|0)) \in \mathcal{Q}$ . Once  $x^*(N_p - 1|0) - g_x(y_a^*) = A_t(\rho(N_p - 1|0))(x^*(N_p|0) - g_x(y_a^*))$ , it follows that the corresponding consecutive error is given by:  $e(1 + N_p|1) = A_t(\rho(N_p - 1|0))e(N_p|0) + (x(N_p + 1|1) - x^*(N_p - 1|0))$ , where  $\theta = (x(N_p + 1|1) - x^*(N_p - 1|0)) \in \mathcal{S}(N_p)$ . Thence, from (C2), we have  $x^*(N_p|0) \in \mathbf{X}_f(\rho(N_p - 1|0)) \Rightarrow x(N_p + 1|1) \in \mathbf{X}_f(\rho(N_p|1))$ .

Generically stating, we obtain  $(x^*(k + j|k), v^*(k + j|k)) \in \mathcal{Z}_\pi(j)$  and  $x^*(k + j + 1|k + 1) = A(\rho(k + j|k + 1))x^c(k + j|k + 1) + B(\rho(k + j|k + 1))v^*(k + j|k + 1)$ , where  $x^*(k + j + 1|k) = A(\rho(k + j|k))x^*(k + j|k) + B(\rho(k + j|k))v^*(k + j|k)$ . Since  $x^*(k + j + 1|k) - x^*(k + j + 1|k + 1) \in \mathcal{V}(j) \oplus \diamond(\mathbf{A}\mathcal{S}(j - 1))$ ,  $\forall j \in \mathbb{N}_{[1, N_p - 1]}$ , it holds that  $(x^+(k + j + 1|k), v^*(k + j + 1|k)) \in \mathcal{Z}_\pi(j + 1) = \mathcal{Z}_\pi(j) \ominus (\mathcal{S}(j) \times \{0\})$ .

Finally, from conditions (C1), (C2), (C4), and (C5) from Theorem 2, we can infer that the generated control signal is well defined. Thus, being  $\Gamma$  a parameter-dependent TRPI set for the closed-loop system evolution, the constraints are fulfilled as the horizon slides and the optimisation is indeed recursively feasible. This concludes the proof.  $\square$

### A.3 Proof of Proposition 2 (Tracking Error Convergence)

Let there be a terminal stage cost  $V(\cdot)$  such that Assumption 5 holds. Leveraging from Lemma 1, assume that  $\lim_{k \rightarrow +\infty} \hat{P}_k \rightarrow P_k$ , i.e. the extrapolated scheduling trajectory converges to the **true** scheduling parameter values. Furthermore, let Theorem 1 be satisfied. Note that since  $\ell(x - x_a, u - u_a)$  is a quadratic stage cost  $\|x - x_a\|_Q^2 + \|u - u_a\|_R^2$ ,  $\alpha_\ell$ ,  $\gamma_x$  and  $\gamma_u$  indeed exists.

Next, consider there exists a solution  $Y(\rho)$  to Theorem 4. Then, the closed-loop is stable due to (C3) of Theorem 2, which conversely ensures that

$$\delta V(k) = V(x(k) - x_a(k), \rho(k)) - V(x(k - 1) - x_a(k - 1), \rho(k - 1)) \leq -\|x(k) - x_a(k)\|_Q + \gamma_V(\bar{w}).$$

Here, we assume that  $\rho(k + j|k) = \rho(k + j)$ , for simplicity. If the scheduling uncertainty is considered, one can account for the related model uncertainty  $\Delta(k + j|k)$ , as gives Eq. (20), and, thanks to the robust constraint satisfaction, consider that the deviation of the nominal predicted state trajectory from the real one is bounded.

Assume that  $\lim_{k \rightarrow +\infty} y_a(k) \rightarrow y_a^o$ . Analogously, use  $\lim_{k \rightarrow +\infty} x_a(k) \rightarrow x_a^o := g_x(y_a^o)$ . Then, thanks to the error dynamics in Eq. (28), we obtain:

$$\|x(k) - x_a(k)\|_Q \leq \beta(\|(x(0) - x_a(0))\|, k) + \gamma(\bar{w}).$$

Since  $Q > 0$  (and positive definite) and  $x_a(0) = 0$  (by design), we have  $\|x(k) - x_a(k)\|_Q \geq \|x(k) - x_a(k)\|$  and thus stability is established.

