THÈSE

pour obtenir le grade de

DOCTEUR DE LA UNIVERSITÉ GRENOBLE ALPES et de l'UNIVERSIDADE FEDERAL DE SANTA CATARINA

Spécialité : **Automatique-Productique**

Arrêté ministériel : 25 mai 2016

Présentée par
Marcelo MENEZES MORATO

Thèse dirigée par:
Julio Elias NORMEY-RICO et
Olivier SENAME

préparée au sein du **GIPSA-Lab**
dans **L'École doctorale Electronique, Electrotechnique, Automatique, Traitement du Signal (EEATS)**

Predictive Control Methods for Linear Parameter Varying Systems

Méthodes de Commande Prédictive pour les Systèmes Linéaires à Paramètres Variants

Métodos de Controle Preditivo para Sistemas Lineares a Parâmetros Variantes

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Morato, Marcelo Menezes
Predictive Control Methods for Linear Parameter Varying Systems / Marcelo Menezes Morato ; orientador, Julio Elias Normey-Rico, coorientador, Olivier Sename, 2023.
297 p.

Tese (doutorado) - Universidade Federal de Santa Catarina, Centro Tecnológico, Programa de Pós-Graduação em Engenharia de Automação e Sistemas, Florianópolis, 2023.

Inclui referências.

1. Engenharia de Automação e Sistemas. 2. Controle preditivo baseado em modelo. 3. Sistemas lineares a parâmetros variantes. 4. Controle robusto. 5. Control não linear. I. Normey-Rico, Julio Elias. II. Sename, Olivier. III. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Engenharia de Automação e Sistemas. IV. Título.

THÈSE

pour obtenir le grade de

DOCTEUR DE LA UNIVERSITÉ GRENOBLE ALPES et de l'UNIVERSIDADE FEDERAL DE SANTA CATARINA

Spécialité : **Automatique-Productique**

Arrêté ministériel : 25 mai 2016

Présentée par
Marcelo MENEZES MORATO

Thèse dirigée par:
Julio Elias NORMEY-RICO et
Olivier SENAME

préparée au sein du **GIPSA-Lab**
dans **L'École doctorale Electronique, Electrotechnique, Automatique, Traitement du Signal (EEATS)**

Méthodes de Commande Prédictive pour les Systèmes Linéaires à Paramètres Variants

Thèse soutenue publiquement le **21 février 2023**,
devant le jury composé de :

M. Mazen ALAMIR
Directeur de Recherche CNRS, Université Grenoble Alpes, France, Président
M. Daniel LIMÓN MARRUEDO
Professeur, Universidad de Sevilla, Espagne, Rapporteur
M. Sorin OLARU
Professeur, Université Paris-Saclay, France, Rapporteur
M. Daniel COUTINHO
Professeur, Universidade Federal de Santa Catarina, Brésil, Membre
M. John-Jairo MARTINEZ-MOLINA
Professeur, Grenoble INP, Université Grenoble Alpes, France, Membre
M. Gustavo ARTUR de ANDRADE
Professeur, Universidade Federal de Santa Catarina, Brésil, Membre
M. Julio Elias NORMEY-RICO
Professeur, Universidade Federal de Santa Catarina, Brésil, Directeur de thèse, Membre
M. Olivier SENAME
Professeur, Université Grenoble Alpes, France, Directeur de thèse, Membre

Queremos saber
O que vão fazer
Com as novas invenções
Queremos notícia mais séria
Sobre a descoberta da antimatéria
E suas implicações
Na emancipação do homem
Das grandes populações
Homens pobres das cidades
Das estepes, dos sertões

Queremos saber
Quando vamos ter
Raio laser mais barato
Queremos de fato um relato
Retrato mais sério
Do mistério da luz
Luz do disco-voador
Pra iluminação do homem
Tão carente e sofredor
Tão perdido na distância
Da morada do senhor

Queremos saber
Queremos viver
Confiantes no futuro
Por isso se faz necessário
Prever qual o itinerário da ilusão
A ilusão do poder
Pois se foi permitido ao homem
Tantas coisas conhecer
É melhor que todos saibam
O que pode acontecer

Queremos saber
Queremos saber
Todos queremos saber

- *Gilberto Gil, 1976.*

Ricardo Beliel, 1976.

