A fast dissipative robust nonlinear model predictive control procedure via quasi-linear parameter varying embedding and parameter extrapolation

Marcelo Menezes Morato, Julio E Normey-Rico, Olivier Sename

Received: Added at production | Revised: Added at production | Accepted: Added at production

DOI: xxx/xxxx

REGULAR ISSUE ARTICLE

A Fast Dissipative Robust Nonlinear MPC Procedure via qLPV Embedding and Parameter Extrapolation

Marcelo Menezes Morato[1,2] | Julio E. Normey-Rico[1] | Olivier Sename[2]

[1]Departamento de Automação e Sistemas, Universidade Federal de Santa Catarina, Florianópolis, Brazil

[2]Univ. Grenoble-Alpes, CNRS, Grenoble INP⊤, GIPSA-Lab, 38000 Grenoble, France

⊤ 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, a robust Model Predictive Control (MPC) procedure for quasi-Linear Parameter Varying (qLPV) systems is proposed. The novelty resides in considering a recursive extrapolation algorithm to estimate the values of the scheduling parameters along the prediction horizon N_p, which fastens the sluggish performances achieved with the robust qLPV MPCs from the literature. The bounds on the estimation errors of the scheduling parameters through N_p are taken into account by the robust MPC, which solves an online min-max problem: firstly, a constrained Convex Program (CP) is resolved in order to determine the worst-case bound on the cost function and, subsequently, a second constrained Quadratic Program (QP) is solved to minimise this worst-case cost function with respect to a control sequence vector. Since the bounds on the estimation error for the scheduling parameters are usually much smaller than the bounds on the actual scheduling parameter, the conservativeness of the solution is quite reduced. Recursive feasibility and stability of the proposed algorithm are demonstrated with dissipativity arguments given in the form of a Linear Matrix Inequality (LMI) remedy, which determines the zone of attraction for which input-to-state stability is ensured. The nonlinear temperature regulation problem of a flat solar collector is considered as a case study. Using a realistic simulation benchmark, the proposed technique is compared to other robust min-max LPV MPC algorithms from the literature, proving itself efficient while achieving good performances.

KEYWORDS:

Robust Model Predictive Control, Linear Parameter Varying Systems, Dissipativity, Quadratic Programming, Solar Collectors.

1 | INTRODUCTION

Model Predictive Control (MPC) is one of the most widespread control techniques, with many industrial applications[1]. Its implementation is relatively simple, requiring the solution of an online optimisation problem, written in terms of a prediction model and operational constraints. MPC is a sliding-horizon paradigm, which means that the algorithm takes into account the process behaviour along the following N_p steps ahead of each sampling instant; as time evolves, the instants are incremented and the horizon slides forward. For the case of processes with linear time-invariant (LTI) models, a state-feedback predictive control action $u(k)$ can be generated by solving a Quadratic Programming Problem (QP) of the following form:

[0]**Abbreviations:** LPV, Linear Parameter Varying; LTI, Linear Time Invariant; MPC, Model Predictive Control.

Problem 1. LTI MPC

$$\min_{U_k} J_k(U_k) = \min_{U_k} \left(\sum_{k=1}^{N_p} \overbrace{\ell\left(x(k+j|k), u(k+j-1|k)\right)}^{\text{Stage Cost}} \right) \tag{1}$$

$$\text{s.t.} \quad \underbrace{x(k+j) = f_{LTI}(x, u, k+j-1)}_{\text{System Model}}, \ \forall j \in \mathbb{N}_{[1,N_p]}, \tag{2}$$

$$u(k+j-1|k) \in \mathcal{U}, \ \forall j \in \mathbb{N}_{[1,N_p]}, \tag{3}$$

$$x(k+j|k) \in \mathcal{X}, \ \forall j \in \mathbb{N}_{[1,N_p]}, \tag{4}$$

$$x(k+N_p|k) \in \mathbb{X}_f. \tag{5}$$

The model function $f_{LTI}(\cdot)$ is generally expressed as $Ax(k+j-1) + Bu(k+j-1)$, being $x(k)$ the vector of controlled states. It is implied that J_k is a quadratic cost on the vector U_k, which defines the sequence of control actions along the horizon: $U_k = \text{col}\{u(k|k), \dots, u(k+N_p-1|k)\}$. The feasibility sets $\mathcal{U}$ and $\mathcal{X}$ define the admissible values for x and u, considering operational constraints of the controlled process. $\mathbb{X}_f$ is a terminal set for the states. We note that, usually, a terminal (offset) cost $V(x(k+N_p|k))$ is added to J_k.

While with great theoretical value, LTI MPC is restricted to processes represented by LTI models. Nevertheless, the majority of systems is nonlinear and when controlled over larger operating conditions (or when heavily affected by external parameters), the nonlinearities become quantitative and, thus, the previous LTI MPC algorithm must be replaced. Accordingly, the concept of Nonlinear MPC (NMPC) grew on progressively literature since the 00's[2]. The major drawback of NMPC algorithms is that their numerical burdon is much greater than what is required in an LTI setting (which renders QPs), since the nonlinear model predictions generate a Nonlinear Programming Problems (NPs), which are not at all easy to solve (and could even be non-convex). Regarding Problem 1, the NMPC version would have a nonlinear system model, i.e. $x(k+j) = f\left(x(k+j-1), u(k+j-1)\right)$, with $f(\cdot)$ as a nonlinear map. Due to this constraint, NMPC algorithms were hardly able to be computed for real-time embedded applications, requiring excessive online computational capacities and thus violating the sampling period constraint[3].

In order to address this complexity drawback of "full-blown" NMPC strategies, literature points out to the option of considering quasi-/Linear Parameter Varying (qLPV/LPV) model structures to embed the nonlinear process. LPV models maintain the linearity property along the input/output channels, since they are linear in the state space, while nonlinear in the parameter space. The LPV state transition depends ona vector of scheduling parameters denoted $\rho \in \mathcal{P}$, which is known and bounded at each sampling instant. The LPV toolkit has progressively become popular to model process with complex dynamics, with a variety of successful results[4,5,6,7]. Many nonlinear processes can be described within a qLPV formalism, as long as Linear Differential Inclusion (LDI) is respected[8,9].

1.1 | Related Works

There are quite a few MPC algorithms specifically conceived for nonlinear systems represented with qLPV/LPV models. A detailed review on this topic has been recently presented[10]. Synthetically, we can sort out these works into two main different categories: (a) robust MPC methods[11,12,13,14,15], which treat the future scheduling variables as bounded uncertainties[1]; and (b) sub-optimal methods[20,21,22,23,24] that replace the actual future scheduling trajectory $\rho(k+j)$ by a prediction guess $\hat{\rho}(k+j)$, thus yielding programs with QP-alike burden.

With regard to these LPV MPC scheme, we highlight some issues:

- The robust framework resides in solving the MPC optimisation w.r.t. the worst-case cost that is derived with the uncertainty caused by the variation of the scheduling variables along the prediction horizon N_p. These algorithms are usually named "min-max" because the cost function is found through $\arg\max_{\rho \in \mathcal{P}} J_k(\cdot)$, and then minimised with respect to U_k. The original papers[11,12] assume that the future scheduling parameters $\rho(k+j)$ vary arbitrarily within the scheduling set $\mathcal{P}$. More recently, many works[13,14,15] have demonstrated that the min-max procedure can be simplified for the case of bounded rates of parameter variations (i.e. $\delta\rho(k) = \rho(k) - \rho(k-1)$ bounded), which is standard in LPV applications. Nevertheless,

[1] We also mention the application of tube-based MPC design. This robust synthesis is able to reduce the numerical complexity of the algorithm[16] by preparing offline tubes that bound the system trajectories. The main advantage is of these methods[17,18,19] is that the numerical toughness grows well-behaved (linearly, in many cases) with the size of the prediction horizon N_p. Nevertheless, tube-based MPC schemes often result in quite conservative performances.

the major drawbacks with these robust tools are that: (i) they are not implementable for real-time applications, due to the complexity of solving the maximisation problem for the whole scheduling set $\mathcal{P}$ (or, the scheduling variation set $\delta\mathcal{P}$); and (ii) the formulations through offline preparations (such as the tube paradigm) are often quite hard to design; the synthesis procedure is usually not trivial and hard to understand, which hinders industrial acceptance.

- The sub-optimal methods can be conceived through iterative[20] or recursive[23] estimation procedures regarding $\rho(k+j)$. The major drawback is that the resulting QPs may find local minima of the original NP, which conversely may lead to poor or insufficient performances. Despite being able to run in real-time, these algorithms lack performance guarantees, which may compromises the overall results.

There is an evident gap in the literature: the lack of robust nonlinear MPC algorithms, with direct and simple-enough online implementation, using the qLPV embedding approach. Therefore, in this paper, we seek a formulation that is able to run in real-time and, yet, maintains optimality concerns, leading to good performance. The bottleneck is for the algorithm to run in the range of milliseconds, whilst taking into account the model nonlinearities.

1.2 | Contributions

In several of the listed papers[20,21,23], estimation algorithms have been used regarding the future values of the scheduling parameters, along the prediction horizon. We progress on this idea by using a recursive extrapolation algorithm with bounded estimation errors $\xi_\rho \in \mathcal{E} \subset \mathcal{P}$. Then, the bounds are used to formulate a robust min/max MPC. Motivated by the previous discussion and the literature gap, our contributions are as follows:

- We present an error-bounded recursive estimation algorithm for the extrapolates of the scheduling parameters along the horizon, i.e. $\rho(k+j)$ for $j = 1, \dots, N_p - 1$;
- We develop an min/max robust qLPV MPC framework, the use of the extrapolation of ρ to make model-based predictions;
- A dissipativity inequality formulation for input-to-state stability (ISS) analysis is applied. This analysis also serves to verify the recursive feasibility property of the proposed algorithm. An LMI-solvable remedy to estimate the ISS zone is obtained.

Remark 1. Some previous papers[21,25] had already conceived NMPC formulations via qLPV embedding. Nonetheless, in the prior, optimality concerns were not ensured, since the extrapolation error is not taken into account in the design procedure. Moreover, these works reframe the NP into a 2^{nd} OCP version (or as a series of sequential QPs), which is also not easy to solve. Therefore, in this paper, we provide a formulation for the problem through, at most, two consecutive Convex Programming Problems (CPs), in order to achieve comparable results (in terms of computational load).

1.3 | Organisation

This paper is structured as follows. In Section 2, the preliminaries and formalities are presented, especially regarding how nonlinear processes can be embedded into a qLPV representation through LDI. Section 3 presents the formulation for recursive extrapolation of the qLPV scheduling parameters along the prediction horizon. Section 4 formulates the robust min-max MPC. Section 5 shows the application of the dissipativity constraints to verify stability and recursive feasibility of the proposed algorithm. Section 6 details the considered case study, of nonlinear temperature control in solar collectors, for which the proposed strategy is tested. Section 6.2 exhibits the achieved simulation results regarding this case study. General conclusions are drawn in Section 8.

1.4 | Notation

The set of nonnegative real number is denoted by $\mathbb{R}_+$, whist the set of nonnegative integers including zero is denoted by $\mathbb{N}$. 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 $\mathbb{I}_j$; $\text{col}\{a, b, c\}$ denotes the vectorisation (collection) of the entries and $\text{diag}\{v\}$ denotes the diagonal matrix generated with the line vector v.

The value of a given variable $v(k)$ at time instant $k+j$, computed based on the information available at instant k, is denoted as $v(k+j|k)$. $\mathcal{K}$ refers to the class of positive and strictly increasing scalar functions that pass through the origin. A given function $f : \mathbb{R} \to \mathbb{R}$ is of class $\mathcal{K}$ if $f(0) = 0$ and $\lim_{\xi \to +\infty} f(\xi) \to +\infty$. A real-valued scalar function $\phi : \mathbb{R}_+ \to \mathbb{R}_+$ belongs to class $\mathcal{K}_\infty$ if it belongs to class $\mathcal{K}$ and it is radially unbounded (this is $\lim_{s \to +\infty} \phi(s) \to +\infty$. A function $\beta : \mathbb{R}_+ \times \mathbb{R}_+ \to \mathbb{R}_+$ belongs to class $\mathcal{KL}$ if, for each fixed $m \in \mathbb{R}_+$, $\beta(\cdot, m) \in \mathcal{K}$ and, for each fixed $s \in \mathbb{R}_+$, $\beta(s, \cdot)$ is non-increasing and holds for $\lim_{m \to +\infty} \beta(s, m) = 0$. $\mathcal{C}^n$ denotes the set of all compact convex subsets of $\mathbb{R}^n$. A convex and compact set $X \in \mathcal{C}^n$ with non-empty interior, which contains the origin, is named a PC-set. A subset of $\mathbb{R}^n$ is denoted a polyhedron if it is an intersection of a finite number of half spaces. A polytope is defined as a compact polyhedron. A polytope can be analogously represented as the convex hull of a finite number of points in $\mathbb{R}^n$. A hyperbox is a convex polytope where all the ruling hyperplanes are parallel with respect to their axes.

In Appendix A, we provide a Nomenclature and Symbology table, which details all acronyms, variables, indexes, and symbols used in this paper. In the sequel, all the necessary definitions for further development are recalled.

1.5 | Definitions

Definition 1. Nonlinear Programming Problem
Consider an arbitrary real-valued nonlinear function $f_c(x_c)$. A Nonlinear Programming Problem (NP) finds the vector x_c that minimises $f_c(x_c)$ subject to $g_i(x_c) \leq 0$, $h_j(x_c) = 0$ and $x_c \in \mathcal{X}_c$, where g_i and h_j are also nonlinear.

Definition 2. Convex Programming Problem
A Convex Programming Problem is a linearly constrained optimisation problem of a convex function. A CP is a particular type of nonlinear programming problem, for which the function $f_c(x_c)$ is inherently convect with respect to x_c and the constraints $g_i(x_c) \leq 0$ and $h_j(x_c) = 0$ are linear on x_c. Any CP can be formulated as $x_c^\star = \arg\min_{x_c \in \mathcal{X}_c} f_c(x_c)$ subject to constraints $A_{ineq} x_c \leq b_{ineq}$ and $A_{eq} x_c = b_{eq}$. It follows that: $A_{ineq} \in \mathbb{R}^{m_c \times n_c}$, $A_{eq} \in \mathbb{R}^{m_c \times n_c}$, $b_{ineq} \in \mathbb{R}^{m_c}$ and $b_{eq} \in \mathbb{R}^{m_c}$. The solution $x_c^\star$ to this kind of problem is found through interior-point algorithms and gradient-based methods.

Definition 3. Quadratic Programming Problem
A Quadratic Programming Problem (or simply Quadratic Problem, QP) is a linearly constrained mathematical optimisation problem of a quadratic function. A QP is a particular type of convex programming problems. The quadratic function may be defined with respect to several variables, all of which may be subject to linear contraints. Considering a $c \in \mathbb{R}^{n_c}$ gradient vector and a symmetric Hessian matrix $Q_c \in \mathbb{R}^{n_c \times n_c}$, the goal of a QP is to determine the vector $x_c \in \mathbb{R}^{n_c}$ that minimises a regular quadratic function of form $\frac{1}{2}\left(x_c^T Q_x x_c + c^T x\right)$ subject to constraints $A_{ineq} x_c \leq b_{ineq}$ and $A_{eq} x_c = b_{eq}$. The solution $x_c^\star$ to this kind of problem is found by many solvers seen in the literature, based on Interior Point algorithms, quadratic search, etc.

Definition 4. Closed-Loop Input-to-State Stability[26]
Consider a generalised discrete-time nonlinear plant regulated under a state-feedback closed-loop structure. This closed-loop has its dynamics ruled by:

$$x(k+1) = \mathcal{F}_{cl}\left(x(k), w(k)\right), \tag{6}$$

where $x(k)$ is the state of the system regulated under closed-loop, while $w(k)$ is a bounded load disturbance variable such that $||w(k)|| \leq w_{max} \, \forall \, k \in \mathcal{N}$. Then, this nonlinear system is said to be input-to-state stable in closed-loop if there exists a pair of $\mathcal{K}$-functions $\beta(\cdot, \cdot)$ and $\sigma(\cdot)$ such that the following inequality holds: $||x(k)|| \leq \beta(x(0), k) + \sigma(w_{max})$.

