



HAL
open science

A predictive fault tolerant control method for qLPV systems subject to input faults and constraints

Marcelo Menezes Morato, Marc Jungers, Julio Elias Normey-Rico, Olivier Sename

► To cite this version:

Marcelo Menezes Morato, Marc Jungers, Julio Elias Normey-Rico, Olivier Sename. A predictive fault tolerant control method for qLPV systems subject to input faults and constraints. *Journal of The Franklin Institute*, 2022, 359 (16), pp.9129-9167. 10.1016/j.jfranklin.2022.09.011 . hal-03787455

HAL Id: hal-03787455

<https://hal.science/hal-03787455v1>

Submitted on 26 Sep 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial - NoDerivatives 4.0 International License



HAL
open science

A Predictive Fault Tolerant Control Method for qLPV Systems Subject to Input Faults and Constraints

Marcelo Menezes Morato, Marc Jungers, Julio Elias Normey-Rico, Olivier Sename

► **To cite this version:**

Marcelo Menezes Morato, Marc Jungers, Julio Elias Normey-Rico, Olivier Sename. A Predictive Fault Tolerant Control Method for qLPV Systems Subject to Input Faults and Constraints. Journal of The Franklin Institute, Elsevier, 2022, 10.1016/j.jfranklin.2022.09.011 . hal-03787455

HAL Id: hal-03787455

<https://hal.archives-ouvertes.fr/hal-03787455>

Submitted on 26 Sep 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Journal Pre-proof

A Predictive Fault Tolerant Control Method for qLPV Systems Subject to Input Faults and Constraints

Marcelo Menezes Morato, Marc Jungers, Julio E. Normey-Rico, Olivier Sename

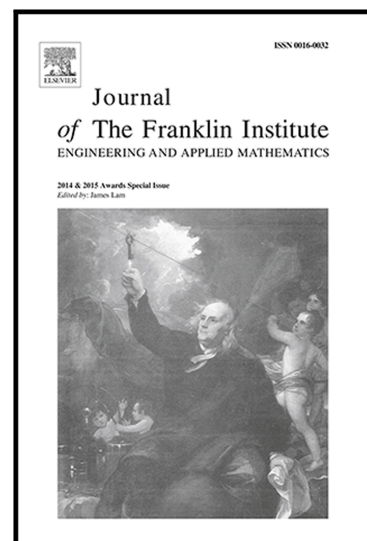
PII: S0016-0032(22)00652-4
DOI: <https://doi.org/10.1016/j.jfranklin.2022.09.011>
Reference: FI 5816

To appear in: *Journal of the Franklin Institute*

Received date: 10 December 2021
Revised date: 16 June 2022
Accepted date: 14 September 2022

Please cite this article as: Marcelo Menezes Morato, Marc Jungers, Julio E. Normey-Rico, Olivier Sename, A Predictive Fault Tolerant Control Method for qLPV Systems Subject to Input Faults and Constraints, *Journal of the Franklin Institute* (2022), doi: <https://doi.org/10.1016/j.jfranklin.2022.09.011>

This is a PDF file of an article that has undergone enhancements after acceptance, such as the addition of a cover page and metadata, and formatting for readability, but it is not yet the definitive version of record. This version will undergo additional copyediting, typesetting and review before it is published in its final form, but we are providing this version to give early visibility of the article. Please note that, during the production process, errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.



A Predictive Fault Tolerant Control Method for qLPV Systems Subject to Input Faults and Constraints

Marcelo Menezes Morato^{a,b}, Marc Jungers^c, Julio E. Normey-Rico^a, Olivier Sename^b

^a*Departamento de Automação e Sistemas, Universidade Federal de Santa Catarina, Florianópolis, Brazil*

^b*Univ. Grenoble Alpes, CNRS, Grenoble INP[†], GIPSA-lab, 38000 Grenoble, France.*

[†]*Institute of Engineering Univ. Grenoble Alpes.*

^c*Université de Lorraine, CNRS, CRAN, F-54000 Nancy, France.*

Abstract

In this paper, we investigate the use of Model Predictive Control (MPC) applications for quasi-Linear Parameter Varying (qLPV) systems subject to faults along the input channels. We propose a Fault Tolerant Control (FTC) mechanism based on a robust state-feedback MPC synthesis, considering polytopic inclusions. In order to alleviate the numerical burden of the robust min-max procedure, we use small prediction horizons, in such a way that the solution becomes viable for real-time systems. The FTC system is able to tolerate time-varying saturation of the actuator, which may happen due to malfunctions. Recursive feasibility and poly-quadratic stability guarantees are ensured through the synthesis of adequate terminal ingredients. Accordingly, we present a catalogue of three different LMI remedies, considering: (a) parameter-independent ingredients, (b) a parameter-dependent terms and (c) a parameter-dependent maps that take into account bounded rates of parameter variation. An autonomous driving car example is used to illustrate the performances of the proposed technique, which is compared to other MPCs from the literature. The proposed FTC method is able to ensure good performances, obtained with reduced computational demand.

Keywords: Fault Tolerant Control, Model Predictive Control, Linear Parameter Varying Systems, Small Horizon, Saturating Actuator.

1. Introduction

Real instrumented control systems are naturally subject to possible faults, failures and component malfunctions [1]. Accordingly, over the last couple of years, a considerable amount of attention has been given to the synthesis of Fault Tolerant Control (FTC) schemes [2, 3, 4]. These algorithms are able to offer increased process availability by avoiding breakdowns and maintaining performances despite faults.

FTC laws can be synthesis in either passive or active formalisms [5]. In active FTC techniques [6, 7], a fault detection (FD) scheme is required to quantify the level of faults, while in passive techniques, no FD layer is required. Currently, there exist different passive FTC approaches, such as fault hiding [8], and fault accommodation [9, 10]. Under the scope of the latter, the use of Model Predictive Control (MPC) schemes has gained special attention [11, 12], especially due to the widespread acceptance and maturity of predictive control.

In MPC, an optimal control action is computed online by the means of a constrained optimisation problem, written in terms of a process model, constraints, and a performance cost. The original MPC algorithms were conceived for Linear Time-Invariant (LTI) prediction models. In this case, MPC generates online Quadratic Programs (QPs), which are solved rapidly with many modern solvers [13, 14], enabling real-time applications. Anyhow, when using nonlinear prediction models, MPC renders Nonlinear Programs (NPs), which are much more complex and difficult to solve [15].

Email address: marcelomzm@gmail.com (Marcelo Menezes Morato)

Anyhow, many nonlinear processes can be modelled through quasi-Linear Parameter Varying (qLPV) structures, for which the state transition depends on a scheduling law [16, 17]. In this paper, we are interested in the application of a passive fault tolerant MPC for these systems. Specifically, we consider the case of input fault along the input channels, which are represented as time-varying input saturation constraints, as in [18, 19]. **The major difficulties that arise when dealing with these systems are: (i) how to counteract the input saturation that occurs due to faults¹, and (ii) how to correctly predict the process behaviour despite the unavailability of the future scheduling variables ρ .**

The recent survey paper [20] discusses the possibilities of issuing (nonlinear) predictive control through qLPV model structures, detailing the available solutions to the synthesis of these control laws for this type of problems. As mentioned therein, the unavailability of ρ adds a complexity-optimality trade-off barrier for the designer: either one chooses to tune a robust MPC [21, 22, 23], which complicates real-time implementation due to numerical complexity, or to disregard stabilisation guarantees and design a sub-optimal QP [24, 25], which makes the implementation as fast as any LTI MPC, but may lead to algorithm infeasibility or performance degradation (in the presence of sudden faults, these may even be unstable).

1.1. Problem Statement

Motivated by this context, the main goal of this paper is to develop a passive fault tolerant control scheme for these qLPV systems subject to input faults. These faults may occur due to malfunctions in the actuator system, thus leading to deteriorated performances if not correctly accounted for by the controller. Therefore, we consider the following control objectives: (1) to operate fast enough for time-critical applications, (2) to ensure closed-loop stability and sufficient performances, despite the input-saturation faults, and (3) to ensure robustness against the unavailable scheduling variables and the fault terms.

1.2. Proposed Method

The proposed method is a state-feedback model predictive control algorithm. The method consists in a robust min-max procedure, using an heuristically-tuned small prediction horizon to alleviate numerical burden. The method considers a fault-free polytopic qLPV model subject to “fault-induced” input saturation constraints, as done in [19, 26]. By using special terminal ingredients with sector conditions related to these saturation constraints, the method is able to ensure poly-quadratic stability and recursive feasibility of the optimisation, despite faults.

1.3. Contributions

The main innovation of this paper lies in providing an input-fault tolerant predictive control framework that serves for time-critical applications, does not resort to sub-optimality, and continuously maintains stability and feasibility properties, despite faults. Accordingly, the main contributions of this paper are:

- A catalogue of three different LMI-solvable remedies for the computation of the MPC terminal ingredients, ensuring poly-quadratic stability of the closed-loop and recursive feasibility of the MPC algorithm, despite input saturation that may appear due to faults/malfunctions (**Theorem 3.8**). This catalogue comprises:
 - (a) parameter independent / quadratic terminal ingredients (**Corollary 3.9**);
 - (b) parameter dependent terminal terms, considering arbitrarily fast rates of parameter variation (**Corollary 3.10**);
 - and (c) parameter dependent terminal maps, taking into account bounded rates of parameter variation (**Corollary 3.11**).

¹As demonstrated in the sequel, we note that these fault-induced input constraints are time-varying and unknown. Thereof, the controller accounts for an estimate of the fault scenario, which corresponds to a constant saturation limit for each input channel.

- Realistic numeric simulation results of the vertical dynamics of a vehicle are presented (Section 4). This application serves to demonstrate the effectiveness and simplicity of proposed FTC algorithm when compared to other MPC algorithms from the literature. Standard indexes are used to quantify performances and certify the reduced computational complexity.

We note that our method does not resort to control reconfiguration when faults occur. **Different from prior fault tolerant MPCs from the literature [11, 6, 7], we use a fixed control algorithm, with no changes on constraints or performance goals during the implementation.** The fault tolerance is made viable through the adequate terminal ingredients (main contribution), which ensure stabilisation even in faulty conditions, expressed as an input-related set. The verification of recursive feasibility and stability, thus, become much simpler than when control reconfiguration happens, which may even unstabilize the system if not correctly accounted for.

1.4. Paper Organisation

The rest of this paper is organised as follows. In Sec. 2, the problem setup is presented, considering the class of polytopic qLPV models with fault-induced input saturation. Sec. 3 provides the catalogue of LMI-solvable remedies used to compute the MPC terminal ingredients. These ingredients also offer an estimate for the set of initial conditions which allow the convergence of the algorithm. The simulation results are provided in Sec. 4. General conclusions are drawn in Sec. 5.

Notation: $\mathbb{Z}_{[a,b]}$ stands for the set integer numbers within $[a, b]$. $\nu(k+i|k)$ is used to represent a predicted value of variable ν for time instant $k+i$, computed at instant k . An ellipsoid is denoted as follows: for a symmetric positive definite matrix $M \in \mathbb{R}^{n_x \times n_x}$, and a positive scalar $\mu > 0$, we obtain an ellipsoid $\mathcal{E}(M, \mu) := \{x \in \mathbb{R}^{n_x} : x^T M x \leq \mu\}$. \mathcal{K} refers to the class of positive and strictly increasing scalar functions that pass through the origin.

2. Problem Setup

2.1. Process Model

We consider the class of discrete-time qLPV processes subject to fault-induced input saturation. These systems are represented by Eq. (1) and henceforth assumed to satisfy the Assumptions presented in the sequel:

$$\begin{aligned} x(k+1) &= A(\rho(k))x(k) + B(\rho(k))\text{sat}_{f(k)}\{u(k)\}, \forall k \in \mathbb{N}, \\ y(k) &= C(\rho(k))x(k) + D(\rho(k))\text{sat}_{f(k)}\{u(k)\}, \forall k \in \mathbb{N}, \\ \rho(k) &= f_\rho(x(k)) \in \mathcal{P} \subseteq \mathbb{R}^{n_\rho}, \forall k \in \mathbb{N}, \\ \partial\rho(k) &= \rho(k) - \rho(k-1) \in \partial\mathcal{P}, \forall k \in \mathbb{N}, \end{aligned} \tag{1}$$

where $x \in \mathbb{R}^{n_x}$ is the vector of system states, $u \in \mathbb{R}^{n_u}$ the vector of control inputs, $y \in \mathbb{R}^{n_y}$ the vector of controlled outputs, and $\rho \in \mathbb{R}^{n_\rho}$ the vector of scheduling parameters. The fault-induced saturation function is

$$\text{sat}_{f(k)}\{u(k)\} = \text{col}\{\text{sign}\{u_j(k)\} \min\{|u_j(k)|, \bar{u}_j f_j(k)\}\}, \forall j \in \mathbb{Z}_{[1, n_u]}.$$

Assumption 2.1. States are measurable for all sampling instants $k \in \mathbb{N}$.

Even though the states $x(k)$ are assumed to be measured, we aim at formulating a state-feedback control policy in such a way that the outputs $y(k)$ behave in a certain way, according to objectives. Furthermore, we stress that these outputs are not necessarily equivalent to $x(k)$, but rather a time-varying combination of them, as gives Eq. (1).

Assumption 2.2. The fault term $f \in \mathbb{R}^{n_u}$ is a vector $\text{col}\{f_1, \dots, f_{n_u}\}$, for which each entry $f_j \in [0, 1]$ represents the loss of effectiveness that is induced at the corresponding j -th control input.

Next, we detail the validity of Assumption 2.2 and how the fault-induced saturation happens: consider that a given system exhibits a fault on its first actuator, quantified by $f_1 = 0.8$. This fault term implies that the first control entry u_1 has its performance compromised by 20%. Therefore, the corresponding control input space is decreased, and thus we observe a saturation phenomenon. In a fault-less condition ($f_1 = 1$), we have $\text{sat}_f\{u\} \in [-\bar{u}_j, \bar{u}_j]$, whereas in the faulty case (with $f_1 = 0.8$), it is implied that $\text{sat}_f\{u\} \in [-0.8\bar{u}_j, 0.8\bar{u}_j]$.

As a concrete example of how such fault-induced actuator saturation phenomenon happens in practice, we refer to the case of electro-rheological suspension dampers subject to faults, as modelled in [27]. When there is an unexpected electro-rheological fluid leakage, the amount of force that the damper delivers gets decreased by a loss of effectiveness factor which is related to the amount of lost fluid. In turn, one can represent the damper force (actuation) as expected input saturated by an upper-bound which is related to the fault (thus fault-induced). In [26], this phenomenon is studied and a fault estimation scheme is developed.

Assumption 2.3. *The scheduling parameters are known, bounded, and have bounded rates of variations, considering $\rho \in \mathcal{P}$ and $\partial\rho \in \partial\mathcal{P}$, where:*

$$\mathcal{P} := \left\{ \rho_j \in \mathbb{R} \mid \underline{\rho}_j \leq \rho_j \leq \bar{\rho}_j, \forall j = 1, \dots, n_\rho \right\}. \quad (2)$$

$$\partial\mathcal{P} := \left\{ \partial\rho_j \in \mathbb{R} \mid \underline{\partial\rho}_j \leq \partial\rho_j \leq \bar{\partial\rho}_j, \forall j = 1, \dots, n_\rho \right\}. \quad (3)$$

Remark 1. Bounded rates of scheduling parameter variations are a fair hypothesis for any practical application. The scheduling sets \mathcal{P} and $\partial\mathcal{P}$ are assumed independent of time, and neither depend on the current state value.

Assumption 2.4. *The matrix pair $(A(\rho), B(\rho))$ from the qLPV system in Eq. (1) is stabilizable for all possible scheduling variable values, i.e. $\forall \rho \in \mathcal{P}$.*

Assumption 2.5. *Matrices $A(\cdot)$, $B(\cdot)$, $C(\cdot)$ and $D(\cdot)$ in Eq. (1) are polytopic on ρ , which means that they can be expressed as follows:*

$$\begin{aligned} A(\rho) &= \sum_{j=1}^{2^{n_\rho}} \gamma_j(\rho) A_j, & B(\rho) &= \sum_{j=1}^{2^{n_\rho}} \gamma_j(\rho) B_j, \\ C(\rho) &= \sum_{j=1}^{2^{n_\rho}} \gamma_j(\rho) C_j, & D(\rho) &= \sum_{j=1}^{2^{n_\rho}} \gamma_j(\rho) D_j, \end{aligned}$$

where A_j , B_j , C_j and D_j are constant matrices and each of the 2^{n_ρ} combination variables $\gamma_j(\rho)$ satisfies $\sum_{j=1}^{2^{n_\rho}} \gamma_j(\rho) = 1$ and that each $\gamma_j(\rho) \in [0, 1]$. For this class of polytopic qLPV systems, it is thus possible to normalise and restrict the set \mathcal{P} to the unit simplex of dimension 2^{n_ρ} . The system polytope of 2^{n_ρ} vertices is henceforth denoted Ω .

The regulation of this process must be such that the state, output, and input variables are always constrained within the following admissibility sets, respectively:

$$\mathcal{X} := \left\{ x \in \mathbb{R}^{n_x} \mid \|x_j\| \leq \bar{x}_j, \forall j \in \mathbb{Z}_{[1, n_x]} \right\}. \quad (4)$$

$$\mathcal{Y} := \left\{ y \in \mathbb{R}^{n_y} \mid \|y_j\| \leq \bar{y}_j, \forall j \in \mathbb{Z}_{[1, n_y]} \right\}. \quad (5)$$

$$\mathcal{U} := \left\{ u \in \mathbb{R}^{n_u} \mid \|u_j\| \leq \bar{u}_j, \forall j \in \mathbb{Z}_{[1, n_u]} \right\}. \quad (6)$$

2.2. Faults and Fault-induced Input Saturation

In this paper, as previously discussed, we consider that the qLPV system is subject to faults along the input channel, as represented by the time-varying saturation function in Eq. (1). These faults may happen due to unpredicted conditions or malfunctions on the actuation system of this process.

The time-varying fault term $f(k)$ in Eq. (1) is unknown by definition. Nevertheless, we assume to know the inferior bounds of each fault entry, this is $\underline{f}_j := \inf_{k \in \mathbb{N}} f_j(k)$. Each inferior bound \underline{f}_j gives the maximal loss of effectiveness of the corresponding j -th input.