Agradecimentos

Neste espaço inicial da minha tese, deposito algumas palavras com a intenção de agradecer todas as pessoas que me tocaram, de alguma forma, durante minha trajetória ao longo do doutorado.

Primeiramente, agradeço à minha mãe, Débora, e ao meu pai, José Rubens, pelo carinho, pelo amor, pelo apoio constante, e por me instigarem, desde pequeno, a curiosidade, a crítica, e o questionamento. Agradeço, pela afeição constante, meus amados avós, Dette e Antônio, minha madrinha, Estera, e meu padrinho, Emílio, além dos primos e primas: Thaís, Guto, Mallu, Tanira, Thiago, Samya, Alyssa, Beni, Biba, Manu, Luan, Gui e Julia. Agradeço, também, o incentivo por parte das minhas tias Mônica, Estela e Cláudia.

À minha parceira Lara, agradeço pela conexão profunda, por cada conversa sem fim, por todo suporte, e por este amor que segue transcorrendo e transformando tempo e espaço, navegando todos os sentidos. Dos afetos que ganhei através da Lara, agradeço, em especial, o acolhimento e o apreço de Denise, Alexandre, Flávia e Guilherme. Também recordo do carinho de Julia, Inez, Romeu, e do pequeno Antônio.

Lembro, aqui, de agradecer, pelos conselhos, risadas e por tanto bem-querer, meus amigos e minhas amigas do coração. Às amizades que mantenho desde adolescente, Thaian, Niterói, Arthur, Morena, Henrique e Breno, e àquelas que desde a graduação me acompanham, Sunshine, Duda, Marina, Cani, Eduardo, Igor, e Alina. Recordo, também, da parceria e do suporte, especialmente ao longo do período em isolamento, de Thaler, André, Renato, Vito, Lê, Jamal, Malu e Zomer, além da amizade em Grenoble de Sabine, Vanessa, Bob, Kazuti, Bia e, em especial, da Monica. Não deixo de rememorar o afeto, à distância, mas sempre presente, de Aninha, Paiva, Petrassi, João Vitor, Maísa, Noaii, Alice, André, Dani, Argenis e Nomar.

Ao Julio, meu orientador e amigo, agradeço enormemente por sempre me acolher e me guiar nos caminhos da pesquisa, deste a graduação. Obrigado pela orientação cuidadosa e, em especial, pelo convite para escrevemos juntos o livro-texto pra a SBA. Não tenho dúvidas que continuaremos trabalhando juntos, da forma extremamente agradável que sempre tem sido.

Ao meu segundo orientador, e também amigo, Olivier, agradeço pelas revisões minuciosas dos textos, as sugestões teóricas a respeito das minhas pesquisas, e por sempre receber-me da melhor forma possível em Grenoble. Agradeço, também, pelo esforço em ajustar as possibilidades de financiamento para o acordo de cotutela, garantindo minha ida à França. Merci :)

Ao longo destes anos de tese, tive a oportunidade de desenvolver diversas colaborações científicas com colegas da comunidade acadêmica. Aproveito, então, para agradecer as parcerias com o Prof. Tito Santos e com o Prof. Marcus Americano, da Universidade Federal da Bahia, que muito me ajudaram no desenvolvimento de MPCs para rastreamento de referência e orientados ao tema da COVID-19, respectivamente. Agradeço, também, o Prof. Carsten Scherer, da Universidade de Stuttgart, por lá me receber tão cordialmente e pelas enérgicas, motivadoras e desafiadoras discussões sobre MPC, IQCs e dissipatividade. Também gostaria de mencionar e agradecer o Prof. Carlos Bordons, da Universidade de Sevilha, com quem pude manter um vínculo de pesquisa desde minha graduação. Lembro de também de citar, aqui, meus amigos e parceiros Dr. Paulo Mendes, do Fraunhofer Instituto, e Prof. José Vergara-Dietrich, da Universidade Tecnológica Federal do Paraná, com quem muito discuti e fustiguei as pesquisas em energias renováveis.

