Sub-optimal recursively feasible Linear Parameter-Varying predictive algorithm for semi-active suspension control

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

A Sub-Optimal Recursively Feasible Linear Parameter Varying Predictive Algorithm for Semi-Active Suspension Control

Marcelo Menezes Morato[a,b], Julio E. Normey-Rico[a], Olivier Sename[b]

[a] *Departamento de Automação e Sistemas (*DAS*), Universidade Federal de Santa Catarina, Florianópolis, Brazil*
[b] *Université Grenoble Alpes, CNRS, GIPSA-lab, Grenoble, France*

Abstract

This paper proposed a control algorithm to enhance the comfort of passengers in a vehicle equipped with Semi-Active Suspension systems, manipulating the force delivered by the Semi-Active damper. For this goal, the vertical dynamics of the car are represented through a quasi-Linear Parameter Varying (qLPV) model. Then, the algorithm resides in solving a set-constrained Model Predictive Control (MPC) problem, embedding a comfort performance index to the MPC optimization function. The sub-optimality of this algorithm resides in the fact that the MPC is synthesized considering a frozen guess for the evolution of the qLPV scheduling parameters along the prediction horizon. Assuming bounds on the variation rates of the qLPV scheduling parameters, the method enables a replacement of the original complex nonlinear LPV MPC optimization by a much simpler Quadratic Program (QP). This QP includes a Lyapunov-decreasing stage cost and embeds set-based terminal ingredients, which guarantee that the domain of attraction of the controller is enlarged and that recursive feasibility can be maintained despite the non-exact model scheduling (and suboptimality). The control structure is tested and compared to other optimal control approaches, such as a clipped Linear Quadratic Regulator. The paper ends with successful realistic nonlinear simulations of a 1/5-scaled car with Electro-Rheological suspensions, which illustrate the overall good operation and behaviour achieved with the proposed regulation algorithm. With the proposed method, the comfort of the passengers is substantially improved.

1. Introduction

1.1. Semi-Active Suspension Systems

The suspension system of a car is the mechanism that acts to enhance the driving performance with respect to roll handling and passenger comfort. Semi-Active (SA) suspensions are today the standard component in many state-of-the-art high-range cars and a good deal of academic and industrial research works have been focused on their control. SA suspensions are well-performing, energy-efficient and altogether less expensive than Active ones, for instance. Further context and details can be found in [1, 2].

Many design algorithms have been proposed for the SA suspension control problem. The main issue is how to handle the dissipativity constraints of the SA dampers while ensuring good performances. Papers [3, 4] detail some the available methods proposed for this goal (see references therein). Some of the most modern techniques have been tested, such as clipped optimal LQRs in [5], H_∞ techniques [6], Linear Parameter Varying (LPV) approaches [7], and nonlinear methods, such as backstepping [8].

1.2. Predictive Control

Although all these approaches achieve good performances, a more natural framework for optimal control of constrained process is Model Predictive Control (MPC) [9]. MPC is indisputably a very well established

Email addresses: marcelomnzm@gmail.com (Marcelo Menezes Morato), julio.normey@ufsc.br (Julio E. Normey-Rico), olivier.sename@gipsa-lab.fr (Olivier Sename)

feedback control technique, with success in many industrial applications. This method allows to explicitly consider input and state constraints in the design process and solves the constrained optimization problem at each sampling instant to determine the control policy. For its formulation, MPC requires a fidel process model, so that the future output predictions can be expressed with respect to current input and state values.

The original MPC framework was attached to the idea of processes with linear time-invariant (LTI) models. This condition is inherently violated for almost all plants with a wide range of operating conditions, which require nonlinear forecasting models. Therefore, MPC has been progressively extended to embed nonlinear model predictions (NMPC) as discusses [10], but this kind of nonlinear design is not trivial and has increased numerical complexity, suffering from issues related to their high complexity when sought to run in real-time. It is a known fact that the computational requirements of MPC may be excessive in many situations, since the optimization problem which has to be solved online, at each sampling instant.

SA suspension control consists, basically, in varying the damping coefficient, which implies in variations on the delivered force. The dissipativity constraints of the damper are, thus, input constraints, and this kind of problem falls into a saturation paradigm which is elegantly dealt with by MPC. Some papers have indeed employed MPC for SA suspension control. In this paper, the focus is given to reduced-order car frameworks (such as the quarter-car or de half-car models), which decouple the vertical dynamics by vehicle corner or side to reduce the complexity of the yielded MPC algorithm (quarter-car models reduce number of states by a third, roughly, with respect to full-car models). The idea of solving the control problem for each vehicle corner (or side) is appealing when passenger comfort is the main concern, because the coupling and load transfer distribution between corners can be neglected as their influence upon comfort is small, as discusses [11]. The following SA suspension control papers that apply MPC reduced-order models are recalled:

- A methodology for performance evaluation of SA suspensions under optimal control algorithm (MPC included) is presented in [12]. The method enables to evaluate passenger comfort and vehicle handling w.r.t. differente road profiles;
- Both [13] and [14] proposed clipped (saturated) versions of optimal control. Since the clipping action is not embedded into the optimization procedure due to nonlinearities that aimed to be avoided, the results do not represent optimality;
- An MPC algorithm is formulated for an LPV quarter-car model in [15]. Anyhow, there are no assessments on feasibility guarantees of the proposed tool, which simply considers as if the scheduling parameters were fixed along the horizon, for the MPC computation;
- In [16], the predictive controller is synthesized for a half-car model with some experimental validation, but the effect of the road disturbances is neglected. The achieved results are interesting and the control policy is implemented within 10 ms.

Remark 1. With respect to the Author's previous works, in [11, 17] fast MPC algorithms were developed considering full car models. The inputed nonlinearity and the dissipativity constraints are adapted using a pre-filter, which makes the model, from the MPC viewpoint, LTI. This pre-filtering technique, for practical purposes, may cause implementation distress, given that a bilinear term $\dot{z}_{def}(k)u(k)$ is converted into a linear term $\dot{z}_{def}(k)u_{nom} + u_f(k)$ by this pre-filter, which means that a division by $\dot{z}_{def}(k)$ is necessary and, for situations when this velocity term approaches zero, the pre-filtering must be adapted. Note that near-zero piston velocity situations are very common in SA suspensions (constant straight road profile, for instance). Moreover, the optimization procedure is adapted using some heuristics (Infeasible Start Newton Method, Primal-Barrier term) so that the MPC can run within 5 ms, which also implies in sub-optimal results. Finally, these two papers lack analysis in the recursive feasibility property of the algorithm, which is required when dealing with multiple road profiles.

1.3. Main Motivation

It seems that the majority of predictive control algorithms applied for automotive SA suspension systems that are able to run in real-time achieve sub-optimal results. This fact does not mean that they do not

enhance the performances of these systems w.r.t. other control frameworks. Indeed, most papers show good performance enhancements. Nonetheless, up to the Author's best knowledge, no paper has presented recursive feasibility assessments on these MPC algorithms, which are very necessary to ensure that the control method can run despite the model simplifications. Therefore, the main motivation of this paper is to present a predictive control algorithm for vehicular SA suspensions that embed the recursive feasibility property.

The majority of reduced-order models for SA suspensions that handle the dissipativity constraints of the dampers have nonlinear characteristics. The pre-filtering method, as discussed in Remark 1 is not such a good option concerning real implementation.

Today, generalized formulations of NMPC are available to deal with such models, but, as highlighted, they are generally not able to run fast enough. Recently, the use of LPV models [18, 19] has been brought to focus to tackle and facilitate the control of nonlinear processes; such models are also nonlinear, but are "coordinated" by bounded, known scheduling parameters ρ. Due to these parameters, LPV models are much simpler to represent than full nonlinear ones, being quite similar to the LTI framework; for this reason, LPV models have become very popular for NMPC control purposes [20]. MPC design based on LPV models has formally been studied since the beginning of the 00's, but this field is still open for investigation (in Section 3, a brief overview of the LPV MPC state-of-the-art is presented).

In fact, feasible LPV MPC algorithms with Quadratic Programming (QP) level complexity are rather scarce. The available methods either rely on heavy offline procedures or are too conservative (solving a robust problem with respect to all possible trajectories for ρ). Moreover, the sub-optimal methods that are not too conservative have a lack o recursive feasibility holds.

1.4. Contributions

Many nonlinear processes can be described within an LPV formalism, as long as linear differential inclusion is respected [21], and so is the case of SA suspension systems.

Motivated by the reasons discussed above, this paper seeks the development of a predictive control policy for SA suspensions that enhances the comfort of the onboard passengers. The proposed algorithm is adapted from [22], is based on an LPV model of the suspension and considers bounded rates of the scheduling parameters, as suggested by [20], and evaluating its recursive feasibility holds. The new method simplifies the computation of the control-invariant sets which are computed using a worst-case scenario for the evolution of the parameters, resulting in a practically implementable version.

Hence, the main contributions of this paper are listed below:

- A control-oriented LPV model for vehicular SA suspension systems is proposed (Section 2). This model is corroborated with respect to the comfort performance indexes proposed by [12]. The dissipativity properties of the SA dampers are embedded to the model as input contraints.

- Using a frozen guess for the future evolution of the scheduling parameters, provided at each sampling instant k, the novel qLPV MPC algorithm is proposed (Section 4). This algorithm is based on a standard QP coupled to contractive terminal set constraints and a Lyapunov-decreasing terminal stage cost. The terminal set is computed using thanks to the bounds on the scheduling parameter variations.

- Considering a high-fidelity nonlinear model for a real vehicle testbed with four Electro-Rheological SA dampers, numerical simulations of the proposed algorithms are presented (Section 5). The proposed method is compared with respect to other algorithms from the literature (as a clipped LQR). The results demonstrate the effectiveness of the proposed tool to enhance the comfort of the onboard passengers; performance indexes from [12] are used for evaluation.

Note that a brief overview of the available LPV MPC methods is presented in Section 3. Moreover, this Section presents the standard MPC design or the case of systems described via qLPV models, making it evident how the evolution of ρ becomes a computational issue, since: *i)* it is (*a priori*) unknown; and *ii)* it transforms the optimization procedure into a nonlinear one. It is also shown how the nonlinear optimization

can be converted into a QP by using a frozen evolution guess for the future values of ρ. General conclusions of the paper and a debate on the achieved results are drawn in Section 6.

Remark 2. Let the difference between LPV and qLPV models be cleared: the scheduling parameters in the LPV case are exogenous, unknown and must be pre-specified, while in the qLPV case they are available during the prediction horizon computations from a (possibly nonlinear) map of states and inputs $f_\rho(x(k), u(k))$. A direct consequence of these issues is that, considering MPC design, stability is typically dealt with in a robust worst-case level for the LPV case, while rendered as nonlinear programs for the qLPV scenario. This work is mainly concerned with the latter class of systems, addressing the issue with a sub-optimal QP method.

2. A Control-Oriented qLPV Corner Model for Vehicle Vertical Dynamics

In this paper, a SA suspension control system is developed for a vehicle with 4 Electro-Rheological (SA) dampers. The control system is composed of four MPC algorithms, one concerned with the performances of each vehicle corner. Indeed, a qLPV representation is provided in this Section for each corner of the vehicle. This qLPV model enables to express the nonlinear dissipativity constraints of the dampers into linear input contraints.

The control-oriented qLPV corner model is adapted from [23]; it serves for design and analysis purposes. The model involves the vertical dynamics of the vehicle, at each corner, considering the chassis dynamics (z_s) and the displacements of the wheel link (z_{us}), which are meddled by the road profile disturbances (z_r). Figure 1 shows a schematic representation of a vehicle corner. This 2DOF model is governed by the following laws:

$$\begin{cases} m_s \ddot{z}_s(t) &= -F_s(t) - F_d(t)\,, \\ m_{us} \ddot{z}_{us}(t) &= F_s(t) + F_d(t) - F_t(t)\,, \end{cases} \tag{1}$$

where $F_s(t)$, $F_d(t)$ and $F_t(t)$ represent, respectively, the force delivered by spring, by the (controlled) damper and by the tire.