Remark 2. We highlight our recent paper [27], wherein we use this fault representation framework to model loss of effectiveness in electro-rheological dampers. Stating that \underline{f}_j is known, in accordance with this framework, given that we can relate the maximal loss on a given actuator due to physical phenomena. For instance, in that studied damper system, $\underline{f}_j = 0.2$ corresponds to the leakage of a related proportion of the electro-rheological fluid from the damper chamber. We also mention that the case of vehicle braking systems [28] also exhibit this kind of fault-related behaviour: the maximal deliverable braking (control input) has an input space which naturally shrinks over time, due to lifespan characteristics. A braking system is considered in this paper as a case study to apply the proposed FTC method (see Sec. 4).

For control purposes, we take into account the bound \underline{f}_j in order to build a “faulty” input set, which is used to represent the system under faulty conditions. By doing so, we are able to use a simpler model in the synthesis step.

Consider the fault-less input admissibility set \mathcal{U} , which corresponds to the “wider” input limits implied by $\text{sat}_{[1, \dots, 1]} \{u\}$. Then, the corresponding faulty set is given by:

$$\mathcal{U}^+ := \{u \in \mathbb{R}^{n_u} \mid \|u_j\| \leq \bar{u}_j^+, \forall j \in \mathbb{Z}_{[1, n_u]}\}, \quad (7)$$

where $\bar{u}_j^+ = \underline{f}_j \bar{u}_j$. Note that \mathcal{U}^+ is a shrunk version of the original admissibility set \mathcal{U} , embedding the effects of a stricter saturation $\text{sat}_f \{u\}$. In practice, the conservative restriction $u \in \mathcal{U}^+$ will be imposed as the larger constraint when the failures occur and when the values for each term f_j are unknown. This is, in fact, the restriction that is robust with respect to the unknown fault. Furthermore, it should be noted that the linear part of the saturation does not depend on the value of the fault. We emphasise that the constraints $\|u_j\| \leq \bar{u}_j^+, \forall j \in \mathbb{Z}_{[1, n_u]}$ are what we call the additional/“fault-induced” input saturation.

Figure 1 illustrates the loss of effectiveness fault-induced saturation phenomenon. In fault-less situations, we observe $f_j = 1$ and $\text{sat}_f \{u\} \in \mathcal{U}$. On the contrary, when any $f_j \neq 1$, a tightened saturation is enacted over the corresponding control input u_j (and thus $\text{sat}_f \{u\} \in \mathcal{U}^+$). Note how \mathcal{U}^+ is a “shrunk” version of the original input admissibility set \mathcal{U} .

Taking the previous discussion into account, we henceforth use two distinct realisations of the system from Eq. (1): (a) a fault-less model, and (b) fault-embedded model based on \underline{f} . Respectively, they are as follows:

- Fault-less conditions (equivalent to $f(k) = \text{diag}\{1, \dots, 1\}$ in Eq. (1)):

$$\begin{aligned} x(k+1) &= A(\rho(k))x(k) + B(\rho(k))u(k), \\ y(k) &= C(\rho(k))x(k) + D(\rho(k))u(k), \end{aligned} \quad (8)$$

- Faulty situations (whenever any $f_j(k) \neq 1$ in Eq. (1)):

$$\begin{aligned} x(k+1) &= A(\rho(k))x(k) + B(\rho(k))\text{sat}_f \{u(k)\}, \\ y(k) &= C(\rho(k))x(k) + D(\rho(k))\text{sat}_f \{u(k)\}. \end{aligned} \quad (9)$$

Note that $\text{sat}_{f(k)} \{\text{sat}_f \{u(k)\}\} = \text{sat}_f \{u(k)\}$, which means that $\text{sat}_{f(k)} \{\cdot\}$ is transparent from the viewpoint of the saturated input $\text{sat}_f \{u(k)\}$. Thus, synthesising a control system based on the worse-case fault-induced saturation model from Eq. (9) implies in a control input that is robust regarding

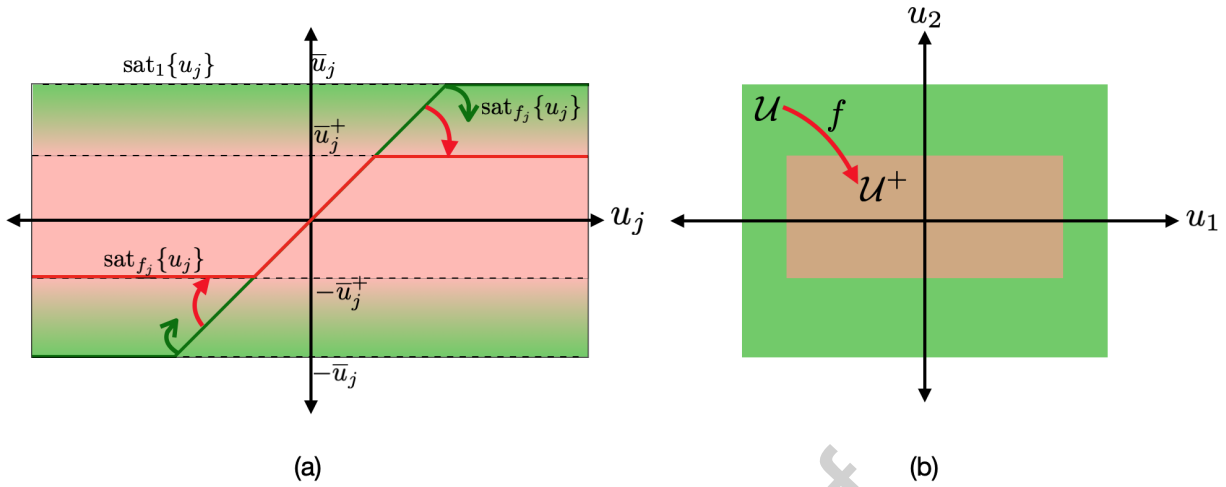


Figure 1: Illustration of the fault-induced saturation phenomenon: (a) depicts the effects over a single input channel u_j , considering a fault-less case ($sat_1\{u_j\}$, wider green set), a time-varying fault-induced saturation ($sat_{f_j}\{u_j\}$, transition zone between the two sets), and the “worst-case” fault-induced saturation ($sat_{f_j}\{u_j\}$, smaller red set). (b) shows the fault-related sets \mathcal{U} and \mathcal{U}^+ over two inputs.

the original fault model from Eq. (1). This is: a control synthesis that stabilises Eq. (9) will ensure the corresponding stabilisation of Eq. (1).

We cannot use both these previous models (fault-less, Eq. (8), and fault-embedded, Eq. (9)) for synthesis. Therefore, despite \mathcal{U}^+ being known, there is no need to consider Eq. (9) as the real dynamics of the system, since given the stricter saturation (i.e. $f_j(k) \neq 1$) only occurs under faulty conditions.

Thus, in this paper, we are interested with the design of fault accommodation controller, which “does not know” when faults happen and what is the value of the fault terms $f(k)$ (we consider no FDI mechanism, and thus a passive FTC). Nevertheless, the controller should ensure the aimed performance goals while satisfying all process constraints. With regard to the previous models, we assume that the time-varying $f(k)$ is neither known nor estimated, whereas only \underline{f} is known (a lower bound metric for these faults).

In any case, we proceed by proposing a control synthesis formulation based on Eq. (8), while using a complementary input $v(k)$ to include the saturation $sat_{f_j}\{\cdot\}$ from Eq. (9). The proposed control system, thus, takes into account the nominal constraints (represented by \mathcal{U}) and the fault-less model (Eq. (8)). Then, stability and recursive feasibility properties of the closed-loop system are analysed altogether with the input saturation caused by faults. Accordingly, in order to overlap the saturation function, we benefit from a sector condition, as discussed in Section 3.

2.3. Formulation of the passive Fault Tolerant MPC

In order to control this process w.r.t. a given performance goal, despite possible faults along the input channel, we propose a fault tolerant control scheme based on robust MPC. This paradigm is illustrated by the block diagram in Fig. 2: we consider that a polytopic qLPV system is subject to input faults, which coordinate the input saturation; the states are measured and, thus, a robust state-feedback predictive control law is used to ensure that performance guidelines are met, given with regard to the output variables. The MPC operates based on nominal constraints (of a fault-less condition), a known performance goal, and terminal ingredients. The latter serve to ensure poly-quadratic stability of the system, as well as recursive feasibility of the algorithm.

In an LTI setting, MPC laws are a result of quadratic optimisation problems, which are “numerically cheap”, being solvable in real-time (within a few microseconds). Anyhow, when qLPV models (as Eq. (1))

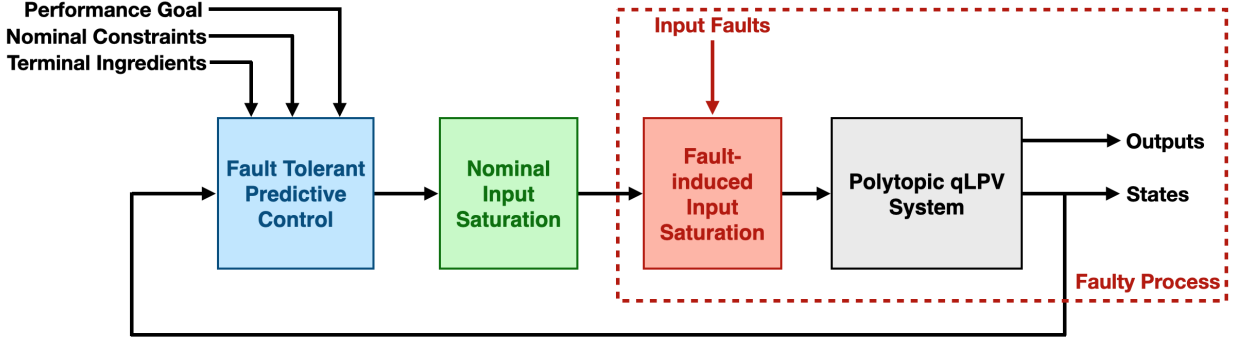


Figure 2: Block diagram of the proposed FTC scheme.

are considered, the MPC optimisation gets much more complicated, since unknown variables appear in the process predictions, e.g., at sampling instant $k + 2$:

$$x(k + 2|k) = A(\overbrace{\rho(k+1)}^{\text{unknown}})A(\rho(k))x(k) + A(\overbrace{\rho(k+1)}^{\text{unknown}})B(\rho(k))u(k|k) + B(\overbrace{\rho(k+1)}^{\text{unknown}})u(k + 1|k).$$

Thus, in order to properly control this process, we consider a min-max MPC procedure. In this robust setting, the future values of the scheduling variables $\rho(k + j)$, $\forall j \in \mathbb{Z}_{[1, N_p]}$ are treated as uncertainties. Thence, the control law is obtained by minimising the “worst-case” performance cost $\bar{J} = \max_{\partial\rho(k+i) \in \partial\mathcal{P}} J$, induced by the uncertainties [29]. The procedure during the implementation is the following:

$$U_k^* = \arg \min_{U_k} \left(\max_{\partial\rho(k+i) \in \partial\mathcal{P}} \left(\overbrace{\sum_{i=1}^{N_p} \ell(x(k+i|k), u(k+i-1|k))}^J + V(x(k+N_p|k)) \right) \right) \quad (10)$$

$$\begin{aligned} \text{s.t.} \quad & \overbrace{\begin{aligned} x(k+i+1|k) &= A(\rho(k+i))x(k+i|k) + B(\rho(k+i))u(k+i|k) \\ y(k+i|k) &= C(\rho(k+i))x(k+i|k) + D(\rho(k+i))u(k+i|k) \end{aligned}}^{\text{qLPV Process Model}}, \forall i \in \mathbb{Z}_{[1, N_p]}, \\ & \overbrace{u(k+i-1|k) \in \mathcal{U}}^{\text{Control Input Admissibility}}, \forall i \in \mathbb{Z}_{[1, N_p]}, \\ & \overbrace{x(k+i|k) \in \mathcal{X}}^{\text{Admissible Process Operation}}, \forall i \in \mathbb{Z}_{[1, N_p]}, \\ & \overbrace{y(k+i-1|k) \in \mathcal{Y}}^{\text{Admissible Output Values}}, \forall i \in \mathbb{Z}_{[1, N_p]}, \\ & \overbrace{x(k+N_p|k) \in \mathbf{X}_f}^{\text{Terminal Set Constraint}}, \end{aligned}$$

where \mathbf{X}_f and $V(\cdot)$ are the so-called “terminal ingredients”. In this min-max setting, it is implied that $\rho(k+i) = \rho(k) + \sum_{j=1}^i \partial\rho(k+j)$, with $\partial\rho(k+i) \in \partial\mathcal{P}$. Therefore, the computational cost is reduced w.r.t. the case of arbitrarily fast variations for ρ (i.e. $\partial\rho$ unbounded) [30]. For regularity purposes, thanks to the definition of the scheduling parameter $\rho(\cdot)$ in Eq. (8), we have simultaneously $\rho \in \mathcal{P}$, $\partial\rho \in \partial\mathcal{P}$ and $\rho + \partial\rho \in \mathcal{P}$. In the sequel, we detail the choice of the prediction horizon N_p and the considered output-related performance objectives.

As previously stated, we use the fault-less model from Eq. (8) in the MPC formulation given in Eq. (10). The fault-related aturation is disregarded in the nominal MPC predictions, but will be later on used to generate appropriate terminal ingredients which provide robustness certificates (regarding corresponding the faulty model from Eq. (9)).

Remark 3. In this paper, we deal with a state-feedback fault tolerant MPC, with output-related steady-state conditions [15]. Thus, the objective of the control law is related to the output variables. Specifically, we aim to find a control law in such a way that the outputs $y(k)$ track a generic output-related target goal y_r . For such, it is implicitly implied by the considered model that the states should be admissibly steered to a steady-state condition x_r , and the control input to a steady-state regime u_r , being x_r and u_r related to y_r (as detailed in the sequel, in Assumption 2.6).

Remark 4. The resulting robust MPC framework holds a close relationship with several robust MPCs for systems with norm-bounded uncertainty, given that, at each sampling instant, the qLPV predictions are linear with unknown (but bounded) future parameter values. However, our formulation allows to have lighter notations and take better into account the fact that each component may have a physical interpretation, which may not have the interval of definition. We exploit Assumption 2.3 in order to take into account non-symmetrical intervals that will be not possible with normed variables ρ without a suitable change of variable.

2.4. Performance Goal

In order to generalise the application of our method to the case of reference tracking of constant steady-points, we consider the following Assumption:

Assumption 2.6. *The qLPV system in Eq. (8) satisfies the following set of nonlinear equations (in a fault-less condition):*

$$\begin{bmatrix} A(f_\rho(x_r, u_r)) - \mathbb{I}_{n_x} & B(f_\rho(x_r, u_r)) \\ C(f_\rho(x_r, u_r)) & D(f_\rho(x_r, u_r)) \end{bmatrix} \begin{pmatrix} x_r \\ u_r \end{pmatrix} = \begin{bmatrix} 0_{n_x \times 1} \\ y_r \end{bmatrix}, \quad (11)$$

where the admissible solution $p_r^x = (x_r, u_r)$ implies in an output steady-state operation point y_r contained inside \mathcal{Y} .

Thus, we tune the proposed fault tolerant control scheme by taking into account an admissible solution (x_r, u_r) that satisfies² Assumption 2.6. Thus, we use a performance cost to induce the convergence of the states $x(k+i|k)$ to the steady-state state reference target x_r , whilst forcing $u(k+i-1|k)$ towards the steady-state input condition u_r . This conversely ensures that the output variables $y(k+i|k)$ track y_r . Thereof, the performance cost of the robust MPC is taken as follows:

$$\ell(x(k+i|k), u(k+i-1|k)) = \|x(k+i|k) - x_r\|_Q^2 + \|u(k+i-1|k) - u_r\|_R^2, \quad (12)$$

being Q and R the positive definite weighting matrices. Since R is positive definite, the induced cost in Eq. (12) is convex in u . On the basis of Assumption 2.1, the control signal can be expressed as state-feedback action, which also implies that this stage cost is convex in x . Due to the linearity in the state transition of the qLPV model in Eq. (8), it holds that the minimisation problem derived under this cost is indeed a QP, since $\rho(k+j)$ are passed from the maximisation NP. The terminal cost $V(x(k+N_p|k), \rho(k+N_p))$ is likewise set to weight the deviances from the target operation, w.r.t. the N_p steps ahead sample:

$$V(x, \rho) = \|x - x_r\|_{P(\rho)}^2 = (x - x_r)^T P(\rho) (x - x_r). \quad (13)$$

The weighting matrix $P(\rho)$ in Eq. (13) is symmetric and positive-definite, while it could be parameter-dependent or parameter-independent with regard to ρ , according to the chosen stabilisation approach. This matter is addressed in further details in the following Section, where we provide a catalogue of LMIs for construction of $P(\cdot)$, according to the degree of information available regarding the controlled process.

²The solution to Eq. (11) can be provided by an offline reference planner algorithm [25]. In the case of time-varying reference targets, we recommend other approaches [15, 31], which are out of the scope of this paper.

2.5. Prediction Horizon Heuristic

The maximisation procedure $\max_{\partial\rho(k+i)\in\partial\rho} J$ subject to constraints is computationally difficult (NP-hard). Indeed, as evidenced in many papers [21, 29], this is a tough complexity barrier of min-max MPCs. The computational complexity of the maximisation procedure grows exponentially with respect to the dimension of the controlled process and the size of the prediction horizon N_p . In many cases, this combination yields algorithms that are not solvable for real-time applications. One alternative is to replace the NP maximisation by an approximated quadratic program [31], with real-time iterations or gradient-based methods. Nevertheless, this complexity reduction comes at the expense of sub-optimality.

In order to maintain optimality of the proposed fault tolerant control method, we use an alternative option to reduce the numerical burden of evaluating $\max_{\partial\rho(k+i)\in\partial\rho} J$: we impose a threshold to the size of the prediction horizon N_p , which is chosen heuristically³. This heuristic has been explored in the Author's paper [32], wherein we discuss and exploit this limitation on N_p for a semi-active suspension system.

As indicated in previous papers [32, 29], there is a Pareto optimality issue behind the choice of N_p [33]: while better performances are obtained with larger horizons (smaller averaged J in Eq. (10)), more computational time is required to implement the min-max strategy (i.e. to solve the optimisation in Eq. (10)). Thus, we choose N_p in such a way that the mean computational time required by the used solver t_c does not surpass the sampling period T_s , in order for the strategy to be implemented in real-time. This is expressed as follows:

$$N_p = \left\{ \arg \min_{N_p} \{J \text{ in Eq. (10)}\} \mid t_c < T_s \right\}, \quad (14)$$

being N_p the maximal horizon size, t_c the average computational time needed by the embedded solver to analytically evaluate Eq. (10) and T_s the sampling period of the discrete-time process. The solution to this heuristic criterion is derived with bisection algorithms, as in [34], since line-search may provide a non-maximal N_p .