Menciono, ademais, os debates e as sugestões, a respeito do meu trabalho, fornecidas pelos professores Hector Bessa, Marcelo de Lellis, Eugênio Castelan, Rodolfo Flesch e Eduardo Camponogara, da Universidade Federal de Santa Catarina. Igualmente, agradeço o Prof. Peter Seiler, da Universidade de Michigan, e a Profa. Antoneta Bratcu, do GIPSA-Lab.

Agradeço pelas discussões e colaborações em conjunto, também, o Dr. Tobias Holicki, da Universidade de Stuttgart, o Prof. Joey Ocon, da Universidade das Filipinas Diliman, o Dr. Emanuel Bernardi, da Universidade Tecnológica Nacional da Argentina, o Dr. Juanjo Marquez e a Prof. Ascensión Zafra-Cabeza, da Universidade de Sevilha, o M.Sc Igor Pataro, da Universidade Federal da Bahia, o Prof. Daniel Cajueiro e o Dr. Saulo Bastos, da Universidade de Brasília, o Prof. João Manoel Gomes, da Universidade Federal do Rio Grande do Sul, o Prof. Marc Jungers, da Universidade de Lorraine, e o Prof. Vladimir Stojanović, da Universidade de Kragujevac. Também agradeço, pelas pelos trabalhos em conjunto, os colegas doutorandos Ariel Medero, Thanh-Phong Pham, Victor Cunha, Hugo Pipino e Igor Pataro, além dos alunos de iniciação científica Guilherme Nascimento e Amir Naspolini, e da mestranda Zaman Yazbeck.

Por fim, aproveito deste espaço pra agradecer as sugestões e as discussões sobre meu trabalho dadas pelos Profs. Mazen Alamir e John-Jairo Martinez-Molina, do GIPSA-Lab, tanto em minha banca de tese quanto como meu comitê de avaliação individual ao longo da tese. Agradeço, igualmente, a Prof. Vilma Oliveira, da Universidade de São Paulo, e ao Prof. Márcio Martins, da Universidade Federal da Bahia, pelas discussões colocadas na minha banca de qualificação. Em especial, agradeço aos Profs. Daniel Coutinho e Gustavo Andrade, da Universidade Federal de Santa Catarina, pelas contribuições e sugestões referentes a esta tese, além da participação na minha banca de defesa. Por esta razão, também agradeço os relatos e os comentários do Prof. Sorin Olaru, da Universidade Paris-Saclay, e do Prof. Daniel Limón, da Universidade de Sevilha. Muito obrigado!

Agradeço, também, as instituições de pesquisa brasileiras e francesas, que me financiaram ao longo destes quatro anos de tese. No nome de Lívia, Marielle, Ênio e Rodrigo, agradeço todos os funcionários e servidores que mantém o dia-a-dia destas instituições. Demarco, por mim, meu compromisso com a pesquisa científica como expressão cultural de livre acesso e difusão e, portanto, pontuo os agradecimentos às plataformas Sci-Hub, Libgen, arXivs e HAL.

Contents

List of Figures

List of Tables

Part I

Preamble

Chapter 1

Introduction

This doctoral thesis was set in a joint co-supervision agreement between Universidade Federal de Santa Catarina (UFSC, Brazil) and Communatué Université Grenoble-Alpes (ComUE UGA, France), under the supervision of Dr. Julio Elias Normey-Rico and Dr. Olivier Sename, respectively.

Predictive control is an optimal framework for the regulation of constrained processes, with applicability to many complex nonlinear systems. Accordingly, linear parameter varying models are often able to describe these dynamics. Therefore, the main topic investigated along this work is the exploitation of model predictive control schemes for linear parameter varying systems. Specifically, I face the problem of how to conceive accurate algorithms without requiring the complete knowledge of the future scheduling parameter trajectories.