Definition 5. Regional ISpS (ISS)[27]
The system in Eq. (6) is said to be ISpS in $\mathcal{X}_{RISS} \subseteq \mathbb{R}^{n_x}$ if there exists a $\mathcal{KL}$-function β_r, a $\mathcal{K}$-function σ_r and a scalar $d_r \in \mathbb{R}_+$ such that, for each $x_0 \in \mathcal{X}_{RISS}$, all $w(k)$ and all $v(k)$, it holds that the corresponding state trajectory satisfies the following inequality $||x(k)|| \leq \beta_r(x(0), k) + \sigma_r(||v(k-1)||) + d_r \, \forall k \in \mathcal{Z}_{[1,+\infty]}$. If the origin lies in the interior of $\mathcal{X}_{RISS}$ and the inequality also holds for $d_r = 0$, the system is said to be ISS in $\mathcal{X}_{RISS}$. Note that the presence of a non-null d_r in the above inequality is a conservative solution, i.e. for a greater $||d_r||$, one finds a smaller ISpS region $\mathcal{X}_{RISS}$.

Definition 6. Recursive Feasibility
An optimisation algorithm is said to be recursively feasible inside the feasibility set if, for any starting condition $x(k_0) = x_0 \in \mathcal{X}_{ISS}$, the optimisation is feasible and remains feasible throughout the following instants $k > k_0$.

Definition 7. Robust Positively Invariant Set[28]
A set $\mathbb{X}_f \subset \mathcal{X}$ is said to be a robust positively invariant (RPI) set for the considered qLPV embedded system given in Eq. (9) with $u(k) = I_1 U_k^\star$ if, for the whole state evolution sequence X_k and for any $x(k) \in \mathbb{X}_f$ and $\rho(k) \in \mathcal{P}$, it follows that $x(k+1) \in \mathbb{X}_f$. This is, the state evolution X_k always lies inside the RPI set $\mathbb{X}_f$.

Definition 8. D/G Scalings[29]
Let $\Theta = \text{diag}\{\rho_1(k)\mathbb{I}_{\text{size}\{\rho_1\}}, \dots, \rho_{n_\rho}(k)\mathbb{I}_{\text{size}\{\rho_{n_\rho}\}}\}$, with $\rho_j \in \mathbb{R}$. Accordingly, the set of D/G-Scalings is defined as follows:

$$\mathcal{M}_{D/G} := \left\{ \begin{bmatrix} M_{11} & M_{12} \\ M_{12}^T & -M_{11} \end{bmatrix} : M_{11} = M_{11}^T \succ 0, M_{12} + M_{12}^T = 0, M_{11}\Theta = \Theta M_{11}, M_{12}\Theta = \Theta M_{12} \right\}.$$

2 | PRELIMINARIES AND FORMALITIES

2.1 | The Nonlinear System and qLPV Embedding

Consider the following generic discrete-time nonlinear system:

$$x(k+1) = f(x(k), u(k), w(k)), \tag{7}$$

where $k \in \mathbb{N}$ represents the sampling instant, $x : \mathbb{N} \to \mathcal{X} \subset \mathbb{R}^{n_x}$ represents the system states, $u : \mathbb{N} \to \mathcal{U} \subset \mathbb{R}^{n_u}$ is the vector of control inputs and $w : \mathbb{N} \to \mathcal{W} \subseteq \mathbb{R}^{n_w}$ stands for load disturbance variables.

This process satisfies the following Assumptions:

Assumption 1. The nonlinear map $f : \mathcal{X} \times \mathcal{U} \times \mathcal{W} \to \mathcal{X}$ is continuous and continuously differentiable with respect to x, i.e. class $\mathcal{C}^\infty$

Assumption 2. The (box-type) sets $\mathcal{X}$ and $\mathcal{U}$ define the feasibility constraints for the system states and the control vector, delimited by the operational (physical) limitations of these variables. These sets yield ultimate bounds on x and u, as follows:

$$\mathcal{X} := \left\{ x \in \mathbb{R}^{n_x} : ||x|| \leq \overline{x} \right\}, \quad \mathcal{U} := \left\{ u \in \mathbb{R}^{n_u} : ||u|| \leq \overline{u} \right\}.$$

Assumption 3. The set $\mathcal{W}$ defines the load disturbances. For regularity purposes, we consider that $\mathcal{W}$ is *a priori* an open set.

Assumption 4. The states are measurable at all sampling instants $k \in \mathbb{N}$, which means that control can be formulated under a state-feedback fashion $u(k) = \kappa(k)x(k)$.

Assumption 5. This nonlinear system satisfies the Linear Differential Inclusion property.

The nonlinear dynamics in Eq. (7) can be given through a qLPV realisation if Assumption 5 is satisfied. The LDI property is as follows[8]: suppose that, for each x, u and w and for every sampling instant k, there exists a matrix $H(x,u,k) : \mathcal{X} \times \mathcal{U} \times \mathbb{N} \to \mathcal{H}$ such that the following equality holds:

$$\left[f(x(k), u(k), w(k)) \right] = H(x,u,k) \begin{bmatrix} x(k) \\ u(k) \\ w(k) \end{bmatrix}. \tag{8}$$

In this case, LDI is verified and, thus, it follows that:

$$G := \begin{cases} x(k+1) = A(\rho(k))x(k) + B(\rho(k))u(k) + B_w(\rho(k))w(k) \\ \rho(k) = f_\rho(x(k), u(k)) \in \mathcal{P} \end{cases}, \tag{9}$$

is a qLPV realisation[2] of Eq. (8) with $f_\rho : \mathcal{X} \times \mathcal{U} \to \mathcal{P} \subset \mathbb{R}^{n_\rho}$ representing the endogenous nonlinear scheduling proxy. Note that $\rho(k)$ is bounded and known online at each instant k, but generally unknown for any future instant $k+j \; \forall j \in \mathbb{N}_{[1,\infty]}$. Ultimate bounds are considered upon ρ, as follows:

$$\mathcal{P} := \left\{ \rho \in \mathbb{R}^{n_\rho} \,|\, ||\rho|| \leq \overline{\rho} \right\}.$$

[2]There is a conceptual difference between proper LPV and qLPV models: for the first class, the scheduling parameters are generally exogenous variables, such as external activation signals, completely independent from x and u; in the second class, there exists some proxy to compute the scheduling parameters as an endogenous (possibly nonlinear) map of states and inputs $f_\rho(x(k))$. We discern qLPV models for LPV ones in order to highlight that the considered embedding has an inherent endogenous scheduling proxy $f_\rho(\cdot)$.

Remark 2. The LDI property, as expressed through Eq. (8), is satisfied for closed sets $\mathcal{X}$ and $\mathcal{U}$ and not for the whole spaces $\mathbb{R}^{n_x}$ and $\mathbb{R}^{n_u}$. For this reason, $\mathcal{H}$ is a subset of $\mathbb{R}^{(n_x)\times(n_x+n_u+n_w)}$.

Remark 3. Through the sequel, for simplicity, we drop the dependency of f_ρ on w and u, simply taking $\rho(k) = f_\rho(x(k))$. Nonetheless, we stress that all developments presented in the sequel can be easily extended to broader case. For the sake of simplicity, we also drop the parameter dependency from B_w, i.e. $B_w(\rho(k)) = B_w$. Note that this can always be done via the inclusion of the parameter dependency into the load disturbance signal, e.g. $B_w(\rho(k))w(k) = B_{w_2}w_2(k)$ with $w_2(k) = f_w(\rho(k), w(k))$.

Complementary, we consider that Assumption 6 is satisfied[3]. This Assumption does not compromise the proposed approach; it serves only to analytically account for model-process mistmatch uncertainties. We also consider that the qLPV embedding of Eq. (9) satisfies Assumptions 7 (local Lipschitz property of $f_\rho(\cdot)$), 8 (bounded rates of variation for ρ) and 9 (stabilisability).

Assumption 6. Matrices $A(\rho(k))$ and $B(\rho(k))$ are affine-dependent on $\rho(k)$, as in: $A(\rho(k)) = A_0 + A_1\rho(k)$ and $B(\rho(k)) = B_0 + B_1\rho(k)$.

Assumption 7. The nonlinear scheduling parameter map $f_\rho : \mathcal{X} \to \mathcal{P}$ agrees to a local Lipschitz condition around any arbitrary point $x \in \mathcal{X}$, this is:

$$||f_\rho(x) - f_\rho(\hat{x})|| \leq \Gamma||(x - \hat{x})||\,, \forall\, x \in \mathcal{X}\,, \forall\, \hat{x} \in \mathcal{X}\,, \tag{10}$$

where the smallest constant Γ that satisfies Eq. (10) is known as the Lipschitz constant for $f_\rho(\cdot)$.

Assumption 8. The deviation of the scheduling parameters is bounded, i.e. $\delta\rho(k) = (\rho(k) - \rho(k-1)) \in \delta\mathcal{P}, \forall k \in \mathbb{N}$.

Assumption 9. The open-loop qLPV model $(A(\rho(k)), B(\rho(k)))$ is structurally stabilizable for all $\rho \in \mathcal{P}$.

2.2 | Model-based Predictions through qLPV embedding

Since the qLPV embedding in Eq. (9) retains the linearity property from inputs to outputs, it is possible to formulate numerically-efficient design procedures using these models. While LPV control is standard in both state-feedback and dynamics output-feedback formulations[6,30,31,32], the design predictive control algorithms for LPV systems is not trivial, since solving the inherent constrained $\arg\min_{U_k} J$ optimisation problem requires the knowledge of future values for the scheduling parameter.

The key difficulty of LPV MPC algorithms is that the prediction of the future state variables $x(k+j|k), \forall j \in \mathbb{N}_{[1,N_p]}$ gets rather complicated from the two-steps ahead prediction onwards, since nonlinear terms appear:

$$\left\{ x(k+2|k) = A(\rho(k+1))A(\rho(k))x(k) + A(\rho(k+1))B(\rho(k))u(k|k) \right. . \tag{11}$$

Note that a cross-product term between different instances of parameter-dependent matrices arises. For the considered qLPV embedding, $\rho(k+j)$ is a function of the states, which converts the previous equation to:

$$\left\{ x(k+2|k) = A(f_\rho(x(k+1|k)))A(\rho(k))x(k) + A(f_\rho(x(k+1|k)))B(\rho(k))u(k|k) \right. , \tag{12}$$

which still is nonlinear and, therefore, it follows that Problem 1 formulated with a qLPV embedded model remains an NP.

We can expand[20] the state predictions from Eq. (11) for the whole sequence of the future states within the prediction horizon. Thus, the complete future state sequence, denoted X_k, is expressed as regular bilinear function of the sampled state $x(k)$, the control sequence and the "scheduling sequence" P_k. This yields:

$$X_k = A_x(P_k)x(k) + B_x(P_k)U_k\,, \tag{13}$$

where the scheduling sequence and the sequence of future state variables are respectively given by:

$$P_k = \left[\, \rho(k) \;\; \rho(k+1) \;\; \rho(k+2) \;\; \ldots \;\; \rho(k+N_p-1) \,\right]^T . \tag{14}$$

$$X_k = \left[\, x(k+1|k) \;\; x(k+2|k) \;\; \ldots \;\; x(k+N_p|k) \,\right]^T . \tag{15}$$

[3] We stress that any other kind of parameter dependency could be used (polynomial, Linear Fractional Transformations, etc.)

Matrices $A_x(P_k)$ and $B_x(P_k)$ in Eq. (13) are given as follows:

$$A_x = \begin{bmatrix} A(\rho(k)) \\ A(\rho(k+1))A(\rho(k)) \\ \vdots \\ A(\rho(k+N_p-1))A(\rho(k+N_p-2))\dots A(\rho(k)) \end{bmatrix}, \tag{16}$$

$$B_x = \begin{bmatrix} B(\rho(k)) & 0 & \dots \\ A(\rho(k+1))B(\rho(k)) & B(\rho(k+1)) & \dots \\ & \vdots & \\ A(\rho(k+N_p-1))\dots A(\rho(k+1))B(\rho(k)) & A(\rho(k+N_p-1))\dots A(\rho(k+2))B(\rho(k+1)) & \dots \end{bmatrix}. \tag{17}$$

2.3 | Extrapolating the Scheduling Sequence

As formulated in previous works[20], guessing mechanisms can be used to estimate the values of P_k. In fact, different formulations for these extrapolation algorithms have been previously presented: through recursive Least-Square minimisation[23] or iterative guessing through the actual qLPV proxy[4] $P_k = f_\rho(X_k)$, computed via sequential QPs, ensuring refined extrapolations as k increases.

In this paper, we follow this same idea, considering an algorithm that estimates the scheduling sequence at each instant, with bounded estimation errors. This is, for $\hat{P}_k$ as the extrapolation guess, the algorithm provides:

$$\hat{P}_k = \text{col}\{\hat{\rho}(k+j)\}\ \forall j \in \mathbb{N}_{[0,N_p-1]} \text{ with} \tag{18}$$

$$\xi_\rho(k+j) = \rho(k+j) - \hat{\rho}(k+j)\,\forall j \in \mathbb{N}_{[0,N_p-1]} \text{ and} \tag{19}$$

$$\xi_\rho(k+j) \in \mathcal{E} := \left\{ \xi_\rho \in \mathbb{R}^{n_p} \,|\, ||\xi_\rho|| \leq \xi_\rho^{\text{bound}} \right\}. \tag{20}$$

Generally, these algorithms[20,23] yield horizon-increasing errors $||\xi_\rho(k+j+1)|| \geq ||\xi_\rho(k+j)||$ for $j \in \mathbb{N}_{[0,N_p-2]}$, due to the fact the more information is available regarding the present instant than the future ones, which depend on future variables which haven't yet been defined.

2.4 | The Generated Sub-Optimal MPC

Regarding the application of the MPC algorithm in Problem 1 simply based on an extrapolation guess $\hat{P}_k$, the following finite-horizon cost is considered:

$$J_k = \left(\sum_{j=1}^{N_p} (\ell(x(k+j), u(k+j-1))) \right), \tag{21}$$

where $\ell(\cdot)$ is the main stage cost. Note that we disregard the use of a terminal cost $V(\cdot)$. This cost function is formalised according to traditional tuning methods[33], with a quadratic form:

$$\ell(\cdot) = (x(k+j|k))^T Q(x(k+j|k)) + (u(k+j-1|k))^T R(u(k+j-1|k))\, j \in \mathbb{N}_{[1,N_p]}. \tag{22}$$

Sub-optimal MPC algorithms[23] have been applied to regulate nonlinear systems with qLPV models. Accordingly, the following constrained QP is applied[5]:

$$\min_{U_k} \quad J_k \tag{23}$$

$$\text{subject to} \quad X_k = A_x(\hat{P}_k)x(k) + B_x(\hat{P}_k)U_k, \tag{24}$$

$$x(k+j) \in \mathcal{X}\,\forall\, j \in \mathbb{N}_{[1,N_p]}, \tag{25}$$

$$u(k+j-1) \in \mathcal{U}\,\forall\, j \in \mathbb{N}_{[1,N_p]}, \tag{26}$$

$$\rho(k+j) = \hat{P}_k\{j\}\,\forall\, j \in \mathbb{N}_{[0,N_p-1]}, \tag{27}$$

[4] Abusive notation is used. In fact, each $\rho(k+j)$ depends individually on $f_\rho(x(k+j))$, which converts into $P_k = \text{col}\{f_\rho(x(k+j))\}$ with $j \in \mathbb{N}_{[0,N_p-1]}$.

[5] Note that $\hat{P}_k\{j\}$ denotes the j-th entry of the scheduling sequence vector $\hat{P}_k$.

This QP can lead to insufficient performances because local minima can be found since, although the qLPV embedding equivalently represents the nonlinear dynamics, the scheduling parameters are unknown along the horizon. Thus, the model predictions conceived with $\hat{P}_k$ are inaccurate, implying in an prediction error on X_k. Therefore, we proceed by adapting this QP in order to include the bounds on the estimation error of the scheduling sequence $\xi_\rho(k+j)$ such that optimality can be maintained.

In general, optimisation problems are represented with Hessian, gradients and inequalities. Accordingly, the previous QP can be re-stated as:

$$\begin{aligned} U_k^\star &= \arg\min_{U_k} \left(\frac{1}{2} U_k^T H(\hat{P}_k) U_k - U_k^T g(\hat{P}_k, x(k)) \right), \\ \text{s.t.} \quad & A_{ineq} U_k \leq b_{ineq}(k), \\ & C_{ineq} U_k = 0, \end{aligned} \tag{28}$$

being $U_k^\star$ the control sequence solution. In this formulation, $H(\hat{P}_k)$ is the Hessian of the quadratic cost function J_k and $g(\hat{P}_k, x(k))$ is its gradient.