The main idea of this heuristic is to obtain a good compromise between performance and computational cost, so that the fault tolerant MPC algorithm can be used in real-time control applications. We note that the horizon size N_p will depend on the processor, on the used solver and on the dimensions of the controlled process. For the majority of time-critical processes (for which T_s is given in the millisecond range), N_p is simply of a few steps.

We note that the size of the horizon is not changed under faulty conditions. In practice, the fault is tolerated by the means of adequate terminal ingredients, and the choice of a small prediction horizon does not compromise stability nor fault tolerance. The only drawback of the proposed heuristic is that performances can be deteriorated w.r.t. those obtained with a "full-blown" nonlinear MPC. Nevertheless, we note that the execution of an NMPC algorithm in real-time is computationally unattractive because of its general nonlinear dependence of the predicted states on the future control inputs and states [25].

3. A Catalogue of Terminal Ingredients

In this Section, we present a catalogue of three different LMI-solvable remedies for the computation of the terminal ingredients ($V(\cdot)$ and \mathbf{X}_f). These ingredients are used to ensure stability of the closed-loop, despite faults, as well as the recursive feasibility of the optimisation. We stress that these ingredients are the tools that ensure the fault tolerance of the proposed method, and, thus, the main contribution of this work.

Robustness against faults is ensured if the controller is able to ensure performance even though there are discrepancies between the model predictions and the real system trajectories. With regard to this matter, we consider three different cases, which should be selected according the amount of information available on the process:

³We acknowledge that some works [15] use two horizons, one for state predictions (N_p) and another one for the control sequence (N_u). Taking $N_u < N_p$ can help alleviating numerical burden, but since we consider constraints on states and outputs, the effect of a long N_p remains quite significant.

1. The parameters are assumed to vary arbitrarily between sampling instants, i.e. $\partial\rho$ unbounded, and when the control signal quadratically stabilises the process. This yields parameter independent terminal ingredients.
2. The parameters vary arbitrarily between the sampling instants, but the control signal poly-quadratically stabilises the process. This yields ρ -dependent terminal ingredients.
3. The parameters are such that there are known bound rates of variations between sampling instants, i.e. $\partial\rho \in \partial\mathcal{P}$, and the terminal ingredients are also ρ -dependent, ensuring poly-quadratically stabilisation of the faulty system.

We consider these three cases in order to expand the generalisation of the proposed fault tolerant MPC for a wide range of applications. In Section 4, we show simulation results considering these three cases in order to demonstrate how further robustness can be guaranteed (a wider basin of attraction) if more information is available regarding the process (third case).

The choice of which of these three cases is considered only influences on the synthesis of the terminal ingredients, made viable through Theorem 3.8, which should be in accordance with each case. Accordingly, we prove three Corollaries (3.9-3.11) that adapt this Theorem to the context of each case. Nevertheless, we first present generic Theorems for recursive feasibility and stability, which are incorporated by Theorem 3.8.

3.1. Recursive Feasibility

Recursive feasibility of the algorithm is essential in order to ensure that if an initial control sequence U_0^* is computationally tractable, the following control sequences will also be tractable, meaning that the fault tolerant control problem is well-posed and continuously solvable. This property is addressed by the following Theorem, from [35], which provides some sufficient conditions.

Theorem 3.1. *Stabilising Recursively Feasible MPC, from [35]*

Let Assumptions 2.1 and 2.4 hold. Assume that a nominal state-feedback control law $u(k) = K(\rho(k))x(k)$ exists. Consider that the MPC policy is formulated through the optimisation problem in Eq. (10), with stage cost $\ell(\cdot)$ that is quadratic and instantaneously convex on x . Assume that there exists a quadratic and convex terminal stage cost $V(x, \rho)$ and a terminal state domain $\mathbf{X}_f(\rho)$. Then, closed-loop stability is ensured if the following conditions hold $\forall \rho \in \mathcal{P}$, under axioms A1 and A2:

(C1) The origin lies in the interior of $\mathbf{X}_f(\rho)$;

(C2) Any consecutive state to x , in closed-loop given by $(A(\rho) + B(\rho)K)x$ lies within $\mathbf{X}_f(\rho)$;

(C3) The discrete-time Lyapunov inequality is verified within this invariant set, this is, $\forall \rho, \rho^+ \in \mathcal{P}$ and $\forall x \in \mathbf{X}_f(\rho)$:

$$V((A(\rho) + B(\rho)K(\rho))x, \rho^+) - V(x, \rho) \leq -x^T Qx - x^T K^T(\rho)RK(\rho)x. \quad (15)$$

(C4) The image of the nominal feedback lies within the admissible control domain: $K(\rho)x \in \mathcal{U}, \forall \rho \in \mathcal{P}$ and $\forall x \in \mathcal{X}$.

(C5) The terminal set $\mathbf{X}_f(\rho)$ is a subset of \mathcal{X} .

Assuming that the initial solution of the MPC problem U_k^* , computed with respect to an initial state $x(0)$, is feasible, then, the MPC algorithm yields recursively feasible programs.

Axioms:

A1) $\ell(x, K(\rho)x) \geq \beta_1(\|x\|), \forall x \in \mathcal{X}, \beta_1(\cdot) \in \text{class } \mathcal{K}$.

A2) $0 \leq V(x, \rho) \leq \beta_2(\|x\|), \forall x \in \mathcal{X}, \beta_2(\cdot) \in \text{class } \mathcal{K}$. Furthermore, $V(x) > 0$ holds for $x \neq 0$.

Proof. This is a standard Theorem, whose Proof is found by ensuring a energy-dissipative decay of the MPC cost function, see e.g. [25, Proof of Theorem 1] and [35, Axioms A1-A4]. \square

Remark 5. As in [25], we drop the non-null steady-state arguments (x_r, u_r) to prove stabilisation and recursive feasibility. This means that we verify Theorem 3.1 with $x_r = 0_{n_x}$ and $u_r = 0_{n_u}$, for simplicity. This change of variables is accompanied by an according change on the scheduling parameter $\rho(x(k))$, which becomes $\tilde{\rho}(k) = \rho(x(k) + x_r)$ in order to preserve the physical interpretation of the system. We also note that if the origin is stabilizable and x_r is contained inside the admissible set \mathcal{X} , it also becomes stabilizable, refer to e.g. [15, Theorems 1, 2 and 3]. This discussion is accounted for in Lemma 3.2.

Remark 6. Condition (C3) in Theorem 3.1 is the only one that is explicitly dependent on the time instant k . This is due to the fact that the terminal cost function $V(x, \rho)$ depends on the time-varying scheduling parameters ρ . Thus, the discrete-time Lyapunov difference inequality (15) requires to consider values for the scheduling parameter at different time steps. $V(\cdot)$ is expressed in this condition as parameter-dependent, i.e. $V(x, \rho)$. This notation is given to encompass both kinds (parameter-dependent and independent) of terminal costs. Regarding the regulation purpose in the polytopic setting, it follows that condition (C3) should hold for all 2^{n_ρ} vertices of the process polytope Ω . If (C3) is solved using a common Lyapunov matrix P , quadratic stability is ensured, which is a more conservative approach. If (C3) is solved under a parameter-dependent matrix $P(\rho)$, a lighter requirement is sought, which leads to a less conservative control strategy.

3.2. Stability

Lemma 3.2. *Closed-loop steady-state, adapted from [15]*

Consider the polytopic qLPV system described by Eq. (8) and subject to the process operational constraints $x(k+j) \in \mathcal{X}$, $u(k+j-1) \in \mathcal{U}$, $y(k+j-1) \in \mathcal{Y}$ and to the terminal constraint $x(k+N_p|k) \in \mathbf{X}_f$. Consider that a nominal feedback gain $K(\rho(k)) \in \mathbb{R}^{n_u \times n_x}$ is used and let $P(\rho(k))$ be a Lyapunov matrix such that (C3) from Theorem 3.1 holds for all vertices of Ω and for some given definite positive matrices $Q \in \mathbb{R}^{n_x \times n_x}$ and $R \in \mathbb{R}^{n_u \times n_u}$. Take an admissible constant steady-state reference condition for $x(k)$, namely $x_r \in \mathcal{X}$. Then, if for a given initial state condition $x(0)$, the optimal solution of the constrained quadratic optimisation problem in Eq. (10) is such that $\|x(0) - x_r^*\|_Q^2 = 0$, which implies that $\|x_r^* - x_r\|_P^2 = 0$.

Theorem 3.3. *Stability, adapted from [15]*

Consider that Assumptions 2.1, 2.4 and 2.6 hold. Let the Lyapunov condition (C3) in Theorem 3.1 also hold. Assume that the recursive feasibility has been verified for the FTC policy. Consider that the output steady-state y_r and state steady-state x_r , respectively belong to \mathcal{Y} and \mathcal{X} (this is, they are admissible targets). Then, for any feasible initial state $x_0 \in \mathcal{X}$, the proposed MPC control policy steers the system states to x_r in an admissible way.

The proofs of Lemma 3.2 and Theorem 3.3 are provided in the Appendixes of this paper, for brevity. It is implied by Theorem 3.3 that the set of admissible steady-states that can be tracked without error is $\mathcal{X}_r \subseteq \mathcal{X}$. Since the evolutions of the system states remain inside the feasibility set \mathcal{X} , any sequence of piecewise admissible targets within \mathcal{X}_r can be tracked without steady-state error. If the desired steady-state target is not admissible, i.e. $x_r \notin \mathcal{X}$, then the controller will not be able to steer the system to it⁴.

3.3. Region of Attraction

We now discuss how recursively feasibility and stability are ensured even in the situations of faults along the input channel. In a “normal situation” (with input saturation $u \in \mathcal{U}$), these properties are directly ensured by the optimisation in Eq. (10). Anyhow, under a “faulty situation”, when an actuator fault that leads to additional input saturation occurs, we must demonstrate that certificates are available despite this additional clipping ($u \in \mathcal{U}^+$, refer to Eq. (7)).

Consider that the MPC is represented by a state-feedback $u(k) = K(\rho(k))x(k)$. Then, the corresponding closed-loop dynamics should be analysed for the broader case, when faults are present. Thus, we must take

⁴Another closest admissible steady-state x_r^* could be fostered to be a stabilisable condition if the MPC cost function includes an extended artificial reference condition, as argued in [15].

the saturation term $\text{sat}_{\underline{f}}\{K(\rho(k))x(k)\}$ into account. For such, consider the following dead-zone function from [36]:

$$\Psi(u(k)) = \text{col}\{u_j(k) - \text{sat}_{\underline{f}}\{u_j(k)\}\} = u(k) - \text{sat}_{\underline{f}}\{u(k)\}. \quad (16)$$

Eq. (16) facilitates the calculation of a guess for the domain of attraction of the proposed fault tolerant MPC method, in faulty conditions. Coupling the dead-zone function to Eq. (9) generates the following state dynamics:

$$x(k+1) = A_{cl}(\rho(k))x(k) - B(\rho(k))\Psi(K(\rho(k))x(k)), \quad (17)$$

$$A_{cl}(\rho(k)) = A(\rho(k)) + B(\rho(k))K(\rho(k)). \quad (18)$$

It holds that $A_{cl_j} = A_j + B_jK$ for each vertex j of the polytope Ω , if the state-feedback gain is independent of ρ . If the state-feedback is parameter-dependent, i.e. $K = K(\rho)$, it follows that $A_{cl_{\{i,j\}}} = A_i + B_iK_j$, where i and j are vertices of Ω in this case, $A_{cl_{\{i,j\}}}$ is defined over a $(n_\rho)^2$ grid, since $A(\rho) = \sum_i \gamma_i A_i$, $B(\rho) = \sum_i \gamma_i B_i$ and $K(\rho) = \sum_j \gamma_j K_j$.

Following the lines of [37], we consider a signal $v(k) = Gx(k)$ as an auxiliary degree-of-freedom used for analysis purposes. This signal represents “a corrective signal”, which, subtracted to the nominal state-feedback, yields a signal that respects the saturation bounds. The set \mathcal{S} is defined as the set within which lie the system states for when $u(k) - v(k)$ is norm bounded by $\text{col}\{\bar{u}_j^+\}$, this is:

$$\mathcal{S} = \{x \in \mathbb{R}^{n_x} \mid \|(K - G)x\| \leq \text{col}\{\bar{u}_j^+\}\}. \quad (19)$$

In order to account for \mathcal{S} , we use the generalised “sector condition” [38], which defines a region for which the signal $(u(k) - v(k))$ does not saturate.

Lemma 3.4. *Sector Condition, Lemma 1 in [38]*

If the signal $(u(k) - v(k)) \in \mathcal{U}^+$ and, equivalently $x(k) \in \mathcal{S}$, then, it holds that:

$$\Psi(u(k))^T S (\Psi(u(k)) - v(k)) \leq 0, \quad (20)$$

for every diagonal positive definite matrix $S \in \mathbb{R}^{n_u \times n_u}$.

Lemma 3.5. *Lyapunov Sector Condition [37, 15]*

In order to ensure condition (C3) of Theorem 3.1 of the predictive control framework, the associated Lyapunov condition must include the Sector Condition from Lemma 3.4, this is, $\forall x \in \mathbf{X}_f$ and $\forall \rho, \rho^+ \in \mathcal{P}$, the following adapted discrete-time Lyapunov inequality must be verified:

$$V((A(\rho) + B(\rho)K)x, \rho^+) - V(x, \rho) + x^T Q x + x^T K^T R K x \leq SC, \quad (21)$$

$$2(\Psi(Kx))^T S (\Psi(Kx) - Gx) = SC. \quad (22)$$

Proof. Consider that inequality (21) is satisfied. If $x \in \mathcal{S}$, then, thanks to Lemma 3.4, the generalised sector condition (20) holds and, thus, (21) implies (C3) in Theorem 3.1. Note that \mathcal{S} implies in a local condition, defined by the region the input saturates. This concludes the proof. \square

Remark 7. We stress that the considered sector condition from Eq. (20) is conceived with regard to the faulty model from Eq. (9). This means that the dead-zone function $\Psi(u)$ given in Eq. (16) is expressed with regard to the constant saturation function $\text{sat}_{\underline{f}}$. Since it is implied that $\underline{f} \leq f(k)$, the used sector condition is a conservative bound (since $\text{sat}_{\underline{f}}$ represents a “stronger” saturation than $\text{sat}_{f(k)}$). Note that SC in (21) is a sufficient upper bound for a corresponding sector condition related to $\text{sat}_{f(k)}$. Nevertheless, since $f(k)$ is not known, this conservative sector is what remains.

Definition 3.6. Region of Attraction

The set of all initial conditions $x_0 = x(0)$ that result in converging trajectories is denoted $\mathcal{R}_A \subseteq \mathbb{R}^{n_x}$, and called the region (basin or domain) of attraction in closed-loop of the fault tolerant MPC controller $u(k) = K(\rho(k))x(k)$.

The task of analytically determining \mathcal{R}_A is not trivial (even for the case of low-order LTI systems), since this set can be non-convex, open and unbounded [39]. Therefore, an origin-centred closed \mathbb{R}^{n_x} -ball \mathcal{R}_E can be taken as an estimate for the actual region of attraction such that $\mathcal{R}_E \subseteq \mathcal{R}_A$.

A reasonable and direct manner that can be used to compute this estimated region is to use level sets [40] of a candidate Lyapunov map associated to the closed-loop expression of Eq. (17). We note that the Lyapunov map is used to describe the terminal stage cost in Theorem 3.1. Considering that there exists a generic parameter-dependent quadratic Lyapunov candidate function $M^{-1}(\rho)$, the associated level set is denoted:

$$\mathcal{L}_V(\mu) = \{x \in \mathbb{R}^{n_x} \mid (x^T M^{-1}(f_\rho(x))x) \leq \mu\}. \quad (23)$$

Finally, as discussed in [40], such level set $\mathcal{L}_V(\mu)$ can be used as a reasonable estimate for the region of attraction of the closed-loop system with a known nominal state-feedback gain $K \in \mathbb{R}^{n_u \times n_x}$, i.e. $\mathcal{R}_E \subset \mathcal{L}_V(\mu)$ for an adequate value of μ .

For $V(\cdot)$ to be indeed an admissible Lyapunov function for all $x(k) \in \mathcal{R}_E$ with state trajectories described by Eq. (17), Theorem 3.1 must be ensured taking $\mathbf{X}_f \subseteq \mathcal{R}_E$ while respecting the sector condition in Eq. (20). For such, we provide a catalogue of three LMI-solvable problems that can be verified in order to compute the terminal ingredients. These are the main contributions of this paper, since the offered terminal ingredients are able to ensure Poly-quadratic Stability of the closed-loop qLPV system regulated under MPC, despite the fault-induced input saturation.

Definition 3.7. Poly-quadratic Stability [41]

The qLPV system with explicit saturation constraints in Eq. (17) is subject to a nominal MPC control policy of state-feedback form $u(k) = K(\rho(k))x(k)$. The closed-loop dynamics are locally poly-quadratically stable if there exists some Lyapunov candidate function which is quadratic in the state $x(k) \in \mathcal{R}_E$, as given in Eq. (13).

3.4. Terminal Ingredients for Local Poly-quadratic Stability

As previously discussed, if the terminal stage cost $V(\cdot)$ is more than a candidate map, but rather an actual admissible Lyapunov function for all $x(k)$ contained inside the region of attraction estimate \mathcal{R}_E , then, it verifies stability and ensures recursive feasibility.

Accordingly, we provide LMI-solvable remedies for the computation of the terminal ingredients of the proposed fault tolerant MPC policy, ensuring poly-quadratic local stability and recursive feasibility. These terminal ingredients can be selected according to the amount of information available regarding the process, with regard to the three possible cases presented in the beginning of this Section. As of this, we provide three Corollaries which adapt Theorem 3.8 to the context of each case:

1. Corollary 3.9 yields parameter-independent Lyapunov maps, corresponding to the case of quadratic stabilisation and arbitrarily varying parameters (case 1);
2. Corollary 3.10 generates parameter-dependent Lyapunov maps, for the case of arbitrary varying scheduling parameters (case 2);
3. Corollary 3.11 results in parameter-dependent Lyapunov maps, for the case of scheduling parameters with bounded rates of variations (case 3).