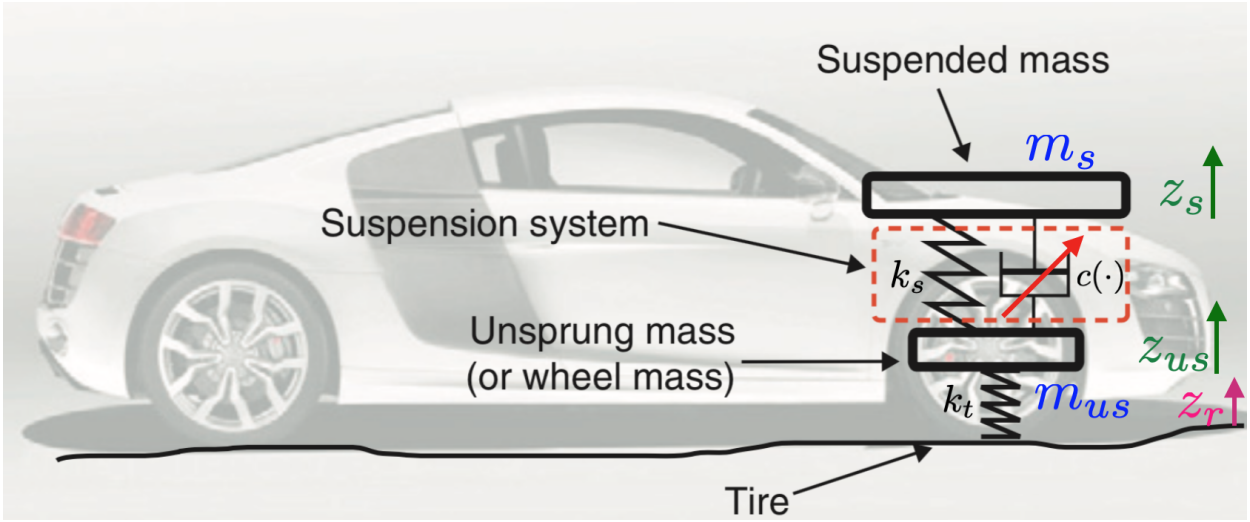

Figure 1: Vehicle Corner with Semi-Active Suspension System

These forces are further detailed: the spring force and the tire force are given as respectively proportional to the suspension deflection ($z_{def} = z_s - z_{us}$) and the wheel deflection, as follows:

$$F_s(t) = k_s z_{def}(t)\,, \tag{2}$$

$$F_t(t) = k_t \left(z_{us}(t) - z_r(t)\right)\,, \tag{3}$$

being k_s and k_t the stiffnesses of the spring and the tire, respectively. Finally, the damper force is given as follows:

$$F_d(t) \;=\; k_0 z_{def}(t) + c(\cdot)\dot{z}_{def}(t)\,, \tag{4}$$

where $c(\cdot)$ is the damping coefficient which is the control input to the SA system.

The dissipativity constraints of the SA damper are set upon $F_d(t)$, which must always lie within a feasibility set whose form is roughly illustrated in Figure 2. This dissipativity set has a hysteresis-like behaviour and can be represented by the following inequalities:

$$\overline{F}_d \;\leq\; F_d(t) \;\leq\; \overline{F}_d\,, \tag{5}$$

$$0 \;\leq\; \underline{c} \leq c(\cdot) \;\leq\; \overline{c}\,. \tag{6}$$

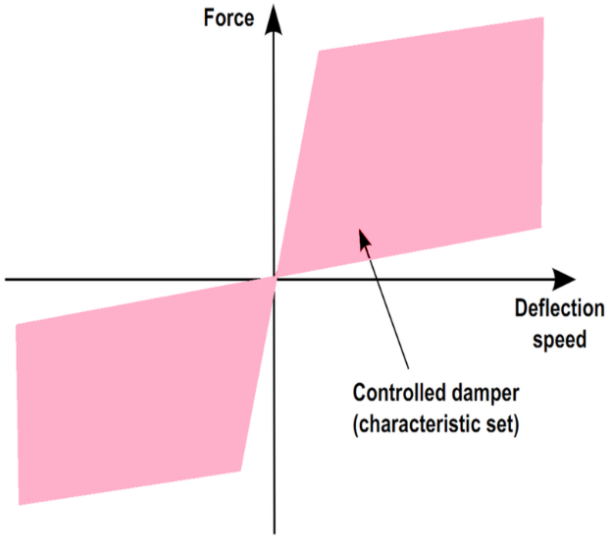

Figure 2: Semi-Active Damper: Feasible Force Region

To incorporate these inequalities into a simpler framework for design purposes, the damper force is traced as the static nonlinear map, as suggests [24], which can be used for both Magneto-rheological or Electro-rheological dampers (the technologies present in majority of SA suspensions), as follows:

$$F_d(t) \;=\; k_0 z_{def}(t) + c_0 \dot{z}_{def}(t) + \rho(t)u(t)\,, \tag{7}$$

where the scheduling parameter

$$\rho(t) \;=\; f_c \tanh\left(k_1 z_{def}(t) + c_1 \dot{z}_{def}(t)\right)$$

directly embeds the hysteresis-like behaviour of the SA damper. Parameters k_0 and c_0 denote the nominal stiffness and damping coefficient of the SA damper; moreover, $u(t)$ denotes the duty cycle of a PWM signal that regulates the voltage input which provides the electrical field upon the damper. This electric field varies the viscosity of the MR/ER fluid. In practice, it is this PWM signal $u(t)$ that acts as the control input to the suspension application. Then, the dissipativity constraints are expressed as simple input constraints: $0 < u(t) < 1$.

The suspension deflection velocity variable $\dot{z}_{def}(t)$ is bounded, due to physical limits (converted as constraints on the system variables), and can be measured or, a least, accurately estimated. Therefore, $\rho(t)$ is also known and bounded at each instant, serving as the scheduling variable for the qLPV model.

Then, the state-space representation of the qLPV consists in re-writing Eq. (1) with

$$x(t) \,=\, [z_s(t)\,,\, \dot{z}_s(t)\,,\, z_{us}(t)\,,\, \dot{z}_{us}(t)]^T$$

as system states:

$$\begin{array}{rcccccc} \dot{x}(t) & = & A_c x(t) & + & B_{c_1}(\rho(t))u(t) & + & B_{c_2} z_r(t) \\ y(t) & = & C_c x(t) & + & D_{c_1}(\rho(t))u(t) & + & D_{c_2} z_r(t) \end{array} \,. \tag{8}$$

Remark 3. In this paper, as done in many practical applications, the measured outputs of the SA suspension system are acceleration variables. These accelerations can be measured using accelerometers/inertial units, that are widely present in top-cars. These sensors are the ones used for the control of vertical dynamic behaviours. No additional sensors are needed, but the on-board ones [23].

Remark 4. As displayed in many papers from the literature [25, 23] , with experimental validation included, observers can be used, using acceleration variables, to estimate the states of SA suspensions, considering corner models. Therefore, in the sequel, it is assumed that the system states are estimated by some observer scheme and available, in real-time, to the control system.

For the reasons discussed above, the two acceleration variables from Eq. (1) are measured, i.e. $y(t) = [\ddot{z}_s(t)\,,\, \ddot{z}_{us}(t)]^T$. Therefore, the matrices in Eq. (8) are the following:

$$\begin{aligned} A_c &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{-(k_s+k_0)}{m_s} & \frac{-c_0}{m_s} & \frac{(k_s+k_0)}{m_s} & \frac{c_0}{m_s} \\ 0 & 0 & 0 & 1 \\ \frac{(k_s+k_0)}{m_{us}} & \frac{c_0}{m_{us}} & \frac{-(k_s+k_0+k_t)}{m_{us}} & \frac{-c_0}{m_{us}} \end{bmatrix}, \\ B_{c_1}(\rho(t)) &= \begin{bmatrix} 0 & \frac{-\rho}{m_s} & 0 & \frac{\rho}{m_{us}} \end{bmatrix}^T, \\ B_{c_2} &= \begin{bmatrix} 0 & 0 & 0 & \frac{k_t}{m_{us}} \end{bmatrix}^T, \\ C_c &= \begin{bmatrix} \frac{-(k_s+k_0)}{m_s} & \frac{-c_0}{m_s} & \frac{(k_s+k_0)}{m_s} & \frac{c_0}{m_s} \\ \frac{(k_s+k_0)}{m_{us}} & \frac{c_0}{m_{us}} & \frac{-(k_s+k_0+k_t)}{m_{us}} & \frac{-c_0}{m_{us}} \end{bmatrix}, \\ D_{c_1}(\rho(t)) &= \begin{bmatrix} \frac{-\rho}{m_s} & \frac{\rho}{m_{us}} \end{bmatrix}^T, \\ D_{c_2} &= \begin{bmatrix} 0 & \frac{k_t}{m_{us}} \end{bmatrix}^T. \end{aligned}$$

Remark 5. Due to physical limits of the SA suspension, constraints are also set upon the system states, considering $n_x = 4$ and $n_u = 1$:

$$x(t) \in \mathcal{X} \;\; := \;\; \{x_j \in \mathbb{R}^{n_x} \,|\, \underline{x}_j \,\leq\, x_j \,\leq\, \overline{x}_j\}\,. \tag{9}$$

The input constraints are:

$$u(t) \in \mathcal{U} \;\; := \;\; \{u \in \mathbb{R}^{n_u} \,|\, 0 \,<\, u \,\leq\, 1\}\,. \tag{10}$$

Conversely, the dissipativity constraints are:

$$F_d(t) \in \mathcal{D} \;\; := \;\; \{F_d \in \mathbb{R}^{n_u} \,|\, \underline{F}_d \,\leq\, F_d \,\leq\, \overline{F}_d\}\,, \tag{11}$$

which are always respected if $x \in \mathcal{X}$ and $u \in \mathcal{U}$.

2.1. Performances Indexes

As discussed by the references of automotive Semi-Active suspensions, the main objective of these systems is to isolate the vehicle body from the disturbances implied by the road through which the car is driven [2]. At the same time, the comfort of the onboard passengers must be enhances.

These two objectives (vehicle body isolation and passenger comfort) are physically conflicting. Stiff/high damping enhances passenger comfort, while smooth/low damping enables easier road holding, see [12]. Anyhow, since this paper consider a quarter-car vehicle corner model, it is assumed that a central controller for the braking system will be concerned with the first objective, of isolating the vehicle body from road trepidations, reducing the roll angle of the car and enhancing handling concerns. This kind of controller is seen in different papers, e.g. [26, 27].

Therefore, the corner controllers are concerned with passenger comfort performances. As proposed in [12], a simple methodology to evaluate the comfort of the onboard passengers is to analyse the car's center-of-gravity (COG) acceleration. At each corner, this analysis is reduced to the acceleration of the sprung-mass (chassis body), given by $\ddot{z}_s(t)$.

The vertical chassis acceleration $\ddot{z}_s(t)$ response to the road disturbances $z_r(t)$ can be evaluated between 0 and 20 Hz for comfort specifications, as discussed in [28]. The two criteria from [12] to evaluate the comfort of the passengers, within these frequencial bounds, are given by:

1. Comfort performance index in the time-domain:

$$J^t_{comfort} \quad = \quad \int_0^\tau \ddot{z}_s^2(t)dt\,, \tag{12}$$

   where τ represents a given time period.

2. Comfort performance index in the frequency-domain:

$$J^f_{comfort} \quad = \quad \frac{\mathcal{C}(f\{\ddot{z}_s^{controlled}\},0,20)}{\mathcal{C}(f\{\ddot{z}_s^{nominal}\},0,20)}\,, \tag{13}$$

   where $\ddot{z}_s^{nominal}$ and $\ddot{z}_s^{controlled}$ refer to the car COG acceleration in a nominal (passive, uncontrolled) situation and when under a control scheme; moreover, $f\{\cdot\}$ represents the frequency response of the signal of interest, and $\mathcal{C} : \mathbb{R}\times\mathbb{R}\times\mathbb{R}\to\mathbb{R}$, denoted $\mathcal{C}(x,\underline{h},\overline{h}) = \int_{\underline{h}}^{\overline{h}} |x(\mu)|^2 d\mu$, where $\overline{h}$ and $\underline{h}$ represent the frequency interval limits of interest.

2.2. Vehicle Testbed

For realistic validation purposes, when running numerical simulations, a full nonlinear vehicle model from [17] is used, which also embeds noises and couplings. This model has been validated and retrieved from a real mechatronic testbed.

This experimental platform from which the validation model has been constructed is the *INOVE Soben-Car*, a (1/5) reduced-size vehicle, show in Figure 3[1]. The SA dampers in this testbed are Electro-Rheological (ER), which means that the PWM signal $u(t)$ controls an electric field which varies the viscosity of an ER fluid inside the damper chamber, increasing or decreasing the delivered force.

Table 1 presents the parameters from the quarter-car model in Eq (8) with respect to this testbed.

The real nonlinear behaviour of the ER SA dampers is shown in Figure 4 through Force vs. Deflection Speed diagrams, showing real data at the left side and fitted data at the right side, considering the use of Eq. (7) to compute the damper force. Clearly, the dissipativity constraints are respected if $u(t)$ is bounded to $\mathcal{U}$ and Eq. (7) is used to compute the damper force.