The MPC policy that results from the online solution of Eq. (28) is generated under a paradigm of a moving-window horizon, which slides along k as time evolves. This means that that at instant k the control sequence $U_k^\star$ is computed considering the system behaviour within the next N_p steps. At the following instant, $k+1$, the problem $\min J_{k+1}$ is solved considering the performances for N_p samples ahead of $k+1$, computing $U_{k+1}^\star$, and so forth. The control policy at each instant is the first entry of the solution the QP, this is:

$$u(k) = \overbrace{\left[\mathbb{I}_{n_u} \; 0_{n_u} \; \dots \; 0_{n_u} \right]_{n_u \times (n_u N_p)}}^{I_1} U_k^\star = \kappa(k) x(k). \tag{29}$$

2.5 | Terminal Ingredients and Dissipativity Constraints

The concept of input-to-state stability (ISS) is used to verify stability and also allows control synthesis for nonlinear systems. We use the concept of ISS generalised for discrete-time nonlinear processes[26], as presented in the prequel. We are concerned with ISS since the considered MPC generates a state-feedback control law, which means that the states should be stabilised.

Recent results regarding ISS and Input-to-State Practical Stability (ISpS), which is a weaker property[6], have been presented regarding min-max nonlinear MPCs, see[34,35,36]. In general, many min-max MPC methods are not able ensure ISS (but simply ISpS) because the effect of non-null disturbance inputs is taken into account by the min-max procedure even if the disturbance vanishes in reality. Anyhow,[27] demonstrates that only a local upper bound on the min-max cost function J_k (instead of a global one, which is more costly to demonstrate) is sufficient to ensure ISS. In this paper, we build from these previous results, specially concerning the feasibility property of the maximisation procedure.

We stress that an ISS sytem is asymptotically stable in the absence of inputs u and w or if the inputs are time-decaying. Note that if the inputs are merely bounded, the evolution of the system states are ultimately bounded to a set whose size depends on the bounds of the inputs, which is quite logical.

In order to verify that the MPC algorithm ensures closed-loop ISS and recursive feasibility of the optimisation procedure, there are two main options:

- To use the so-called "terminal ingredients"[37], this is: verify some conditions with respect to the terminal stage cost $V(\cdot)$ and the terminal constraint $\mathbb{X}_f$. Essentially, the terminal set must be an RPI set, while $\ell(\cdot)$ must be $\mathcal{K}$-class lower bounded, and $V(\cdot)$ must be $\mathcal{K}$-class upper bounded. A Lyapunov-decreasing inequality must also be satisfied.

- The second option is to use dissipativity arguments[38]. The main characteristic of this second stability-verification path is that LMI formulations are yielded for *a priori* verification. This is the path followed in this paper, following the lines of a previous works[38,29].

[6] ISpS does not impose asymptotic stability for null disturbance inputs.

3 | THE RECURSIVE EXTRAPOLATION ALGORITHM

In this Section, we propose a simple method to recursively construct the extrapolation for the scheduling sequence along the horizon. The method resides in a first-order Taylor expansion of the scheduling proxy $f_\rho(x(k))$ around the state deviation. We denote $\Delta x(k+j) = (x(k+j+1) - x(k+j))$ as the incremental state deviation, which is naturally bounded due to the bounds on $x(k)$, i.e. $\Delta x(k+j) \in \Delta\mathcal{X}\ \forall\, j \in \mathbb{N}$. We consider that the following Assumptions are satisfied.

Assumption 10. The state deviations are ultimately bounded. This is:

$$\begin{aligned}\Delta x(k) &= A(\rho(k))x(k) + B(\rho(k))u(k) - x(k) = \left(A(\rho(k)) - \mathbb{I}_{n_x}\right)x(k) + B(\rho(k))u(k)\,.\\ \|\Delta x(k)\| &= \|\left(A(\rho(k)) - \mathbb{I}_{n_x}\right)x(k) + B(\rho(k))u(k)\| \leq \underbrace{\|(A(\overline{\rho}) - \mathbb{I}_{n_x})\overline{x} + B(\overline{\rho})\overline{u}\|}_{\overline{\Delta x}}\,.\end{aligned}$$

Assumption 11. The static map $f_\rho(\mu)$ can be approximated by a first-order Taylor expansion for any arbitrary μ:

$$f_\rho(\mu) = f_\rho(\mu)|_{\mu^\star} + \left.\frac{\partial f_\rho}{\partial \mu}\right|_{\mu^\star}(\mu - \mu^\star) + \xi_\rho\,, \tag{30}$$

being $\mu^\star$ the expansion point and ξ_ρ a residual noise which inherits the discrepancy between the real static map and its approximate.

Assumption 12. The static map $f_\rho(\cdot)$ is class $\mathcal{C}^1$, i.e. first-order differentiable with respect to x, for all x points in $\mathcal{X}$.

Assumption 13. The differentiation function $f_\rho^\partial(k)$ is ultimately bounded.

Thus, on the basis of Assumption 11, the following expression is written:

$$f_\rho(x(k+j)) = f_\rho(x(k+j-1)) + \xi_\rho(k+j-1) + \underbrace{\left.\frac{\partial f_\rho}{\partial x(k+j)}\right|_{x(k+j-1)}}_{f_\rho^\partial(k+j-1)}\Delta x(k+j-1)\,. \tag{31}$$

Expanding Eq. (31) along a prediction horizon of N_p steps and embedding it to $\rho(k+1) = \rho(k) + f_\rho(x(k+1)) - f_\rho(x(k))$, gives:

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

It is a fact that, $\rho(k)$ and $\Delta x(k)$ are known variables at each instant k, whereas $f_\rho^\partial(k)$ can be numerically evaluated. Nonetheless, in practice, the values for $f_\rho^\partial(k+j)$ considering $j \in \mathbb{N}_{[1,N-2]}$ is unknown.

Proposition 1. For simplicity, one can assume that the partial derivative $f_\rho^\partial(k)$ stays constant along the prediction horizon, i.e. $f_\rho^\partial(k+j) = f_\rho^\partial\ \forall\, j \in \mathbb{N}_{[1,N-2]}$, where f_ρ^∂ denotes the partial derivative evaluated at instant k.

Based on the previous Proposition, the extrapolation at a given instant k can be given as a function of the previous extrapolation, this is: $P_k = P_{k-1}^\star + f_\rho^\partial \Delta X_k^\star + \Xi_k$, where

$$P_{k-1}^\star = \begin{bmatrix}\rho(k-1) & \rho(k) & \rho(k+1|k-1) & \dots & \rho(k+N_p-2|k-1)\end{bmatrix}$$

represents the previous estimation for the scheduling trajectory with the first term corrected (since $\rho(k|k-1)$ is known at instant k and $\Delta X_k^\star$ represents the state deviations along the horizon, corrected with the known value $\Delta x(k)$; Ξ_k represents a bias residual vector, which "corrupts" the extrapolation, but is obviously not known.

Since the bias residual cannot be accounted for, the extrapolation for this scheduling sequence is simply given by:

$$\hat{P}_k = \hat{P}_{k-1}^\star + f_\rho^\partial \Delta X_k^\star\,. \tag{32}$$

For such, $\Delta X_k^\star$ is computed by adapting Eq. (13), based on the corrected sequence of control inputs computed at the last sample holds, this is: $\tilde{U}_{k-1} = \begin{bmatrix}u(k) & \dots & u(k+N_p|k-1)\end{bmatrix}^T_{n_u \times N_p}$, and based on the previous, corrected scheduling trajectory guess $P_{k-1}^\star$. It follows: $\Delta X_k = A_x(P_{k-1}^\star)\Delta x(k) + B_x(P_{k-1}^\star)\tilde{U}_{k-1}$.

The corrections upon variables $\hat{P}_{k-1}$ and ΔX_k are:

$$\hat{P}^{\star}_{k-1} = \lambda \hat{P}_{k-1} + \nu\rho(k)\,, \tag{33}$$

$$\Delta X^{\star}_{k} = \lambda \Delta X_{k} + \nu\Delta x(k)\,, \tag{34}$$

$$\text{with } \lambda = \begin{bmatrix} 0 & \mathbb{I} & \dots & \mathbb{I} \end{bmatrix} \text{ and } \nu = \begin{bmatrix} \mathbb{I} & 0 & \dots & 0 \end{bmatrix}.$$

The dimensions of λ and ν in Eqs. (33)-(34) should be in accordance with n_p and n_x. If sought, a forgetting factor can be added to the algorithm, replacing the identity matrices in λ with exponentially decaying terms, such as $\mathbb{I}e^{-k/k_{max}}$. This forgetting factor attenuates the amount of mistaken information passed from on scheduling sequence estimate $\hat{P}_k$ to the following $\hat{P}_{k+1}$.

In order to ensure that the recursive extrapolation procedure holds, Assumptions 12 and 13 must be satisfied. Therefore, $f_\rho(\cdot)$ should be at least class $\mathcal{C}^1$, so that the derivative $f^{\partial}_{\rho}(k)$ exists for all $x(k+j) \in \mathcal{X}$, in order to construct the first-order Taylor expansion. Moreover, Assumption 13 is necessary so that every extrapolated term $\hat{\rho}(k+j)$ is bounded.

Finally, we mention that the recursive law in Eq. (32) does not ensure that the extrapolation guess abides to the scheduling parameter set $\mathcal{P}$. Therefore, in order to take into account that the variation of the parameters $\delta\rho(k) \in \delta\mathcal{P}$, each extrapolation vector $\hat{P}_k$ is "clipped" with respect to $\mathcal{P}$ and $\delta\mathcal{P}$.

3.1 | Convergence and Estimation Error Bounds

Proposition 2. This recursive extrapolation algorithm, as given through Eq. (32), indeed converges. This is, after a finite amount of steps k_c it holds that $\lim_{k\to k_c} \hat{P}_k \to P_{k_c}$.

Proof. Refer to Appendix B. □

Proposition 3. The ultimate bound of the estimation error $\xi^{\text{bound}}_{\rho}$ achieved with this algorithms yields an error set $\mathcal{E}$ which is a subset of $\mathcal{P}$. This ultimate bound is given by $\xi^{\text{bound}}_{\rho} = \left(\Gamma + \overline{f^{\partial}_{\rho}}\right)\overline{\Delta x}$ and it known *a prior*, i.e. there is no need to execute the algorithm to check it.

Proof. Refer to Appendix C. □

4 | PROPOSED MIN/MAX QLPV MPC ALGORITHM

Considering that $\xi_\rho \in \mathcal{E} \subset \mathcal{P}$, the sub-optimal MPC algorithm in Eq. (23) is adapted in order to ensure robustness. We seek performances guarantees despite the uncertainties introduced by the scheduling sequence estimation error.

As previously discussed, solving a single QP w.r.t. a scheduling sequence guess as in Eq. (28) does not ensure performances, since the solution $U^{\star}_k$ may represent a local minima of J_k. Anyhow, we know that the actual nonlinear process model in Eq. (7) differs from the $\hat{P}_k$-based prediction in Eqs. (13)-(14) due to the discrepancy variable ξ_ρ. Then, as gives Eq. (32), these model-process mismatches along the horizon can be treated robustly, providing a worst-case bound $J^{\text{bound}}_k > J_k$. Then, as done in robust min/max LPV MPC procedures, the QP is formulated with respect to J^{bound}_k, ensuring the overlap of local minima and robust performances.

Based on Assumption 6 and Eq. (19), we can thus expand the LPV model along the prediction horizon:

$$\begin{aligned} x(k+j+1) = {} & A(\hat{\rho}(k+j))x(k+j) + B(\hat{\rho}(k+j))u(k+j) + B_w w(k+j) \\ & + \underbrace{\left(A_1\xi_\rho(k+j)x(k+j) + B_1\xi_\rho(k+j)u(k+j)\right)}_{\sigma(k+j)}\,. \end{aligned} \tag{35}$$

The uncertainties introduced due to the model-process mismatch (extrapolation of the scheduling sequence) are denoted henceforth as $\sigma(k+j)$, which belongs to compact set $\mathcal{S}$ whose bounds can be computed offline, w.r.t. $\xi^{\text{bound}}_{\rho}$, $\mathcal{X}$ and $\mathcal{U}$:

$$\mathcal{S} := \left\{\sigma \in \mathbb{R}^{n_x} \mid \|\sigma\| \leq \overline{\sigma}\right\}. \tag{36}$$

We concatenate $\sigma(k+j)$ along the horizon as follows:

$$\Sigma_k \;=\; \left[\, \sigma(k|k) \;\ldots\; \sigma(k+N_p-1|k) \right] \,. \tag{37}$$

Remark 4. In regular min-max LPV MPC algorithms [11], $\mathcal{S}$ is computed w.r.t. much larger possible variations for ρ (the whole set $\mathcal{P}$). In works [15] that consider bounded rates of variations for ρ, the uncertainty set $\mathcal{S}$ is computed as if $\rho(k+j) \;=\; \rho(k) \pm j\delta\rho(k)$, which yields a smaller uncertainty set $\mathcal{S}$, but in general also larger than the set considered in this paper.

Embedding the uncertainty to the process predictions, we obtain:

$$X_k \;=\; A_x(\hat{P}_k)x(k) + B_x(\hat{P}_k)U_k + \Sigma_k \,. \tag{38}$$

Then, the core idea of the proposed method is quite simple, following the lines of the original min/max algorithms, but formulating the worst-case cost function with respect to the uncertainties introduced by the estimation errors ξ_ρ. It follows that $\Sigma_k^\star$, which induces the worst-case bound on the cost function J_k^{bound}, is found by solving the following maximisation CP:

$$\begin{aligned} \max_{\Sigma_k} \quad & J_k \\ \text{subject to} \quad & \text{constraints in Eqs. (24), (25), (26) and (27)} \,. \end{aligned} \tag{39}$$

Then, the solution that derives from this QP, $\Sigma_k^\star$, is plugged to the regular minimisation MPC CP, given in Eq. (23). The complete solution achieved with the proposed tool is $U_k^\star \;=\; \arg\min_{U_k} \max_{\Sigma_k} J_k$ subject to constraints (24)-(27). This solution resides in the sequential operation of both CPs: maximisation with respect to Σ_k (CP) followed by minimisation w.r.t. U_k (QP).

4.1 | The Implementation

Regarding the proposed min/max method, its implementation is performed according to the following guideline:

1. Offline Procedure:
   - Firstly, one should verify if the considered nonlinear process should satisfies Assumptions 1 to 5.
   - LDI should be performed, finding the LPV model G as in Eq. (9).
   - Regarding model G, one should verify if Assumptions 6 to 9 are satisfied.
   - The smallest Lipschitz constant Γ in Eq. (7) should be defined and so should the bounds on Δx.
   - With the aid of simulation tools, the recursive extrapolation algorithm of Eq. (32) should be tested and the forgetting factors λ and ν should be adequately tuned.
   - Compute the worst-case bound on the estimation error with respect to Eq. (C7).
   - Compute the compact uncertainty set $\mathcal{S}$ due to the wrong scheduling guess as of Eq. (36).
   - Prepare the MPC procedure by tuning the cost weighting matrices Q and R.
   - Compute the nominal cost function J_k in the Hessian-gradient form of Eq. (28).
2. Online Procedure: solve Algorithm 1.

5 | RECURSIVE FEASIBILITY ANALYSIS

This Section is concerned with the recursive feasibility properties of both CPs, and ISS of the closed-loop system regulated by the proposed MPC paradigm. We proceed by demonstrating the asymptotic stability of the closed-loop system and estimating the region of attraction of each CP. The zone of attraction for the complete algorithm is given by the smallest intersection of the two regions. Note that asymptotic ISS is demonstrated for a given region $\mathcal{X}_{ISS}$. Then, it is proved that for any starting condition within this region, the algorithm is recursively feasible.

Algorithm 1 Proposed Robust min/max NMPC Algorithm

Initialise: $x(0) = x_0$, $\rho(0) = \rho_0$, $k = 0$.
Require: P_0, λ, ν; **Require**: Q, R, N_p, $\mathcal{S}$;
Loop:

- Step (1): Measure the states $x(k)$ and get the scheduling parameters $\rho(k)$;
- Step (2): Evaluate the derivative f_ρ^∂;
- Step (3): Compute the extrapolation of the scheduling parameters along the horizon through Eq. (32);
- Step (4): Solve the maximisation CP, computing $\max_{\Sigma_k} J_k$ subject to constraints (24)-(27), finding $\Sigma_k^\star$;
- Step (5): Solve the minimisation QP, computing $\min_{U_k} J_k$ subject to constraints (24)-(27) with $\Sigma_k^\star$, finding $U_k^\star$
- Step (6) Apply the local control policy $u(k)$ as in Eq. (29);
- Step (7): Increment k, i.e. $k \leftarrow k + 1$.

end

5.1 | The Maximisation CP

In order to demonstrate the recursive feasibility property of the maximisation CP, we follow closely discussions of previous works[27,34,35].