Now we consider the convergence of  $V_O(y_a(k) - y_r)$  such that the limit  $\lim_{k \rightarrow +\infty} y_a(k) \rightarrow y_a^o$  holds. Let us define  $\hat{y}_a = (1 - \alpha)y_a(k) + \alpha y_a^o$ , where  $\alpha \in [0, 1]$  is the optimal solution from Eq. (27). From the convexity of  $V_O(\cdot)$ , we obtain:

$$V_O(\hat{y}_a - y_r) \leq (1 - \alpha)V_O(y_a(k) - y_r) + \alpha V_O(y_a^o - y_r).$$

We can use the Lipschitz continuity of the map  $x_r := g_x(y_r)$  in order to obtain  $\|x_a(k) - \hat{x}_a\| \leq L_x \|y_a(k) - \hat{y}_a\|$ , where  $L_x > 0$  is the Lipschitz constant of  $g_x(\cdot)$ . Consider  $(y_a(k) - \hat{y}_a) = \alpha(y_a(k) - y_a^o)$ .

Since the closed-loop is stable, it follows that the total MPC cost dissipates over time, which implies in:

$$\begin{aligned} V_O(y_a(k) - y_r) - V_O(\hat{y}_a - y_r) &\leq V(x_a(k) - \hat{x}_a) + \gamma_{V_o}(\bar{w}) \\ &\leq a_1 \|x_a(k) - \hat{x}_a\|^\sigma + \gamma_{V_o}(\bar{w}) \leq a_1 (L_x \|y_a(k) - \hat{y}_a\|)^\sigma + \gamma_{V_o}(\bar{w}) \\ &\leq a_1 L_x^\sigma \alpha^\sigma \|y_a(k) - y_a^o\|^\sigma + \gamma_{V_o}(\bar{w}). \end{aligned}$$



Then, from the convexity of  $V_O(\cdot)$ , we have:

$$V_O(\hat{y}_a - y_r) \leq (1 - \alpha)V_O(y_a(k) - y_r) + \alpha V_O(y_a^o - y_r),$$

Thus, since  $\frac{1}{\alpha} > 1$ , we obtain:

$$V_O(y_a(k) - y_r) \leq a_1 L_x^\sigma \alpha^{\sigma-1} \|y_a(k) - y_a^o\|^\sigma + \gamma_{V_O}(\bar{w}) + \left(\frac{1-\alpha}{\alpha}\right) V_O(y_a(k) - y_r) + V_O(y_a^o - y_r).$$

Finally, since  $\lim_{k \rightarrow +\infty} \left(\frac{1-\alpha}{\alpha}\right) V_O(y_a(k) - y_r) \rightarrow \gamma_y(\bar{w})$ , we obtain  $V_O(y_a(k) - y_r) - V_O(y_a^o - y_r) \leq a_1 L_x^\sigma \alpha^{\sigma-1} \|y_a(k) - y_a^o\|^\sigma + \gamma_n(\bar{w})$ , with  $\sigma > 1$  and  $a_1 > 0$  as constant scalars. Taking the limit at both sides of this inequality leads to:

$$V_O\left(\left(\lim_{k \rightarrow +\infty} y_a(k)\right) - y_r\right) \leq V_O(y_a^o - y_r) + \gamma_n(\bar{w}).$$

Note that  $V_O(\cdot)$  is a weighted quadratic cost by definition, thus  $|\|y_a\| - \|y_r\|| \leq \|y_a - y_r\|_T \leq V_O(y_a - y_r)$ . Thus:

$$\begin{aligned} & \left| \left\| \lim_{k \rightarrow +\infty} y_a(k) \right\| - \|y_r\| \right| \leq \|y_a^o - y_r\|_T + \gamma_n(\bar{w}), \\ & -(\|y_a^o - y_r\|_T + \gamma_n(\bar{w})) + \|y_r\| \leq \lim_{k \rightarrow +\infty} \|y_a(k)\| \\ & \leq \underbrace{\|y_a^o - y_r\|_T + \gamma_n(\bar{w}) + \|y_r\|}_{\mu(y_a^o, y_r, \bar{w})}. \end{aligned}$$