Remark 6. The *INOVE Soben-Car* interprets control laws using a fixed sampling frequency of $f_s = 200$ Hz. This condition is quite restrictive in terms of implementation purposes, since the controller must always compute the control signal within 5 ms. Note that this sampling rate is realistic and adequate for actual top-cars [3].

[1] Refer to full details in *www.gipsa-lab.fr/projet/inove*.

Figure 3: *INOVE Soben-Car* Mechatronic Testbed

Table 1: Vehicle Model Parameters

Parameter	Description	Value	Unit
m_s	Sprung mass	2.27	kg
m_{us}	Unsprung mass	0.32	kg
k_s	Spring stiffness	1396	N/m
k_t	Tire stiffness	12270	N/m
k_0	Passive damper stiffness	170.4	N/m
k_1	Hysteresis displacement coefficient	218.16	N/m
c_0	Viscous damping coefficient	68.83	N.s/m
c_1	Hysteresis velocity coefficient	21	N.s/m
f_c	Dynamic yield force of the ER fluid	28.07	N

2.3. Discrete-time qLPV Model

Since the SA suspension experimental testbed is evaluated with a fixed sampling frequency of $f_s = 200\,\text{Hz}$, the controller must be synthesized with respect to a discrete-time model that embeds the $T_s = 5\,\text{ms}$ sampling period. Therefore, considering an Euler discretization method, the model is given by:

$$\begin{aligned} x(k+1) &= Ax(k) + B_1(\rho(k))u(k) + B_2 w(k)\,, \\ y(k) &= Cx(k) + D_1(\rho(k))u(k) + D_2 w(k)\,, \\ \rho(k) &= f_c \tanh\left(A_\rho x(k)\right)\,, \end{aligned} \tag{14}$$

where[2] $w(k) = z_r(k)$, $A_\rho = \begin{bmatrix} k_1 & c_1 & -k_1 & -c_1 \end{bmatrix}$, $A = \mathbb{I}_{n_x} + T_s A_c$, $B_1(\rho(k)) = T_s B_{c_1}(\rho(k))$ $B_2 = T_s B_{c_2}$, $C = C_c$, $D_1 = D_{c_1}$ and $D_2 = D_{c_2}$.

3. MPC Design for Systems with qLPV Models

The objective of this paper is to propose a control algorithm to enhance the comfort performances of the onboard passengers. This algorithm must be realizable and run within the 5 ms sampling period of the

[2]In this paper, $\mathbb{I}_j$ denotes the identity matrix of dimension j.

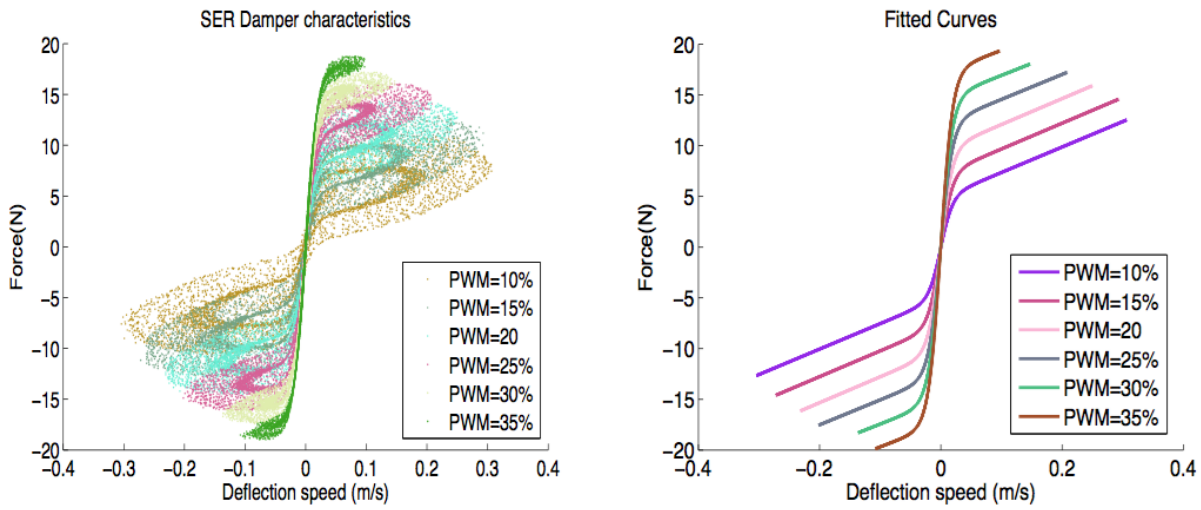

Figure 4: Force-Speed Characteristics - ER Semi-Active Dampers

considered SA suspension system. Moreover, the control policy must ensure input and state contraints are respected.

As discussed in the Introduction, MPC is a very elegant option for constrained processes and it is the method used in this paper. In this Section, a brief review on the available LPV MPC works is presented, highlighting why there is an intrinsic computational necessity increase with the size of the prediction horizon. The standard LPV MPC method is presented and the available approachs that lead to sub-optimality but overcome the numerical burden are presented.

3.1. Literature Overview

LPV MPC works are investigated since the beginning of the 00's; the majority of the methods consider that the scheduling parameter is an uncertain variable along the prediction horizon, solving the MPC problem robustly with respect to it. Some key papers are recalled:

- Explicit methods with stability and optimality guarantees were investigated in [29]. The downside is that, since the future values of the scheduling parameters are unknown, the algorithm ensures the constraints are satisfied for all possible system trajectories, which leads to conservative performances and (numerical-wise) high-demanding QPs.
- Dynamic output feedback algorithms have also been developed [30, 31]. Some of these papers use an LPV Input/Output representation. Anyhow, they are all robust towards ρ, solving worst-case (usually referred to as "min./max.") optimization procedures and resulting in conservative results.
- Other papers [20, 32] present a major advance by considering bounded rates of the scheduling parameters. This simple constraint simplifies the optimization procedure, treating the evolution of the scheduling parameters offline, via Linear Matrix Inequalities (LMIs) and ellipsoidal constraints. Their main downside is that the offline procedures are not necessarily simple to perform.
- Papers [33, 34] also consider bounded rates of the scheduling parameters, but the problem is formulated robustly with the use of "tubes" to deal with the uncertainty introduced by the scheduling parameters along the horizon. Recursive feasibility and stabilizability are demonstrated with respect to the tube formulation.
- That are also another group of papers that must be mentioned, those that parametrize the control inputs in finite amount of possible discrete values, solving a search algorithm to find the smallest cost instead of the actual optimization procedure [35, 36]

With regard to the previous methods, robust procedures or constraints are embedded to the MPC problem. To run these algorithms within 5 ms is, today, impossible. Therefore, some sub-optimal MPC works are recalled:

1. Both [37] and [38] use a frozen scheduling parameter trajectory guess that iterates according to measurements, and transform the nonlinear optimization problem into a linear one. The issue that resides with such methods is that the results may be sub-optimal and that the system trajectory might not be inside the region of attraction of the MPC, resulting in infeasibility.

2. In previous works, the Authors have also developed a QP version of the LPV MPC algorithm using a frozen parameter trajectory guess in [15, 39] and a Least-Squares (LS) identified parameter model in [22]. In the latter, a fictional set-point variable is used to enlarge the domain of attraction of the closed-loop system.

In this paper, the method developed in [22] is further extended and oriented towards the case of SA suspensions. In this paper, a Lyapunov-decreasing terminal cost and a the use of control invariant sets are added to the method to ensure that recursive feasibility is maintained for any starting condition within the constraints set $\mathcal{X}$, which addresses the issue of possible infeasibility from the previous work.

3.2. Why Embed Sub-Optimality?

Now, it is demonstrated why sub-optimal LPV MPC are needed for real-time implementation purposes. The regular MPC procedure is recalled:

MPC policies are essentially derived by solving an optimization procedure that takes into account constraints on the states, outputs and control actions. With some bland assumptions, it is possible to guarantee closed-loop asymptotic stability for the LTI case [40]. Predictive control is widely used to achieve reference tracking and disturbance rejection performances in process control [9], by solving:

Problem 1. *Standard MPC Procedure*

$$\begin{aligned} \min_{U} J_{N_p} &= \min_{U} \sum_{i=1}^{N_p} \ell\left(x,u\right) + V(x(k+N_p|k)) && (15)\\ s.t. \quad & x(k+i+1) = f(x(k+i),u(k+i),w(k+i))\,, && (16)\\ & u(k+i-1|k) \in \mathcal{U}\,, && (17)\\ & x(k+i|k) \in \mathcal{X}\,, && (18) \end{aligned}$$

where $U = col\{u(k|k),\ldots,u(k+N_p-1|k)\}$ *is the sequence of actions inside the prediction horizon* N_p[3]*. The MPC optimization cost* J_{N_p} *is comprised of the sum of a stage cost* $\ell(\cdot)$ *along the horizon and may also include a terminal stage value* $V(x(k+N_p|k))$*.* J_{N_p} *is usually Lyapunov-decreasing to ensure recursive feasibility. It is implied:* $x \in \mathbb{R}^{n_x}$ *and* $u \in \mathbb{R}^{n_u}$*, with* $\mathcal{X}$ *and* $\mathcal{U}$ *as the set contraints that define their respective feasible values (operation). The number of states is* n_x *and the number of control inputs is* n_u*.*

Remark 7. The load disturbances $w(k) \in \mathbb{R}^{n_w}$ are assumed to be (partially) known for the future N_p steps. This is reasonable for the case of the automotive suspensions, as it will be further in the sequel. The number of disturbance variables is n_w.

When the system model is LTI, function $f(\cdot)$ is inherently linear and the optimization problem becomes a regular constrained QP, which is easily tackled by standard solvers. For the studied case, since the system model is qLPV, the optimization becomes nonlinear.

[3]Notation $(k+i|k)$ stands for a predictions for instant $k+i$, from the viewpoint of instant k. Consider $\overline{U}$ and $\underline{U}$, respectively, as the maximal and minimal values this vector may assume.

Definition 3.1. Nonlinear Programming Problem
Consider an arbitrary real-valued nonlinear function $f_c(x_c)$. A nonlinear programming problem finds the vector x_c that minimizes $f_c(x_c)$ subject to $g_i(x_c) \leq 0$, $h_j(x_c) = 0$ and $x_c \in \mathcal{X}_c$, where g_i and h_j are also nonlinear.

Definition 3.2. Quadratic Programming Problem
A Quadratic Programming Problem (or simply Quadratic Problem, QP) is a linearly constrained mathematical optimization problem of a quadratic function. A QP is a particular type of nonlinear programming problems. The quadratic function may be defined with respect to several variables, all of which may be subject to linear contraints. Considering a $c \in \mathbb{R}^{n_c}$ vector, a symmetric matrix $Q_c \in \mathbb{R}^{n_c \times n_c}$, a real matrix $A_{ineq} \in \mathbb{R}^{m_c \times n_c}$, a real matrix $A_{eq} \in \mathbb{R}^{m_c \times n_c}$, a vector $b_{ineq} \in \mathbb{R}^{m_c}$ and another vector $b_{eq} \in \mathbb{R}^{m_c}$, the goal of a QP is to determine the vector $x_c \in \mathbb{R}^{n_c}$ that minimizes a regular quadratic function of form $\frac{1}{2}\left(x_c^T Q_x x_c + c^T x\right)$ subject to constraints $A_{ineq} x_c \leq b_{ineq}$ and $A_{eq} x_c = b_{eq}$. The solution x_c to this kind of problem is found by many solvers seen in the literature, based on Interior Point algorithms, quadratic search, etc.

Consider a generic discrete-time qLPV model $x(k+1) = A(\rho(k))x(k) + B(\rho(k))u(k)$, being the scheduling parameters endogenous, as gives $\rho(k) = f_\rho(x(k), u(k))$. These qLPV scheduling parameters are also possibly expressed through a dynamic recursive equation, i.e. $\rho(k) = f_\rho^m(\rho(k-1), \rho(k-2), \dots)$. The vector of future scheduling policies, from instant k, is given by:

$$\Gamma_k \quad = \quad \text{col}\{\rho(k+1)\,,\, \rho(k+2)\,,\, \dots\,,\, \rho(k+N_p-1)\}\,. \tag{19}$$

Then, departing from an arbitrarily feasible initial condition $x(k) = x_k$, Problem 1 has to internally elaborate constraint (16), which exhibits nonlinearities from the second iteration onward:

$$\begin{aligned} x(k+2|k) &= A(\rho(k+1))A(\rho(k))x_k \\ +A(\rho(k+1))B(\rho(k))u(k|k) &+ B(\rho(k+1))u(k+1|k). \end{aligned} \tag{20}$$

and so forth, up to the N_p-th prediction. This results, therefore, in non-QP version of Problem 1.