Remark 5. In this previous papers, ISS and ISpS properties are verified for the whole min-max CP through the use of terminal ingredients. In this paper, we follow a dissipativity formulation, since we do not make use of RPI sets as terminal constraints nor of terminal stage costs in our formulation. Anyhow, the analysis of recursive feasibility of the maximisation step can be maintained.

Remark 6. Some of the following steps are easier to follow if the weak duality property of CPs is considered[39]: an maximisation CP can be equivalently written as a minimisation CP over the same variables with adjusted slack variables.

The considered maximisation CP solves $\Sigma_k^\star = \arg\max_{\Sigma_k} J_k(\cdot)$ subject to the inequality constraints (24)-(27) and based on the available scheduling sequence guess $\hat{P}_k$, being Σ_k the sequence of uncertainties along the horizon, as gives Eq. (37). Note that it can be adequately re-written in a generalised formulation with respect to Σ_k, this is:

$$\Sigma_k^\star(\hat{P}_k) = \arg\max_{\Sigma_k} \left(\frac{1}{2}\Sigma_k^T H_\sigma \Sigma_k - \Sigma_k^T g_\sigma(\hat{P}_k, x(k)) \right). \tag{40}$$

Replacing X_k in Eq. (40), it follows[7] that $H_\sigma = 2\breve{Q}$ and $g_\sigma = -2\breve{Q}\left(A_x(\hat{P}_k)x(k) + B_x(\hat{P}_k)U_k\right)$.

Remark 7. Note that, if constraints are disregarded, for rationale purposes only, it is direct to evaluate that the maximal value for J_k, w.r.t. Σ_k would be found with $\Sigma_k^\star = (H_\sigma)^{-1} g_\sigma$. Since g_σ is linear on U_k, the value for U_k would be $\overline{U} = \text{col}\{\overline{u}\}$ (a sequence of maximal control signals), this is: $\Sigma_k^\star = -(2\breve{Q})^{-1}\left(A_x(\hat{P}_k)x(k) + B_x(\hat{P}_k)\overline{U}\right)$. Regarding the CP constraints, it follows that (27) adds no difference to this possible result. Moreover, constraints (25) and (26) are only box-type operations over x and u, respectively. Therefore, it follows directly that for any starting condition x_0 within the feasibility set $\mathcal{X}$, this maximisation CP is recursively feasible since J_k is never be unbounded w.r.t. Σ_k due to its regular quadratic formulation on Σ_k, operated through Eq. (40). Nonetheless, this property only remains true if and only if Q^{-1} exists, since $(H_\sigma)^{-1} = (2\breve{Q})^{-1}$.

In order to demonstrate the recursive feasibility property of the CP in Eq. (40), we consider that:

- The minimisation QP is also feasible. This is quite logical because the min-max formulation resides in the operation of both these CPs consecutively.

[7] Notation $\breve{Q}$ and $\breve{R}$ denote block-diagonal matrices with Q and R repeated N_p times in the diagonal, respectively.

- The stage cost $\ell(\cdot)$ is lower bounded for all $x \in \mathcal{X}_{\text{Max CP}}$ (this set denotes the feasibility region for the maximisation CP). Indeed, it follows directly from Eqs. (21)-(22):

$$\ell(\cdot) \geq a_{\ell}(\|x\|)\,, \tag{41}$$

being $a_{\ell}(\|x\|)$ a $\mathcal{K}$-class function.

Lemma 1. Based on these previous conditions, of a feasible minimisation QP and of Eq. (41), it follows that the worst-case cost function J_k^{bound}, computed w.r.t. $\Sigma_k^{\star}$, is upper-bounded, considering that the uncertainties are described as of Eq. (36), such that:

$$J_k^{\text{bound}} \leq \alpha_J(\|x\|) + \beta_J(\overline{\sigma})\,, \tag{42}$$

where α_J and β_J are $\mathcal{K}_{\infty}$-class functions.

Proof. Refer to Appendix D. □

Remark 8. Since the previous Lemma requires that the min. QP to be recursively feasible, it is implied that $\mathcal{X}_{\text{Max CP}} := \mathcal{X}_{\text{Min QP}}$, where $\mathcal{X}_{\text{Min QP}}$ is the feasibility set of the second CP. For simplicity, we henceforth denote the maximisation CP using an abstraction: as an operator Υ upon the measured state vector $x(k)$, as follows:

$$\Sigma_k^{\star} := \Upsilon(\hat{P}_k)x(k)\,. \tag{43}$$

5.2 | The Minimisation QP

The analysis of the ISS property of the minimisation QP is more complex. This CP solves $U_k^{\star} = \arg\min_{U_k} J_k(\cdot)$ subject to constraints (24)-(27), based on the available scheduling sequence guess $\hat{P}_k$ and on the uncertainty vector $\Sigma_k^{\star}$. Figure 1 gives a graphical block-diagram interpretation of the system, considering both CPs (1) and (39) and the extrapolation algorithm, where G represents the open-loop LPV embedding of Eq. (9).

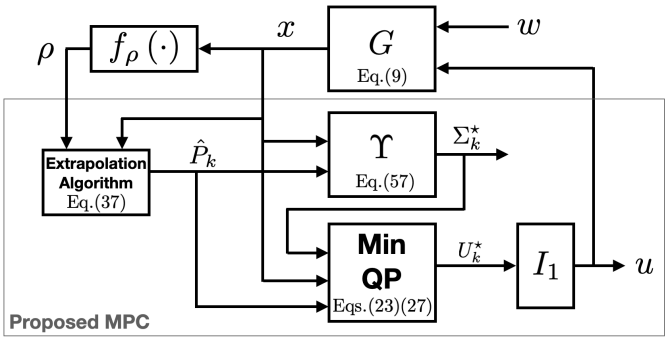

FIGURE 1 Graphical Representation of the LPV Embedded Nonlinear System and Proposed Algorithm.

We replace X_k of Eq. (38) in Eq. (28), which leads to the following Hessian and gradient[8]:

$$H(\hat{P}_k) = 2\left(\breve{R} + B_x(\hat{P}_k)^T \breve{Q} B_x(\hat{P}_k)\right)\,, \tag{44}$$

$$g(\hat{P}_k, x(k)) = -2B_x(\hat{P}_k)^T \breve{Q}\left(A_x(\hat{P}_k)x(k) + \Sigma_k^{\star}\right)\,. \tag{45}$$

Thus, in order to verify ISS, we proceed by defining a nonlinear static map $\phi : g \rightarrow U_k^{\star}$ implied by the constrained minimisation QP in its regular form of Eq. (28).

[8]For notation compactness, we denote henceforth simply $g_k = g(\hat{P}_k, x(k))$ and $\phi_k = \phi(g(\hat{P}_k, x(k)))$.

As demonstrated in previous works[38], the stability of the closed-loop system with $u(k) = I_1 U_k^\star$ deriving from Eq. (28) can be verified if the following sector boundary inequality is satisfied:

$$\phi_k^T H(\hat{P}_k)\phi_k - \phi_k^T g_k \leq 0 \, \forall g_k \, . \tag{46}$$

Regarding this stability inequality, we show a graphical interpretation of the considered system, presented in Figure 2. The proposed MPC policy is divided by the upper ϕ block, which comprises the minimisation QP, and by the lower Υ block, which embeds the maximisation CP. Regarding, Figure 1, the output of the minimisation QP $U_k^\star$ is now replaced by the nonlinear operator ϕ_k. Moreover, the main open-loop process in Figure 2 is represented by G_{I_1}, which is a compacted operator comprising the open-loop plant and matrix I_1, since $u(k) = I_1 U_k^\star$. It follows that $x =: G(u, w)$, $u = I_1\phi$, and, $x := G_{I_1}(\phi, w)$.

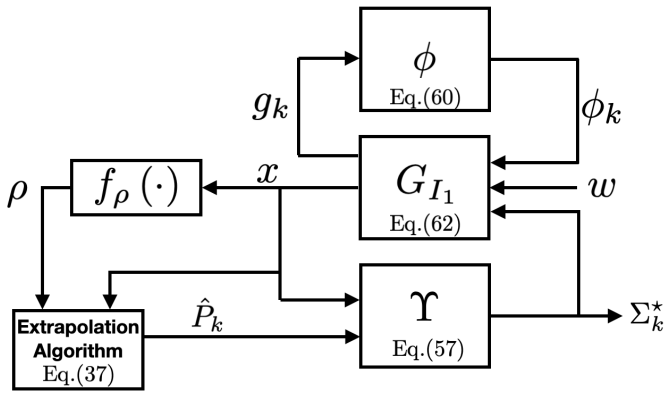

FIGURE 2 Graphical Representation of the Closed-Loop System.

In order to evaluate the previous stability inequality, a factorisation of the Hessian is necessary so that parameter-dependency can be smartly dropped. We define the block-diagonal compact set $\breve{\mathcal{P}} \subset \mathbb{R}^{N_p \times n_p}$ as the compact set within which $\hat{P}_k$ lies (recall that each entry of this vector, $\hat{\rho}(k+j)$ is bounded to $\mathcal{P}$. The factorisation is the following:

$$\begin{bmatrix} 0 & 0 \\ 0 & H(\hat{P}_k) \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & B_x(\hat{P}_k) \\ 0 & \mathbb{I} \end{bmatrix}^T}_{H_P(\hat{P}_k)^T} \underbrace{\begin{bmatrix} 2\breve{Q} & 0 \\ 0 & 2\breve{R} \end{bmatrix}}_{H_0} \underbrace{\begin{bmatrix} 0 & B_x(\hat{P}_k) \\ 0 & \mathbb{I} \end{bmatrix}}_{H_P(\hat{P}_k)} . \tag{47}$$

Based on the prior factorisation, we re-write inequality (46) as follows:

$$\begin{bmatrix} * \end{bmatrix}^T \underbrace{\left(\left[\begin{array}{c|c} 0 & 0 \\ \hline -\mathbb{I} & 0 \end{array}\right] + H_P(\hat{P}_k)^T H_0 H_P(\hat{P}_k) \right)}_{\Pi(\hat{P}_k)} \begin{bmatrix} g_k \\ \phi_k \end{bmatrix} \leq 0 \, . \tag{48}$$

As provided in previous works[40,41], the above parameter-dependent quadratic constraint can be cast into a regular multiplier form $z_g^T M_z z_g \leq 0$, where z_g is the output of a bounded linear operator $\Psi(\hat{P}_k)$ which factorises $\Pi(\hat{P}_k)$, this is: $\Pi(\hat{P}_k) = \left(\Psi(\hat{P}_k)\right)^* M_z \Psi(\hat{P}_k)$. The operator $\Pi(\cdot)$ stands for the "filling" of the previous inequality (48). Thus, we continue by using the previous factorisation to write $\Pi(\cdot)$ in a multiplier form, as follows:

$$\Pi(\hat{P}_k) = \Psi^*(\hat{P}_k) \underbrace{\left[\begin{array}{c|c} 0 & \begin{bmatrix} 0 & 0 \end{bmatrix} \\ \hline \begin{bmatrix} 0 \\ -\mathbb{I} \end{bmatrix} & H_0 \end{array}\right]}_{M_z} \underbrace{\left[\begin{array}{c} \begin{bmatrix} \mathbb{I} & 0 \end{bmatrix} \\ \hline H_P(\hat{P}_k) \end{array}\right]}_{\Psi(\hat{P}_k)} . \tag{49}$$

Figure 3 gives a graphical interpretation of the extraction of parameter-dependency through Ψ. It follows that the multiplier form of $\Pi(\cdot)$ is built with:

$$M_z = \left[\begin{array}{c|cc} 0 & 0 & 0 \\ \hline 0 & 2\breve{Q} & 0 \\ -\mathbb{I} & 0 & 2\breve{R} \end{array}\right], \; \Psi(\hat{P}_k) = \left[\begin{array}{cc} \mathbb{I} & 0 \\ \hline 0 & B_x(\hat{P}_k) \\ 0 & \mathbb{I} \end{array}\right]. \tag{50}$$

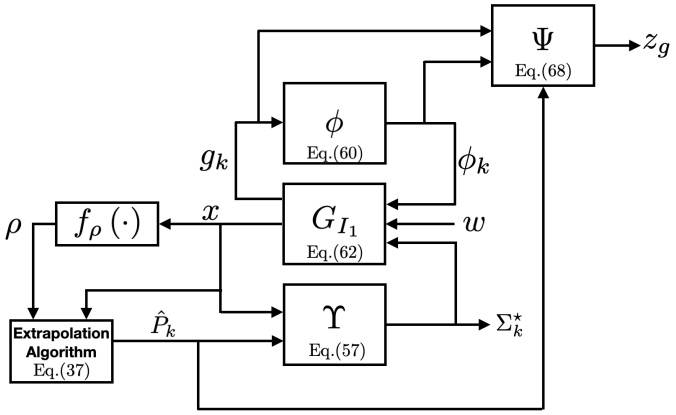

FIGURE 3 Graphical Representation of the Parameter-dependency Extraction.

Through the above development, we express the dissipativity inequality (46) simply as:

$$\left[*\right]^T \Psi(\hat{P}_k)^T M_z \Psi(\hat{P}_k) \left[\begin{array}{c} g_k \\ \phi_k \end{array}\right] \leq 0\,, \tag{51}$$

which is can be compacted as:

$$\left[*\right]^T M_z z_g \leq 0 \, \forall \hat{P}_k \in \breve{\mathcal{P}}\,, \tag{52}$$

being $z_g := \Psi(\hat{P}_k) \left[\begin{array}{cc} g_k & \psi_k \end{array}\right]^T$ the output of the $\Psi(\cdot)$ operator.

The parameter-dependency has been dropped through the previous factorisation procedures. Therefore, we can perform a Linear Fractional Transformation (LFT) to extract the LPV scheduling parameter dependency as an upper Θ-block (which is connected to an LTI nominal block). Considering z_g as an output the lifted system, we graphically illustrate the LFT in Figure 4, where $G^n_{I_1}$ is an LTI nominal model of the augmented plant, as follows:

$$G^n_{I_1} := \left\{\begin{array}{rcl} x(k+1) &=& A^n x(k) + B^n_w w(k) + B^n_\phi \phi_k + B^n_\rho u_\rho(k) \\ y_\rho(k) &=& C^n_\rho x(k) + D^n_{\rho,\phi} \phi_k + D^n_\rho u_\rho(k) \\ z_g(k) &=& C^n_z x(k) + D^n_{z,\phi} \phi_k + D^n_{z,\rho} u_\rho(k) + D^n_{z,\Sigma} \Sigma^\star_k \end{array}\right., \tag{53}$$

where $u_\rho(k) := \Theta y_\rho(k)$ makes the interconnection between this nominal LTI block and the LPV-lifted upper Θ-block and $\Sigma^\star_k$ appears now as an input to the $G^n_{I_1}$ block, since it is present in g_k as gives Eq (45).

Finally, in order to check if the system is ISS, it remains to verify the following Lemma[29], which ensures that the lower $G^n_{I_1}$-Θ block is stable despite the upper ϕ transfer.

Lemma 2. Adapted from a previous paper[29]
The closed-loop system given in the LFT form in Eq. (53), regulated under the proposed min./max MPC law in the form of $U^\star_k = \arg\min_{U_k} J_k(\cdot)$ subject to constraints (24)-(27) and based on the available scheduling sequence guess $\hat{P}_k$ and on the uncertainty $\Sigma^\star_k$, is quadratically stable, verifying the dissipativity inequality (46), if there exists a positive-definite matrix

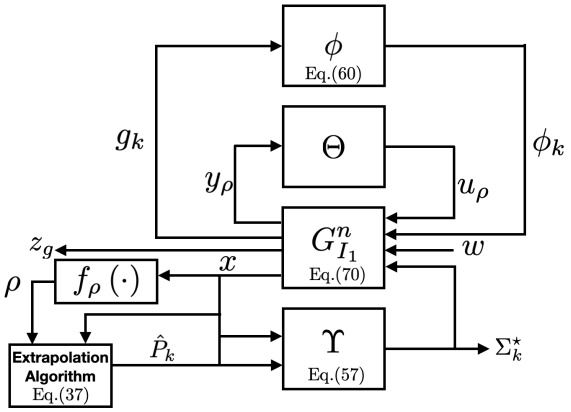

FIGURE 4 Graphical Representation of the Closed-Loop with LPV-Dependency Extracted.