Thus, in order to proceed debating the investigated topic, I begin this document by presenting a brief overview of (nonlinear) predictive control methods and the most recent advances on this topic. Then, I span the general context of linear parameter varying systems and detail how these realisations are able to describe nonlinear, time-varying dynamics. I bridge these two fields of research by presenting the available linear parameter varying predictive control methods, debating their scholastic interest in the context of robust and nonlinear control. Finally, I lay out the available investigation threads within the detailed scope and present the overall objectives of the thesis, as well as the organisation of the remainder of this document.

Before moving on to proper details, I note[1] that a concrete and thorough survey on model predictive control schemes conceived on the basis of linear parameter varying models is available in [Morato, Normey-Rico, and Sename 2020a]. Therefore, only a concise overview is presented next; the Reader is invited to refer to the latter reference for a more comprehensive debate.

[1] In this Chapter, I intentionally used the singular first-person active voice, since I explain the overall lines of my doctoral thesis. Nevertheless, in the following Chapters I prefer to politely use the plural first-person voice. Many of the advances on predictive control that I present in the sequel were also partially discussed and elaborated in conjunction with my colleagues and supervisors in published material: thus, "we". In the final Chapter, I return to the use of the first-person voice, for general conclusions.

1.1 Why is Model Predictive Control relevant?

Control systems literature has profoundly blossomed over the course of the last five decades. Nevertheless, until the late 80's, the industrial practice of control theory consisted on the use of simple Proportional-Integral (PI) and Proportional-Integral-Derivative (PID) controllers for the vast majority of applications, despite all theoretical advances. It was only with development of Model Predictive Control (MPC), as proposed in the original papers by Cutler, Clark and fellow colleagues [Cutler and Ramaker 1980; Clarke, Mohtadi, and Tuffs 1987], that more advanced control techniques were implemented in real industrial contexts[2]. Since then, predictive control has become an active and field of research, with widespread industrial acceptance [Camacho and Bordons 2013]: a wide range of successful MPC implementations is nowadays recognised (for the most diverse kinds of systems!) [Alamir 2013].

1.1.1 General lines

The main reason why predictive control is so well-established resides on the fact that it has the ability to jointly consider performance optimisation and constraint satisfaction under a relatively simple (and intuitive) synthesis framework. For the sake of the argument, consider a system with known dynamical *behaviour*[3] $\mathfrak{B}$. Then, when applying a predictive control algorithm in order to regulate this system, one extracts an optimal control action u from the solution of an optimisation problem[4] that includes the performance objectives and the systems' constraints. This optimisation solution is repeated online during the implementation: at each discrete unit of time (sample), a new optimisation is solved, and thus leading to a corresponding new control input.

Consider that a system Σ is linear time-invariant[5] (LTI). Furthermore, assume that its behaviour $\mathfrak{B}$ can be partitioned in a single-input single-output (SISO) channel $u \Leftrightarrow y$ in such a way that $\mathfrak{B}_{u \Leftrightarrow y} := \left\{ \exists y \in \mathbb{R}\,,\, (u, y) \in \mathfrak{B} \,|\, y = \sum_{i=1}^{n_y} a_i z^{-i} y + \sum_{i=1}^{n_u} b_i z^{-i} u \right\}$. Furthermore, consider the following performance goal and constraints: (a) the output y should be brought to a steady-state regime target denoted y_r, while (b) the control signal must be bounded to the convex operational set $[0, 1]$. Accordingly, one can formulate an MPC based on the following optimisation, to be solved at each discrete-time instant k:

$$\mathfrak{P}_k \;:=\; \left\{ \begin{array}{lrcl} \min_{U_k} & & \sum_{i=1}^{N_p} \|z^i y - y_r\| & \\ \text{s.t.} & (z^i y, z^{i-1} u) & \in \;\mathfrak{B}_{u \Leftrightarrow y}, & \forall i \in \mathbb{N}_{[1,N_p]}\,, \\ & z^{i-1} u & \in \;[0, 1], & \forall i \in \mathbb{N}_{[1,N_p]}\,, \end{array} \right. \tag{1.1}$$

being $U_k := \text{col}\{u, \dots z^{N_p - 1} u\}$ the control trajectory along the future prediction horizon N_p. Note that the optimisation problem $\mathfrak{P}_k$ minimises the deviance of the output channel from the steady-state regime, thus seeking to ensure the satisfaction of the performance goal "(a)".