3.3. Frozen Scheduling Guess

Nonetheless, notice that these model-based predictions as in Eq. (20) would also be linear if ρ was known for every iteration inside the N_p horizon. To say one has knowledge of the complete future scheduling vector Γ_k is obviously false, since only $\rho(k)$ is known. But, if a frozen guess was to be used, as done in [15], substituting Γ_k by $\hat{\Gamma}_k$, Problem 1 would be translated into a QP version, but with sub-optimal results due to model-process mismatches, since the linear model use for predictions would be a frozen version of the qLPV process. Through the sequel, it is considered that the following Assumptions holds:

Assumption 3.3. *Some algorithm provides a guess for the evolution of the scheduling parameters along the horizon. This guess is denoted*

$$\hat{\Gamma}_k = col\{\hat{\rho}(k), \hat{\rho}(k+1), \dots\ \hat{\rho}(k+N_p-1)\}\,.$$

Remark 8. It has been demonstrated in [22] that a recursive LS algorithm can be used to "predict" the scheduling parameters of a SA suspension system reasonably well.

Assumption 3.4. *As done in [20, 33], it holds that the ρ has a bounded variation rate, this is:*

$$\rho \;\in\; \mathcal{P} := [\underline{\rho}\,,\, \overline{\rho}] \;\; and \;\; \dot{\rho} \;\in\; \delta\mathcal{P} := [\underline{\delta\rho}\,,\, \overline{\delta\rho}]\,. \tag{21}$$

Remark 9. In fact, as of Eq. (7), this is true, since $f_c \tanh(A_\rho x(k))$ is always bounded due to the fact that z_{def} is bounded. The derivative / difference is also inherently bounded.

From the viewpoint of k, the minimal and maximal prediction evolution guesses would be:

$$\hat{\Gamma}_k^{min} = \text{col}\{\rho(k) + \underline{\delta\rho}, \ldots, \rho(k) + (N_p - 1)\underline{\delta\rho}\}, \tag{22}$$

$$\hat{\Gamma}_k^{max} = \text{col}\{\rho(k) + \overline{\delta\rho}, \ldots, \rho(k) + (N_p - 1)\overline{\delta\rho}\}. \tag{23}$$

These bounds are taken into account by the LS algorithm as saturation limits. It is directly implied that $\hat{\Gamma}_k^{min} \subset \hat{\Gamma}_k \subset \hat{\Gamma}_k^{max}$

Let the generic discrete-time qLPV model be extended for the next j steps ahead of k[4]:

$$\begin{aligned} x(k+j) &= \overbrace{\coprod_{n=0}^{j-1} A(\rho(k+n))}^{A^j(\Gamma_k)} x_k \\ &+ \underbrace{\sum_{m=1-k}^{j-k} \left(\left(\coprod_{n=k+1}^{m-1} A(\rho(n)) \right) B(\rho(j-m))u(j-m) \right)}_{B^j(\Gamma_k)U}. \end{aligned} \tag{24}$$

Then, since the scheduling prediction is always limited to the bounds given by $\hat{\Gamma}_k^{min}$ and $\hat{\Gamma}_k^{max}$, the model-process mismatches that arise by using a frozen model are also bounded. Take μ_j as these mismatches, due to the differences between the real state $x(k+j)$, which is a function of Γ_k, and the predicted state $\hat{x}(k+j)$, which is a function of the scheduling guess $\hat{\Gamma}_k$, as gives[5]:

$$\begin{aligned} \mu_j &= \hat{x}(k+j) - x(k+j) \\ &= \left(A^j(\hat{\Gamma}_k - \Gamma_k) \right) x_k \\ &+ \left(B^j(\hat{\Gamma}_k - \Gamma_k) \right) U, \end{aligned} \tag{25}$$

which leads to:

$$\begin{aligned} \underline{\mu}_j &\leq \mu_j \leq \overline{\mu}_j, \\ \underline{\mu}_j &= A^j(\hat{\Gamma}_k - \hat{\Gamma}_k^{min})x_k + B^j(\hat{\Gamma}_k - \hat{\Gamma}_k^{min})\underline{U}, \\ \overline{\mu}_j &= A^j(\hat{\Gamma}_k - \hat{\Gamma}_k^{max})x_k + B^j(\hat{\Gamma}_k - \hat{\Gamma}_k^{max})\overline{U}, \end{aligned} \tag{26}$$

where $\overline{U}$ and $\underline{U}$ represent, respectively, a sequence maximal and minimal control inputs.

Notice how these mismatches are bounded by the saturation conditions implied by the input constraints and a sequence of minimal or maximal scheduling parameter variations. Also, remark that μ_j increases along with the prediction horizon N_p, departing from $\mu_0 = 0$. This issue is rather interesting, since the MPC procedure will re-calculate the control sequences and predictions at each sampling instant, meaning that if the algorithm is recursively feasible, the effects of the model-process mismatches upon the controlled outputs will relieved over time.

4. Set-Constrained Recursively Feasible qLPV MPC Procedure

Based on the discussion of the sub-optimal qLPV MPC design based on (bounded) frozen scheduling parameter evolution guesses $\hat{\Gamma}_k$ provided in the previous Section and the considered SA suspension application detailed in Sec. 2, this Section develops the proposed qLPV MPC algorithm for passenger comfort enhancement.

[4] $\coprod$ stands for the left-side matrix product.

[5] Matrices $A^j(\cdot)$ and $B_1^j(\cdot)$ are affine in Γ_k. Therefore, $A^j(\hat{\Gamma}_k) - A^j(\Gamma_k) = A^j(\hat{\Gamma}_k - \Gamma_k)$.

A recursive saturated LS will provide, at each sampling instant k, an approximate guess for the evolution the scheduling parameters along the horizon. This frozen prediction $\hat{\Gamma}_k$ is passed to the MPC, that can be computed using a single QP.

Then, the design of such predictive controller must integrate some tools to guarantee reference-tracking despite the model uncertainties μ_j (that grow along the horizon). Essentially, what is done in this paper is to guarantee an enlarged domain of attraction of the system under closed-loop control, so that any initial condition that lies inside this domain can be steered as envisioned. Due to the horizon-increasing model-process mismatches μ_j, it must also guaranteed that the system is not driven out of an stability region, which can never be allowed. The recursive feasibility property must be verified to ensure that, although leading in sub-optimal results, the algorithm will stabilize and converge.

The tools used to address this matter are adapted to the qLPV case, respectively, from two prominent works [41, 40]: 1) the use of pseudo-reference tracking allows an enlargement of the domain of attraction of MPC policies, finding more options of stable closed-loop equilibrium points and 2) the usage of a terminal set that contracts along the horizon and a Lyapunov decreasing terminal stage cost, which together guarantee that, even with bounded uncertainties, the controlled system is able to meet performance goals whilst stability and feasibility are maintained.

Remark 10. These tools have previously been applied to the case of nonlinear systems; in [42], robust assessments are presented in terms of formal guarantees of recursitivy and feasibility.

4.1. Pseudo-Reference Tracking

The "MPC for Tracking" method from [41] is considered in the sequel: this control design is used to ensure that the controller can asymptotically steer the process a steady-state reference x_s in an admissible manner from any feasible initial state x_0. The approach consists basically in adapting the standard MPC cost function (i.e. weighting the quadratic difference between output and reference). The use of the "MPC for Tracking" design for qLPV models has been previously done in [43], where the scheduling trajectory Γ_k is taken as frozen, based on the known value $\rho(k)$ (no guess is performed).

Remark 11. The "MPC for Tracking" design includes an artificial reference x_a and sets the system to track it, while, at the same time, makes it track the actual reference x_s, which altogether ensures an enlarged domain of attraction. The target operation point $p_t = (x_s, u_s)$ is an admissible steady-state, which is possible if Eq. (14) is LPV-stabilizable (refer to the definition in [44]). Anyhow, this tool still does not guarantee the convergence of the qLPV system to the target point p_t because the model uncertainty μ_j is obviously non-null and horizon-increasing, as previously discussed. Therefore, a contractive terminal set constraint must also be used.

Assumption 4.1. *Consider: (1) $Q \in \mathbb{R}^{n_x \times n_x}$ and $R \in \mathbb{R}^{n_u \times n_u}$ as positive definite matrices; and (2) $\kappa \in \mathbb{R}^{n_u \times n_x}$ as an arbitrary stabilizing state-feedback control gain. For these matrices, it is implied that, for the generic discrete-time qLPV model,*

$$(A(\rho(k)) + B(\rho(k))\kappa)$$

is Schur. Then, there exists another positive definite matrix $P \in \mathbb{R}^{n_x \times n_x}$ such that

$$(A(\rho_k) + B(\rho_k)\kappa)^T P (A(\rho_k) + B(\rho_k)\kappa) - P = -(Q + \kappa^T R \kappa)$$

holds for all $\rho_k \in \Gamma_k$.

Then, as long as the previous Assumption holds, the MPC Problem is formulated with the following adjusted optimization cost:

$$J_{N_p} = V(\cdot) \tag{27}$$
$$+ \sum_{i=1}^{N_p} \underbrace{\left(||x(k+i|k) - x_a||_Q^2 + ||u(k+i-1|k) - u_s||_R^2 \right)}_{\text{Main Cost } \ell(\cdot)},$$

where the terminal stage cost is given by:

$$V(\cdot) \;=\; V_o^{x_a} + \overbrace{||x(k+N_p|k) - x_a||_P^2}^{V_o^x}\,, \tag{28}$$

with $x_a \in \mathcal{X}$ and $u_s \in \mathcal{U}$ being artificial variables for the set-point and for the control signal. The quadratic offset function $V_o^{x_a}$ penalizes the deviation between the artificial reference x_a and the target operation point x_s (actual set-point). The inclusion of a suitable penalization of the terminal state (terminal cost term V_o^x) can lead to asymptotic stability with good performances, as demonstrated in [42]. As pointed out by the latter, the offset cost $V_o^{x_s}$ must be convex and respect:

$$\beta_1||x_a - x_s||_1 \;\leq\; V_o^{x_a}(x_a, x_s) \;\leq\; \beta_2||x_a - x_s||_1\,, \tag{29}$$

where β_1, β_2 are positive real constants. The artificial tracking point is given by $p_s = (x_a, u_s)$.

Proposition 4.2. *If the stage cost weights Q and R are adequately chosen, is it possible to use an MPC algorithm, formulated with a quadratic stage cost of the form in Eq. (27), to optimize and enhance the comfort of onboard passengers, with respect to nominal (uncontrolled) situations.*

Proof. Indeed, MPC as a SA suspension control system can act to ensure a better comfort of the onboard passengers. The MPC will, at each sampling instant, act to minimize the primary control objective ℓ along the control horizon.

To do so, the time-domain index given in Eq. (12) is embedded to ℓ through Q and R. This index is re-written with respect to the discrete-time qLPV model in Eq. (14):

$$J_{comfort}^{N_p} = \sum_{j=0}^{N_p} \ddot{z}_s^2(k+j|k)T_s\,.$$

This finite sum approximates the integral in Eq. (12).

Then, assuming that (x_a, u_s) will converge to $(x_s, u_s) = (0, 0)$ (real set-point and respective control signal), it follows that[6]:

$$\begin{aligned}\sum_{j=0}^{N_p} \ddot{z}_s^2(k+j|k)T_s \;&=\; \sum_{j=0}^{N_p} \ell(\cdot) \\ &=\; \sum_{j=0}^{N_p} ||x(k+j|k)||_Q^2 + ||u(k+j-1|k)||_R^2\,.\end{aligned} \tag{30}$$

From Eq. (14), it follows that:

$$\begin{aligned}\ddot{z}_s^2(k+j|k)T_s = (C\{1,:\}x(k+j-1|k) \\ + D_1\{1,:\}(\rho(k+j-1|k))u(k+j-1|k))^2\,T_s\,.\end{aligned} \tag{31}$$

Thus, if Q and R are chosen, respectively, as:

$$Q \;=\; \big(C\{1,:\}\big)^T T_s \big(C\{1,:\}\big)\,, \tag{32}$$

$$R \;=\; \big(D\{1,:\}(\rho_k)\big)^T T_s \big(D\{1,:\}(\rho_k)\big)\,, \tag{33}$$

where $\rho(k+j-1|k)$ is replaced by $\rho_k = \rho(k)$, for simplicity, the MPC policy with main cost $\ell(\cdot)$ will act to minimize $\ddot{z}_s^2(t)$ and enhance comfort performances. □

[6] $M\{l,:\}$ denotes the vector formed by the l^{th} line of matrix M; moreover, w is neglected from the sequence, since the control law has no measures over it (it cannot be minimized, since it is an external variable).