$P = P^T > 0$ and a constant $\tau > 0$ such that:

$$\begin{bmatrix} (A^n)^T P A^n - P & (A^n)^T P B^n_\phi & (A^n)^T P B^n_\rho & 0 \\ \star & (B^n_\phi)^T P B^n_\phi & (B^n_\phi)^T P B^n_\rho & 0 \\ \star & \star & (B^n_\rho)^T P B^n_\rho & 0 \\ \star & \star & \star & 0 \end{bmatrix} - \tau \Pi_\phi + \Pi_\Theta \prec 0 \tag{54}$$

where

$$\Pi_\phi = [*]^T \left[\begin{array}{c|ccc|c} 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 2\breve{Q} & 0 & 0 & 0 \\ -\mathbb{I} & 0 & 2\breve{R} & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 \end{array}\right] \begin{bmatrix} C^n_z & D^n_{z,\phi} & D^n_{z,\rho} & D^n_{z,\Sigma} \end{bmatrix}, \tag{55}$$

$$\Pi_\Theta = [*]^T M_\Theta \begin{bmatrix} C^n_\rho & D^n_{\rho,\phi} & D^n_\rho & D^n_{\rho,\Sigma} \\ 0 & 0 & \mathbb{I} & 0 \end{bmatrix}, \tag{56}$$

and $M_\Theta \in \mathcal{M}_{D/G}$.

Proof. Refer to Appendix E. □

The positive definite matrix P found through Lemma 2 defines the following set:

$$\mathcal{X}_{\text{Min CP}} := \left\{ x \in \mathbb{R}^{n_x} \mid x^T P x \leq 1 \right\}. \tag{57}$$

Thus, for any starting condition x_0 contained in the interior of $\mathcal{X}_{\text{Min CP}}$, the minimisation QP ensures (local) asymptotic stabilisation to the origin. Since the proposed MPC is made of two consecutive CPs, the complete set within which ISS is verified is given by:

$$\mathcal{X}_{ISS} := \mathcal{X}_{\text{Max CP}} \cap \mathcal{X}_{\text{Min QP}}. \tag{58}$$

Since $\mathcal{X}_{\text{Max CP}} := \mathcal{X}_{\text{Min QP}}$, it follows that $\mathcal{X}_{ISS} = \mathcal{X}_{\text{Min QP}}$.

6 | APPLICATION EXAMPLE

In this Section, we present a case-study for which the robust dissipative MPC method is applied. As discussed in energy systems literature [42,43,44], the addition of renewable energy sources to power plants can be a good route to reduce greenhouse gas emissions and environmental impact. Anyhow, an inherent problem to be solved is how to integrate these energy sources without loosing efficiency and dispatchability of energy plants.

6.1 | Solar-thermal system, phenomenological model, and control problem

We consider modern solart-thermal (ST) systems, which are structures that integrate collector fields, accumulation tanks and gas heaters. Each subsystem has independent dynamics that strongly influence the total output. These ST units are controlled in order to ensure efficiency despite variations on the energy input caused due to cloudy periods of the day. We assume that the global ST coordination as well as the control of the tanks and gas heaters are regularly working: the heated fluid is accumulated on the tanks to compensate for the lack of heated flow coming from the solar collectors in cloudy periods. Moreover, if the outlet temperature is not enough to comply with demands, the gas heater is used to further heat the outlet. The heated fluid is used to attend the heating demands of a separate industrial process.

The focus of the control system is solely to the regulate the temperature of the ST collector panel. Accordingly, the collector outlet flow temperature signal must track a constant steady-state reference, despite instaneous variations on the solar irradiance or on the external temperature. Figure 5 illustrates the considered ST system.

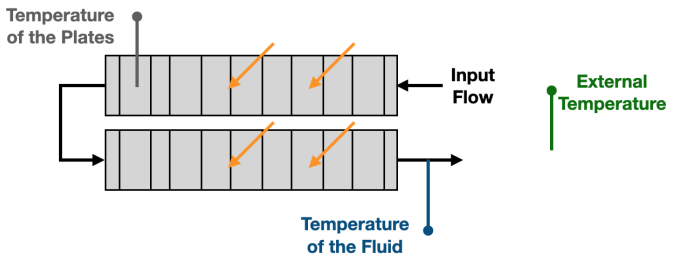

FIGURE 5 Schematic Illustration of a solar-thermal collector field.

Complete phenomenological models have previously been derived for ST collector fields[43], with according model-validation[45]. These models are derived on the basis of the following set of assumptions:

- The fluid flow through the solar collector is incompressible (with density ϵ_f), with uniform pressure along the field; the heat transfer capacity of the fluid is constant and denoted C_f;
- The heat transfer capacity of the collector plates is constant and denoted C_m; the density of these metal plates is also constant and denoted ϵ_m;
- The balance of energy equations assume a constant thermal loss coefficient ν, with respect to the thermal energy that derives from the incident solar radiance;
- The heat transfer coefficient of the absorver (external temperature to plates), denoted h_0, is constant, while the heat transfer coefficient of the fluid (fluid to plates), denoted $h_i(\cdot)$, varies positively according to the temperature of the plates.

Then, the following partial-differential dynamics arise due to balance of energy equations, where t represents the time variable and s the space variable:

$$\epsilon_m C_m A_e \frac{dT_p}{dt}(t) = d_e \pi \nu I(t) - d_e \pi h_0 (T_p(t) - T_e(t)) - d_i \pi h_i (T_p(t))(T_p(t) - T_f(t)), \tag{59}$$

$$\epsilon_f C_f A_i \frac{\partial T_f}{\partial t}(t,s) = -u(t)\epsilon_f C_f \frac{\partial T_f}{\partial s}(t,s) + d_i \pi h_i (T_p(t))(T_p(t) - T_f(t)). \tag{60}$$

In these temperature gradient dynamics of Eqs. (59)-(60), $I(t)$ stands for solar radiance focused upon the collectors (which is a load disturbance from a control viewpoint); T_p, T_e and T_f are, respectively, the collector plate, the external (load disturbance as well) and the fluid temperatures; u is the inlet fluid flow, which is the control input of the system; finally, A_i and A_e are, respectively, the internal and external surfaces of the pipes, that have (internal and external) diameters of d_i and d_e.

For application purposes, the space-derivative term $\frac{\partial T_f}{\partial s}(t,s)$ can be replaced[43,45,46] by either a nonlinear function or an apparent transport delay. In this paper, it is approximated by the following nonlinearity:

$$\frac{\partial T_f(t,s)}{\partial s} \approx \frac{1-e^{-\frac{T_f(t)}{T_f^{max}}}}{(1-e^{-1})}, \tag{61}$$

which means that the diffusion of the thermal energy of the fluid flowing along the flat collectors increases with respect to its temperature $T_f(t)$ until a certain level is attained T_f^{max}, after which the diffusion is constant, i.e. the whole fluid inside the flat collector is at the same temperature. This approximation is quite reasonable with respect to the ST application and in accordance with the literature[43].

The heat transfer coefficient of the fluid $h_i\left(T_p(t)\right)$ is given according to the following nonlinear equation:

$$h_i\left(T_p(t)\right) = \overline{h}_i\left(\frac{1-e^{\frac{-T_p(t)}{T_p^{max}}}}{1-e^{-1}}\right), \tag{62}$$

where $\overline{h}_i$ is the maximal heat transfer coefficient of fluid, attained for $T_p(t) = T_p^{max}$.

Regarding the nonlinear model of Eqs. (59)-(60) with the relaxations of Eqs. (61)-(62), the parameters have been identified and adjusted for the *CIESOL ST* plant, located in the *CIESOL-ARFR-ISOL R&D* Centre of the University of Almería, Spain. The numerical values for these parameters, from paper[43], are given in Table 1.

TABLE 1 Model Parameters of the ST Process in Eqs. (59)-(60).

ϵ_m	$1100\,\mathrm{kg/m^3}$	C_m	$440\,\mathrm{J/(kg^oC)}$
ϵ_f	$1000\,\mathrm{kg/m^3}$	C_f	$4018\,\mathrm{J/(kg^oC)}$
A_e	$0.0038\,\mathrm{m^2}$	A_i	$0.0013\,\mathrm{m^2}$
d_i	$0.04\,\mathrm{m}$	d_e	$0.07\,\mathrm{m}$
h_0	11	$\overline{h}_i$	800
ν	3.655	–	–

6.2 | Problem under consideration

The goal of this ST system is to track outlet temperature references to cover a certain heat demand, which is done by varying the inlet fluid flow u. This collector field has a $160\,\mathrm{m^2}$ surface area, distributed in ten parallel rows composed of eight collectors per row.

In terms of performances, the temperature set-point tracking should be done as fast as possible, while respecting the maximal temperature of $300\,^{\mathrm{o}}\mathrm{C}$ that the inlet fluid can tolerate. Moreover, the temperature of the plates should not surpass $600\,^{\mathrm{o}}\mathrm{C}$. These performances can be evaluated using usual reference-tracking indexes, such as the integral of the average tracking error. Through the sequel, we denote T_p^{sp} and T_f^{sp} as the constant steady-state temperature references to the collector plate and to flowing fluid, respectively. The considered steady-state targets for reference tracking are:

$$T_p^{sp} = 109.93\,^{\mathrm{o}}\mathrm{C}\,,\ \mathrm{T_f^{sp}} = 97\,^{\mathrm{o}}\mathrm{C}\,.$$

The inlet flow (control signal) should be always positive, since no fluid can be extracted from the ST units, only injected, and abide to a upper bound of $0.35\,\mathrm{m^3/s}$. Moreover, the control policy has to be evaluated within $T_s = 0.01\,\mathrm{s}$, which is the considered sampling period.

We stress that the dynamics of this ST process exhibit average settling periods in the order of 100 s. In practice, many control schemes have been tuned considering a sampling period of a few seconds, e.g.[43,45]. Nevertheless, we choose a tighter sampling period for illustration purposes, in order to verify whether the proposed method could serve for embedded real-time applications.

The disturbances to this system (the solar radiance and external temperature variables) are assumed to be measurable from a control viewpoint. This is quite reasonable, given that accurate estimations for the future behaviour of these disturbances can be indeed obtained[42]. These estimation results (for solar radiance and outside temperature) are easily provided with Neural Network tools[47].

Table 2 resumes the state and input constraints. Note that the fluid and plate temperatures are lower-bounded by external temperature to the ST system, $T_e(t)$. If there is no sun, the ST system reaches a thermal equilibrium with $T_e(t)$. For simplicity, since $T_e(t) > 0$, the lower bounds on T_p and T_f can be taken as 0.

TABLE 2 Constraints of the considered ST system.

$u(t) \in \mathcal{U}$	$\mathcal{U} := \left\{u \in \mathbb{R} \,\middle|\, 0 \leq u \leq 0.35\,\mathrm{m}^3/\mathrm{s}\right\}$
$T_p(t) \in \mathcal{T}_p$	$\mathcal{T}_p := \left\{T_p \in \mathbb{R} \,\middle|\, T_e(t) \leq T_p \leq T_p^{max}\right\}, T_p^{max} = 600\,^{\mathrm{o}}\mathrm{C}$
$T_f(t) \in \mathcal{T}_f$	$\mathcal{T}_f := \left\{T_f \in \mathbb{R} \,\middle|\, T_e(t) \leq T_f \leq T_f^{max}\right\}, T_f^{max} = 300\,^{\mathrm{o}}\mathrm{C}$

6.3 | qLPV-embedded Model

Since this paper is concerned with the application of MPC technique, the ST nonlinear phenomenological model of Eqs. 59-60, with the relaxations of Eqs. (61)-(62), is Euler-discretised with the sampling period of $T_s = 0.01$ s. This procedure yields a nonlinear discrete-time model. Given that the proposed min/max MPC method is conceived for qLPV embedded nonlinear models, and due to the fact that the LDI property holds for the yielded discrete-time model, a qLPV model is obtained. We consider the following system states:

$$x(k) = \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} = \begin{bmatrix} T_p(k) - T_p^{sp} \\ T_f(k) - T_f^{sp} \end{bmatrix}, \tag{63}$$

and the scheduling parameters as $\rho = [\rho_1, \rho_2]^T$, which are respectively derived directly from the nonlinearities added to the balance of energy equations due to the time-varying thermal loss term given in Eq. (62) and due the partial derivative approximation given in Eq. (61):

$$\begin{bmatrix} \rho_1(k) \\ \rho_2(k) \end{bmatrix}^T = f_\rho(x(k)) = \begin{bmatrix} d_i \pi \overline{h_i} \left(\frac{1 - e^{-\frac{x_1(k)}{(T_p^{max} - T_p^{sp})}}}{1 - e^{-1}} \right) \\ \frac{1 - e^{-\frac{x_2(k)}{(T_f^{max} - T_f^{sp})}}}{(1 - e^{-1}) A_i} \end{bmatrix}. \tag{64}$$

Evidently, each of the scheduling parameters is bounded to a convex set:

$$\rho_1 \in [\underline{\rho_1}, \overline{\rho_1}] = [0, d_i \pi \overline{h_i}] \text{ and} \tag{65}$$

$$\rho_2 \in [\underline{\rho_2}, \overline{\rho_2}] = \left[0, \frac{1}{A_i}\right], \tag{66}$$

which means that $\rho \in \mathcal{P}$. Furthermore, note that the time-derivatives of ρ, denoted $\delta\rho$ are also available and ultimately bounded in a convex set $\delta\mathcal{P}$. Accordingly, the following qLPV realisation is obtained:

$$x(k+1) = A(\rho(k))x(k) + B(\rho(k))u(k) + B_w(\rho(k))w(k), \tag{67}$$

$$\rho(k) = f_\rho(x(k)). \tag{68}$$

Note that $[A(\rho), B(\rho), B_w(\rho)]$ are affine on the scheduling vector ρ. The vector of load disturbances is given as follows $w(k) = \left[\begin{array}{c|cc} I(k) & T_e(k) & T_p^{sp} \;\; T_f^{sp} \end{array}\right]$. The model matrices are:

$$A(\rho(k)) = \mathbb{I}_{n_x} + T_s \begin{bmatrix} -\frac{d_e \pi h_0}{\epsilon_m C_m A_e} - \frac{1}{\epsilon_m C_m A_e}\rho_1(k) & \frac{1}{\epsilon_m C_m A_e}\rho_1 \\ \frac{1}{\epsilon_f C_f A_i}\rho_1 & -\frac{1}{\epsilon_f C_f A_i}\rho_1 \end{bmatrix} \tag{69}$$

$$B(\rho(k)) = T_s \begin{bmatrix} 0 \\ -\rho_2 \end{bmatrix}, \tag{70}$$

$$B_w(\rho(k)) = T_s \left[\begin{array}{cc|cc} \frac{d_e \pi \nu}{\epsilon_m C_m A_e} & \frac{d_e \pi h_0}{\epsilon_m C_m A_e} & -\frac{d_e \pi h_0}{\epsilon_m C_m A_e} - \frac{1}{\epsilon_m C_m A_e}\rho_1(k) & \frac{1}{\epsilon_m C_m A_e}\rho_1 \\ 0 & 0 & \frac{1}{\epsilon_f C_f A_i}\rho_1 & -\frac{1}{\epsilon_f C_f A_i}\rho_1 \end{array}\right]. \tag{71}$$

6.4 | Offline MPC Preparations

The system is conceived for a steady-state reference tracking goal with the aforementioned $T_p^{sp} = 109.93\,^{\circ}\text{C}$ and $T_f^{sp} = 97\,^{\circ}\text{C}$. Regarding this matter, we note that:

- The box-type set for the states, $\mathcal{X}$, is defined with the following ultimate bound: $\overline{x} = \begin{bmatrix} 490 & 203 \end{bmatrix}^T\,^{\circ}\text{C}$.
- The deviation of the states Δx is, thus ultimately bounded by: $\overline{\Delta x} = \begin{bmatrix} 0.162 & 0.2637 \end{bmatrix}^T\,^{\circ}\text{C}$.
- The differentiation function $f_\rho^\partial(k)$ is ultimately bounded:

$$\left\| \frac{\partial f_\rho}{\partial x} \right\| = \left\| \begin{bmatrix} (\frac{d_i \pi \overline{h_i}}{(1-e^{-1})}) \frac{1}{490} e^{-\frac{x_1}{490}} \\ (\frac{1}{A_i(1-e^{-1})}) \frac{1}{203} e^{-\frac{x_2}{203}} \end{bmatrix} \right\| \leq 0.3246 \;\forall\; x \in \mathcal{X}. \tag{72}$$