Theorem 3.8. Terminal Ingredients

Consider the discrete-time polytopic qLPV system described by Eq. (9) being controlled under the fault tolerant MPC action nominally given by $u(k) = K(\rho)x(k)$. The complementary control input used to ensure the sector condition is nominally given by $v(k) = G(\rho)x(k)$. Assume that Axioms A1 and A2 hold. Then, conditions (C1)-(C5) from Theorem 3.1 are satisfied if there exist a positive definite matrix $H \in \mathbb{R}^{n_x \times n_x}$, a symmetric parameter-dependent positive definite matrix $Y(\rho) : \mathbb{R}^{n_p} \rightarrow \mathbb{R}^{n_x \times n_x}$, a parameter-dependent square matrix $X(\rho) : \mathbb{R}^{n_p} \rightarrow \mathbb{R}^{n_x \times n_x}$, four parameter-dependent rectangular matrices $L(\rho) : \mathbb{R}^{n_p} \rightarrow \mathbb{R}^{n_u \times n_x}$, $M(\rho) : \mathbb{R}^{n_p} \rightarrow \mathbb{R}^{n_u \times n_x}$, $Z(\rho) : \mathbb{R}^{n_p} \rightarrow \mathbb{R}^{n_u \times n_u}$ and $W(\rho) : \mathbb{R}^{n_p} \rightarrow \mathbb{R}^{n_u \times n_x}$, and a definite positive diagonal matrix $T \in \mathbb{R}^{n_u \times n_u}$ such that $Y(\rho) = P^{-1}(\rho) > 0$, $L(\rho) = K(\rho)X(\rho)$, $W(\rho) = G(\rho)X(\rho)$ and that the following LMI optimisation holds for all $\rho, \rho^+ \in \mathcal{P}$:

$$\begin{aligned} \min_{Y(\rho), L(\rho), W(\rho), T, H, X(\rho)} \quad & H \\ \text{subject to} \quad & \begin{bmatrix} Y_i & \mathbb{I} \\ \star & H \end{bmatrix} \geq 0, \forall i \in 0, \dots, n_p, \\ \text{and} \quad & \text{LMIs (25), (26), (27) and (28)}. \end{aligned} \quad (24)$$

$$\left[\begin{array}{cc|ccc} Y(\rho) & \star & \star & \star & \star \\ -W(\rho) & 2T & \star & \star & \star \\ \hline A(\rho)Y(\rho) + B(\rho)L(\rho) & 0 & Y(\rho^+) & \star & \star \\ Y(\rho) & 0 & 0 & Q^{-1} & \star \\ L(\rho) & 0 & 0 & 0 & R^{-1} \end{array} \right] > 0, \quad (25)$$

$$\left[\begin{array}{c|c} Y(\rho) & \star \\ \hline I_i L(\rho) & \bar{u}_j^2 \end{array} \right] > 0, \quad (26)$$

$$\left[\begin{array}{c|c} Y(\rho) & \star \\ \hline I_i (L(\rho) - W(\rho)) & (\bar{u}_j^+)^2 \end{array} \right] > 0, \quad (27)$$

$$\left[\begin{array}{c|c} \bar{x}_j^2 & \star \\ \hline I_j^T Y^{-1}(\rho) & Y(\rho) \end{array} \right] > 0, \quad (28)$$

where I_j denotes the j -th row of the identity matrix \mathbb{I} . In LMIs (26) and 27, it is given w.r.t. to an identity \mathbb{I}_{n_u} , i.e. these LMIs must hold for all $i \in \mathbb{Z}_{[1, n_u]}$. In LMI (28), it is given w.r.t. to an identity \mathbb{I}_{n_x} , i.e. this LMI must hold for all $j \in \mathbb{Z}_{[1, n_x]}$.

Then, for any admissible initial condition $x(0) = x_0 \in \mathcal{R}_A$, the proposed fault tolerant MPC method ensures that the controlled process will be locally and poly-quadratically stabilised at the aimed steady-state operation (x_r, u_r) . Moreover, recursive feasibility is verified for the admissible sequence of solutions U_k and that all closed-loop trajectories departing from inside \mathcal{R}_E will converge to its interior domain, taking $\mathcal{R}_E \subset \mathcal{L}_V(\cdot, \cdot)(\mu) = \{x_0 \in \mathbb{R}^{n_x} \mid (x_0^T P(f_\rho(x_0)) x_0) \leq \mu\}$.

We stress that the proposed MPC, which operates by the means of the optimisation in Eq. (10), generates closed-loop trajectories that remain within a (terminal) region of attraction \mathcal{R}_E . This region of attraction, in practice, is directly related to the solution of the LMIs in Theorem 3.8 and thus affected by the level of faults, as quantified by \underline{f} and \mathcal{U}^+ . For systems subject to harder faults (i.e. terms f_j closer to 0), the constraint input upper bounds u_j^+ also gets closed to 0, which imply in a smaller region mapped by $P(\rho)$. Notice that LMI (27) imposes an upper bound \bar{u}_j^+ upon the projection of each control input $u_j = I_j(K(\rho) - G(\rho))x$ along the ellipsoid $x^T P(\rho)x$, and thus shrinks the available input space with regard to this bound.

If a feasible solution to Theorem 3.8 exists, then proposed fault tolerant MPC is able to ensure that the closed-loop dynamics tolerate faults as strong as quantified by the worst-case lower bounds $f_j > 0$, i.e. any $f_j(k) \in [f_j, 1]$. In practice, the maximal level of faults tolerated by the proposed scheme (and thus the existence of feasible terminal ingredients) depends on the system model and constraints on states, inputs and scheduling variables.

We adapt Theorem 3.8 with respect to the three considered cases. As of this, the variables $Y(\cdot)$, $W(\cdot)$, and $L(\cdot)$ are treated in different manners (as parameter-dependent or independent functions). We stress that the same LMI problem formulation is kept for less conservative or more conservative solutions, which differ according to the level of knowledge of the designer regarding the process. The following Corollaries take into account each of the considered cases.

Corollary 3.9. *Constant Lyapunov Map*

Assume that the scheduling parameters vary arbitrarily along the discrete-time iterations, and that the system is to be quadratically stabilised. Therefore, the LMIs (25), (26), (27) and (28) in Theorem 3.8 are solved over the vertices of the polytope, as follows:

$$\left[\begin{array}{cc|ccc} Y & \star & \star & \star & \star \\ -W_i & 2T & \star & \star & \star \\ \hline A_j Y + B_j L_i & 0 & Y & \star & \star \\ Y & 0 & 0 & Q^{-1} & \star \\ L_i & 0 & 0 & 0 & R^{-1} \end{array} \right] > 0, \forall (i, j), \quad (29)$$

$$\left[\begin{array}{c|c} Y & \star \\ \hline I_i L_i & \bar{u}_j^2 \end{array} \right] > 0, \forall i, \quad (30)$$

$$\left[\begin{array}{c|c} Y & \star \\ \hline I_i (L_i - W_i) & (\bar{u}_j^+)^2 \end{array} \right] > 0, \forall i, \quad (31)$$

$$\left[\begin{array}{c|c} \bar{x}_j^2 & \star \\ \hline I_j^T Y^T & Y \end{array} \right] > 0, \quad (32)$$

which yields a single ellipsoid $\mathcal{E}(Y-1, 1)$ regarding the level set definition in Eq. (23).

Corollary 3.10. *Parameter-dependent Lyapunov Map*

Assume that the scheduling parameters vary arbitrarily along the discrete-time iterations, and that the system is to be poly-quadratically stabilised. Therefore, the LMIs (25), (26), (27) and (28) in Theorem 3.8 can be solved for all vertices of the polytope:

$$\left[\begin{array}{cc|ccc} Y_i & \star & \star & \star & \star \\ -W_i & 2T & \star & \star & \star \\ \hline A_j Y_i + B_j L_i & 0 & Y_j & \star & \star \\ Y_i & 0 & 0 & Q^{-1} & \star \\ L_i & 0 & 0 & 0 & R^{-1} \end{array} \right] > 0, \forall (i, j), \quad (33)$$

$$\left[\begin{array}{c|c} Y_i & \star \\ \hline I_i L_i & \bar{u}_j^2 \end{array} \right] > 0, \forall i, \quad (34)$$

$$\left[\begin{array}{c|c} Y_i & \star \\ \hline I_i (L_i - W_i) & (\bar{u}_j^+)^2 \end{array} \right] > 0, \forall i, \quad (35)$$

$$\left[\begin{array}{c|c} \bar{x}_j^2 & \star \\ \hline I_j^T Y_i^T & Y_i \end{array} \right] > 0, \forall i. \quad (36)$$

This procedure results in $2^{n_\rho} + 1$ ellipsoids $\mathcal{E}(Y_i^{-1}, 1)$.

Corollary 3.11. *Parameter-dependent Lyapunov Map, Bounded Rates of Parameter Variations*

Assume that the scheduling parameters have bounded rates of variations, i.e. $\partial \rho \in \partial \mathcal{P}$, and that the system is to be poly-quadratically stabilised. Consider that $\underline{\partial \rho} \leq \rho^+ - \rho \leq \bar{\partial \rho}$, i.e. $\underline{\partial \rho} + \rho \leq \rho^+ \leq \bar{\partial \rho} + \rho$. Thus, define the following polytope for (ρ, ρ^+) :

$$\begin{pmatrix} \rho \\ \rho^+ \end{pmatrix} = H_\theta \Theta, \quad (37)$$

where Θ is a vector given within an n_h simplex and H_θ depends on the bounds $\underline{\rho}$, $\bar{\rho}$ and $\underline{\partial\rho}$ and $\bar{\partial\rho}$.

Take $\tilde{Y}(\Theta)$ at the place of $Y(\rho)$ and $\hat{Y}(\Theta)$ at the places of $Y(\rho^+)$, as follows:

$$\begin{aligned} Y(\rho) &= \sum_{i=1}^{2^{n_\rho}} \gamma_i Y_i = \sum_{i=1}^{2^{n_\rho}} ([I_i \ 0_i] Y_i) H_\theta \Theta \\ &= \sum_{j=1}^{n_h} \underbrace{\left(\sum_{i=1}^{2^{n_\rho}} Y_i [I_i \ 0_i] \right)}_{\tilde{Y}_i(\Theta)} h_{\theta_j} \Theta_j, \end{aligned} \quad (38)$$

$$Y(\rho^+) = \sum_{j=1}^{n_h} \underbrace{\left(\sum_{i=1}^{2^{n_\rho}} Y_i [0_i \ I_i] \right)}_{\hat{Y}_i(\Theta)} h_{\theta_j} \Theta_j, \quad (39)$$

Therefore, the LMIs (25), (26), (27) and (28) in Theorem 3.8 can be solved over the vertices of the polytope defined in terms of $\Theta_i \in \Theta$, as follows:

$$\left[\begin{array}{cc|ccc} \tilde{Y}_i & \star & \star & \star & \star \\ -\tilde{W}_i & 2T & \star & \star & \star \\ A_j \tilde{Y}_i + B_j \tilde{L}_i & 0 & \tilde{Y}_i & \star & \star \\ \tilde{Y}_i & 0 & 0 & Q^{-1} & \star \\ L_i & 0 & 0 & 0 & R^{-1} \end{array} \right] > 0, \forall (i, j), \quad (40)$$

$$\left[\begin{array}{c|c} \tilde{Y}_i & \star \\ \hline I_i \tilde{L}_i & \bar{u}_j^2 \end{array} \right] > 0, \forall i, \quad (41)$$

$$\left[\begin{array}{c|c} \tilde{Y}_i & \star \\ \hline I_i (\tilde{L}_i - \tilde{W}_i) & (\bar{u}_j^+)^2 \end{array} \right] > 0, \forall i, \quad (42)$$

$$\left[\begin{array}{c|c} \bar{x}_j^2 & \star \\ \hline I_j^T \tilde{Y}_i^T & \tilde{Y}_i \end{array} \right] > 0, \forall i. \quad (43)$$

This procedure results in $2^{n_\rho} + 1$ ellipsoids $\mathcal{E}(Y_i^{-1}, 1)$.

The proof of Theorem 3.8 (and Corollaries 3.9, 3.10, and 3.11) is provided in the Appendixes. We note that the solution of the LMI problem presented in Theorem 3.8 ensures a positive definite parameter dependent matrix $P(\rho)$, which can be used to compute the MPC terminal ingredients $V(\cdot)$ and \mathbf{X}_f such that input-to-state stability of the closed-loop is guaranteed, verifying the conditions of Theorem 3.1. Furthermore, when the MPC algorithm uses these terminal ingredients, for whichever initial condition $x(0) \in \mathbf{X}_f$ it starts with, it remains recursively feasible for all consecutive discrete time instants $k > 0$.

Remark 8. For the qLPV systems as in Eq. (1), for which the scheduling parameters ρ are not defined within a simplex, the application of Theorem 3.8 (and the subsequent Corollaries) depends on an adaptation of these parameters. The solution holds through by replacing the parameter dependency on ρ by the convex weighted sum of $\gamma_i(\rho)$, with $\sum_i \gamma_i(\rho) = 1$. Instead of using a physically-related scheduling parameter, the LMIs are solved over a parameter that describes an adapted polytope. Note that this change of variables also implies in a new polytope w.r.t. the scheduling parameter derivatives, since these variables are affine on ρ and allow time-domain derivation. As an example, a qLPV system with one physically-related scheduling parameter ρ , which yields $\gamma_1 = \frac{\bar{\rho} - \rho}{\bar{\rho} - \underline{\rho}}$ and $\gamma_2 = \frac{\rho - \underline{\rho}}{\bar{\rho} - \underline{\rho}}$.

Remark 9. As a final comment, we stress that the terminal ingredients generated by any of the proposed solutions (Theorem 3.8 or its variations under Corollaries 3.9, 3.10, and 3.11) are explicitly affected by

the scheduling parameters ρ , while implicitly influenced by the maximal fault level \underline{f} . With regard to the scheduling variables, the terminal ingredients are nominally ρ -dependent, and synthesised with regard the available information regarding the behaviour of these variables (as gives each Corollary), accordingly implying in a more or less conservative control scheme. In terms of the fault information $f(k)$, we note that the lower bound \underline{f} is used to define the maximal saturated control entries, as illustrated by Fig. 1 and given in Eq. (7). These control signal limits, e.g. $u_j^+, \forall j \in \mathbb{Z}_{[1, n_u]}$, are used to construct the sector condition from Ineq. (21), which is embedded to the proposed solution (specifically, the sector conditions appears in LMIs (27), (31), (35) and (42)). Note that the sector condition acts by shrinking the available input space with regard to the level of faults. In practice, a system subject to higher degree of faults will have a corresponding stricter terminal region related to the control input.

3.5. Overall FTC Process

Before presenting application results of the proposed passive FTC method, we briefly recap the main elements and present an algorithm, for implementation ease. Overall, the following aspects are of importance:

- We consider the class of discrete-qLPV processes subject to fault-induced input saturation, as given by Eq. (1). This system must be controlled in such way that the outputs y meet certain performance criteria, while the process constraints ($u \in \mathcal{U}$, $x \in \mathcal{X}$, and $y \in \mathcal{Y}$) are satisfied, despite the presence of faults f .
- For such, a parameter-dependent state-feedback MPC law is formulated $u(k) = K(\rho(k))$, enable by the solution of an online min-max optimisation.
- The proposed MPC works as a robust passive FTC by considering the uncertainties implied in the dynamics by the future behaviour of scheduling parameters, and by handling the fault-induced saturation by stabilising terminal ingredients. Furthermore, the MPC operates with an heuristically-chosen (small) prediction horizon N_p .
- Specifically, the MPC optimisation is based on a fault-free prediction model of the system, as gives Eq. (8), while requiring the terminal ingredients $V(\cdot)$ and \mathbf{X}_f , which dependent on the level of faults.
- These terminal ingredients can be generated according to the information available regarding the process, as gave Theorem 3.8, Corollaries 3.9-3.11.
- Algorithm 1 summarises the offline preparation and the online implementation of the proposed passive FTC method.

4. Results: Application to Vehicle Dynamics

In this Section, we demonstrate how the proposed fault accommodation technique can be applied in practice. Accordingly, we compare the proposed robust MPC method with other MPC algorithms from the literature, which are not designed to take the issue of faults into account. The difference between our method and the others is that the novel terminal ingredients provided in Section 3 enable recursive feasibility and stabilisation even when the process is subject to the fault-induced time-varying saturation along the input channels.

We proceed by evaluating the obtained performances using standard indexes, such a root-mean-square (RMS) metric and the average computational time required to solve the optimisation, during implementation (denoted t_c). Through the sequel, we refer to “tracking performances/results” as the regulation of the error trajectories of given variables to the origin. We use an output reference model, which is constant and known.

Specifically, we compare the proposed technique with a “full-blown” nonlinear MPC (denoted “NMPC”), which solves a nonlinear optimisation problem, using $x(k+j+1) = f_x(x, u, k+j)$ and $y(k+j) =$

Algorithm 1 Proposed Passive FTC**Input:** Fault-less model $(A(\cdot), B(\cdot), C(\cdot), D(\cdot), f_\rho(\cdot))$ **Input:** Process constraints $(\mathcal{X}, \mathcal{Y}, \mathcal{U}, \mathcal{P}, \delta\mathcal{X})$ **Input:** Fault level (f_1, \dots, f_{n_u}) **Input:** Output steady-state target (y_r) **Offline Preparations:**

- 1: (1) Generate the state and input targets (x_r, u_r) , via Eq. (11);
- 2: (2) Generate the terminal ingredients through Theorem 3.8;
- 3: (3) Compute the MPC horizon, solving Eq. (14).

Online Implementation:

- 4: **for** $k = 0$ to end of implementation **do**
- 5: Measure $x(k)$;
- 6: Compute $\rho(k) = f_\rho(x(k))$;
- 7: Solve the MPC optimisation from Eq. (10);
- 8: From the solution U_k^* , apply the first entry $u^*(k|k) = K(\rho(k))x(k)$.
- 9: **end for**

$f_y(x, u, k + j)$ to predict the future behaviour of the states and outputs, as in [42, Chapter 7]. This is the optimal theoretical solution in a nominal case, without any fault-induced saturation (i.e. Eq. (8)). We use this technique as a comparison benchmark because it allows us to evaluate the enhancements provided with our method with respect to: (1) fault accommodation, since the NMPC does not analytically consider the effects of faults into account; and (2) computational load, since the nonlinear program is possibly impractical from an implementation perspective.

4.1. Vehicle Dynamics Control

Automotive/vehicular systems include complex dynamics of motion. One can enhance the vertical, roll and pitch dynamics of a car, which are related to comfort performances of the passengers, see [43]. Moreover, safety characteristics of a vehicle can be enhanced when longitudinal, lateral and yaw movements are controlled. **What is done in (industrial and academic) practice is that two sets of control systems are synthesised: one concerning the suspension system of the car [44], which embeds the first dynamics, and another one regarding the steering and braking systems, which relates to the safety cautions [28].**

In this Section, the simulation results of the proposed MPC algorithms will be presented focusing on the safety problem of an automotive system, concerning, more specifically, lateral and yaw dynamics. In the recent literature, it has been shown that the joint use of braking and steering systems can highly enhance these lateral safety performances. In this perspective, interesting recent developments have been presented, mostly involving MPC with linearised models, as in [45, 46, 47].