Remark 12. The objective of the inclusion of the artificial target point p_s works as follows. Consider that the system evolves as predicted (with $\mu_j = 0$) and that the actual target point $p_t = (x_s, u_s) = (0, 0)$ is an admissible point contained inside the tracking set $\mathcal{T} := \mathcal{X} \times \mathcal{U}$. Then, p_t is an asymptotically stable point in closed-loop, since the MPC will ensure convergence to it. Otherwise, the achieved closed-loop equilibrium is given by $p_s^\star = (x_a^\star, u_s^\star) = \arg\min_{x_a} V_o^{x_a}(x_a, x_s)$. Moreover, the inclusion of the artificial reference ensures recursively feasibility properties of the algorithm and that the achieved closed-loop equilibrium $x_a^\star$ is the closest possible to x_s when it is feasible.

4.2. Contractive Sets

Consider the following definitions presented in [45]:

Definition 4.3. 1-Step Robust Set:
The 1-Step set of Υ, $\mathcal{Q}_1\{\Upsilon\}$, stands for the set of states which can be steered in one sampling instant into the target set Υ by an admissible control action, despite $\mu_j \neq 0$.

Definition 4.4. Robust Controlled Positively Invariant Set:
A set $\Upsilon \subset \mathbb{R}^{n_x}$ is said to be *control invariant* for the qLPV system in Eq. (14) if, for all possible $x_k \in \Upsilon$, there exists an admissible input $u = \kappa(x) \in \mathcal{U}$ so that $x(k+1)$ lies inside Υ despite $\mu_j \neq 0$. This is valid iff $\Upsilon \subseteq \mathcal{Q}_1\{\Upsilon\}$.

Definition 4.5. N_r-Step Control Invariant Sequence:
A sequence of N_r steps $\mathcal{S}_{N_r} := \{\Upsilon_j\}$ is the set sequence through which x can be steered through, leaping from one set Υ_j to the following Υ_{j-1}, with feasible control actions, until finally reaching the target invariant set Υ.

MPC design coupled to the use of control set sequences is used to make sure the algorithm guarantees asymptotic convergence despite model-process uncertainties μ_j, which is the case of this work[7]. To compute a reachable set sequences for qLPV models, the bounds on the variation rate of the scheduling parameters $\dot{\rho}$ must be taken into account: as gives Eq. (26), from the viewpoint of instant k, $x(k+1)$, for whichever $\rho(k+1) \in \mathcal{P}$, is, at most, equal to $x^\star(k+1) = A(\rho^\star)x_k + B_1(\rho^\star)u(k)$ where $\rho^\star = \rho(k) + \overline{d\rho}$ or $\rho^\star = \rho(k) - \underline{d\rho}$. Therefore, Υ must be computed from x_k, abiding to:

$$\Upsilon^{\text{Max}} \subseteq \Upsilon \subseteq \Upsilon^{\text{Min}} \text{ and } p_t \in \Upsilon\,, \tag{34}$$

where Υ^{Max} and Υ^{Max} are the sets achieved with admissible control laws and, respectively, a sequence of N_r maximal and minimal scheduling parameter variations $\hat{\Gamma}_k^{max}$ and $\hat{\Gamma}_k^{min}$.

Then, for each iteration k, a sequence of reachable sets is computed as the intersection of the min./max. wider sets, found with $\hat{\Gamma}_k^{max}$ and $\hat{\Gamma}_k^{min}$, respectively. This is, for $j = \max\{N_r - k, 0\}, \ldots, 0$:

$$\mathcal{S}_{N_r} := \left\{\text{col}\{\Upsilon_j\} \,|\, \Upsilon_j = (\Omega_j^{max} \cap \Omega_j^{min})\right\}\,. \tag{35}$$

$$\Omega_j^{max} = \left(A^{j+1}(\hat{\Gamma}_k^{max})x_k + B^{j+1}(\hat{\Gamma}_k^{max})U\right)\,, \tag{36}$$

$$\Omega_j^{min} = \left(A^{j+1}(\hat{\Gamma}_k^{min})x_k + B^{j+1}(\hat{\Gamma}_k^{min})U\right)\,. \tag{37}$$

With these definitions in mind, to guarantee that within N_r steps from the initial instant k_0 the controlled qLPV system (14) reaches a terminal control invariant set Υ_0 in Eq. (34) which contains the target equilibrium p_t, the following contractive terminal set constraint is included to the design:

$$x(k_0 + N_r) \in \Upsilon_j\,,\; j = \max\{N_r - k\,,\, 0\}\,, \tag{38}$$

assuming $\mathcal{S}_{N_r}$ is available from Eq. (35). Note that this terminal set Υ_j is equal to the larger Υ_{N_r} at the initial instant k_0 being shrinked subsequently until, at $k_0 + N_r$, it becomes the smallest set Υ_0.

[7] For this development, $w(k+j)$ is suppressed, since it is a known variable throughout the N_p horizon.

Remark 13. This constraint makes the MPC method intrinsically time-varying, since, at least for the first N_r samples, the sets are contracting. In this paper, it is considered that $N_r \geq N_p$.

When the above terminal constraint is coupled to the MPC optimization procedure, there is indeed an enlargement of its domain of attraction, giving further holds on stability and feasibility, which are needed due to model-plant differences μ_j. The sequence of control invariant sets makes sure the terminal constraints contracts and the states converge to the desired target p_t (or as closely as possible, due to the pseudo-reference technique). Therefore, the main idea of the design method used in this paper is to use a constrained, finite horizon MPC to regulate the SA suspension system described by the discrete-time qLPV model in Eq. (14), from any admissible initial condition $x_0 \in \mathcal{X}$ to the target goal p_t using a fixed $\hat{\Gamma}_k$ by minimizing the adjusted objective function in Eq. (27) with $x(0, \rho(k)) \equiv x_0$ subject to the original constraints in Problem 1 coupled to the contractive constraint in Eq. (38).

4.3. Quadratic Stabilizability and Recursive Feasibility

Since the terminal stage cost, terminal ingredient and primary optimization have been defined, it follows to verify if the proposed controller ensures quadratic stabilizability and recursive feasibility, which are envisioned properties of the algorithm.

Definition 4.6. Recursive Feasibility of MPC Algorithms[40]
Consider that the terminal set constraint on x exists such that $\Upsilon_0 \subset \mathcal{X}$, with $\mathcal{X}$ closed, convex and compact and that the origin lies within the interior of Ω, for Ω being the largest admissible set[8] such that $\Omega \subseteq \mathcal{X}$. Then, essentially, the following axioms verify if the MPC terminal cost function is Lyapunov-decreasing along the control horion:

- *A*1) $\ell(\cdot) \geq \beta_1(||x||), \forall x \in \Omega, \forall u \in \mathcal{U}, \forall \rho \in \mathcal{P}$, for $\beta_1(\cdot)$ of class $\mathcal{K}$[9].
- *A*2) $V(\cdot) \leq \beta_2(||x||), \forall x \in \Upsilon_0, \forall \rho \in \mathcal{P}$, for $\beta_2(\cdot)$ of class $\mathcal{K}$[10].
- *A*3) $V(x(k+1)) - V(x(k) + \ell(x(k), u(k)) \leq 0, \forall x \in \Upsilon_0, \forall u \in \mathcal{U}, \forall \rho \in \mathcal{P}, \forall k$[11].

If these three axioms hold. the MPC will be recursively feasible for any starting condition $x_0 \in \mathcal{X}$.

The terminal set Υ_0 is given by $\{x \in \mathbb{R}^{n_x} \,|\, V(x) \leq \alpha_s\}$ such that $\Upsilon_0 \subset \Omega$. Moreover, α_s is some scalar such that for all $x \in \Omega$, $f_m(x(k), u(k)) \in \Upsilon_0$.

Definition 4.7. Quadratic Stabilizability inside the Feasibility Region
The considered qLPV system given in Eq. (14) is said to be stabilizable if there exists a positive definite map $V : x(k) \to x(k)^T P x(k)$, where $P = P^T \succ 0$ and $P \in \mathbb{R}^{n_x \times n_x}$ and a state-feedback control policy of fashion $u(k) = \kappa x(k)$, with $\kappa \in \mathbb{R}^{n_u \times n_x}$, such that the following inequality:

$$V\left(A(\rho(k))x(k) + B(\rho(k))\kappa x(k)\right) - V\left(x(k)\right) \leq -x(k)^T \left(Q + \kappa^T R \kappa\right) x(k) \tag{39}$$

holds for all $x \in \mathcal{X}$ and $\rho \in \mathcal{P}$, with $Q = Q^T \succ 0$ and $R = R^T \succ 0$. Then, the origin is globally exponentially stable within the feasibility region $\mathcal{X}$ is globally exponentially stable for $x(k+1) \,\forall\, \rho \in \mathcal{P}$ and any initial condition $x_0 \in \mathcal{X}$.

Remark 14. The above notion of quadratic stabilizability "inside the feasibility region" is slightly smoother than the notion of pure quadratic stabilizability, which would require the verification of the inequality for all $x \in \mathbb{R}^{n_x}$. The notion of the feasibility regionalization implies that only $\mathcal{X}$ must be considered, which may be *a priori* a smaller proper $\mathcal{C}$ set than $\mathbb{R}^{n_x}$ (inequality must hold for all $x \in \mathcal{X}$ instead of $\mathbb{R}^{n_x}$).

[8] In fact, this set must also be positively control invariant, such that Ω is the one-step-step from Υ_0.
[9] This axiom implies that ℓ is function-wise lower bounded.
[10] This axiom implies that V is function-wise upper bounded.
[11] This axiom implies that V decreasing along the horizon.

Definition 4.8. $\mathcal{K}$ refers to the class of positive and strictly increasing scalar functions that pass through the origin. A given function $f : \mathbb{R} \to \mathbb{R}$ is of class $\mathcal{K}$ if $f(0) = 0$ and $\lim_{\xi \to +\infty} f(\xi) \to +\infty$.

Assumption 4.9. *(i) There exists a $\mathcal{K}$ function $\beta_1(||x||)$ that lower bounds the horizon cost $\ell(x)$; and (ii) there exists another $\mathcal{K}$ function $\beta_2(||x||)$ that upper bounds the terminal cost $V(x(k+N_p))$.*

Proposition 4.10. *Quadratic Stabilizability inside the Feasibility Region*
The considered qLPV system, when regulated by the MPC policy conducted through $u(k) = \kappa x(k)$ is quadratic stabilizable.

Remark 15. The MPC policy yields a quadratic stabilizability property if the third recursive feasibility axiom verifies, with $V = x(k)^T P x(k)$, as demonstrated in the sequel. Moreover, full conditions for the satisfaction of $A3$ are demonstrated in [46], where LMI-solvable remedy to ensure this axiom is satisfied is proposed, concerning the case of scheduling parameter assumed as if they were held constant over N_p.

Proposition 4.11. *Recursive Feasibility*
The proposed algorithm is recursively feasible inside the feasibility set for any starting condition $x(k_0) = x_0 \in \mathcal{X}$.

Proof. Since only the measured (state-feedback from an observer) variable $x(k)$ and scheduling sequence $\hat{\Gamma}_k$ are used to solve Problem 1, at the following discrete-time instants $k > k_0 = 0$, the online optimization is not related to any disturbance variables and, thus, the recursive feasibility property can be analysed albeit disturbances (feedforward compensation is neglected in this proof).

Assume that Problem 1 is feasible for an initial condition x_0, based on a Γ_{k_0} scheduling sequence, resulting in $U^\star_{k_0}$ as the optimal sequence of control action which solves Eq. (15) at instant k_0; this optimal control policy leads to a minimal state sequence with respect to the cost function $J_{N_p}(\cdot)$. It holds that $x^\star(k_0+i) \in \Upsilon_j$ and $u(k_0+i-1|k_0) \in \mathcal{U} \forall i \in \mathbb{N}_{[1,N_p]}$. Moreover, it is implied that $x^\star(k_0+N_p) \in \Upsilon_0$, which is a positive invariant set for the qLPV model.