- With respect to $\mathcal{X}$, the smallest local Lipschitz constant for the nonlinear map $f_\rho(\cdot)$ is found for:

$$\left\| \begin{bmatrix} (\frac{d_i \pi \overline{h_i}}{(1-e^{-1})}) \left(e^{-\frac{x_1}{490}} - e^{-\frac{\hat{x}_1}{490}} \right) \\ (\frac{1}{A_i(1-e^{-1})}) \left(e^{-\frac{x_2}{203}} - e^{-\frac{\hat{x}_2}{203}} \right) \end{bmatrix} \right\| \leq \Gamma \left\| \begin{pmatrix} x_1 - \hat{x}_1 \\ x_2 - \hat{x}_2 \end{pmatrix} \right\|, \tag{73}$$

  where

$$\Gamma = \left| \begin{matrix} (\frac{d_i \pi \overline{h_i}}{(1-e^{-1})}) \frac{e^1}{490} \\ (\frac{1}{A_i(1-e^{-1})}) \frac{e^1}{230} \end{matrix} \right| = 0.8825\,. \tag{74}$$

- The worst-case scheduling sequence estimation error is given by:

$$\xi_\rho^{\text{bound}} = \left(\Gamma + \overline{f_\rho^\partial} \right) \overline{\Delta x} = \begin{bmatrix} 0.046 & 0.0015 \end{bmatrix}^T. \tag{75}$$

- The uncertainties σ introduced due to the model-process mismatches, thus, are bounded to the compact set $\mathcal{S}$, defined as:

$$\mathcal{S} := \{ \sigma \in \mathbb{R}^{n_x} \,|\, \|\sigma\| \leq 4.89^{\text{o}}\text{C} \}. \tag{76}$$

Notice, for comparison purposes, that the uncertainty set computed as if the scheduling parameters varied arbitrarily inside $\mathcal{P}$ (as done in the original min/max LPV MPC design algorithms[11]) is given by:

$$\mathcal{S}^{\text{Cao et al., 2005}} := \{ \sigma \in \mathbb{R}^{n_x} \,|\, \|\sigma\| \leq 599^{\text{o}}\text{C} \}. \tag{77}$$

while the uncertainty set computed taking the rates of variations of the scheduling parameters ($\delta\rho$) into account, as done in[13], for a control horizon of $N_p = 30$ steps, is given by:

$$\mathcal{S}^{\text{Li et al., 2010}} := \{ \sigma \in \mathbb{R}^{n_x} \,|\, \|\sigma\| \leq 489^{\text{o}}\text{C} \}. \tag{78}$$

Evidently, these two sets are much wider than the one with the proposed method. This means that the online computational effort to solve the maximisation CP is smaller with the proposed method. This is tested and demonstrated in the following Section.

7 | SIMULATION RESULTS AND ANALYSIS OF THE MPC SCHEME

Now, the proposed dissipative fast robust MPC method for nonlinear systems is applied to the ST collector system. The following simulations are performed in Matlab, with the aid of Yalmip software, Gurobi and fmincon solvers, on a 2.4 GHz, 8 GB RAM Macintosh computer.The considered process is emulated through the nonlinear high-fidelity phenomenological partial-differential model given in Eqs. (59)-(60), with parameters given by Table 1.

The proposed method is implemented with the uncertainties defined by the set $\mathcal{S}$ in Eq. (76). The solutions of the CPs are obtained with fmincon (maximisation CP, through an interior-point mechanism) and Gurobi (minimisation QP) solvers.

Through the sequel, the proposed control scheme is denoted "Proposed qLPV MPC". For comparison purposes, it is compared to the following key methods from the literature:

- A full-blown NMPC algorithm[2], which embeds the complete nonlinear model predictions. To solve the resulting NP, fmincon solver is used; this method is referred to as "Full-Blown NMPC".
- The original min/max LPV MPC algorithm[11], defined with respect to the uncertainty set given in Eq. (77). Its solution comprises a CP (maximisation, via fmincon) and a QP (minimisation, via Gurobi); it is henceforth denoted "min-max (Cao et. al, 2005)".
- The min/max LPV MPC scheme considering bounded rates of parameter variations[13], defined with respect to the uncertainty set given in Eq. (78). This approach is also resolved via fmincon and Gurobi; it is denoted "min-max (Li et. al, 2010)".
- The qLPV-embedding NMPC method[25], which uses a scheduling sequence estimation and solves sequential QPs, solved via through iterated uses of Gurobi. This last method is henceforth marked as "qLPV MPC (Cisneros & Werner, 2020)".

All these controllers are synthesised with the same cost function J_k and prediction horizon $N_p = 30$ samples. The cost function is set to further force the regulation of the fluid temperature variable, with the following weights:

$$Q = \begin{bmatrix} 0.2 & 0 \\ 0 & 0.8 \end{bmatrix}, \; R = 10^{-6}. \tag{79}$$

We proceed by depicting the obtained results in terms of reference tracking, i.e. regulation of the system states to the origin. These results comprise 950 s of simulation of the considered solar-thermal unit. The load disturbances (solar irradiance and environment temperature) are shows in Figure 6. Once again, we remark that the reference tracking goals are taken as constant values, this is:

$$T_p^{sp} = 109.93\,^{\circ}\mathrm{C}, \; \mathrm{T_f^{sp}} = 97\,^{\circ}\mathrm{C}. \tag{80}$$

7.1 | Analysis of the Region of Attraction

Firstly, we aim to demonstrate that the proposed method is indeed recursively feasible, yielding an ISS region of attraction $\mathcal{X}_{ISS}$. According to the steps detailed in Sec. 5, the LMI in Lemma 2 yields a positive definite matrix P and a constant τ that verify the dissipativity conditions of the proposed min-max algorithm. This is, indeed there exist P and τ such that the cost function of the minimisation QP decays over the simulation run; they are numerically given:

$$P = \begin{bmatrix} 0.18746 & 0.00011 \\ 0.87199 & 24.00050 \end{bmatrix} 10^{-4}, \; \tau = 1.67939\,10^{-7}. \tag{81}$$

Thus, for whichever starting condition x_0 found inside the ellipsoidal set $\mathcal{X}_{ISS} := \left\{x_0 \in \mathbb{R}^{n_x} \mid x_0^T P x_0 \leq 0\right\}$, input-to-state stability is ensured. Accordingly, this is shown in Figure 7, where the elipsoid $\mathcal{X}_{ISS}$ is depicted altogether with the evolution of the systems states $x(k)$ (obtained with the Proposed qLPV MPC method).

Complementary, Figure 8 displays the decrease of the cost function $J_k(\cdot)$ over the simulation run. We note that it has an asymptotic behaviour towards zero; the instantaneous increasing moments stand for those where the appear harsh variations of solar irradiance (refer to Fig. 6) and, thus, the scheduling sequence extrapolation does not get such accurate gets and the maximisation CP computes larger uncertainties Σ_k. We note that the cost function is compared to that obtained with the Full Blown NMPC method, which is obviously smoother since it accounts for the complete NP.

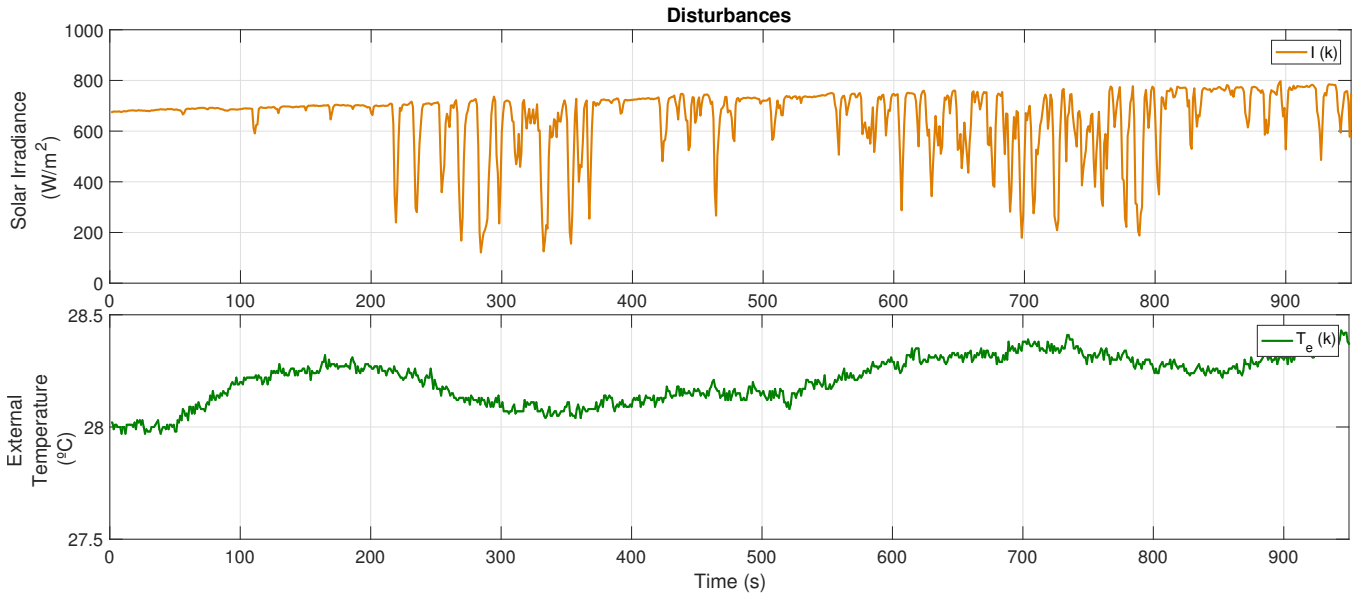

FIGURE 6 Disturbances.

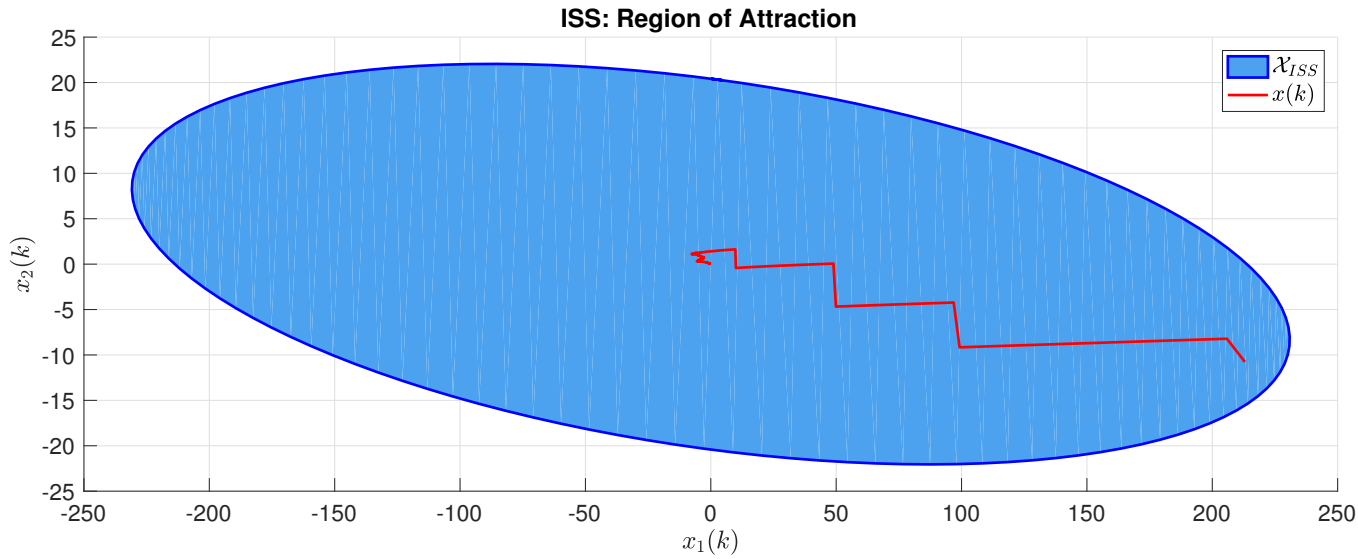

FIGURE 7 ISS Region of Attraction.

7.2 | Extrapolation of Scheduling Sequence

In Figure 9, we present the results concerning the extrapolation of the scheduling parameters ρ_1 and ρ_2 along the prediction horizon N_p. In this Figure, the dashed black line depicts the actual variation of $\rho(k)$, whilst the full blue line shows different snippets of scheduling sequences extrapolated according to the recursive algorithm in Eq. (32). The estimation error is quite small. Furthermore, the average time needed to solve the algorithm is of 0.41 ms, much smaller than the considered sampling period of 10 ms.

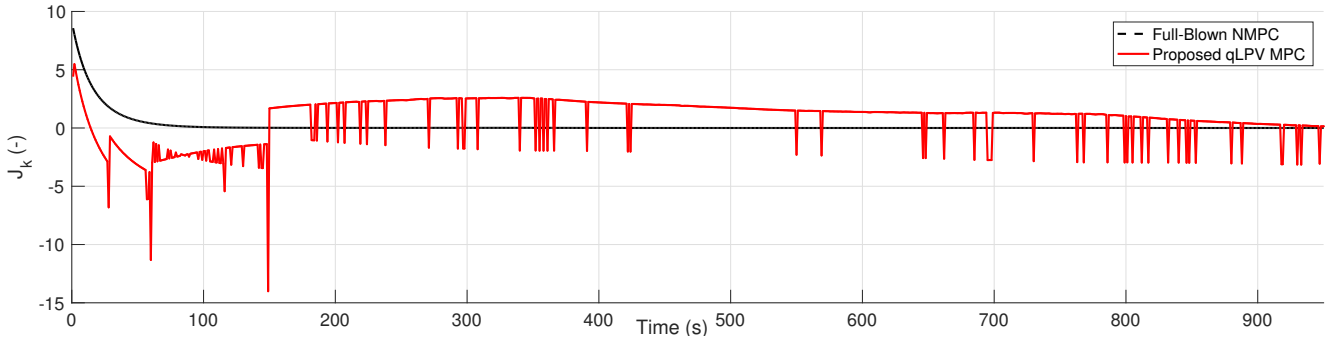

FIGURE 8 Decay of $J_k(\cdot)$ over time.

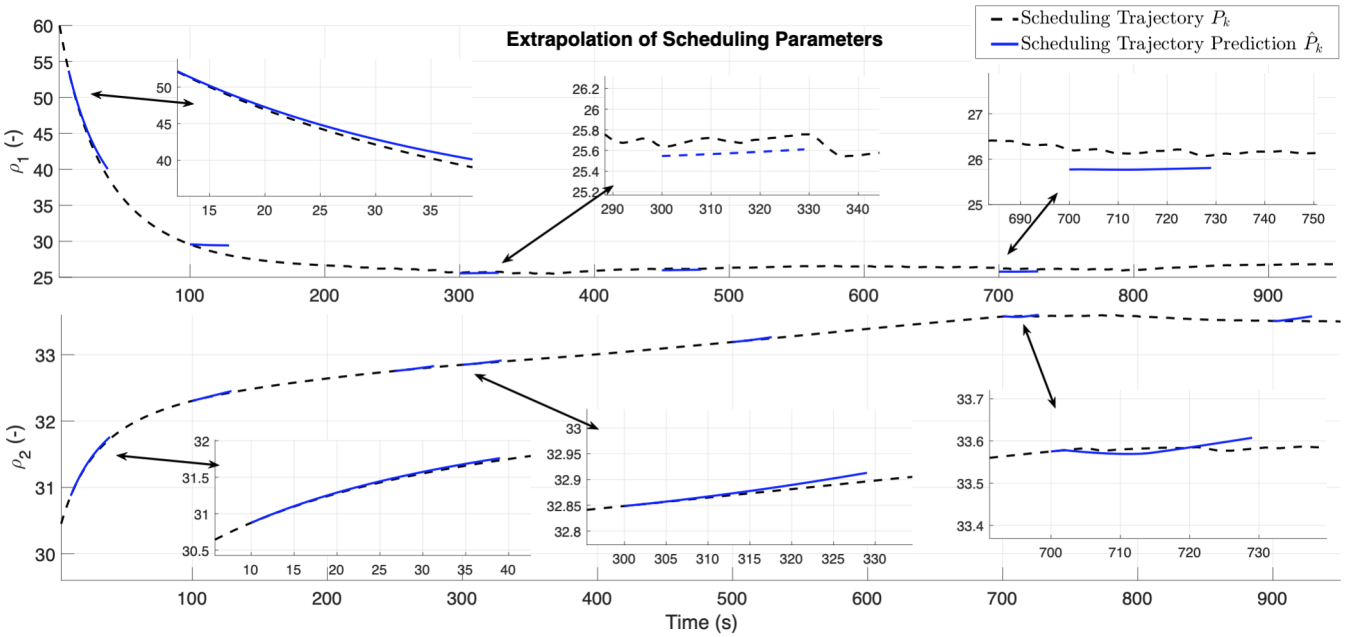

FIGURE 9 Scheduling Sequence Extrapolation.