Anyhow, for realistic implementation, full nonlinear models should be considered in order to represent the complete dynamics, as done in [48]. Nevertheless, these models can imply in control laws that are excessively demanding from a computational perspective. The main interest of using a full nonlinear model is that it embeds the nonlinear load transfer between the vehicle corners and the fast dynamics present in the tire force behaviour. These characteristics are especially interesting in dangerous driving situations, for which a linearised model may neglect behaviours and thus lead to possible accidents.

In terms of the lateral behaviour, the main nonlinear dynamics under interest are the sideslip of the car $\beta(t)$ and the yaw $\psi(t)$ dynamics. These dynamics are included in what is usually referred to as a “bicycle model” [49, 50], which models the vehicle from an horizontal perspective, as illustrated in Figure 3.

The dynamic bicycle model is governed by the following nonlinear equations:

$$mv\dot{\beta}(t) = F_{ty_f}(t) + F_{ty_r}(t) + mv\dot{\psi}(t), \quad (44)$$

$$Z_z\ddot{\psi}(t) = l_f(-F_{tx_f}(t)\sin(\delta_f(t)) + F_{ty_f}(t)\cos(\delta_f(t))) - l_r F_{ty_r}(t) - \Delta F_{tx_r}(t)t_r + M_{dz}(t), \quad (45)$$

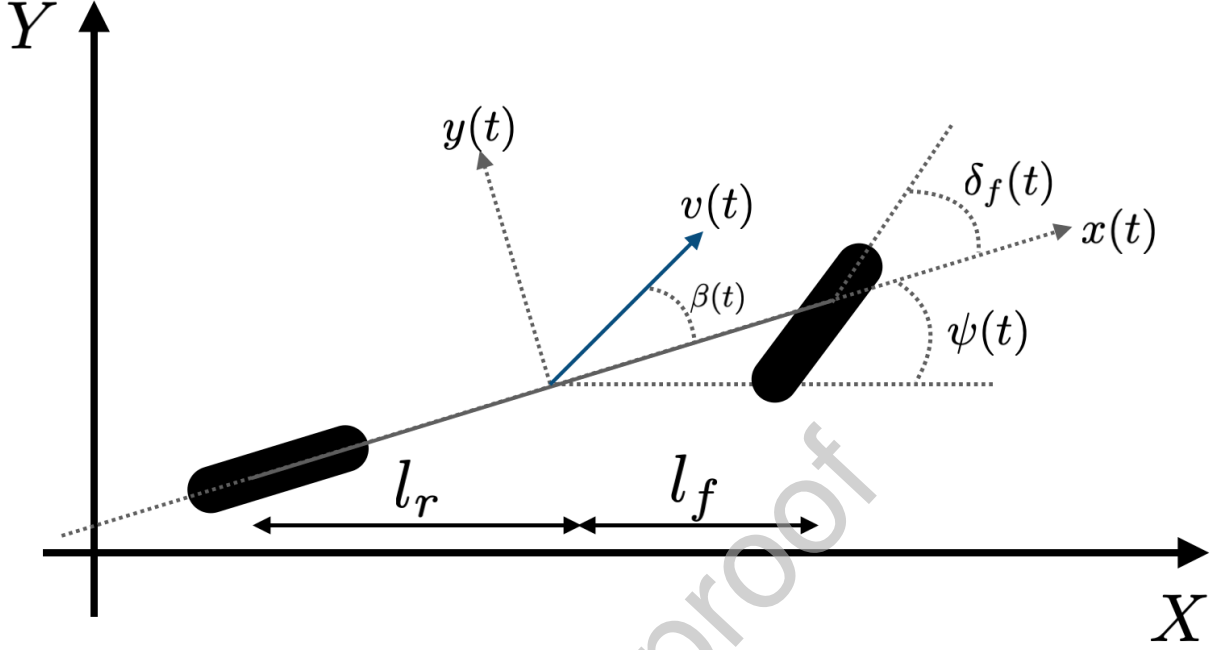


Figure 3: Bicycle model of a car.

where $F_{ty_f}(t)$, $F_{ty_r}(t)$ and $\Delta F_{tx_r}(t)$ are the front, rear and tire lateral forces and the longitudinal rear differential forces, respectively; these forces depend on the sideslip angle $\beta(t)$ and on the slip ratio $\lambda(t)$. The car's longitudinal speed is given by $v(t) = \sqrt{(\dot{X}(t))^2 + (\dot{Y}(t))^2}$, its total mass is denoted m and its inertia along the vertical z axis is denoted Z_z . Finally, $\delta_f(t)$ denotes the steering angle and $M_{dz}(t)$ the yaw moment disturbance.

For low slip values (which is regular for controlled situations), i.e. $|\lambda(t)| < 0.1$, the longitudinal rear differential forces can be expressed as:

$$\Delta F_{tx_r}(t) = \frac{\mu_c R m_r g}{2} (T_{b_{rl}}(t) - T_{b_{rr}}(t)), \quad (46)$$

being m_r the mass of the rear part of the car, g the gravitational constant, $T_{b_{rj}}(t)$ the rear braking torque at the left ($j = l$) and right ($j = r$) sides.

Moreover, under these same conditions, the front and rear forces along the y -axis are given by:

$$F_{ty_f}(t) = -\mu_c C_f \beta(t) - \frac{\mu_c C_f l_f}{v} \dot{\psi}(t) + C_f \sin(\delta_f(t)), \quad (47)$$

$$F_{ty_r}(t) = -\mu_c C_r \beta(t) + \frac{\mu_c C_f l_r}{v} \dot{\psi}(t). \quad (48)$$

Note that for low steering angles, $F_{tx_f}(t)$ renders very little effects upon $\dot{\psi}(t)$ and, therefore, it will be neglected for simplification purposes.

The first actuator for this system is the braking (EMB) system, which provides a continuously variable braking torque, as follows:

$$\dot{T}_{b_{rl}}(t) = 2\pi\bar{\omega} (u_1(t) - T_{b_{rl}}(t)), \quad (49)$$

$$\dot{T}_{b_{rl}}(t) = 2\pi\bar{\omega} (u_2(t) - T_{b_{rl}}(t)), \quad (50)$$

where $\bar{\omega} = 10\text{Hz}$ is the actuator cut-off frequency and $u_1(t)$ and $u_2(t)$ are the control inputs, which are constrained to:

$$u_1(t) \in \mathcal{U}_1 := \{u_1(t) \in \mathbb{R} \mid 0 \leq u_1(t) \leq u_1^{max}\}, \quad (51)$$

$$u_2(t) \in \mathcal{U}_2 := \{u_2(t) \in \mathbb{R} \mid 0 \leq u_2(t) \leq u_2^{max}\}. \quad (52)$$

Remark 10. In the majority of the aforementioned papers, a second actuator is considered for active steering conditions. In this case, the front steering angle $\delta_f(t)$ is given as the sum of the angle provided by the driver and an additional steering angle provided by a vehicle stability controller, see [48]. In this paper, we consider an hierarchical control scheme, meaning that a vehicle stability controller is responsible to generate $\delta_f(t) = f_\delta(x(t))$ in order to maintain stability, while a braking controller is concerned with sideslip and yaw rate tracking. We design herein braking control system. Assuming that $\delta_f(t) = f_\delta(x(t))$ complicates the control problem, since a full nonlinear prediction of the states is computationally complex.

In terms of possible faults that may occur in this system, we note that the braking torque upper bounds (u_1^{max} and u_2^{max} in Eqs. (51)-(51), respectively) vary over time. The braking system naturally deteriorates over its lifespan, which reduces the available braking torque. Other malfunctions in components of the braking system may also degrade its effectiveness, thus also decreasing these upper bounds. These fault-related phenomena are taken into account using the fault-induced saturation model, as given in Eq. (1). These details are explained in the sequel.

The proposed fault tolerant control method can be used either as an Advanced Driver-Assistance System (ADAS) or even as a driving control for autonomous vehicles. In these two different contexts, the bounds over ρ and $\delta\rho$ differ. Table 1 gathers all the symbology and parameter values for the dynamic bicycle model, considering a *Renault Mégane Coupé* vehicle. Model parameters have been obtained from [48], which exhibits an ADAS application.

Table 1: Bicycle Model Parameters.

Symbol	Value	Physical Meaning
m	1535 kg	Vehicle mass
m_r	648.3 kg	Vehicle rear mass
Z_z	2149 kgm ²	Vehicle yaw inertia
C_f	42042 N/rad	Lateral tire front cornering stiffness
C_r	43671 N/rad	Lateral tire rear cornering stiffness
R	0.3 m	Tire radius
l_f	1.4 m	Distance front axle - COG
l_r	1 m	Distance rear axle - COG
t_r	1.4 m	Rear axle length
u_1^{max}	120 Nm	Maximal front braking torque
u_2^{max}	120 Nm	Maximal rear braking torque
μ_c	[0.4, 1]	Tire/road contact friction interval
v	[50, 120] km/h	Vehicle velocity interval
g	9.81 m/s ²	Gravitational constant

4.2. qLPV Model

In this work, it is assumed that the sideslip angle $\beta(t)$, the total front steering angle $\delta_f(t)$ and the yaw variation $\dot{\psi}(t)$ are known and measurable. Moreover, one considers only constant velocity scenarios for $v = 30\text{m/s}$ and a tire/road contact friction of $\mu_c = 1$. Therefore, the following qLPV model is obtained.

For synthesis and control design, it is discretised with a T_s sampling period. This model is obtained by plugging Eqs. (46)-(50) into Eqs. (44)-(45) and considering an Euler discretisation method.

$$\dot{x}(t) = A(\rho(t))x(t) + B_1 \text{sat}_{f(t)}\{u(t)\} + B_2(\rho(t))w(t), \quad (53)$$

$$y(t) = Cx(t), \quad (54)$$

where $x(t) = [\beta(t) \ \dot{\psi}(t) \ T_{b_{rl}}(t) \ T_{b_{rr}}(t)]'$ is the system state vector, $y(t) = [\beta(t) \ \dot{\psi}(t)]'$ is the output vector, $u(t) = [u_1(t) \ u_2(t)]'$ is the control input vector, and $w(t) = [\delta_f(t) \ M_{dz}(t)]'$ is the disturbance vector.

The saturation term $\text{sat}_{f(t)}\{u(t)\}$ represents the fault-induced loss of effectiveness that occurs on each braking system (rear-left and rear-right wheels). Accordingly, in the following scenarios, we use $\underline{f}_j = 0.6$ as the lower bound for the faults on both actuators. This means that the fault-induced saturation $\text{sat}_{f(t)}\{u(t)\}$ implies in a maximal decrease of 40% on the effectiveness of each braking torque.

The scheduling parameters $\rho(t) = [\rho_1(t) \ \rho_2(t)]^T$, chosen according to [48], are given by:

$$\rho_1(t) = \cos(\delta_f(t)) \in [0.5, 1], \quad (55)$$

$$\rho_2(t) = \frac{\sin(\delta_f(t))}{\delta_f(t)} \in [0.8, 1], \quad (56)$$

being $\delta_f(t) \in [-60, 60]^\circ$.

For the control paradigms developed in this paper, bounds are needed on the variation of the scheduling parameters. These bounds are:

$$\partial\rho_1(t) = -\sin(\delta_f(t))\dot{\delta}_f(t) \in [-0.86, 0.86]\overline{\dot{\delta}_f}, \quad (57)$$

$$\partial\rho_2(t) = \dot{\delta}_f(t) \left(\frac{\cos(\delta_f(t))\delta_f(t) - \sin(\delta_f(t))}{(\delta_f(t))^2} \right) \in [0.2, 1.31]\overline{\dot{\delta}_f}, \quad (58)$$

where $\overline{\dot{\delta}_f} = \|\dot{\delta}_f\|_\infty$ is the upper bound on the variation of the frontal steering angle.

Finally, the matrices in the qLPV model in Eq. (53) are affine on the scheduling parameter $\rho(t)$, as follows:

$$A(\rho(t)) = \begin{bmatrix} -\left(\frac{\mu_c}{mv}(C_f + C_r)\right) & \left(1 + \frac{\mu_c}{mv^2}(l_r C_r - l_f C_f)\right) & 0 & 0 \\ \left(\frac{\mu_c}{Z_z}(l_r C_r - l_f C_f \rho_1(t))\right) & -\left(\frac{\mu_c}{Z_z v}(l_r^2 C_r + l_f^2 C_f \rho_1(t))\right) & -\left(\frac{\mu_c R m_r g t_r}{2Z_z}\right) & \left(\frac{\mu_c R m_r g t_r}{2Z_z}\right) \\ 0 & 0 & -(2\pi\bar{\omega}) & 0 \\ 0 & 0 & 0 & -(2\pi\bar{\omega}) \end{bmatrix},$$

$$B_1 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 2\pi\bar{\omega} & 0 \\ 0 & 2\pi\bar{\omega} \end{bmatrix}, \quad B_2(\rho(t)) = \begin{bmatrix} \frac{C_f \rho_2(t)}{Z_z} & 0 \\ \frac{l_f C_f \rho_2(t)}{Z_z} & \frac{1}{Z_z} \\ 0 & 0 \\ 0 & 0 \end{bmatrix},$$

$$C = [I_2 \ 0_2], \quad D = 0_{2 \times 2}.$$

4.3. Control Goals

Considering that $\delta_f(t)$ is determined by some other active steering control system, Eq. (53) becomes a qLPV representation of a vehicle braking system. The considered performance objective are the following, despite faults:

1. To sufficiently reduce the yaw rate error with respect to a reference model, for comfort or sport driving performances;

2. To track smooth sideslip performance signals, also given by some reference model;
3. To ensure that the braking control signal can be achieved by the considered Electro-Mechanical Brakes (EMB actuators), i.e., respecting the control constraints given in Eqs. (51)-(52).

4.4. Problem Solution and Simulation Scenario

In the sequel, we use a nonlinear model validated on a real *Renault Megané Coupé* car (refer to [48]). The qLPV model in Eq. (53) is Euler-discretised with a sampling period of $T_s = 5$ ms, which is the usual operational sampling of embedded automotive control systems [51].

The results presented in the sequel were achieved using Matlab software, Yalmip toolbox and *fmincon* and *SDPT3* solvers. All numerical simulations in this paper were performed on the same 2.4 GHz, 8 GB RAM Macintosh computer.

The weights used for the synthesis of all MPC controllers are $Q = \text{diag}\{ 10 \ 100 \ 0.1 \ 0.1 \}$ and $R = \text{diag}\{ 0.5 \ 0.5 \}$, since the main control priority/goal is the yaw rate performances.

For simplicity of illustration, we show only the results achieved with Corollary 3.11, which is the “complete” case of Lyapunov parameter-dependency and bounded rates of parameter variation (more information is available regarding the process). More conservative results are obtained with the other Corollaries⁵.

We provide full nonlinear simulation results in the sequel. The following results correspond to 5 s simulation under an avoidance manoeuvre.

Figure 4 presents the disturbances applied to the system (the front steering angle δ_f and the yaw moment disturbance M_{dz}) together with the scheduling parameters (and their variation rates). Recall that the front steering angle is also used to compute the references to β and ψ , using an adequate performance model. We use a step-like signal $f(t)$ in order to represent loss of effectiveness faults of 20% ($f_1 = f_2 = 0.8$) and 40% ($f_1 = f_2 = 0.6$) on both braking systems. These faults may happen due to malfunctions which restraint the amount of braking effort available.

4.5. Tuning of the Prediction Horizon

First, we tune the size of the prediction horizon w.r.t. to the heuristic criterion given in Eq. (14). The chosen horizon given in number of discrete-time samples is the one that maximises performances, while abiding to the sampling period threshold of 5 ms.

With regard to this matter, Table 2 shows the different N_p -dependent computational stress index values of the min-max procedure, being $t_c = t_c^{\text{Min}} + t_c^{\text{Max}}$, i.e. the first term is respective to the min. QP while the later is respective to the max. NP. This table also shows the RMS indexes for the variables of interest, $(\beta - \beta_r)$ and $(\dot{\psi} - \dot{\psi}_r)$.

We note that smaller indexes indicate better performances, although for these to be attained, more computational time is required to implement the min-max strategy. This is a Pareto optimality issue [33], since better performances are provided with longer horizons, whereas the optimisation becomes more demanding as they increase. Complementary Figure 5 illustrates two Pareto planes, showing how the NRMS index of the variables of interest w.r.t. to different horizon sizes and the yielded computational stress t_c .

Therefore, w.r.t. the previous discussions, we proceed with the application of a horizon of $N_p = 6$ steps for this vehicle dynamics problem, which enables its real-time operation under the threshold of $T_s = 5$ ms.

4.6. Performance Evaluation

Now, we show the regulation results with the proposed method against those obtained with a nonlinear MPC tool. The full-blown NMPC is synthesised with a larger control horizons of $N_p = 25$, as suggests [51]. Furthermore, SSMPC denotes the proposed FTC method with $N_p = 6$ (we use “SS” as an abbreviation for

⁵This is quite reasonable, since cases 1 and 2 allow less knowledge regarding the controlled process. The first case (Corollary 3.9 implies in quadratic stabilisation, which is the most conservative of the three. The second case (Corollary 3.10) implies in poly-quadratic stabilisation using parameter-dependent terminal ingredients, but bounded rates of the variations of the scheduling variables between samples are not considered.

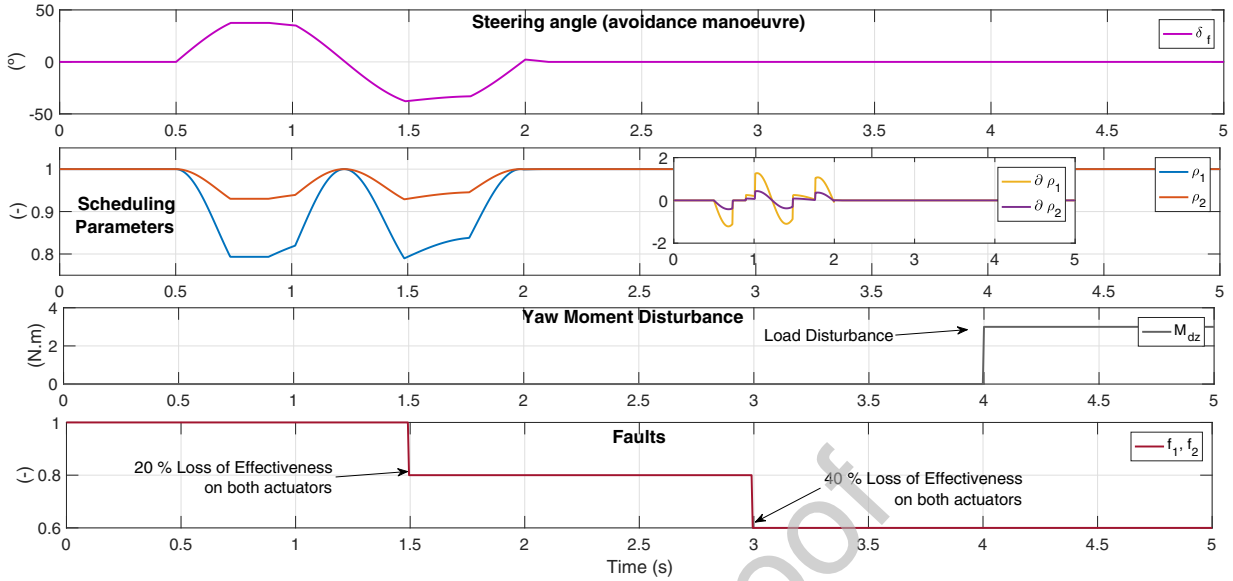


Figure 4: Faults, Disturbances and Scheduling Parameters.