Note that, in nominal conditions (reachable reference  $y_r \in \mathcal{Y}$  and null disturbances), we obtain  $\gamma_{V_O}(\bar{w}) = 0$  and  $y_a^o = y_r$ , and thus  $\lim_{k \rightarrow +\infty} V_O(y_a(k) - y_r) \rightarrow 0$ , which means the steady-state target is reached. In the case the reference isn't reachable and there are disturbances, we can only infer that  $\lim_{k \rightarrow +\infty} y_a(k)$  exists within  $(-\mu(y_a^o, y_r, \bar{w}), +\mu(y_a^o, y_r, \bar{w}))$ . Nevertheless, it is implied that  $\lim_{k \rightarrow +\infty} V_O(y_a(k) - y_r)$  is bounded, which means it converges to a neighbourhood of the origin. This concludes the proof.  $\square$

## AUTHOR BIOGRAPHIES



**Marcelo Menezes Morato** is a PhD Candidate in Automation and Systems Engineering at UFSC/Brazil and Automation and Production and UGA/France. He is an ad-hoc referee to multiple journals, including *Automatica*, *Control Engineering Practice*, and *Renewable Energy*. He received the 2019 Green Talents award from the German government for young high-potential scientists due to his relevant research in sustainability.



**Victor Moreira Cunha** is a master candidate in Electrical Engineering at Federal University of Bahia (UFBA), Brasil. He received B.Sc in Electrical Engineering from UFBA in 2019, with an exchange period in EIGSI, La Rochelle. He received the prize for best masters' paper in the Brazilian Congress of Automation (CBA 2020). His main research interests are Robust Model Predictive Control, Nonlinear Control, Optimization and State Estimation.



**Tito Luis Maia Santos** received B.Sc in Electrical Engineering from Federal University of Bahia (Brazil) in 2006, M.Eng. in Electrical Engineering and Dr.Eng. in Automation and Systems in 2008 and 2011 respectively from Federal University of Santa Catarina (Brazil). In 2011, he joined the Department of Electrical and Computing Engineering, Federal University of Bahia (Brazil) as Adjunct Professor (Associate Professor from 2019). He stayed one year as research scholar at the Department of Systems Engineering and Automatics at Seville University (2010/2011) and another year as Visiting Scholar at Process Systems and Control Laboratory at University of California, Berkeley (2017/2018). His main research interests are Model Predictive Control, Stochastic Model Predictive Control, Process Control, Time-delay Systems, and Dead-Time Compensators.



**Julio Elias Normey-Rico** received his Ph.D. degree from University of Seville, Spain, in 1999. He is currently a Full Professor of the Dept. of Automation and Systems Engineering in Federal University of Santa Catarina (UFSC), in Brazil and head researcher of Renewable Energies Research Team (GPER/UFSC), lead research group in this topic in Latin America. He is the director of several research partnerships with energy industries and international cooperation agreements (Argentina, Uruguay, Spain, Chile and Italy). He is the author of over 260 conference and journal papers, and published four book chapters and a full textbook on *Control of Dead-Time Processes* (Springer). He has supervised over 60 PhD/MSc. candidates. He was associated Editor of *Control Engineering Practice* from 2007 to 2018 and Editor of the *International Renewable Energy Congress* since 2014. Since 2000 he integrates the NOC of several national conferences related with automatic control, and recently, he was the General Chair of the IFAC Symposium DYCOPS 2019.



**Olivier Sename** received a Ph.D. degree from Ecole Centrale Nantes in 1994. He is now Professor at the Grenoble Institute of Technology within GIPSA-lab. His main research interests include Linear Parameter Varying systems and automotive applications. He is the (co-)author of 1 book, more than 50 international journal papers, and more than 200 international conference papers. He has led several industrial (Delphi, Renault, Volvo Trucks, JTEKT) and international (Mexico, Italy, Hungary) collaboration projects. He has supervised 30 Ph.D. students.