[2]In fact, these original MPC algorithms were originally coined for chemical systems and oil refineries.

[3]$\mathfrak{B}$ is the vector space that collects all maps from a time *axis* to the signal space.

[4]This optimisation is expressed in terms of the future dynamic behaviour of the system along a prediction horizon of N_p steps, i.e. $z^i \mathfrak{B}$, $\forall i \in \mathbb{N}_{[0,N_p]}$, being z the time-shift operator.

[5]That is, whose behaviour $\mathfrak{B}$, satisfies $z\mathfrak{B} \subseteq \mathfrak{B}$ for any time-shift $z \geq 1$.

The static solution of the optimisation $\mathfrak{P}_k$ in Eq. (1.1) (i.e. solving it only once, instead of doing this repeatedly, at each sample) is, by itself, already of practical interest. Indeed, the application of the corresponding control sequence $U_k^\star$ is usually called optimal control. Linear Quadratic Regulators (LQRs), for instance, can be represented within this framework using an infinite horizon formulation, i.e. finding the feedback which solves the static $\lim_{N_p \to +\infty} \mathfrak{P}_0$.

Yet, in contrast to a static solution of the optimisation, MPC operates under a sliding-horizon paradigm. This means that, at each discrete instant k, the constrained optimisation problem $\mathfrak{P}_k$ is solved, and the first entry of its minimiser $U_k^\star$ is applied to the process (the remainder is disregarded). Such rolling-horizon mechanism procedure provides inherent robustness properties to MPC, as argues [Allan et al. 2017], since $\mathfrak{P}_k$ is updated according to the available measurements of the process. Figure 1.1 provides a block diagram representation of the implementation of MPC scheme, in order to further elucidate the current debate[6]

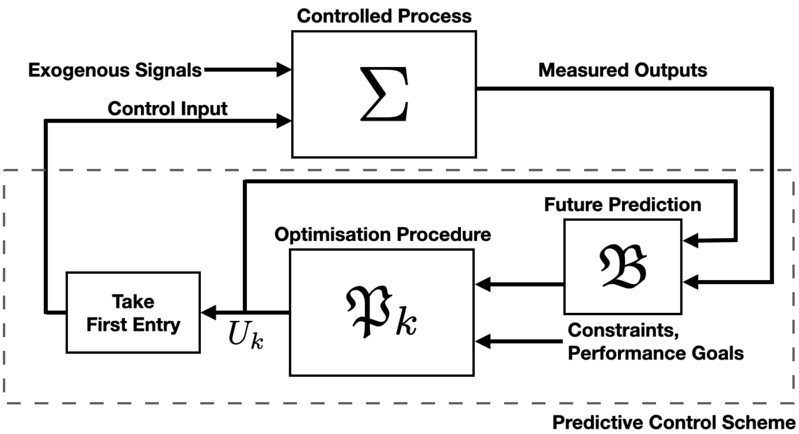


Figure 1.1: Block diagram: Implementation of an MPC algorithm.

Next, I present some examples in order to motivate the contributions presented in this thesis. Specifically, I aim in highlighting how MPC can, indeed, be an interesting control option for the several types of systems[7], given the fact that it allows optimal performances, from the viewpoint each sampling instant. These examples are presented with little mathematical details, bringing to focus the conveniences[8] of predictive control: it takes account the effect of measurements online, counter-acts the effect of uncertainties and disturbances by construction, and also allows the designer to explicitly consider constraints in the synthesis.

Example 1. *Consider a vehicular semi-active suspension system, as illustrated in Figure 1.2. A vehicle suspension comprises a controllable damper and a spring, connecting each wheel*

[6]In Chapter 2, a broader overview of the theoretical background on MPC is presented. This includes discussions on how to ensure closed-loop stability using terminal ingredients, optimality, and recursive feasibility of the MPC optimisation. As a side-note, I recommend the following references for the interested Reader: [Allgöwer and Zheng 2012; Grüne and Pannek 2017; Limon et al. 2018].