Table 2: Prediction Horizon Tuning.

N_p	$\text{RMS}\{\beta - \beta_r\}$	$\text{RMS}\{\psi - \psi_r\}$	$t_c^{\text{Min.}}$	$t_c^{\text{Max.}}$	t_c
2	0.4834	0.3176	0.24 ms	3.40 ms	3.64 ms
5	0.4721	0.1268	0.29 ms	3.45 ms	3.75 ms
6	0.4718	0.1256	0.31 ms	4.12 ms	4.43 ms
10	0.4646	0.1244	0.42 ms	10.42 ms	10.85 ms
20	0.4624	0.1206	0.52 ms	19.09 ms	19.61 ms
25	0.4623	0.1206	0.68 ms	27.83 ms	28.51 ms

short-sighted, due to the small horizon size). In terms of the considered fault events, the braking torques are limited by 20% from $t = 1.5\text{s}$ onwards (i.e. $\bar{u}_j^+ = 96\text{N.m}$) and by 40% from $t = 3\text{s}$ onwards (i.e. $\bar{u}_j^+ = 72\text{N.m}$). We recall that the FTC is synthesised to tolerate a fault degradation of up to 40%, since the LMIs are solved for $f_j = 0.6$.

First off, we compare the proposed algorithm to the NMPC in terms of the average computational stress required to generate the control law. With respect to this matter, Table 3 summarises the average time required by the algorithms to solve the resulting optimisation programs. As we can see, the “full-blown” NMPC would not be able to run in a real-time, embedded in an on-board vehicle micro-controller, since its average computational effort is much longer than the sampling period of 5 ms. Nonetheless, the proposed method enables real-time operation due to the heuristic criterion used to choose N_p , which is a key feature and major advantage for time-critical systems.

With respect to the aforementioned control objectives, Figure 6 depicts the achieved results with these two controllers in terms of angle and yaw rate reference tracking⁶. The curves labeled “No Control” are those obtained in an open-loop scenario, with only active steering and no coordinated braking (i.e. taking

⁶A reference model x_r is used and the corresponding tracking error trajectories $x - x_r$ are steered by the control to the origin.

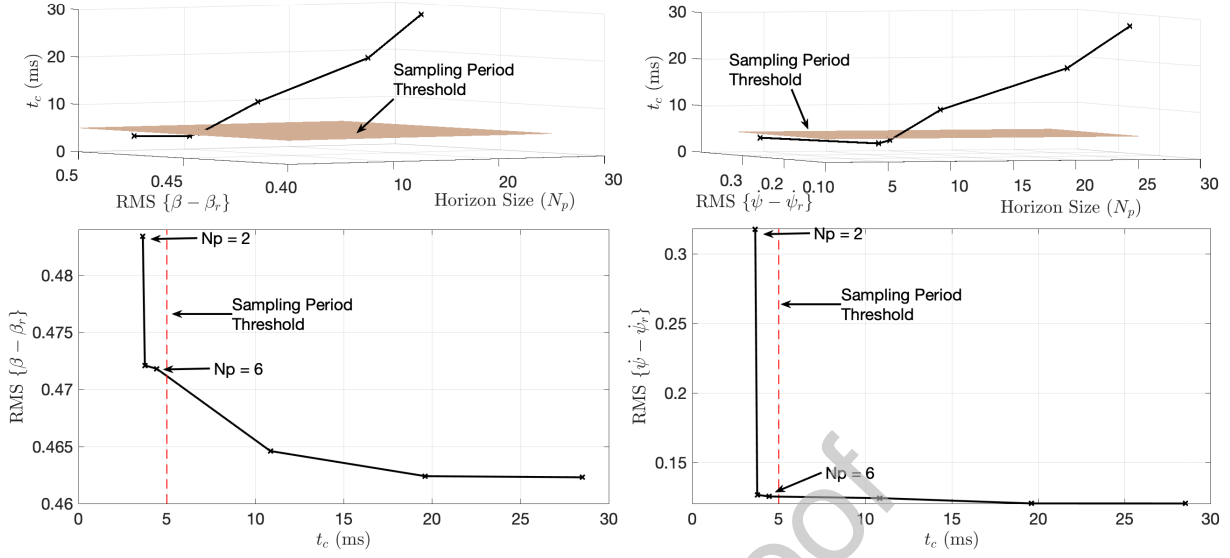
Figure 5: Pareto Planes $(\beta - \beta_r) \times t_c$ and $(\dot{\psi} - \dot{\psi}_r) \times t_c$.

Table 3: Average Computational Stress of the MPCs.

Method	Average Computational Stress
NMPC	60.08 ms
SSMPC (FTC)	4.53 ms

$u = 0$).

In terms of sideslip angle β tracking (which is the second minimisation priority in the MPC cost function, according to the chosen value for matrix Q), we can argue that the both control strategies present similar results. Numerically, we assess effectiveness of the controllers using an RMS index of the error trajectories, which are presented in Table 4. The proposed method slightly outperforms the NMPC, but the effects of fault on the sideslip angle are minor. We indicate the RMS index with regard to the fault-less conditions (before the fault-induced saturation occurs) and those subject to faults.

In terms of the main control priority, which is yaw rate $\dot{\psi}$ tracking, we can notice the advantages of the proposed method much clearer. The effect of the fault-induced saturation becomes more visible in this output, as marked (and zoomed) in Fig. 6, which shows the advantages of the fault accommodation ensured by the robustness provided with the terminal ingredients. The effectiveness of the algorithms are once again measured in terms of an RMS index, given in Table 4, with regard to faulty and fault-less situations. When the fault-induced saturation is enacted, we are able ensure 28% better performances, which is quite significant.

Finally, Fig. 7 shows the clipped control actions derived with the controllers. The presence of fault-induced saturation is emulated from $t = 1.5$ s onwards. With the proposed terminal ingredients, the FTC method is able to smoothly accommodate faults and thus maintains stability and recursive feasibility properties.

We stress that the abrupt peaks in the control actions, derived with the FTC strategy, occur due to the maximisation part of the robust MPC formulation, which computes harsher variations of the future scheduling parameters at these moments. These peaks, nevertheless, do not compromise the achieved results.

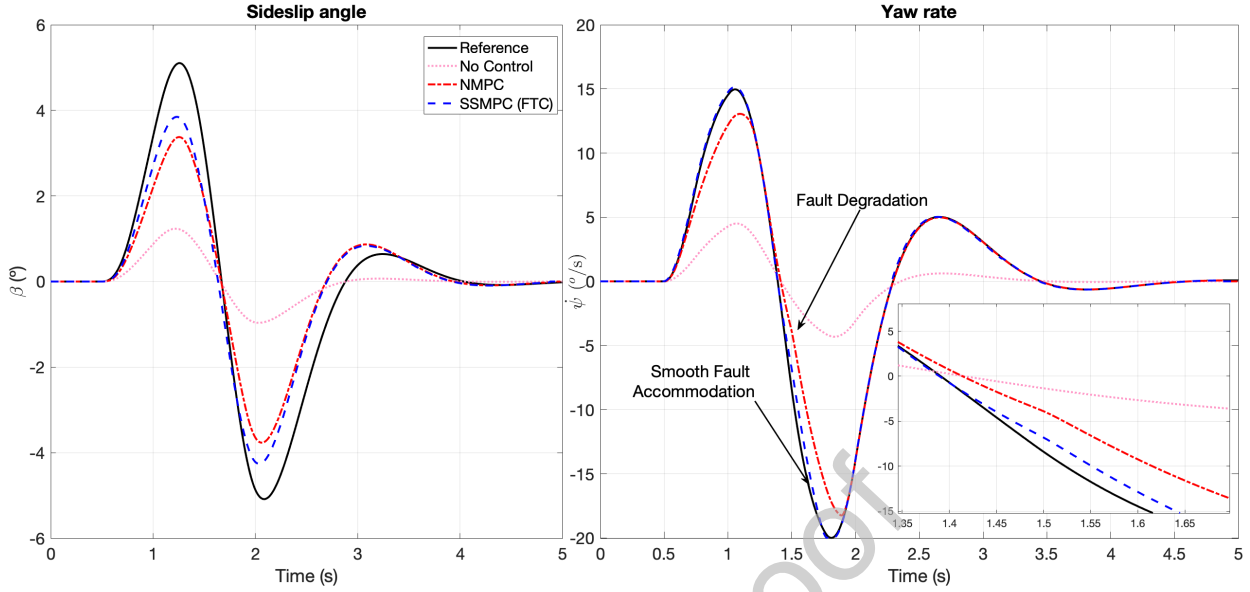
Figure 6: Results: Sideslip Angle β Tracking (left) and Yaw rate $\dot{\psi}$ Tracking (right).

Table 4: RMS of the Error Trajectories

Sideslip angle tracking	NMPC	SSMPC (FTC)	Enhancement
Fault-less conditions: $t \in [0, 1.5]$ s	0.7331	0.7299	0.4 %
Faulty conditions: $t \in [1.5, 3]$ s	0.5923	0.5829	1.58 %
Yaw rate tracking	NMPC	SSMPC (FTC)	Enhancement
Fault-less conditions: $t \in [0, 1.5]$ s	0.2966	0.2746	7.41 %
Faulty conditions: $t \in [1.5, 3]$ s	0.2865	0.2051	28.41 %

One could consider a slew rate constraint to further smooth this control signal. We note that the state constraints are respected by all algorithms. Furthermore, these variations on the control do not affect the stability of the vehicle, nor maneuvering concerns, as discussed next.

In terms of the stability of the vehicle itself, we show in Fig. 8 the obtained trajectories for β and $\dot{\psi}$ with the FTC method with regard to the driving stability regions presented in [47] and the steering manoeuvre stability region given in [28]. As it can be seen, the obtained trajectories (dashed blue lines) are stable in a driving sense.

4.6.1. Poly-Quadratic Stability and Region of Attraction

In order to conclude, we present the estimates for the region of attraction of the proposed method. Accordingly, we present each estimated domain \mathcal{R}_E , as defined in Eq. (23), considering parameter-dependent and independent-cases, and bounded rates of parameter variations (cases 1, 2, and 3, which depend on the amount of information regarding the controlled process).

We note that, for all cases, the Lyapunov cases are defined through Theorem 3.8, being symmetric and holding a Lyapunov condition with sector constraints for the whole scheduling polytope Ω . Theorem 3.8, Corollary 3.9 yields a quadratic P (case 1), Theorem 3.8, Corollary 3.10 yields a parameter-dependent $P(\rho)$ (case 2) and Theorem 3.8, Corollary 3.11 yields a parameter-dependent $P(\rho)$ for bounded rates of parameter variation $\partial\rho$ (case 3). The previous results were shown for the broader case (case 3).

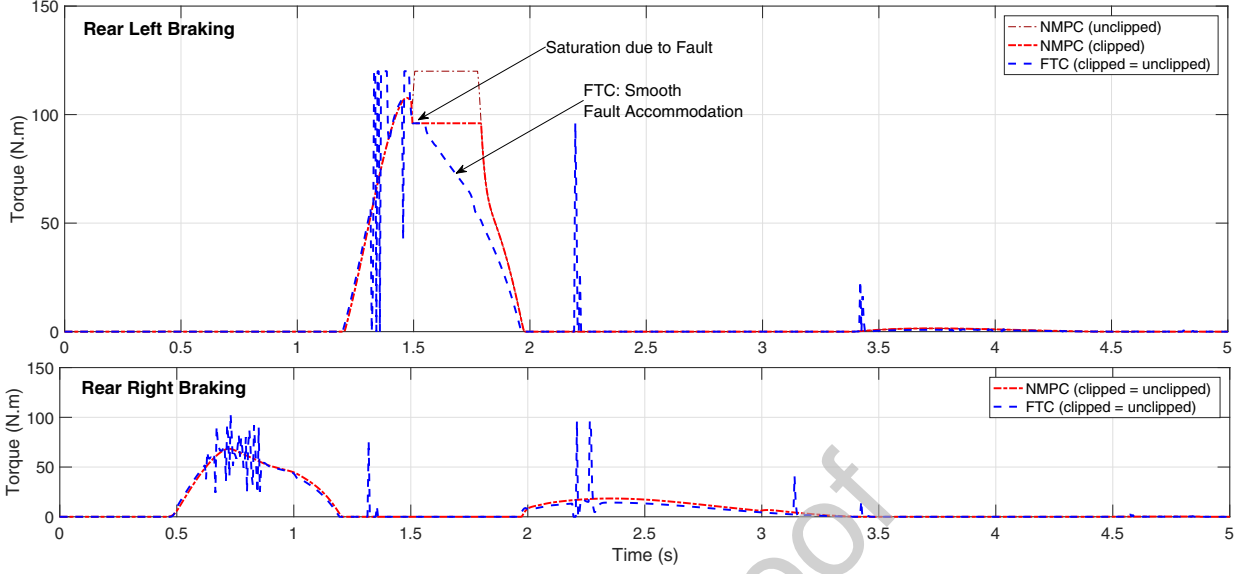


Figure 7: Control Actions.

Below, Figure 9 presents a phase diagram of tracking variables w.r.t. the first and second states (variables of interest $\beta - \beta_r$ and $\dot{\psi} - \dot{\psi}_r$) of the controlled system and the respective Lyapunov matrices⁷ P . The ellipsoids encompass the three cases provided by the catalogue of terminal ingredients. We note that the estimated regions of attraction through each Corollary are presented as the intersections of each ellipsoid pair. As it can be clearly seen, the evolution of the trajectories of these states always lies within the ellipsoids, for initial conditions that also departed from their interiors. This means that, indeed, local poly-quadratic asymptotic stability is verified for the proposed method, together with recursive feasibility. We must stress that the same conjecture remains true for the other states of the system.

Complementary, this figure also shows the sector condition from Eq. (20), where the ellipsoid is computed w.r.t. S in Theorem 3.8. We note that the fault-related saturation only occurs w.r.t. the second control signal, since the MPC always computes some $u_1 \leq u_1^+$ in the considered simulations.

Below, we present the numerical values obtained for the Lyapunov matrices $P(\cdot) = Y^{-1}(\cdot)$ and the sector ellipsoids S , with each Corollary. We note that only a pair of Lyapunov matrices are made necessary since, in the considered model of Eq. (53), $A(\rho)$ is only dependent on ρ_1 and $B(\cdot)$ is, in fact, parameter-independent. The parameter-dependency on ρ_2 only appears w.r.t. $B_2(\rho)$, which does not appear in the proposed LMIs.

$$Y_{\text{Corollary 3.9}} = \left(\begin{bmatrix} 6.85 & -3.17 & 0.03 & -0.02 \\ \star & 1.67 & -0.01 & 6 \cdot 10^{-3} \\ \star & \star & 3 \cdot 10^{-3} & -2 \cdot 10^{-3} \\ \star & \star & \star & 4 \cdot 10^{-5} \end{bmatrix} \cdot 10^{-3} \right)^{-1}, \quad (59)$$

⁷Indeed, P is of 4×4 ; what is shown, in fact, is $T_x^T P T_x$, where T_x is a transformation matrix such that $(\beta(k) \quad \dot{\psi}(k)) = T_x x(k)$.