The MPC control policy $u(k_0) = u^\star(k_0|k_0)$ is applied to the process and steers the system from the initial state x_0 to a successor state $x(k_0+1) = \hat{x}^\star(k_0+1|k_0) = x^\star(k_0+1|k_0) = x^\star$. Next, it is demonstrated that, at instant $k_1 = k_0 + 1$, for initial condition $x_1 = x(k_1)$ and scheduling sequence Γ_{k_1}, there exists a feasible solution to Problem 1. The feasibility of the solution at instant k_0 is used to construct a feasible solution at this following sample k_1.

Take the bounds of the variation of the scheduling parameters into account, as follows:

$$x(k_1+1) \quad = \quad A(\rho(k_1))x_1 + B_1(\rho(k_1))u(k_1), \tag{40}$$

since $u(k_1) = \kappa x_1 = \kappa x^\star$ and $\rho(k_1) = \rho(k_0) + \sum_{l=0}^{k_1-k_0} \partial\rho(l)$, one arrives at:

$$\begin{aligned} x(k_1+1) &= A(\rho(k_0)+\partial\rho(k_0))x^\star + B_1(\rho(k_0)+\partial\rho(k_0))\kappa x^\star \\ &= [A(\rho(k_0)+\partial\rho(k_0)) + B_1(\rho(k_0)+\partial\rho(k_0))\kappa]\, x^\star \\ &= \underbrace{(A(\rho(k_0)+B_1(\rho(k_0))\kappa)}_{A_{cl}(\rho(k_0),x^\star)} x^\star \\ &+ \underbrace{(A(\partial\rho(k_0)) + B_1(\partial\rho(k_0))\kappa)\, x^\star}_{w(k_1)} . \end{aligned}$$

Since $w(k_1)$ is ultimately bounded due to its construction (given that μ_j is bounded and residing on the fact that scheduling parameters variation rates are also bounded), $\underline{\partial\rho} \leq \partial\rho(k) \leq \overline{\partial\rho}$, and the bounds on x (i.e. $x \in \mathcal{X}$), it holds that $x(k_1+1)$ is indeed an admissible point, contained inside the feasibility set for x (it does not diverge), this is: $A_{cl}(\rho(k_0), x^\star)x^\star + w(k_1) \in \mathcal{X}$ (.

Finally, apart from this induction development, the three recursive feasibility axioms will also be verified individually:

A1) Indeed $\ell(x)$ is $\mathcal{K}$-class lower bounded, i.e.[12]:

$$\begin{aligned}
\ell(x) &= \left(x(k+i)^T Q x(k+1) + u(k)^T R u(k)\right) \\
&= \left(A(\rho)x + B_1(\rho)u\right)^T Q \left(A(\rho)x + B_1(\rho)u\right) + x^T \kappa^T R \kappa x \\
&= x^T (A(\rho)^T Q A(\rho) + 2A^T(\rho) Q B_1(\rho)\kappa \\
&\quad + \kappa^T B^T(\rho) Q B_1(\rho)\kappa + \kappa^T R \kappa) x \\
&= x^T \left(A_\ell(\rho, x)\right) x \geq x^T \beta_1 x = \beta_1(||x||),
\end{aligned} \tag{41}$$

which means that it is always possible to choose a real constant scalar β_1 which is $\leq A_\ell(\underline{\rho}, \underline{x}) \forall\, x \in \Omega = \Upsilon_1$.
A2) Indeed the terminal stage cost $V(\cdot)$ is $\mathcal{K}$-class upper bounded; departing from $x(0) = x$, it follows[13]:

$$\begin{aligned}
V(x(k+N_p)) &= x^T(k+N_p) P x(k+N_p) \\
&= \left(A^{N_p}(\Gamma_k)x + B_1^{N_p}(\Gamma_k)\kappa \overrightarrow{X}_{k,N_p}\right)^T P \\
&\quad \left(A^{N_p}(\Gamma_k)x + B^{N_p}(\Gamma_k)\kappa \overrightarrow{X}_{k,N_p}\right) \\
&= x^T \overbrace{\left(A^{N_p}(\Gamma_k) P A^{N_p}(\Gamma_k)\right)}^{A_V(\Gamma_k)} x \\
&+ 2x^T \overbrace{\left((A^{N_p}(\Gamma_k))^T P B^{N_p}(\Gamma_k)\kappa\right)}^{h(\Gamma_k, \overrightarrow{X}_{k,N_p})} \overrightarrow{X}_{k,N_p} \\
&+ \overrightarrow{X}^T_{k,N_p} \underbrace{\left(\kappa^T (B^{N_p}(\Gamma_k))^T P B^{N_p}(\Gamma_k)\kappa\right)}_{B_V(\Gamma_k, \overrightarrow{X}_{k,N_p})} \overrightarrow{X}_{k,N_p} \\
&\leq x^T \beta_2 x = \beta_2(||x||),
\end{aligned} \tag{42}$$

which means that it is always possible to find a real constant scalar

$$\begin{aligned}
\beta_2 &\geq (A_V(\Gamma_k) + 2h(\Gamma_k^{max}, \overrightarrow{\overrightarrow{X}}_{k,N_p}), \overrightarrow{\overrightarrow{A}}_{k,N_p}) \\
&+ \overrightarrow{\overrightarrow{A}}^T_{k,N_p} B_V(\Gamma_k^{max}, \overrightarrow{\overrightarrow{X}}_{k,N_p}) \overrightarrow{\overrightarrow{A}}_{k,N_p}), \forall\, x \in \Upsilon_0.
\end{aligned}$$

A3) Finally, the last axiom is verified: consider $x(k+j+1) = x(k+1)$ and $x(k+j) = x$, for notation simplicity, which is valid for all $j \in 0, \ldots, N_p - 1$.The terminal ingredient should be decrescent along the solution of x. Of course, the use of the contracting terminal constraints $x(k+N_p|k) \in \Upsilon_j$, implies that the state trajectories are steered further towards theQP target set goal as k increases, this, by itself, should guarantee that V is decrescent[47, 41]. The decay of $V(\cdot)$ is demonstrated:

$$\begin{aligned}
V(x(k+1)) - V(x) + \ell(x) &\leq 0 \\
x(k+1)^T P x(k+1) - x^T P x + x^T (A_\ell(\Gamma_k, x)) x &\leq 0 \\
x^T (A(\rho(k))^T P A(\rho(k)) + 2A(\rho(k)) P B(\rho(k))\kappa & \\
+ \kappa^T B^T(\rho(k)) P B(\rho(k))\kappa) x & \\
- x^T P x + x^T (A_\ell(\Gamma_k, x)) x &\leq 0,
\end{aligned}$$

[12] Notation is simplified, the $(k+i)$ is dropped.
[13] Notation $\overrightarrow{X}_{k,j}$ denotes the collection of sequence of states from $x(k)$ to $x(k+j)$.

which is equivalent to:

$$\begin{aligned}\big(A^T(\rho(k))(P+Q)A(\rho(k))-P\big)\\ +(\kappa^T B^T(\rho(k))(P+Q)A(\rho(k))\\ +A^T(\rho(k))(P+Q)B(\rho(k))\kappa)\\ +\big(\kappa^T B^T(\rho(k))(P+Q)B(\rho(k))\kappa\big)\\ \leq \kappa^T R\kappa,\end{aligned}$$

which is inherently verified due to the choice of P under Assumption 4.1, being Q also full-rank and positive definite by definition. This concludes proof. □

4.4. Implementation Remarks

This Section presented an MPC design procedure for the control of SA suspension systems, aiming to enhance passanger comfort performances. This procedure converts the nonlinear optimization problem with the original qLPV model into an LTI-scheduled model and QP complexity. To make sure the simplifications of using a scheduling trajectory guess do not compromise the control performances, a pseudo-reference and control invariant set sequences are used s.t. feasibility is guaranteed. This MPC design is sub-optimal due to model-process mismatches, but it has a major advantage of using a single QP, which makes it computationally practicable under the 5 ms sampling period of the vehicle.

For the implementation of the algorithm, it is assumed that the road profile (load disturbances) $w(k)$ are known for the future N_r steps. This information can be pursued with different schemes from the literature, such as frequencial preview loops, adaptative estimation schemes or even extended observers that estimate the road together with the states. Some options for these algorithms are available in [48, 5, 17, 49]. Notice that when computing the terminal set sequences, the road profile information is embedded. The implementation of the proposed MPC algorithm is described in Algorithm 1.

5. Numerical Results

In the Section, numerical simulation results are presented to illustrate the performances of a SA suspension system under the control of the proposed qLPV MPC algorithm. The following results have been obtained with textitMatlab, *Yalmip* toolbox and *Gurobi* solver. The simulation is performed with a realistic, validated, full vehicle nonlinear model of the experimental testbed of a vehicle equipped with four ER dampers.

Once again, recall that the control input for the SA suspension system is the PWM signal $u(t)$, which varies the damping coefficient of the ER dampers by changing the electric field applied over them, which varies the amount of force that is delivered.

Recall that the primary control objective $\ell(\cdot)$ is taken in order to minimize chassis accelerations, to ensure that a smoother ride is provided and the comfort of the passengers is enhanced. The indexes provided in Section 2.1 will be used to evaluate the enhancement provided by the proposed control scheme.

In the following Figures, the proposed method is denoted "qLPVMPC", while "COLQR" denotes a clipped optimal LQR, computed with the LTI versions (frozen $\hat{\Gamma}_k$) of the qLPV process and the same weights Q, R and P. The results obtained with a purely passive, uncontrolled SA damper are marked as "PDamp" (for this, u is taken as null).

According to [17], the prediction horizon N_p is taken as 10 samples, while the contractive horizon N_r is taken as 25, meaning that the positively invariant control sets shrink 2.5 times slower than the sliding horizon. For evaluation of the control strategy itself, the computational processing time for the sequence of sets is excluded from the nominal elapsed time of the algorithm, since they could have been performed all offline, as done in [50].

The following results consider the SA suspension at the front-left corner of the vehicle. Similar results were obtained for the other three corners. The chosen road profile $z_r(t) = w(t)$ stands for a car running in a straight line on a dry road, when it encounters ($t' = 0.5$ s) a sequence of 5 mm bumps on all its wheels,

Algorithm 1 LPV MPC for Passenger Comfort Enhancement

1. Use some estimation algorithm to get the future values for the road profile disturbances w along the next N_r steps;
2. Use a saturated recursive LS procedure to estimate the scheduling parameters of the system along the horizon $\hat{\Gamma}_k$, these parameters represent a hyperbolic tangent function of the deflection velocity and suspension deflection variables;
3. Compute the LTI model that approximates the process along the horizon, based on the scheduling evolution guess $\hat{\Gamma}_k$ for $j = 1, \ldots, N_p$.
4. Compute the final set that contains the desired reference Υ_0 according to Eq. (34). This step has also a QP complexity, see [41];
5. Loop with $i = 1 : 1 : N_r$, from $k = k_0$:
6. Compute the sequence Υ_j of $(N_r - i)$ sets, according to Eq. (35). These sequences of sets are computed using relaxations/contractions from the present state until the target final set Υ_0.
7. Solve the following QP:

$$\begin{aligned} \min_{U} \quad & J_N(x,u) \\ \text{s.t.} \quad & \text{System Evolution: Eq. (14)}, \\ & u(k+i-1|k) \in \mathcal{U}, \\ & x(k+i|k) \in \mathcal{X}, \\ & x(k+N_r|k) \in \Upsilon_j, \; j = \max\{N_r - k, 0\}, \end{aligned} \tag{43}$$

8. From U, take the first entry $u(k|k) = \kappa x(k)$ and apply it to the process.

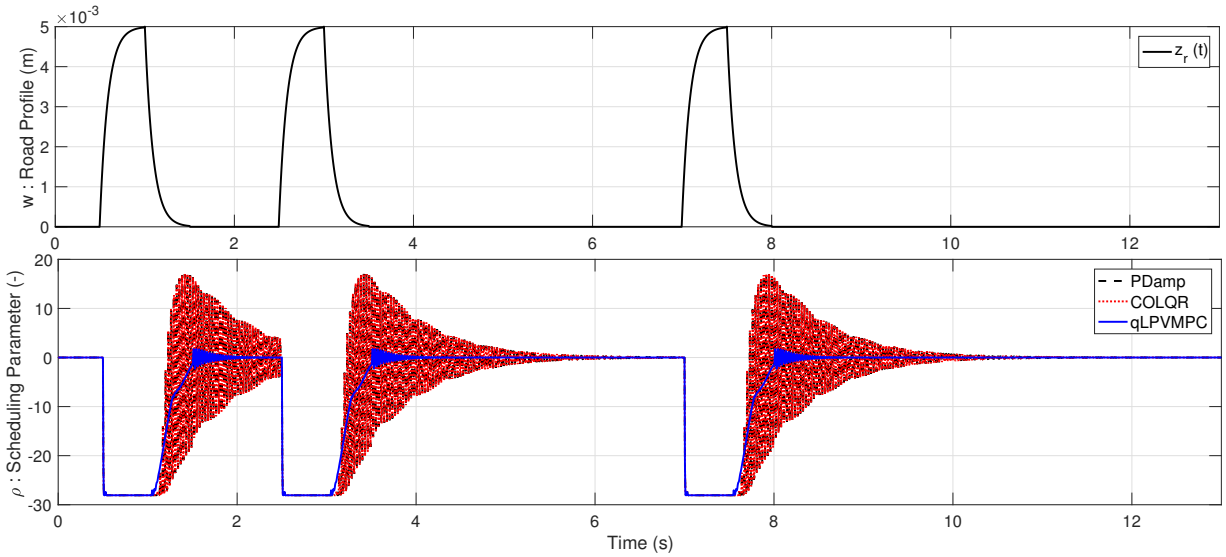

Figure 5: Road Profile and Scheduling Parameters

exciting the bounce motion, which must be counteracted by the suspension controller. This simulation scenario comprises 13 s. Figure 5 shows these bumps and the scheduling parameters along the simulation.