7.3 | Regulation/Tracking Results

The results concerning the regulation of $x(k)$, with all the tested methodologies, are presented in Figure 10. We stress that all methods ensure state and control constraints ($x \in \mathcal{X}$ and $u \in \mathcal{U}$). The regulation of the states to the origin is not thoroughly ensured by the min-max methods by Cao et al. and Li et al., since their respective uncertainty sets $\mathcal{S}^{\text{Cao et al., 2005}}$ and $\mathcal{S}^{\text{Li et al., 2010}}$ are too large with respect to $\mathcal{X}$. We note that the first min-max method stabilised x to $(-66.05\,,\,-96.7)\,^{\circ}\text{C}$, while the second (bounded-rates) method brought the state trajectories to $(-65.95\,,\,-96.34)\,^{\circ}\text{C}$. The smoother performances seem to be the ones attained the Full-blown MPC algorithm, while the proposed method and the one by Cisneros & Werner yield quite comparable performances. We remark that the control action also acts to attenuate the effect of the load disturbances; this is especially evident after $t = 500\,\text{s}$, when both disturbances vary abruptly (see Figure 6).

The proposed method is able to ensure adequate results since its uncertainty set $\mathcal{S}$ is relatively small. Moreover, the uncertainty vector $\Sigma_k^\star$ computed through the maximisation CP norm-decreases over the simulation, as the extrapolation of the scheduling sequences get better (see Figure 9).

We proceed by investigating these performances through performance indexes: Tables 3 and 4 show, respectively, the root-mean square (RMS) and integral-of-the-absolute-error (IAE) indexes applied to $x_1(k)$ (plate temperature tracking) and $x_2(k)$

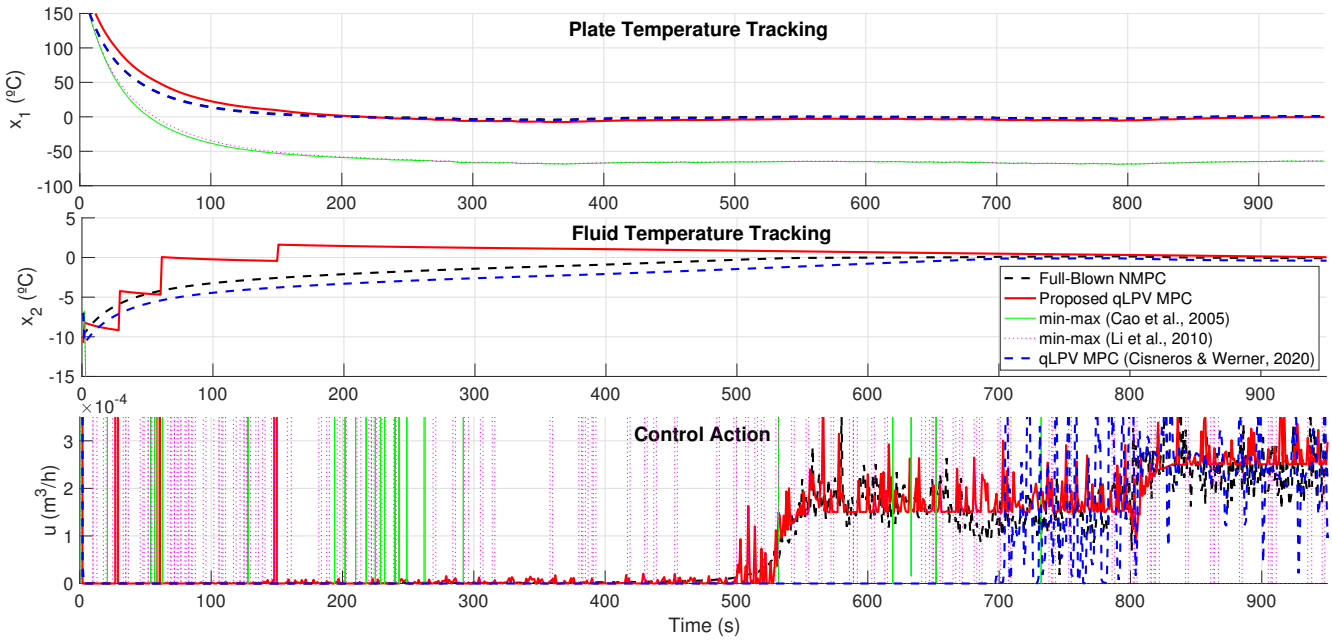

FIGURE 10 ST Unit: Reference Tracking and Control Signal.

TABLE 3 Performances Indexes: Plate Temperature Tracking (x_1).

Method	**IAE** ($.10^{-3}$)	**RMS**
Full-Blown NMPC[2]	7.6803	24.5295
Proposed qLPV MPC	11.5762	29.7982
min-max (Cao et al., 2005)[11]	60.5221	64.6373
min-max (Li et al., 2010)[13]	59.9912	64.2002
qLPV MPC (Cisneros & Werner, 2020)[25]	7.9062	24.6305

TABLE 4 Performances Indexes: Fluid Temperature Tracking (x_2).

Method	**IAE** ($.10^{-3}$)	**RMS**
Full-Blown NMPC[2]	1.1509	2.0402
Proposed qLPV MPC	1.0354	1.8946
min-max (Cao et al., 2005)[11]	93.0361	96.2502
min-max (Li et al., 2010)[13]	91.5767	94.8112
qLPV MPC (Cisneros & Werner, 2020)[25]	1.9396	2.8030

(fluid temperature tracking). We note that smaller IAE and RMS values indicate better performances, which conversely means that the references are tracked faster and with less steady-state error.

With respect to the regulation of x_1, these tables show that the performances achieved with the Full-Blown NMPC and the qLPV MPC by Cisneros & Werner are roughly equivalent in terms of RMS and IAE. The proposed method does not stay far behind, having slightly slower tracking in the first few seconds, which results in the settling seen by $t = 200$ s in Figure 10. It is important to notice that this fact resides in the maximisation procedure, which implies the robustness by finding larger uncertainty vectors $\Sigma_k^\star$ in these first moments, which reflect on the solution found by the minimisation QP and the slight difference to the other methods. Anyhow, we stress that the performances are comparable.

Owing to the regulation of x_2, it is seen that the IAE and RMS indexes indicate that the best tracking performances are obtained with the proposed method. As seen in Figure 10, the Full-Blown NMPC and the qLPV MPC method by Cisneros & Werner yield comparable results.

7.4 | Analysis of the Control Signal

Table 5 presents the TV index, which computes the total variance of the control input over time, this is:

$$\text{TV} := \sum |\delta u(k)| = \sum |u(k+1) - u(k)| . \tag{82}$$

Bigger values for the TV index shows that more variation is applied to the control along the simulation; therefore, values closer to zero indicate better (smoother) control strategies in terms of the use of the actuator.

It is seen that the smoother control values are obtained by the Full-Blown NMPC method, with the proposed method and the method by Cisneros & Wener not standing far behind. The min-max methods by Cao et al. and Li et al. present negligible results, at least for this ST application for which ρ has a big variation set $\mathcal{P}$ with also large possible variation rates (i.e. $\delta\mathcal{P}$ is also large).

TABLE 5 Total Variance of the Control Signal.

Method	**TV**
Full-Blown NMPC[2]	0.1987
Proposed qLPV MPC	0.2800
min-max (Cao et al., 2005)[11]	16.8449
min-max (Li et al., 2010)[13]	68.97871
qLPV MPC (Cisneros & Werner, 2020)[25]	0.2789

7.5 | Analysis of the Computational Stress

With respect to these results, we present a very important issue: the average computational time needed to solve the optimisation procedure of the methods are synthesised in Table 6. We recall that the sampling period of the system is of 10 ms (which is the computational time upper bound). Evidently, the Full-Blown NMPC needs a lot of time to solve its inherent NP, which means that this method is not applicable in practice for processes with small sampling periods. The results obtained with this method are purely numeric and would not be able to be applied in practice. The qLPV MPC method by Cisneros & Werner solves, in average, 5 QPs (it iterates the QPs to compute the extrapolation guess $\hat{P}_k$). The proposed method operates, in average, within 6.3 ms, spending 0.41 ms to make the extrapolation guess $\hat{P}_k$, 4.24 ms to solve the maximisation CP via fmincon and 1.65 ms via Gurobi. These are very interesting results, meaning that the proposed solution is indeed fast and able to operate for embedded applications. The performances of the proposed method are equivalent to the method by Cisneros & Werner, which operates in the millisecond range as well as the available modern NMPC solutions, such as ACADO and GRAMPC[48,49].

We stress that the obtained time performance depends on the operating computer machine and on the size of the controlled system. In this paper, the considered system is a 2×2 system, for which the max. CP and the min. QP are evaluated simply enough. For larger order models, sub-optimal solutions might be necessary; refer to a previous discussion on this matter[50].

TABLE 6 Computational Performance of the Controllers.

Method	Average Computational Time
Full-Blown NMPC[2]	776.50 ms
Proposed qLPV MPC	6.3 ms
min-max (Cao et al., 2005)[11]	7.2 ms
min-max (Li et al., 2010)[13]	7.5 ms
qLPV MPC (Cisneros & Werner, 2020)[25]	8.72 ms

8 | CONCLUSIONS

In this paper, a novel MPC algorithm for nonlinear system is proposed. The nonlinear system is embedded into a qLPV formulation and its scheduling parameters ρ are extrapolated using a recursive Taylor expansion law. The predictive control algorithm is based on a min-max optimisation procedure, written with respect to the uncertainty set derived by wrong estimates of ρ. The dissipativity of the proposed method is verified via an LMI-solvable remedy which ensures the Lyapunov-decrease of the stage cost and an Input-to-state stability region. The method is applied to the nonlinear temperature control problem of solar-thermal collector plates, exhibiting good performances.

With respect to the obtained results, some key points are mentioned:

- Full-blown nonlinear programming NMPC are not applicable for embedded applications of processes with fast sampling rates, since the average time needed to solve the NP is usually larger than the available sampling period. Recent literature has shown how approximated NMPC methods (such as CaSaDi, GRAMPC and ACADO[48]) and qLPV-embedding MPC algorithms[25] are able to efficiently solve such complex control problem in the range of milliseconds.

- For the considered case study, through IAE and RMS indexes, the reference tracking performances obtained with the proposed qLPV-embedding min-max MPC method are equivalent to these fast modern nonlinear MPC methods[25]. The numerical operability of the proposed. method is similar to previous works[48,25]. We note that the complexity of the problem grows with the order of the system.

- The proposed method solves the maximisation convex programming problem with respect to the error regarding the estimation of the scheduling parameters along the prediction horizon. We note that any kind of algorithm with bounded estimation errors could be used in the place of the Taylor expansion one proposed in this paper. An alternative and elegant option could be the use of the iterated mechanism[25], which uses the state sequence computed with the minimisation QP to compute the evolution of ρ along the horizon.

- The proposed method is compared to two keystone min-max LPV MPC algorithms from the literature[11,13], which consider, respectively, that ρ can vary arbitrarily inside $\mathcal{P}$ and considers bounded rates of variations for ρ. Since the variations of the scheduling parameters and its convex set are quite large for the considered application, the results obtained with these methods are quite poor. The uncertainty set with the proposed method is much smaller (by a factor of $1/100$). Furthermore, as time control law progresses, the extrapolation method gets better estimations of ρ, which also makes the uncertainty output of the maximisation problem to converge to zero, as the state trajectories converge.

- Finally, the method has ensured input-to-state stability for a larger regional domain $\mathcal{X}_{ISS}$. This property is ensure together with recursive feasibility through a dissipativity verification framework, solved via LMIs. We note that the advantage of this framework is that it does not require the use of terminal ingredients (constraints and costs) on the optimisation problem, which may be quite hard to compute online for LPV systems. Therefore, the MPC cost function is quadratic on x and u (and quite simple), which allows its fast operation.

ACKNOWLEDGMENTS

This work has been supported by CNPq project $304032/2019-0$ and ITEA3 European project 15016 EMPHYSIS (Embedded Systems With Physical Models in the Production Code Software).

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. Camacho EF, Bordons C. *Model predictive control*. Springer Science & Business Media . 2013.
2. Allgöwer F, Zheng A. *Nonlinear model predictive control*. 26. Birkhäuser . 2012.
3. Camacho EF, Bordons C. Nonlinear model predictive control: An introductory review. In: Springer. 2007 (pp. 1–16).
4. Tóth R. *Modeling and identification of linear parameter-varying systems*. 403. Springer . 2010.
5. Tóth R, Willems JC, Heuberger PS, Hof V. dPM. The behavioral approach to linear parameter-varying systems. *IEEE Transactions on Automatic Control* 2011; 56(11): 2499–2514.
6. Mohammadpour J, Scherer CW. *Control of linear parameter varying systems with applications*. Springer Science & Business Media . 2012.
7. 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.
8. Boyd S, El Ghaoui L, Feron E, Balakrishnan V. *Linear matrix inequalities in system and control theory*. 15. Siam . 1994.
9. Abbas HS, Toth R, Petreczky M, Meskin N, Mohammadpour J. Embedding of nonlinear systems in a linear parameter-varying representation. *IFAC Proceedings Volumes* 2014; 47(3): 6907–6913.
10. 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.
11. Cao YY, Lin Z. Min-max MPC algorithm for LPV systems subject to input saturation. *IET Proceedings-Control Theory and Applications* 2005; 152(3): 266–272.
12. Besselmann T, Löfberg J, Morari M. Explicit LPV-MPC with bounded rate of parameter variation. *IFAC Proceedings Volumes* 2009; 42(6): 7–12.
13. Li D, Xi Y. The feedback robust MPC for LPV systems with bounded rates of parameter changes. *IEEE Transactions on Automatic Control* 2010; 55(2): 503–507.
14. Jungers M, Oliveira RC, Peres PL. MPC for LPV systems with bounded parameter variations. *International Journal of Control* 2011; 84(1): 24–36.
15. Bumroongsri P. An offline formulation of MPC for LPV systems using linear matrix inequalities. *Journal of Applied Mathematics* 2014; 2014.
16. 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.
17. Hanema J, Tóth R, Lazar M. Tube-based anticipative model predictive control for linear parameter-varying systems. In: IEEE. ; 2016: 1458–1463.
18. Abbas HS, Männel G, Hoffmann nCH, Rostalski P. Tube-based model predictive control for linear parameter-varying systems with bounded rate of parameter variation. *Automatica* 2019; 107: 21–28.
19. Hanema J, Lazar M, Tóth R. Heterogeneously parameterized tube model predictive control for LPV systems. *Automatica* 2020; 111: 108622.

20. Cisneros PS, Voss S, Werner H. Efficient nonlinear model predictive control via quasi-LPV representation. In: IEEE. ; 2016: 3216–3221.

21. 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.

22. Mate S, Kodamana H, Bhartiya S, Nataraj PSV. A Stabilizing Sub-Optimal Model Predictive Control For Quasi-Linear Parameter Varying Systems. *IEEE Control Systems Letters* 2019.

23. Morato MM, Normey-Rico JE, Sename O. Novel qLPV MPC Design with Least-Squares Scheduling Prediction. *IFAC-PapersOnLine* 2019; 52(28): 158–163.

24. Alcalá E, Puig V, Quevedo J. LPV-MPC Control for Autonomous Vehicles. *IFAC-PapersOnLine* 2019; 52(28): 106–113.

25. 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; n/a(n/a).

26. Jiang ZP, Wang Y. Input-to-state stability for discrete-time nonlinear systems. *Automatica* 2001; 37(6): 857–869.

27. Magni L, Raimondo DM, Scattolini R. Regional input-to-state stability for nonlinear model predictive control. *IEEE Transactions on automatic control* 2006; 51(9): 1548–1553.

28. Blanchini F, Miani S. *Set-theoretic methods in control*. Springer . 2008.

29. Cisneros PS, Werner H. A dissipativity formulation for stability analysis of nonlinear and parameter dependent MPC. In: IEEE. ; 2018: 3894–3899.

30. Shamma JS. An overview of LPV systems. In: Springer. 2012 (pp. 3–26).

31. Sename O, Gaspar P, Bokor J. *Robust control and linear parameter varying approaches: application to vehicle dynamics*. 437. Springer . 2013.