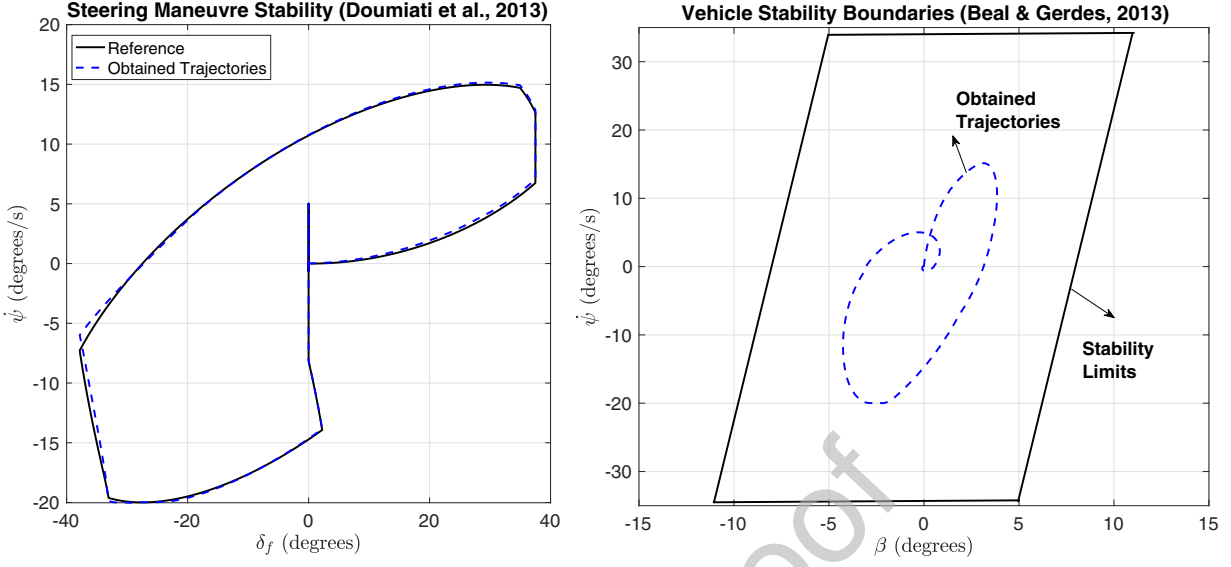


Figure 8: Vehicle Stability Regions.

$$Y_{\text{Corollary 3.10}}(\rho) = \gamma_1(\rho) \left(\begin{bmatrix} 9.26 & -5 \cdot 10^{-3} & 3 \cdot 10^{-5} & -1 \cdot 10^{-3} \\ * & 0.74 & -1 \cdot 10^{-3} & -1 \cdot 10^{-6} \\ * & * & 2 \cdot 10^{-4} & -2 \cdot 10^{-5} \\ * & * & * & 4 \cdot 10^{-6} \end{bmatrix} \cdot 10^{-3} \right)^{-1} \quad (60)$$

$$+ \gamma_2(\rho) \left(\begin{bmatrix} 7.97 & 5 \cdot 10^{-3} & -3 \cdot 10^{-6} & -8 \cdot 10^{-4} \\ * & 0.64 & -1.1 \cdot 10^{-3} & 1 \cdot 10^{-5} \\ * & * & 2 \cdot 10^{-4} & -1 \cdot 10^{-5} \\ * & * & * & 4 \cdot 10^{-6} \end{bmatrix} \cdot 10^{-3} \right)^{-1},$$

$$Y_{\text{Corollary 3.11}}(\rho) = \gamma_1(\rho) \left(\begin{bmatrix} 4.7 & -0.05 & 1 \cdot 10^{-4} & -7 \cdot 10^{-4} \\ * & 0.58 & -1 \cdot 10^{-3} & 2 \cdot 10^{-5} \\ * & * & 1 \cdot 10^{-4} & -1 \cdot 10^{-5} \\ * & * & * & 3 \cdot 10^{-6} \end{bmatrix} \cdot 10^{-3} \right)^{-1} \quad (61)$$

$$+ \gamma_2(\rho) \left(\begin{bmatrix} 5.2 & -0.05 & 1 \cdot 10^{-4} & -8 \cdot 10^{-4} \\ * & 0.64 & -1 \cdot 10^{-3} & 2 \cdot 10^{-6} \\ * & * & 2 \cdot 10^{-4} & -1 \cdot 10^{-5} \\ * & * & * & 4 \cdot 10^{-6} \end{bmatrix} \cdot 10^{-3} \right)^{-1},$$

where:

$$\gamma_1(\rho) = \left(\frac{\bar{\rho}_1 - \underline{\rho}_1}{\bar{\rho}_1 - \underline{\rho}_1} \right) \quad \text{and} \quad \gamma_2(\rho) = \left(\frac{\rho_1 - \underline{\rho}_1}{\bar{\rho}_1 - \underline{\rho}_1} \right). \quad (62)$$

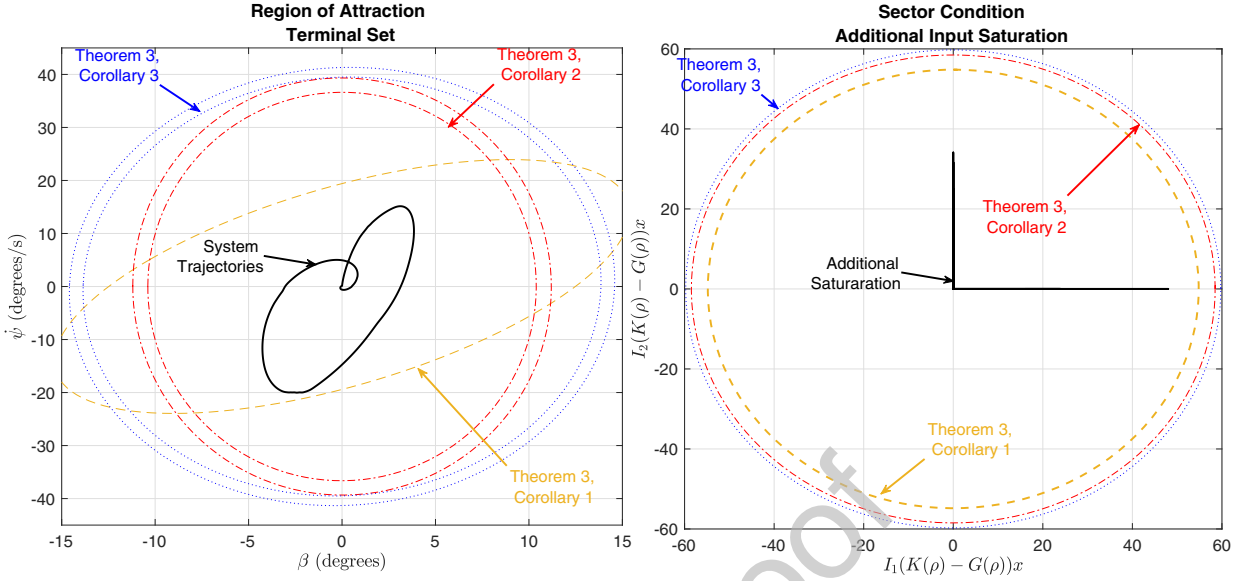


Figure 9: Regions of Attraction, Sector Condition.

$$S_{\text{Corollary 3.9}} = \begin{bmatrix} 0.33 \cdot 10^{-3} & 0 \\ \star & 0.33 \cdot 10^{-3} \end{bmatrix}, \quad (63)$$

$$S_{\text{Corollary 3.10}} = \begin{bmatrix} 0.28 \cdot 10^{-3} & 0 \\ \star & 0.28 \cdot 10^{-3} \end{bmatrix}, \quad (64)$$

$$S_{\text{Corollary 3.11}} = \begin{bmatrix} 0.29 \cdot 10^{-3} & 0 \\ \star & 0.29 \cdot 10^{-3} \end{bmatrix}. \quad (65)$$

4.6.2. Discussion

Based on the results shown in the prequel, we present some concluding remarks:

- Indeed, a full-blown NMPC method that uses nonlinear realistic predictions yields good results for the considered application, despite faults (represented by time-varying input saturation). Nevertheless, this kind of method is not easily applicable in real-time embedded control units, since the average computational time needed to compute the control policy is longer than the sampling period of the system. Moreover, if the fault-induced saturation effects are not taken into account in the design procedure, the performances are affected.
- Regarding performances, we stress that the proposed fault tolerant MPC scheme is able to obtain good results in both faulty and fault-less conditions, and for the resulting algorithm can be evaluated within the sampling period threshold of 5 ms. Moreover, the proposed method ensures poly-quadratic stability guarantees and recursive feasibility property, even in the case of additional saturation (due to input-faults in the front and rear braking systems). When faults occur, we demonstrate that a significant enhancement is made viable regarding yaw rate tracking, which tracks the reference trajectory with a 28% smaller RMS.
- The proposed methodology is, in its core, a passive Fault Tolerant Control method, considering that the actuators present possible losses of effectiveness, which are conversely translated as additional input saturation. The MPC algorithm itself does not have to be “warned” when extra saturation

occurs and naturally ensures stabilisation, leveraging from the Lyapunov condition augmented with the sector arguments.

5. Conclusions

In this article, we presented a fault tolerant control method for qLPV systems subject input faults, which are represented by the means of a fault-induced time-varying saturation effect. The proposed algorithm is based on a robust min-max MPC, using small prediction horizons in order to ensure that the optimisation procedure does not violate the sampling period threshold. A new Theorem is presented for the construction of terminal ingredients which ensure closed-loop stabilisation and recursive feasibility of the algorithm, despite the input saturation caused by faults on the actuators. As evidenced by the realistic simulation results applied to a vehicle braking system, the proposed method is able yield effective reference tracking performances, nearly as good as a full nonlinear MPC method. Some key points of this work should be emphasised:

- The proposed control algorithm is able to operate in real-time, which is a significant advantage over regular robust MPCs. The proposed scheme was applied in a real-time application with a sampling frequency of 200 Hz and tested through realistic simulation scenarios, considering complex nonlinearities.
- The terminal ingredients are provided through LMI-solvable problems, ensuring local poly-quadratic stability of the closed-loop system regulated by the proposed short-sighted algorithm. These LMIs also verify recursive feasibility and enable one to estimate the region of attraction of these algorithms.
- The method inherently ensures stabilisation despite the fault-induced saturation, maintaining performances and avoiding degradation.

For further works, an interesting theme is to further compare the proposed methods considering different kinds of applications. Furthermore, our method could be extended to the case of admissible time-varying references, which can be treated through online reference planners or time-varying terminal sets. Validation tests are also expected to be performed considering a real vehicle testbed.

Acknowledgements

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] V. Puig, Fault diagnosis and fault tolerant control using set-membership approaches: Application to real case studies, *Int. J. Appl. Math. Comput. Sci* 20 (4) (2010) 619–635.
- [2] S. Yin, H. Luo, S. X. Ding, Real-time implementation of fault-tolerant control systems with performance optimization, *IEEE Transactions on Industrial Electronics* 61 (5) (2013) 2402–2411.
- [3] D. Rotondo, A. Cristofaro, T. A. Johansen, F. Nejjari, V. Puig, Robust fault and icing diagnosis in unmanned aerial vehicles using LPV interval observers, *International Journal of Robust and Nonlinear Control* 29 (16) (2019) 5456–5480.
- [4] Y. Wang, D. Rotondo, V. Puig, G. Cembrano, Fault-tolerant control based on virtual actuator and sensor for discrete-time descriptor systems, *IEEE Transactions on Circuits and Systems I: Regular Papers* 67 (12) (2020) 5316–5325.
- [5] M. Blanke, M. Kinnaert, J. Lunze, M. Staroswiecki, J. Schröder, *Diagnosis and fault-tolerant control*, Vol. 2, Springer, 2006.
- [6] M. M. Morato, O. Sename, L. Dugard, LPV-MPC fault tolerant control of automotive suspension dampers, *IFAC-PapersOnLine* 51 (26) (2018) 31–36.
- [7] M. M. Morato, P. R. Mendes, J. E. Normey-Rico, C. Bordons, LPV-MPC fault-tolerant energy management strategy for renewable microgrids, *International Journal of Electrical Power & Energy Systems* 117 (2020) 105644.
- [8] D. Rotondo, V. Puig, F. Nejjari, J. Romera, A fault-hiding approach for the switching quasi-LPV fault-tolerant control of a four-wheeled omnidirectional mobile robot, *IEEE Transactions on Industrial Electronics* 62 (6) (2014) 3932–3944.
- [9] J. D. Boskovic, R. K. Mehra, Fault accommodation using model predictive methods, in: *Proceedings of the 2002 American Control Conference*, Vol. 6, IEEE, 2002, pp. 5104–5109.
- [10] C. Lebreton, C. Damour, M. Benne, B. Grondin-Perez, J.-P. Chabriat, Passive fault tolerant control of PEMFC air feeding system, *International Journal of Hydrogen Energy* 41 (34) (2016) 15615–15621.
- [11] P. Mhaskar, Robust model predictive control design for fault-tolerant control of process systems, *Industrial & engineering chemistry research* 45 (25) (2006) 8565–8574.
- [12] C. Hu, X. Wei, Y. Ren, Passive fault-tolerant control based on weighted LPV tube-MPC for air-breathing hypersonic vehicles, *International Journal of Control, Automation and Systems* 17 (8) (2019) 1957–1970.
- [13] Y. Wang, S. Boyd, Fast model predictive control using online optimization, *IEEE Transactions on control systems technology* 18 (2) (2009) 267–278.
- [14] C. N. Jones, A. Domahidi, M. Morari, S. Richter, F. Ullmann, M. Zeilinger, Fast predictive control: Real-time computation and certification, *IFAC Proceedings Volumes* 45 (17) (2012) 94–98.
- [15] D. Limon, A. Ferramosca, I. Alvarado, T. Alamo, Nonlinear MPC for tracking piece-wise constant reference signals, *IEEE Transactions on Automatic Control* 63 (11) (2018) 3735–3750.
- [16] J. Mohammadpour, C. W. Scherer, *Control of linear parameter varying systems with applications*, Springer Science & Business Media, 2012.
- [17] C. Hoffmann, H. Werner, A survey of linear parameter-varying control applications validated by experiments or high-fidelity simulations, *IEEE Transactions on Control Systems Technology* 23 (2) (2014) 416–433.
- [18] Q. Hu, B. Xiao, M. Friswell, Robust fault-tolerant control for spacecraft attitude stabilisation subject to input saturation, *IET Control Theory & Applications* 5 (2) (2011) 271–282.
- [19] M. Chen, B. Jiang, R. Cui, Actuator fault-tolerant control of ocean surface vessels with input saturation, *International Journal of Robust and Nonlinear Control* 26 (3) (2016) 542–564.
- [20] M. M. Morato, J. E. Normey-Rico, O. Sename, Model predictive control design for linear parameter varying systems: A survey, *Annual Reviews in Control* 49 (2020) 64 – 80.
- [21] Y.-Y. Cao, Z. Lin, Min-max MPC algorithm for LPV systems subject to input saturation, *IET Proceedings-Control Theory and Applications* 152 (3) (2005) 266–272.
- [22] H. S. Abbas, J. Hanema, R. Tóth, J. Mohammadpour, N. Meskin, A new approach to robust MPC design for LPV systems in input-output form, *IFAC-PapersOnLine* 51 (26) (2018) 112–117.
- [23] M. M. Morato, J. E. Normey-Rico, O. Sename, A fast dissipative robust nonlinear model predictive control procedure via quasi-linear parameter varying embedding and parameter extrapolation, *International Journal of Robust and Nonlinear Control* 31 (18) (2021) 9619–9651.
- [24] M. M. Morato, J. E. Normey-Rico, O. Sename, Novel qLPV MPC design with least-squares scheduling prediction, *IFAC-PapersOnLine* 52 (28) (2019) 158–163.
- [25] P. S. González Cisneros, H. Werner, Nonlinear model predictive control for models in quasi-linear parameter varying form, *International Journal of Robust and Nonlinear Control* 30 (10) (2020) 3945–3959.
- [26] M. M. Morato, O. Sename, L. Dugard, M. Q. Nguyen, Fault estimation for automotive electro-rheological dampers: LPV-based observer approach, *Control Engineering Practice* 85 (2019) 11–22.
- [27] M. M. Morato, T.-P. Pham, O. Sename, L. Dugard, Development of a simple ER damper model for fault-tolerant control design, *Journal of the Brazilian Society of Mechanical Sciences and Engineering* 42 (10) (2020) 1–22.
- [28] M. Doumiati, O. Sename, L. Dugard, J.-J. Martinez-Molina, P. Gaspar, Z. Szabo, Integrated vehicle dynamics control via coordination of active front steering and rear braking, *European Journal of Control* 19 (2) (2013) 121–143.
- [29] Y. Su, K. K. Tan, T. H. Lee, Tube-based quasi-min-max output feedback MPC for LPV systems, in: *Preprints of the 8th IFAC Symposium on advanced control of chemical processes*, IFAC, 2012, pp. 10–13.
- [30] M. Jungers, R. C. Oliveira, P. L. Peres, MPC for LPV systems with bounded parameter variations, *International Journal of Control* 84 (1) (2011) 24–36.
- [31] J. Köhler, M. A. Müller, F. Allgöwer, A nonlinear tracking model predictive control scheme for dynamic target signals, *Automatica* 118 (2020) 109030.

- [32] M. M. Morato, J. E. Normey-Rico, O. Sename, Short-sighted robust LPV model predictive control: Application to semi-active suspension systems, in: 2021 European Control Conference (ECC), IEEE, 2021, pp. 1525–1530.
- [33] T. Başar, G. J. Olsder, Dynamic noncooperative game theory, SIAM, 1998.
- [34] M. Hamann, B. Strasser, Graph bisection with pareto optimization, Journal of Experimental Algorithmics 23 (2018) 1–34.
- [35] D. Q. Mayne, J. B. Rawlings, C. V. Rao, P. O. Scokaert, Constrained model predictive control: Stability and optimality, Automatica 36 (6) (2000) 789–814.
- [36] S. Tarbouriech, G. Garcia, J. M. G. da Silva Jr, I. Queinnec, Stability and stabilization of linear systems with saturating actuators, Springer Science & Business Media, 2011.
- [37] L. S. Figueiredo, M. J. Larcercda, V. J. S. Leite, Local poly-quadratic stabilization of systems with saturating actuators (Text in Portuguese), in: Proceedings of the 14th Brazilian Symposium of Intelligent Automation (SBAI), Ouro Preto, Brazil, Oct. 27-30, 2019, SBA, 2019, pp. 1–6. doi:10.17648/sbai-2019-111476.
- [38] J. G. Da Silva, S. Tarbouriech, Antiwindup design with guaranteed regions of stability: an LMI-based approach, IEEE Transactions on Automatic Control 50 (1) (2005) 106–111.
- [39] D. Peaucelle, S. Tarbouriech, M. Ganet-Schoeller, S. Bannani, Evaluating regions of attraction of LTI systems with saturation in IQS framework, IFAC Proceedings Volumes 45 (13) (2012) 242–247.
- [40] M. Jungers, E. B. Castelan, Gain-scheduled output control design for a class of discrete-time nonlinear systems with saturating actuators, Systems & Control Letters 60 (3) (2011) 169–173.
- [41] A. Pandey, M. C. de Oliveira, Quadratic and poly-quadratic discrete-time stabilizability of linear parameter-varying systems, IFAC-PapersOnLine 50 (1) (2017) 8624–8629.
- [42] F. Allgöwer, A. Zheng, Nonlinear model predictive control, Vol. 26, Birkhäuser, 2012.
- [43] U. Kiencke, L. Nielsen, Automotive control systems: for engine, driveline, and vehicle, Springer Science & Business Media, 2005.
- [44] C. Pousot-Vassal, C. Spelta, O. Sename, S. M. Savaresi, L. Dugard, Survey and performance evaluation on some automotive semi-active suspension control methods: A comparative study on a single-corner model, Annual Reviews in Control 36 (1) (2012) 148–160.
- [45] P. Falcone, F. Borrelli, J. Asgari, H. E. Tseng, D. Hrovat, Predictive active steering control for autonomous vehicle systems, IEEE Transactions on control systems technology 15 (3) (2007) 566–580.
- [46] P. Falcone, H. E. Tseng, J. Asgari, F. Borrelli, D. Hrovat, Integrated braking and steering model predictive control approach in autonomous vehicles, IFAC Proceedings Volumes 40 (10) (2007) 273–278.
- [47] C. E. Beal, J. C. Gerdes, Model predictive control for vehicle stabilization at the limits of handling, IEEE Transactions on Control Systems Technology 21 (4) (2012) 1258–1269.
- [48] C. Pousot-Vassal, O. Sename, L. Dugard, S. Savaresi, Vehicle dynamic stability improvements through gain-scheduled steering and braking control, Vehicle System Dynamics 49 (10) (2011) 1597–1621.
- [49] J. Kong, M. Pfeiffer, G. Schildbach, F. Borrelli, Kinematic and dynamic vehicle models for autonomous driving control design, in: 2015 IEEE Intelligent Vehicles Symposium (IV), IEEE, 2015, pp. 1094–1099.
- [50] P. Polack, F. Althé, B. d’Andréa Novel, A. de La Fortelle, The kinematic bicycle model: A consistent model for planning feasible trajectories for autonomous vehicles?, in: 2017 IEEE Intelligent Vehicles Symposium (IV), IEEE, 2017, pp. 812–818.
- [51] M. M. Morato, M. Q. Nguyen, O. Sename, L. Dugard, Design of a fast real-time LPV model predictive control system for semi-active suspension control of a full vehicle, Journal of the Franklin Institute.
- [52] J. Löfberg, Oops! I cannot do it again: Testing for recursive feasibility in MPC, Automatica 48 (3) (2012) 550–555.