Most importantly, Figure 6 depicts both controlled outputs $(\ddot{z}_s, \ddot{z}_{us})$ and the delivered damping force (as well as the dissipativity constraints $\mathcal{D}$). Clearly, it is evident that the proposed predictive controller yields the smoother results, further minimizing the control objective $\ell(\cdot)$. Due to saturation effects, the LQR strategy achieved almost the results as those with a passive damper (in open-loop).

To better evaluate these results, the time index $J^t_{comfort}$ is computed through a normalized root-mean-square (RMS) function of the acceleration variables. Table 2 shows the RMS obtained for the passive case for both $\ddot{z}_s(t)$ and $\ddot{z}_{us}(t)$ and the enhancements achieved with the COLQR and qLPVMPC methods with respect to the passive condition. As evidenced, the proposed method yields a 14.25 % passenger comfort enhancement in terms of the chassis acceleration variable. This is quite significant as it the algorithm is computed for a scaled vehicle model.

Table 2: Performance Enhancement

RMS	Method	Value	Enhancement
Reference	**PDamp**	0.4650	0 %
Tracking	**COLQR**	0.4646	0.08 %
$\ddot{z}_s(t) \to 0$	**qLPVMPC**	0.3984	14.32 %
Reference	**PDamp**	3.008	0 %
Tracking	**COLQR**	3.005	0.1 %
$\ddot{z}_{us}(t) \to 0$	**qLPVMPC**	2.611	13.18 %

Concerning the frequencial index $J^f_{comfort}$, Figure 7 shows the FFT results of $\ddot{z}_s(t)$ for both PDamp and qLPVMPC cases, under the $0-20$ Hz frequency range. As discussed by [12], the important issue is to reduce the peaks caused when a road profile income appears. In numerical terms, the peak with the qLPVMPC method is 16.25 % smaller than the one with the passive damper, which demonstrates furthermore the enhancement provided to the passengers.

It must be remarked that the proposed method with its sets computations can be performed using parallel

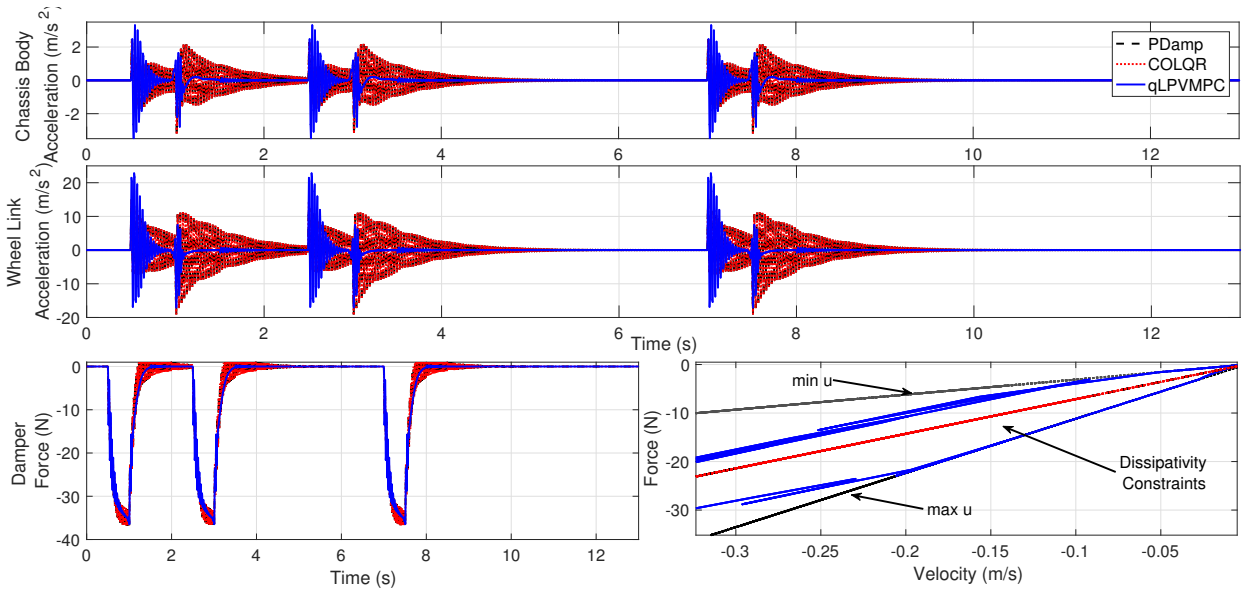

Figure 6: Controlled Outputs and Control Signal

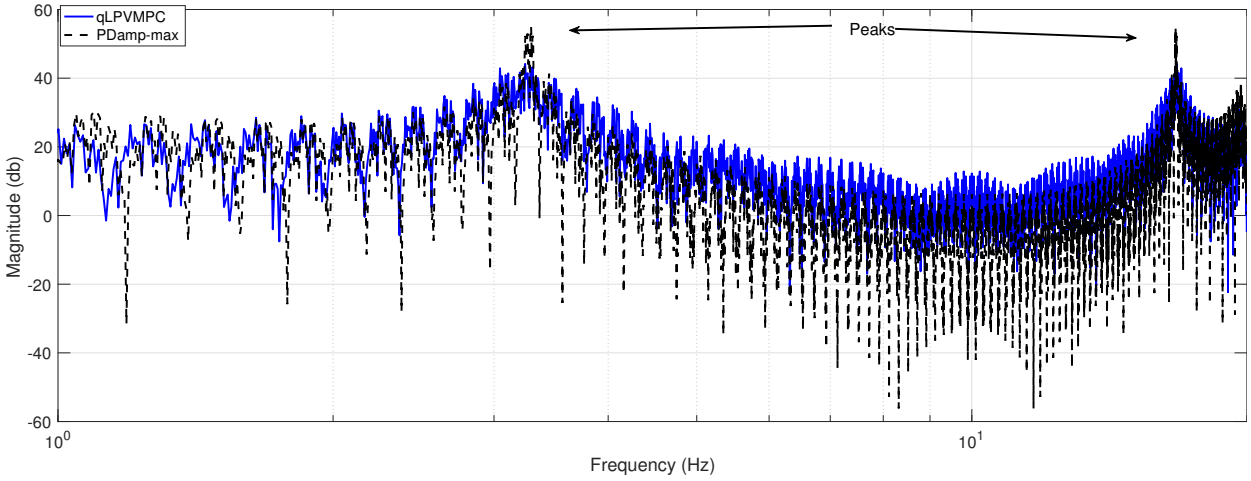

Figure 7: FFT: $J^f_{comfort}$

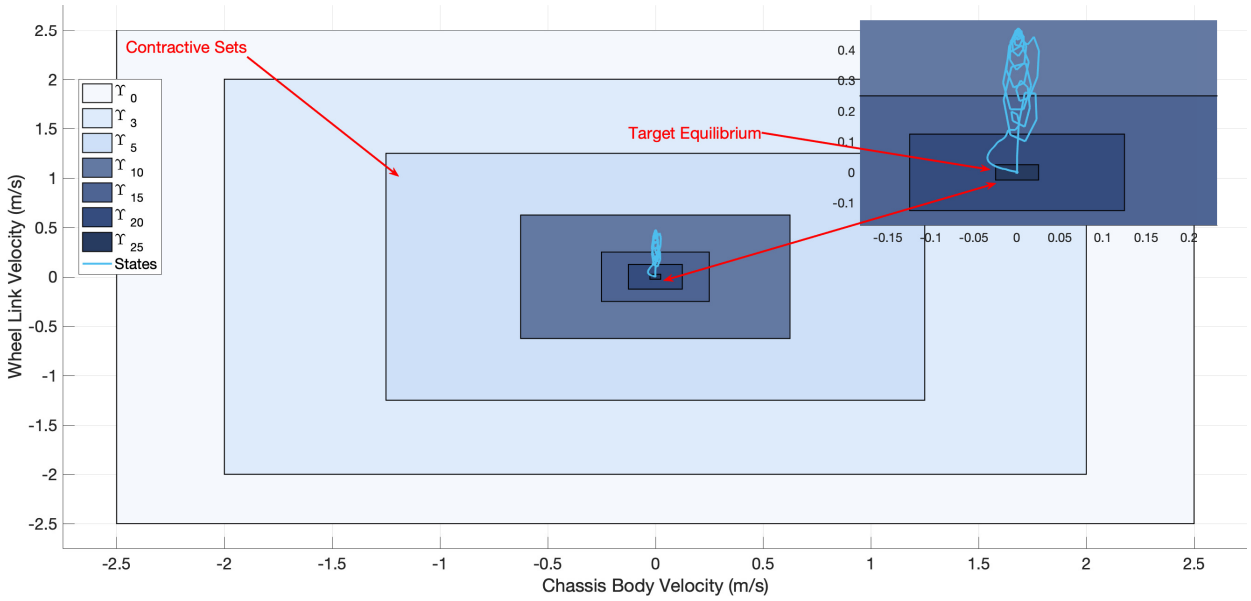

Figure 8: Controlled Outputs Tracking Plane

computing tools; in fact, this QP step that is used to compute the control law (Step 7 of the Algorithm in Section 4.4) allows to achieve a quite reasonable numerical effort (elapsed within $0.035\,\mathrm{s} < \mathrm{T_s}$). If the original nonlinear programming problem was to be considered (without the frozen guess for the scheduling parameter), much greater effort would be necessary and the law would not be able to be implemented for real-time purposes.

Concerning the qLPV results, some additional results must be presented: Figure 8 shows a $\dot{z}_s(t) \times \dot{z}_{us}(t)$ plane and the (2D-cut) contracting sets Υ_j. Clearly, the sets tool works to constrain the convergence of theses velocities to a (target) final region, despite the disturbance and model-process mismatches. Finally, Figure 9 shows a 3D polyhedra cut version of Υ_{N_r} (which is, in fact, 4D) and the polyhedra obtained for the evolution of the last three states; this Figure demonstrates how the initial polyhedra contains the state evolutions, which finally converge to Υ_0. This result also corroborates with the validity of the recursive feasibility axioms (specially $A3$).

6. Conclusions

This paper elaborated a novel MPC method for the enhancement of passenger comfort using SA suspension systems. The suspension is modelled within a qLPV framework, and the damping force is modelled through a nonlinear hyperbolic tangent function, as suggested by the literature. The method embed the nonlinearities within a scheduling parameter, which is estimated through the prediction horizon at each sampling instant. The frozen scheduling evolution guess is used to transform the nonlinear prediction problem into a linear QP, which can be solved within some mili-seconds. Set-based tools (terminal ingredient and stage cost) are included to the MPC so that it maintains quadratic stability and recursive feasibility, despite model-process mismatches. These properties are analytically demonstrated. The optimization cost function of the MPC is shown to embed comfort constraints, with regard to performance indexes from the literature. The algorithm is successfully applied to the control of a Semi-Active suspension system via realistic simulation, achieving good results compared to existing control optimal control methods.