32. Scorletti G, Fromion V, De Hillerin S. Toward nonlinear tracking and rejection using LPV control. *IFAC-PapersOnLine* 2015; 48(26): 13–18.

33. Mayne DQ. Model predictive control: Recent developments and future promise. *Automatica* 2014; 50(12): 2967–2986.

34. Limón D, Alamo T, Salas F, Camacho EF. Input to state stability of min–max MPC controllers for nonlinear systems with bounded uncertainties. *Automatica* 2006; 42(5): 797–803.

35. Lazar M, De La Peña DM, Heemels W, Alamo T. On input-to-state stability of min–max nonlinear model predictive control. *Systems & Control Letters* 2008; 57(1): 39–48.

36. He DF, Huang H, Chen QX. Quasi-min–max MPC for constrained nonlinear systems with guaranteed input-to-state stability. *Journal of the Franklin Institute* 2014; 351(6): 3405–3423.

37. Mayne DQ, Rawlings JB, Rao CV, Scokaert PO. Constrained model predictive control: Stability and optimality. *Automatica* 2000; 36(6): 789–814.

38. Seiler P, Packard A, Balas GJ. A dissipation inequality formulation for stability analysis with integral quadratic constraints. In: IEEE. ; 2010: 2304–2309.

39. Löfberg J. Oops! I cannot do it again: Testing for recursive feasibility in MPC. *Automatica* 2012; 48(3): 550–555.

40. Megretski A, Rantzer A. System analysis via integral quadratic constraints. *IEEE Transactions on Automatic Control* 1997; 42(6): 819–830.

41. Scherer CW. LPV control and full block multipliers. *Automatica* 2001; 37(3): 361–375.

42. Camacho EF, Berenguel M, Rubio FR, Martínez D. Control Issues in Solar Systems. In: Springer. 2012 (pp. 25–47).

43. Pasamontes M, Álvarez J, Guzman J, Berenguel M, Camacho E. Hybrid modeling of a solar-thermal heating facility. *Solar Energy* 2013; 97: 577–590.

44. Morato MM, Mendes PR, Normey-Rico JE, Bordons C. LPV-MPC fault-tolerant energy management strategy for renewable microgrids. *International Journal of Electrical Power & Energy Systems* 2020; 117: 105644.

45. Ampuño G, Roca L, Gil JD, Berenguel M, Normey-Rico JE. Apparent delay analysis for a flat-plate solar field model designed for control purposes. *Solar Energy* 2019; 177: 241–254.

46. Pipino HA, Morato MM, Bernardi E, Adam EJ, Normey-Rico JE. Nonlinear temperature regulation of solar collectors with a fast adaptive polytopic LPV MPC formulation. *Solar Energy* 2020; 209: 214–225.

47. Vergara-Dietrich JD, Morato MM, Mendes PRC, Cani AA, Normey-Rico JE, Bordons C. Advanced Chance-Constrained Predictive Control for the Efficient Energy Management of Renewable Power Systems. *Journal of Process Control* 2019.

48. Quirynen R, Vukov M, Zanon M, Diehl M. Autogenerating microsecond solvers for nonlinear MPC: a tutorial using ACADO integrators. *Optimal Control Applications and Methods* 2015; 36(5): 685–704.

49. Englert T, Völz A, Mesmer F, Rhein S, Graichen K. A software framework for embedded nonlinear model predictive control using a gradient-based augmented Lagrangian approach (GRAMPC). *Optimization and Engineering* 2019; 20(3): 769–809.

50. Zhang Y, Li S, Liao L. Near-optimal control of nonlinear dynamical systems: A brief survey. *Annual Reviews in Control* 2019.

How to cite this article: M. M. Morato, J.E. Normey-Rico, and O. Sename (2021), A Fast Dissipative Robust Nonlinear MPC Procedure via qLPV Embedding and Parameter Extrapolation, *International Journal of Robust and Nonlinear Control*.

APPENDIX

A NOMENCLATURE AND SYMBOLOGY

Acronyms	
MPC	Model Predictive Control
LPV	Linear Parameter Varying
LTI	Linear Time-Invariant
QP	Quadratic Programming Problem
NMPC	Nonlinear Model Predictive Control
NP	Nonlinear Programming Problem
qLPV	Quasi-linear Parameter Varying
LDI	Linear Differential Inclusion
ISS	Input-to-State Stability
ISpS	Input-to-State Practical Stability
LMI	Linear Matrix Inequality
2^{nd}OCP	Second-order Cone Programming Problem
CP	Convex Programming Problem

Variables	
N_p	Prediction horizon
J_k	MPC quadratic cost
Q, R	MPC weighting matrices
ℓ	Stage cost

x	States
Δx	Deviation of the states
u	Control inputs
ρ	Scheduling parameters
$\delta\rho$	Scheduling parameters' rates of variations
w	Load disturbances
$\mathcal{X}$	State admissibility set
$\mathcal{U}$	Control input admissibility set
$\mathcal{P}$	Scheduling set
$\mathcal{W}$	Disturbance set
$\mathbf{X}_f$	Terminal state set
V	Terminal offset cost
U_k	Future control actions vector
$H(x,u,k)$	LDI matrix
G	Process representation
$f_\rho(\cdot)$	Scheduling proxy
Γ	Lipschitz constant
X_k	Future state evolution
P_k	Future scheduling sequence
ξ_ρ	Scheduling sequence estimation error
$H(P_k)$	Hessian of J_k
$g(P_k,x)$	Gradient of J_k
f_ρ^∂	Scheduling proxy derivative
λ, ν	Forgetting factors
σ	Model-process mismatch uncertainties
$\mathcal{S}$	Uncertainty set
Σ_k	Uncertainty vector
$\mathcal{X}_{ISS}$	ISS set
ϕ_k	Nonlinear MPC operator
$\Psi(P_k)$	Parameter-dependency extraction operator
$G^n_{I_1}$	Nominal LTI model of augmented system
Θ	Scheduling dependency operator
C_f	Fluid heat transfer capacity
ϵ_f	Fluid density
C_m	Collector plates heat transfer capacity
ϵ_m	Metal plate density
ν	Thermal loss coefficient
h_0, h_i	Heat transfer coefficients
d_e, d_i	Diameters
T_f, T_p	Fluid and plate temperatures
I	Solar irradiance
T_e	External temperature

B PROOF OF PROPOSITION 2

The convergence property can be demonstrated with the aid of the residual term $\xi_\rho(k+j)$, which should turn null. We demonstrate this for $j = 1$; the proof for the following steps is equivalent. Considering that the system is regulated by an MPC controller, and since it is robustly stable despite the residual term, it holds that $\lim_{k\to\infty} x(k+1) = x(k)$. Then, take $\xi_\rho(k) = f_\rho(x(k+1)) - f_\rho(x(k)) - f_\rho^\partial \Delta x(k)$. Due to the stabilisation implied by the MPC, it directly follows that $\lim_{k\to\infty} f_\rho(x(k+1)) = \lim_{k\to\infty} f_\rho(x(k))$ and $\lim_{k\to\infty} \Delta x(k) = 0$. Then, $\lim_{k\to\infty} \xi_\rho(k) = -\lim_{k\to\infty} f_\rho^\partial \Delta x(k) \to 0$. This concludes the proof.

C PROOF OF PROPOSITION 3

Firstly, note that at any given moment ahead of k, the scheduling parameters can be given as:

$$\rho(k+j+1) = \rho(k+j) + f_\rho(x(k+j+1)) - f_\rho(x(k+j)) \;\forall\, k+j \in \mathbb{N}, \tag{C1}$$

whereas the "extrapolated" value is:

$$\hat{\rho}(k+j+1) = \rho(k+j) + \hat{f}_\rho(x(k+j+1)) - f_\rho(x(k+j)) \;\forall\, k+j \in \mathbb{N}, \tag{C2}$$

where $\hat{f}_\rho(\cdot)$ represents the Taylor expansion. The residual ξ_ρ is given by the difference between Eqs. (C1) and (C2), as follows:

$$\begin{aligned} \xi_\rho(k+j+1) &= f_\rho(x(k+j+1)) - \hat{f}_\rho(x(k+j+1)), \\ &= f_\rho(x(k+j+1)) - f_\rho(x(k+j)) - f_\rho^\partial \Delta x(k+j). \end{aligned} \tag{C3}$$

Computing the ultimate bounds on the estimation error:

$$\|\xi_\rho(k+j+1)\| = \|f_\rho(x(k+j+1)) - f_\rho(x(k+j)) - f_\rho^\partial \Delta x(k+j)\|. \tag{C4}$$

Due to the Triangular inequality:

$$\|\xi_\rho(k+j+1)\| \leq \|f_\rho(x(k+j+1)) - f_\rho(x(k+j))\| + \|f_\rho^\partial \Delta x(k+j)\|. \tag{C5}$$

Due to the local Lipschitz property of $f_\rho(\cdot)$ (Assumption 7), it follows:

$$\|\xi_\rho(k+j+1)\| \leq \Gamma\|\Delta x(k+j)\| + \|f_\rho^\partial \Delta x(k+j)\|. \tag{C6}$$

Since f_ρ^∂ is ultimately bounded (Assumption 13), i.e. $\|f_\rho^\partial\| \leq \overline{f_\rho^\partial}$, it holds that:

$$\xi_\rho^{\text{bound}} = \left(\Gamma + \overline{f_\rho^\partial}\right)\overline{\Delta x}. \tag{C7}$$

Note that the ultimate bound on the state deviation is computed with respect to Assumption 10. Moreover, since $\xi_\rho^{\text{bound}} < \overline{\rho}$, $\mathcal{E} \subset \mathcal{P}$. This concludes the proof.

D PROOF OF LEMMA 1

The compactness of $\mathcal{X}$, $\mathcal{U}$ and $\mathcal{P}$ imply that any predicted evolution of the system states X_k and sequence of control actions U_k are bounded. This fact guarantees that the optimal (maximised) cost J_k^{bound} is upper bounded, i.e. there exists a finite real value $\overline{J_k}$ s.t. $J_k^{\text{bound}} \leq \overline{J_k}$ for all $x \in \mathcal{X}_{\text{Max CP}}$.

In the virtue of the previous discussion and by optimality, it is implied that the sequence of control inputs U_k is feasible (note that the min. QP is assumably feasible). Thence, we denote $\Delta J_k^{\text{bound}}$ as the $\left(J_k^{\text{bound}} - J_{k-1}^{\text{bound}}\right)$, i.e. the difference between the worst-case cost functions at two consecutive instants. For all $x \in \mathcal{X}_{\text{Max CP}}$, we consider that the MPC is formulated under a feasible time-varying state-feedback $u(k) = \kappa(k)x(k)$, as gives Assumption 4 and Eq. (29), and, thus:

$$\Delta J_1^{\text{bound}} \leq \max_{\Sigma_k}\left\{\frac{1}{2}\Sigma_k^T H_\sigma \Sigma_k - \Sigma_k^T g_\sigma(\hat{P}_k, x(k))\right\} \leq \beta_J(\overline{\sigma}).$$

Assume that $\Delta J_k^{\text{bound}} \leq \beta_J(\overline{\sigma})$ for all $x \in \mathcal{X}_{\text{Max CP}}$. Consider that the control action is well defined (resulting from the min QP). Then, it follows that:

$$J_{k+1}^{\text{bound}} \leq \max_{\Sigma_k}\left\{\frac{1}{2}\Sigma_k^T H_\sigma \Sigma_k - \Sigma_k^T g_\sigma(\hat{P}_k, x(k)) + J_k^{\text{bound}}\right\},$$

and, thus, we arrive at:

$$\begin{aligned} \Delta J_{k+1}^{\text{bound}} &\leq \max_{\Sigma_k}\left\{\frac{1}{2}\Sigma_k^T H_\sigma \Sigma_k - \Sigma_k^T g_\sigma(\hat{P}_k, x(k)) + J_k^{\text{bound}}\right\} - \max_{\Sigma_k}\left\{\frac{1}{2}\Sigma_k^T H_\sigma \Sigma_k - \Sigma_k^T g_\sigma(\hat{P}_k, x(k)) + J_{k-1}^{\text{bound}}\right\} \\ &\leq \max_{\Sigma_k}\left\{\frac{1}{2}\Sigma_k^T H_\sigma \Sigma_k - \Sigma_k^T g_\sigma(\hat{P}_k, x(k)) + J_k^{\text{bound}} - \frac{1}{2}\Sigma_k^T H_\sigma \Sigma_k + \Sigma_k^T g_\sigma(\hat{P}_k, x(k)) - J_{k-1}^{\text{bound}}\right\} \\ &= \max_{\Sigma_k}\left\{\Delta J_k^{\text{bound}}\right\} \leq \beta_J(\overline{\sigma}). \end{aligned}$$

Therefore, by induction, it is inferred that the decay of the worst-case cost function is upper bounded for all $k \geq 0$, considering that the evolution of x belongs to the feasibility set $\mathcal{X}_{\text{Max CP}}$.

Consider that the system state $x(k)$ is measured and the min-max algorithm computes a control action $u(k) = \kappa(k)x(k)$. Then, the system is driven to $x(k+1)$. Since we consider that $x(k) \in \mathcal{X}_{\text{Max CP}}$ and the model-process uncertainties are upper-bounded (by $\overline{\sigma}$), it is clear that the $x(k+1) \in \mathcal{X}_{\text{Max CP}}$. This is valid for a feasible min. QP (which is demonstrated in the sequel). By the monotonicity results, it also follows that:

$$\begin{aligned}
\Delta J_{k+1}^{\text{bound}} &= J_{k+1}^{\text{bound}} - \max_{\Sigma_k} \left\{ \frac{1}{2}\Sigma_k^T H_\sigma \Sigma_k - \Sigma_k^T g_\sigma(\hat{P}_k, x(k)) + J_k^{\text{bound}} + J_{k-1}^{\text{bound}} \right\} \\
&= \Delta J_k^{\text{bound}} - \frac{1}{2}\Sigma_k^T H_\sigma \Sigma_k + \Sigma_k^T g_\sigma(\hat{P}_k, x(k)) \\
&\leq -\frac{1}{2}\Sigma_k^T H_\sigma \Sigma_k + \Sigma_k^T g_\sigma(\hat{P}_k, x(k)) + \beta_J(\overline{\sigma}) \,.
\end{aligned}$$

E PROOF OF LEMMA 2

Firstly, we define the quadratic Lyapunov function $V = x^T P x$. Then, we pre-multiply and post-multiply the LMI (54) by $\begin{bmatrix} x^T & \phi^T & u_\rho^T \end{bmatrix}$ and $\begin{bmatrix} x^T & \phi^T & u_\rho^T \end{bmatrix}^T$, respectively. This yields the following inequality:

$$(V(k+1) - V(k)) - \left(\tau z_g^T M_z z_g\right) + \left(\begin{bmatrix} u_\rho(k) & y_\rho(k) \end{bmatrix}^T \begin{bmatrix} \mathbb{I} \\ 0 \end{bmatrix}^T M_\Theta \begin{bmatrix} \mathbb{I} \\ 0 \end{bmatrix} \begin{bmatrix} u_\rho(k) & y_\rho(k) \end{bmatrix} \right) < 0 \,. \tag{E8}$$

Then, we can substitute $u_\rho(k) := \Theta y_\rho(k)$ which yields:

$$(V(k+1) - V(k)) - \tau \left(z_g^T M_z z_g\right) + \left(u_\rho(k)^T \begin{bmatrix} \mathbb{I} \\ \Theta \end{bmatrix}^T M_\Theta \begin{bmatrix} \mathbb{I} \\ \Theta \end{bmatrix} u_\rho(k) \right) < 0 \,. \tag{E9}$$

Due to inequality (52) the term $\left(z_g^T M_z z_g\right)$ is implied as negative. Equivalently,

$$\left(u_\rho(k)^T \begin{bmatrix} \mathbb{I} \\ \Theta \end{bmatrix}^T M_\Theta \begin{bmatrix} \mathbb{I} \\ \Theta \end{bmatrix} u_\rho(k) \right)$$

is implied as positive due to the structure of M_Θ. Therefore, for any $\tau > 0$, it holds that:

$$V(k+1) - V(k) < 0 \,, \tag{E10}$$

which means that the propose storage function is a Lyapunov function for the system and, thus, for any starting condition $x_0 \in \mathcal{X}_{ISS}$, local asymptotical stabilisation to origin of the state-space is ensured by the MPC policy.

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 International Journal of Electrical Power & Energy Systems, 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.

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.