Appendix A. Proof of Lemma 1

Assume that $x(0) = x_r^*$ is some initial steady-state and that the reached steady-state $x_r^* \neq x_r$. Since the system at instant k_0 has stabilised at x_r^* , x_r^* is a steady-state state condition for the polytopic qLPV system described by Eq. (1). The control sequence U_{k_0-1} that gave the steady input u_r^* that leads to x_r^* is the optimal solution of the constrained QP and, therefore, $J = \|x(k_0|k_0-1) - x_r\|_Q^2 + \|u_r^* - u_r\|_R^2 + \|x(k_0 + N_p|k_0-1) - x_r\|_P^2$. Take $x_r^* \neq x_r$. This leads to the hypothesis that there exist two ambiguous steady-state conditions $x_r^* \in \mathcal{X}$ and $x_r \in \mathcal{X}$ (and respective inputs $u_r^* \in \mathcal{U}$ and $u_r \in \mathcal{U}$) such that the control sequence U_{k_0} derived from the control law $u(k_0) = K(\rho(k_0))x(k_0)$ is indeed admissible. It yields from A2 in Theorem 3.1 that $V(x_r^* - x_r) \leq \beta_2(\|x_r^* - x_r\|) < \|x_r^* - x_r\|_P^2$. Then, take $\|x_r^* - x_r\|_P^2 = V^*(x_r^* - x_r)$ as another optimal cost. Note that if $V^*(x_r^* - x_r)$ is indeed an optimal cost, the above inequality contradicts the optimality of $V(\cdot)$. Therefore, it can only remain that $x_r^* = x_r$, which imperiously means that $x(k_0) = x_r$. This proves that the reached steady-state is indeed the steady-state reference x_r .

Appendix B. Proof of Theorem 2

Consider $u^*(k)$ as the optimal solution obtained from solving Eq. (10) at time k . The internal model-based prediction associated to the optimisation is denoted $x^*(k+1) = A(\rho(k))x^*(k) + B(\rho(k))u^*(k)$, for

which $x^*(0) = x_k$ is the initial value taken for these predictions departing from a measured real value $x(k) = x_k$. *Feasibility*: Assuming that recursive feasibility has been verified, it is easy to see that if the state at the current time is admissible, i.e. $x(k) \in \mathcal{X}$, and that the optimal solution $u^*(k)$ derives from an optimal cost J^* , then the actual states for the future instant $x(k+1)$ is conducted by a future sequence of control inputs given by U_{k+1} , whose first entry is $u^*(k+1|k+1)$. This means that $u^*(k+1) = u^*(k+1|k+1)$ is feasible due to the feasibility of the optimal solution at instant k , which consequently steers the states to an admissible $x(k+1) \in \mathcal{X}$. *Convergence*: Considering that recursive feasibility is verified, the solution $u^*(k+1)$ for all $k+1 \in \mathbb{Z}$ exists. Assume that the target steady-state reference is fixed along the horizon given by x_r . Then, it is implied that Condition (C3) holds due to the recursive feasibility property, which equivalently leads to:

$$\begin{aligned} V^*(x(k+1)) &\leq V(x(k)) - \ell(x(k), u(k-1)) \\ &\leq V^*(x(k)) - \|x(k) - x_r^*\|_Q^2 - \|u(k-1) - u_r\|_R^2. \end{aligned} \quad (\text{B.1})$$

It is reasonable to verify that $\|u(k-1) - u_r\|_R^2$ has a constant value, since $u(k-1)$ is a control policy that has already been applied to the process. Moreover, note that the above inequality has a positive definite optimal cost V^* with non-increasing evolution along k . Therefore, we can directly infer that $\lim_{k \rightarrow \infty} \|x_k - x_r^*\|_Q = 0$ and, from the use of Lemma 3.2, and it is implied that $\lim_{k \rightarrow \infty} \|x_r^* - x_r\|_P = 0$. Consequently, it is true that $x(k)$ is steered to x_r if this is indeed an admissible steady-state state condition. This concludes the proof.

Appendix C. Proof of Theorem 3, Corollaries 1, 2, and 3

We proceed by demonstrating the validity of this Theorem in a generic sense, considering parameter-dependent matrices. We note that the actual solutions depend on the formulations regarding the knowledge of the process, as gave the previous Corollaries and their according matrix manipulations.

We show that each of the five conditions from Theorem 3.1 are satisfied through the LMI problem in Eq. (24), from (C1) to (C5). This proof takes $(x_r, u_r) = (0_{n_x}, 0_{n_u})$, for notation ease, as done in the prior.

Firstly, we note that condition (C1) trivially holds due to the form of $\mathbf{X}_f(\cdot)$. As gives Eq. (13), this terminal set is defined as an ellipsoid, which holds the origin of \mathbb{R}^{n_x} in its interior, by construction.

The second condition (C2), of control invariance of the terminal set $\mathbf{X}_f(\cdot)$, is verified due to the fact that this set is taken as a sub-level set of the terminal cost $V(\cdot)$. Therefore, if condition (C3) is verified, (C2) is consequently ensured.

The fourth condition (C4) is verified through the LMI in Eq. (26). Firstly, we replace $L(\rho)$ and $Y(\rho)$ by $K(\rho)P^{-1}(\rho)$ and $P^{-1}(\rho)$, respectively, which gives:

$$\left[\begin{array}{c|c} P^{-1}(\rho) & \star \\ \hline I_i K(\rho) P^{-1}(\rho) & \bar{u}_i^2 \end{array} \right] > 0. \quad (\text{C.1})$$

Pre and post-multiplying this LMI by $\text{diag}\{P(\rho), 1\}$ and its transpose, respectively, yields:

$$\left[\begin{array}{c|c} P(\rho) & \star \\ \hline I_i K(\rho) & \bar{u}_i^2 \end{array} \right] > 0. \quad (\text{C.2})$$

Applying a Schur complement to this LMI leads to:

$$P(\rho) - (I_i K(\rho))^T \left(\frac{1}{\bar{u}_i^2} \right) (K(\rho) I_i) > 0. \quad (\text{C.3})$$

This inequality is a sufficient condition for $\frac{\|I_i K(\rho)x\|^2}{(\bar{u}_i)^2} \leq x^T P(\rho)x, \forall x$. Due to the fact that the maximum norm of the projection Fx of an x that belongs to some ellipsoid $x^T P x \leq 1$ is given by $\sqrt{F^T (P^{-1}) F}$, it holds that Eq. (C.3) implies that the projection $I_i K(\rho)x$ (i.e. i -th control signal) is upper-bounded, in norm, by \bar{u}_i . Refer to [52]. This satisfies (C4).

Finally, (C5) is verified in a similar procedure to the last step of the verification of (C4). Applying the Schur complement to (28), it follows that:

$$I_j^T(Y(\rho))I_j < \bar{x}_i^2. \quad (\text{C.4})$$

Eq. (C.4) ensures that the projection $I_j x$ (i.e. j -th state) is norm-bounded by \bar{x}_j , which satisfies condition (C5).

Now, we show that LMI (27) yields a the agreement of the sector condition of Eq. (19). Note that $x^T P(\rho)x \leq 1$ and $\|I_i(K(\rho) - G(\rho))x\|^2 \leq (\bar{u}^+)^2$ is a sufficient condition for $\text{SC} \leq 0$. Thus, we aim to ensure that $(K(\rho) - G(\rho))x$ is norm bounded by \bar{u}^+ for all x inside the ellipsoid. We use a sufficient condition for $\frac{\|I_i(K(\rho) - G(\rho))x\|^2}{(\bar{u}^+)^2} \leq x^T P(\rho)x$:

$$(I_i(K(\rho) - G(\rho)))^T \left(\frac{1}{(\bar{u}^+)^2} \right) (I_i(K(\rho) - G(\rho))) - P(\rho) < 0, \quad (\text{C.5})$$

which can be rearranged as:

$$P(\rho) - (I_i(K(\rho) - G(\rho)))^T \left(\frac{1}{(\bar{u}^+)^2} \right) (I_i(K(\rho) - G(\rho))) > 0. \quad (\text{C.6})$$

Applying a Schur complement to this last inequality leads to:

$$\left[\begin{array}{c|c} P(\rho) & \star \\ \hline I_i(K(\rho) - G(\rho)) & \frac{1}{(\bar{u}^+)^2} \end{array} \right] > 0, \quad (\text{C.7})$$

Pre and post-multiplying this inequality by $\text{diag}\{P^{-1}(\rho), 1\}$ and its transpose, respectively, leads to:

$$\left[\begin{array}{c|c} Y(\rho) & \star \\ \hline I_i(K(\rho) - G(\rho))Y(\rho) & \frac{1}{(\bar{u}^+)^2} \end{array} \right] > 0. \quad (\text{C.8})$$

Using $L(\rho) = K(\rho)Y(\rho)$ and $W(\rho) = G(\rho)Y(\rho)$ in inequality (C.8), we get LMI (27), which ensures that the projection of $I_i(K(\rho) - G(\rho))x$ is upper-bounded, in norm, by the stricter input saturation given by (\bar{u}_i^+) . The satisfaction of this sector condition is equivalent to the satisfaction of (C4) through LMI (26).

Regarding the discrete-time Lyapunov condition (C3), as adapted to include the Sector Condition as of Lemma 3.5, we would like to have a generic LMI condition that ensures:

$$V(A_{cl}(\rho), \rho^+) - V(x, \rho) + x^T Qx + x^T K^T(\rho)RK(\rho)x - 2\Psi^T(x)S(\Psi(x) - G(\rho)x) \leq 0. \quad (\text{C.9})$$

We note that inequality (C.9) is equivalent to:

$$x^T A_{cl}^T(\rho)P(\rho^+)A_{cl}(\rho)x - x^T P(\rho)x + x^T Qx + x^T K^T(\rho)RK(\rho)x - 2\Psi^T(x)S(\Psi(x) - G(\rho)x) \leq 0, \quad (\text{C.10})$$

or, likewise, in matrix form:

$$\begin{pmatrix} x \\ \Psi(x) \end{pmatrix}^T \left[\begin{array}{cc|ccc} A_{cl}^T(\rho)P(\rho^+)A_{cl}(\rho) - P(\rho) + Q + K^T(\rho)RK(\rho) & \star & & & \\ & SG(\rho) & & & \\ \hline & & P^{-1}(\rho^+) & \star & \star \\ & & 0 & Q^{-1} & \star \\ & & 0 & 0 & R^{-1} \end{array} \right] \begin{pmatrix} x \\ \Psi(x) \end{pmatrix} \leq 0, \quad \forall (x, \Psi(x)). \quad (\text{C.11})$$

A sufficient condition for inequality (C.11) is:

$$\left[\begin{array}{cc|ccc} P(\rho) - A_{cl}^T(\rho)P(\rho^+)A_{cl}(\rho) - Q - K^T(\rho)RK(\rho) & \star & & & \\ & -SG(\rho) & & & \\ \hline & & P^{-1}(\rho^+) & \star & \star \\ & & 0 & Q^{-1} & \star \\ & & 0 & 0 & R^{-1} \end{array} \right] > 0. \quad (\text{C.12})$$

Applying two consecutive Schur complements to inequality (C.12), we get:

$$\left[\begin{array}{cc|ccc} P(\rho) & \star & \star & \star & \star \\ -SG(\rho) & 2S & \star & \star & \star \\ \hline A_{cl}(\rho) & 0 & P^{-1}(\rho^+) & \star & \star \\ \text{I} & 0 & 0 & Q^{-1} & \star \\ K(\rho) & 0 & 0 & 0 & R^{-1} \end{array} \right] > 0. \quad (\text{C.13})$$

We can pre and post-multiply inequality (C.13) by $\text{diag}\{P^{-1}(\rho), \mathbb{I}, \mathbb{I}, \mathbb{I}, \mathbb{I}\}$ and its transpose, respectively, which leads to:

$$\left[\begin{array}{cc|ccc} P^{-1}(\rho) & \star & \star & \star & \star \\ -SG(\rho)P^{-1}(\rho) & 2S & \star & \star & \star \\ \hline A_{cl}(\rho)P^{-1}(\rho) & 0 & P^{-1}(\rho^+) & \star & \star \\ P^{-1}(\rho) & 0 & 0 & Q^{-1} & \star \\ K(\rho)P^{-1}(\rho) & 0 & 0 & 0 & R^{-1} \end{array} \right] > 0. \quad (\text{C.14})$$

Using $Y(\rho) = P^{-1}(\rho) > 0$, $L(\rho) = K(\rho)Y(\rho)$ and $W(\rho) = G(\rho)Y(\rho)$, we get:

$$\left[\begin{array}{cc|ccc} Y(\rho) & \star & \star & \star & \star \\ -SW(\rho) & 2S & \star & \star & \star \\ \hline A(\rho)Y(\rho) + B(\rho)L(\rho) & 0 & Y(\rho^+) & \star & \star \\ Y(\rho) & 0 & 0 & Q^{-1} & \star \\ L(\rho) & 0 & 0 & 0 & R^{-1} \end{array} \right] > 0. \quad (\text{C.15})$$

We note that the polytope derived through inequality (C.15) is written in terms of $Y(\rho) = P^{-1}(\rho)$ and not in terms of $P(\rho)$. In the general parameter-dependent case, we take $Y(\rho) = \sum_{i=0}^{2^{n_\rho}} \gamma_i Y_i$, which translates to $P(\rho) = \left(\sum_{i=0}^{2^{n_\rho}} \gamma_i Y_i \right)^{-1}$, meaning that we find not a polytope over P but the inverse of a summation in terms of Y .

Likewise, the polytopes on L and W are respectively given as: $L(\rho) = L_0 + \sum_{i=0}^{2^{n_\rho}} \gamma_i L_i$ and $W(\rho) = W_0 + \sum_{i=0}^{2^{n_\rho}} \gamma_i W_i$. Therefore, we get: $K(\rho) = L(\rho)Y^{-1}(\rho) = \left(L_0 + \sum_{i=0}^{2^{n_\rho}} \gamma_i L_i \right) \left(\sum_{j=0}^{2^{n_\rho}} \gamma_j Y_j \right)^{-1}$ and $G(\rho) = W(\rho)Y^{-1}(\rho) = \left(W_0 + \sum_{i=0}^{2^{n_\rho}} \gamma_i W_i \right) \left(\sum_{j=0}^{2^{n_\rho}} \gamma_j Y_j \right)^{-1}$.

Note, thus, that inequality (C.15) can be converted into an LMI that is somehow simple and more common, whereas with more complex variables $P(\rho)$, $K(\rho)$ and $G(\rho)$. Note that $A(\rho)$ and $B(\rho)$ are known, which means that only the nonlinear term $SW(\rho)$ should be eliminated. Anyhow, since $S > 0$, we can pre and post-multiply inequality (C.15) by $\text{diag}\{\mathbb{I}, S^{-1}, \mathbb{I}, \mathbb{I}, \mathbb{I}\}$ and use a change of variables as $T = S^{-1}$, which leads to:

$$\left[\begin{array}{cc|ccc} Y(\rho) & \star & \star & \star & \star \\ -W(\rho) & 2T & \star & \star & \star \\ \hline A(\rho)Y(\rho) + B(\rho)L(\rho) & 0 & Y(\rho^+) & \star & \star \\ Y(\rho) & 0 & 0 & Q^{-1} & \star \\ L(\rho) & 0 & 0 & 0 & R^{-1} \end{array} \right] > 0. \quad (\text{C.16})$$

Note that inequality (C.16) is indeed an LMI over Y , W , L and T . This LMI is a sufficient condition for the Lyapunov inequality in (C3), while respecting the sector condition.

Finally, we note that the LMI problem proposed in this Theorem is such that the level set $\mathcal{L}_V(1)$, as defined by the intersection of level sets in Eq. (23), is maximised. This is, by taking $Y(\rho) = \sum_{i=1}^{n_\rho} \rho_i Y_i$ with $\sum_{i=1}^{n_\rho} \rho_i = 1$, we choose:

$$\mathcal{R}_E = \bigcap_{i=1}^{n_\rho} \mathcal{E}(Y_i^{-1}, 1) = \bigcap_{i=1}^{n_\rho} \{x \in \mathbb{R}^{n_x} \mid (x^T Y_i^{-1} x) \leq 1\}. \quad (\text{C.17})$$

Indeed, we show that $\mathcal{R}_E \subset \mathcal{L}_{V(\cdot, \cdot)}(1)$. Note that, if $x \in \mathcal{R}_E$, then, $\forall i = 1, \dots, n_\rho$, we have that $(x^T Y_i^{-1} x) \leq 1$. Since each $Y_i^{-1} > 0$ is invertible, by definition, we can apply a Schur complement to this inequality, yielding:

$$\begin{bmatrix} 1 & \star \\ x & Y_i \end{bmatrix} \geq 0, \forall i, \quad (\text{C.18})$$

Thence, for $\rho_0 = f_\rho(x)$, with $\sum_{i=1}^{n_\rho} \rho_{0_i} = 1$, we have that:

$$\sum_{i=1}^{n_\rho} \rho_{0_i} \begin{bmatrix} 1 & \star \\ x & Y_i \end{bmatrix} = \begin{bmatrix} 1 & \star \\ x & Y(\rho) \end{bmatrix} \geq 0, \forall i, \quad (\text{C.19})$$

for which another Schur complement can be applied, owing to $x^T Y^{-1}(\rho_0)x \leq 1$, which leads to $x^T P(\rho_0)x \leq 1$ and, consequently, to $x \in \mathcal{L}_{V(\cdot, \cdot)}(1)$.

Therefore, the level sets can be directly taken as the intersection of the ellipsoids in terms of each Y_i . Thus, we write the LMI problem in order to enlarge each ellipsoid $\mathcal{E}(Y_i^{-1}, 1)$ in the trace sense, as follows:

$$\begin{bmatrix} Y_i & \mathbb{I} \\ \star & H \end{bmatrix} \geq 0, \forall i. \quad (\text{C.20})$$

This concludes the proof.

Journal Pre-proof

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

All persons who have made substantial contributions to the work reported in the manuscript. All funding has been reported in the Acknowledgements section of the manuscript.

Journal Pre-proof