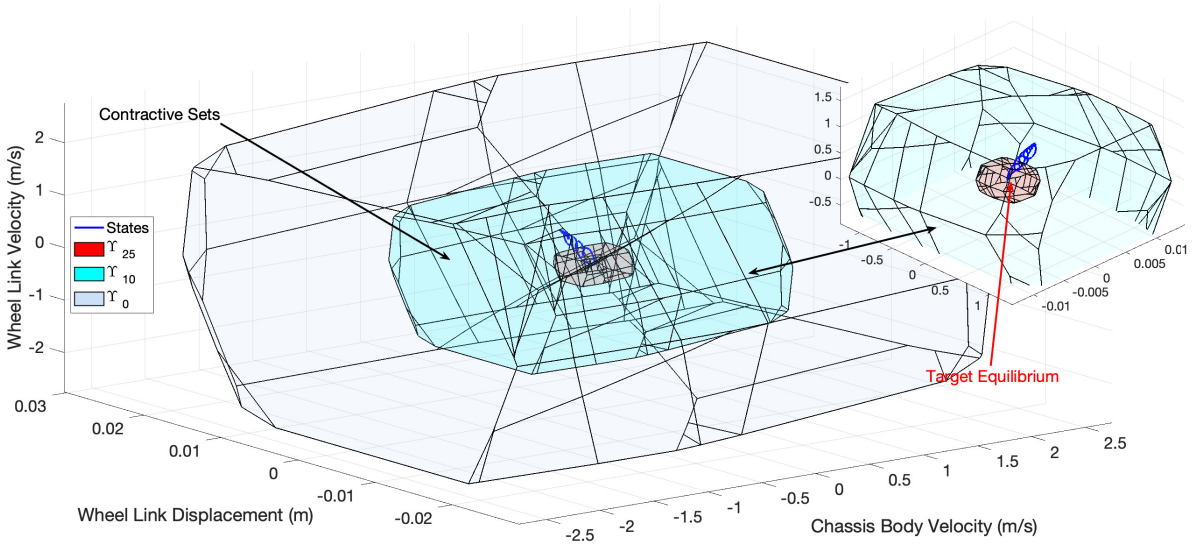

Figure 9: Polyhedra Υ_j and states

Acknowledgments

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

References

[1] Fischer, D., Isermann, R.: 'Mechatronic semi-active and active vehicle suspensions', *Control engineering practice*, 2004, **12**, (11), pp. 1353–1367

[2] Savaresi, S.M., Poussot.Vassal, C., Spelta, C., Sename, O., Dugard, L.: 'Semi-active suspension control design for vehicles'. Butterworth-Heinemann. Elsevier, 2010)

[3] Poussot.Vassal, C., Spelta, C., Sename, O., Savaresi, S.M., Dugard, L.: 'Survey and performance evaluation on some automotive semi-active suspension control methods: A comparative study on a single-corner model', *Annual Reviews in Control*, 2012, **36**, (1), pp. 148–160

[4] Tseng, H.E., Hrovat, D.: 'State of the art survey: active and semi-active suspension control', *Vehicle system dynamics*, 2015, **53**, (7), pp. 1034–1062

[5] Unger, A., Schimmack, F., Lohmann, B., Schwarz, R.: 'Application of LQ-based semi-active suspension control in a vehicle', *Control Engineering Practice*, 2013, **21**, (12), pp. 1841–1850

[6] Nguyen, M.Q., da Silva, J.G., Sename, O., Dugard, L.: 'A state feedback input constrained control design for a 4-semi-active damper suspension system: a quasi-LPV approach', *IFAC-PapersOnLine*, 2015, **48**, (14), pp. 259–264

[7] Poussot.Vassal, C., Sename, O., Dugard, L., Gaspar, P., Szabo, Z., Bokor, J.: 'A new semi-active suspension control strategy through LPV technique', *Control Engineering Practice*, 2008, **16**, (12), pp. 1519–1534

[8] Pang, H., Zhang, X., Xu, Z.: 'Adaptive backstepping-based tracking control design for nonlinear active suspension system with parameter uncertainties and safety constraints', *ISA transactions*, 2019, **88**, pp. 23–36

[9] Camacho, E.F., Bordons, C.: 'Model predictive control'. Springer Science & Business Media, 2013)

[10] Allgöwer, F., Zheng, A.: 'Nonlinear model predictive control'. vol. 26. Birkhäuser, 2012)

[11] Nguyen, M.Q., Canale, M., Sename, O., Dugard, L. 'A model predictive control approach for semi-active suspension control problem of a full car'. In: IEEE 55th Conference on Decision and Control. IEEE, 2016. pp. 721–726

[12] Poussot.Vassal, C., Savaresi, S.M., Spelta, C., Sename, O., Dugard, L. 'A methodology for optimal semi-active suspension systems performance evaluation'. In: 49^{th} IEEE Conference on Decision and Control. IEEE, 2010. pp. 2892–2897

[13] Giorgetti, N., Bemporad, A., Tseng, H.E., Hrovat, D.: 'Hybrid model predictive control application towards optimal semi-active suspension', *International Journal of Control*, 2006, **79**, (05), pp. 521–533

[14] Brezas, P., Smith, M.C., Hoult, W.: 'A clipped-optimal control algorithm for semi-active vehicle suspensions: Theory and experimental evaluation', *Automatica*, 2015, **53**, pp. 188–194

[15] Morato, M.M., Sename, O., Dugard, L.: 'LPV-MPC fault tolerant control of automotive suspension dampers', *IFAC-PapersOnLine*, 2018, **51**, (26), pp. 31–36

[16] Beal, C.E., Gerdes, J.C.: 'Model predictive control for vehicle stabilization at the limits of handling', *IEEE Transactions on Control Systems Technology*, 2013, **21**, (4), pp. 1258–1269
[17] Morato, M.M., Nguyen, M.Q., Sename, O., Dugard, L.: '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*, 2018,
[18] Sename, O., Gaspar, P., Bokor, J.: 'Robust control and linear parameter varying approaches: application to vehicle dynamics'. vol. 437. Springer, 2013)
[19] Mohammadpour, J., Scherer, C.W.: 'Control of linear parameter varying systems with applications'. Springer Science & Business Media, 2012)
[20] Jungers, M., Oliveira, R.C., Peres, P.L.: 'MPC for LPV systems with bounded parameter variations', *International Journal of Control*, 2011, **84**, (1), pp. 24–36
[21] Abbas, H.S., Toth, R., Petreczky, M., Meskin, N., Mohammadpour, J.: 'Embedding of nonlinear systems in a linear parameter-varying representation', *IFAC Proceedings Volumes*, 2014, **47**, (3), pp. 6907–6913
[22] Morato, M.M., Normey.Rico, J.E., Sename, O. 'Novel qLPV MPC design with least-squares scheduling prediction'. In: Proceedings of the 3^{th} IFAC Workshop on Linear Parameter Varying Systems, Eindhoven, The Netherlands, Nov. 4-6, 2019.
[23] Morato, M.M., Sename, O., Dugard, L., Nguyen, M.Q.: 'Fault estimation for automotive electro-rheological dampers: LPV-based observer approach', *Control Engineering Practice*, 2019, **85**, pp. 11–22
[24] Guo, S., Yang, S., Pan, C.: 'Dynamic modeling of magnetorheological damper behaviors', *Journal of Intelligent material systems and structures*, 2006, **17**, (1), pp. 3–14
[25] Ren, H., Chen, S., Zhao, Y., Liu, G., Yang, L.: 'State observer-based sliding mode control for semi-active hydro-pneumatic suspension', *Vehicle System Dynamics*, 2016, **54**, (2), pp. 168–190
[26] Falcone, P., Tseng, H.E., Asgari, J., Borrelli, F., Hrovat, D.: 'Integrated braking and steering model predictive control approach in autonomous vehicles', *IFAC Proceedings Volumes*, 2007, **40**, (10), pp. 273–278
[27] Beal, C.E., Gerdes, J.C.: 'Model predictive control for vehicle stabilization at the limits of handling', *IEEE Transactions on Control Systems Technology*, 2012, **21**, (4), pp. 1258–1269
[28] Hrovat, D.: 'Survey of advanced suspension developments and related optimal control applications', *Automatica*, 1997, **33**, (10), pp. 1781–1817
[29] Besselmann, T., Lofberg, J., Morari, M.: 'Explicit MPC for LPV systems: Stability and optimality', *IEEE Transactions on Automatic Control*, 2012, **57**, (9), pp. 2322–2332
[30] Abbas, H.S., Hanema, J., Tóth, R., Mohammadpour, J., Meskin, N.: 'A new approach to robust MPC design for LPV systems in input-output form', *IFAC-PapersOnLine*, 2018, **51**, (26), pp. 112–117
[31] Ding, B., Ping, X., Pan, H.: 'On dynamic output feedback robust MPC for constrained quasi-lpv systems', *International Journal of Control*, 2013, **86**, (12), pp. 2215–2227
[32] Casavola, A., Famularo, D., Franzè, G., Garone, E.: 'A fast ellipsoidal MPC scheme for discrete-time polytopic linear parameter varying systems', *Automatica*, 2012, **48**, (10), pp. 2620–2626
[33] Hanema, J., Tóth, R., Lazar, M. 'Stabilizing non-linear MPC using linear parameter-varying representations'. In: 56^{th} Annual Conference on Decision and Control. IEEE, 2017. pp. 3582–3587
[34] Hanema, J., Lazar, M., Tóth, R.: 'Stabilizing tube-based model predictive control: Terminal set and cost construction for LPV systems', *Automatica*, 2017, **85**, pp. 137–144
[35] Rathai, K.M.M., Alamir, M., Sename, O., Tang, R.: 'A parameterized NMPC scheme for embedded control of semi-active suspension system', *IFAC-PapersOnLine*, 2018, **51**, (20), pp. 301–306
[36] Alamir, M.: 'A framework for real-time implementation of low-dimensional parameterized NMPC', *Automatica*, 2012, **48**, (1), pp. 198–204
[37] Cisneros, P.S., Sridharan, A., Werner, H.: 'Constrained predictive control of a robotic manipulator using quasi-LPV representations', *IFAC-PapersOnLine*, 2018, **51**, (26), pp. 118–123
[38] Alcalá, E., Puig, V., Quevedo, J.: 'LPV-MPC control for autonomous vehicles', *IFAC-PapersOnLine*, 2019, **52**, (28), pp. 106–113
[39] Morato, M.M., Mendes, P.R.C., Normey.Rico, J.E., Bordons, C.: 'LPV-MPC fault-tolerant energy management strategy for renewable microgrids', *International Journal of Electrical Power & Energy Systems*, 2020, **117**, pp. 105644
[40] Mayne, D.Q., Rawlings, J.B., Rao, C.V., Scokaert, P.O.: 'Constrained model predictive control: Stability and optimality', *Automatica*, 2000, **36**, (6), pp. 789–814
[41] Limón, D., Alvarado, I., Alamo, T., Camacho, E.F.: 'MPC for tracking piecewise constant references for constrained linear systems', *Automatica*, 2008, **44**, (9), pp. 2382–2387
[42] Ferramosca, A., Limon, D., Alvarado, I., Alamo, T., Camacho, E.F. 'MPC for tracking of constrained nonlinear systems'. In: Proceedings of the 48th IEEE Conference on Decision and Control held jointly with 28th Chinese Control Conference, 2009. pp. 7978–7983
[43] Cisneros, P.G., Werner, H.: 'Fast nonlinear MPC for reference tracking subject to nonlinear constraints via quasi-LPV representations', *IFAC-PapersOnLine*, 2017, **50**, (1), pp. 11601–11606
[44] Shamma, J.S. 'An overview of lpv systems'. In: Control of linear parameter varying systems with applications. Springer, 2012. pp. 3–26
[45] Blanchini, F.: 'Set invariance in control', *Automatica*, 1999, **35**, (11), pp. 1747–1767
[46] Mate, S., Kodamana, H., Bhartiya, S., Nataraj, P.S.V.: 'A stabilizing sub-optimal model predictive control for quasi-linear parameter varying systems', *IEEE Control Systems Letters*, 2019,
[47] Limon, D., Alamo, T., Camacho, E.F.: 'Enlarging the domain of attraction of MPC controllers', *Automatica*, 2005, **41**, (4), pp. 629–635

[48] Tudón.Martínez, J.C., Fergani, S., Sename, O., Martinez, J.J., Morales.Menendez, R., Dugard, L.: 'Adaptive road profile estimation in semiactive car suspensions', *IEEE Transactions on Control Systems Technology*, 2015, **23**, (6), pp. 2293–2305
[49] Cvok, I., Deur, J., Tseng, H.E., Hrovat, D. 'Comparative performance analysis of active and semi-active suspensions with road preview control'. In: The IAVSD International Symposium on Dynamics of Vehicles on Roads and Tracks. Springer, 2019. pp. 1808–1818
[50] Bumroongsri, P., Kheawhom, S.: 'An off-line robust MPC algorithm for uncertain polytopic discrete-time systems using polyhedral invariant sets', *Journal of Process Control*, 2012, **22**, (6), pp. 975–983