[7]Along this thesis, I have studied indeed MPC application for distinct system dynamics and descriptions.

[8]Furthermore, I highlight that MPC is, *a priori*, able to consider any kind of process description (linear, nonlinear, multivariable, etc). In practice, the only requirement is that the model predictions can be made solely based on the available measurements.

to the chassis; for a comprehensive debate on suspension system analysis, refer to [Savaresi et al. 2010]. In [Morato et al. 2018a], an MPC is tuned to minimise the influence of the road upon the comfort of the passengers within vehicle; the control actuator is the suspension damper, which varies stiffness along time to diminish chassis' vibration. The main advantage of using a prediction of the dynamics along a rolling horizon is that, at each instant, the effects of the future road disturbances are considered (a road preview is used), which makes the controller take anticipating measures to counter-act the effects caused on the chassis' vibration. Accordingly, Figure 1.3 shows a car chassis' acceleration behaviour when running over a bumpy road. These results consider the application of a fast MPC scheme, a static optimal controller (equivalent to an LQR approach), and no closed-loop law (i.e. passive damping behaviour using minimal or maximal damping). In practice, MPC is able to provide better comfort to the onboard passengers, making an efficient use of the suspension system. I note that the resulting performance with the MPC can be understood as better[9] because the variation of acceleration signal is smaller, smoother, and with reduced peaks, when compared to what is obtained with the LQR.

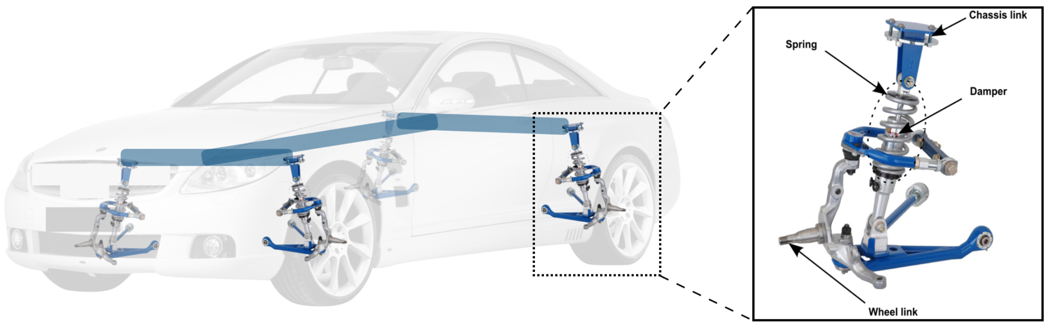


Figure 1.2: Example 1: Automotive suspension systems.

Complementary, I make reference to [Morato, Normey-Rico, and Sename 2021c], where a similar robust MPC scheme is tested for a suspension system against a robust LQR method, which also takes into account constraints (via saturation, clipping) and uncertainties. Figure 1.4 shows the corresponding chassis' and wheels' accelerations. Once again, the MPC approach is able to outperform the LQR method, providing subtler responses.

Example 2. *As another illustration of the application of MPC, I mention the recent works on planning social distancing measures for the COVID-19, e.g. [Morato et al. 2020c; Morato et al. 2020a]. In these papers, the optimisation problem $\mathfrak{P}_k$ includes the problem of minimising the amount of individuals infected by the SARS-CoV-2 virus, while also seeking to lighten social distancing guidelines and considering a hard constraint for the amount of hospitalised people (which should not saturate the availability for treatment). Thereof, the resulting control action ensures that a social health catastrophe does not happen (i.e. the heath system remains able to treat all infected individuals). Again, MPC is an interesting approach because, due to the future predictions, it acts by providing social distancing guidelines that act to revert an infection peak that is previewed to happen in the future. In Figure 1.5, I present some*

[9] This is merely a qualitative assessment. In the end of this Chapter, quantitative performance metrics are presented.

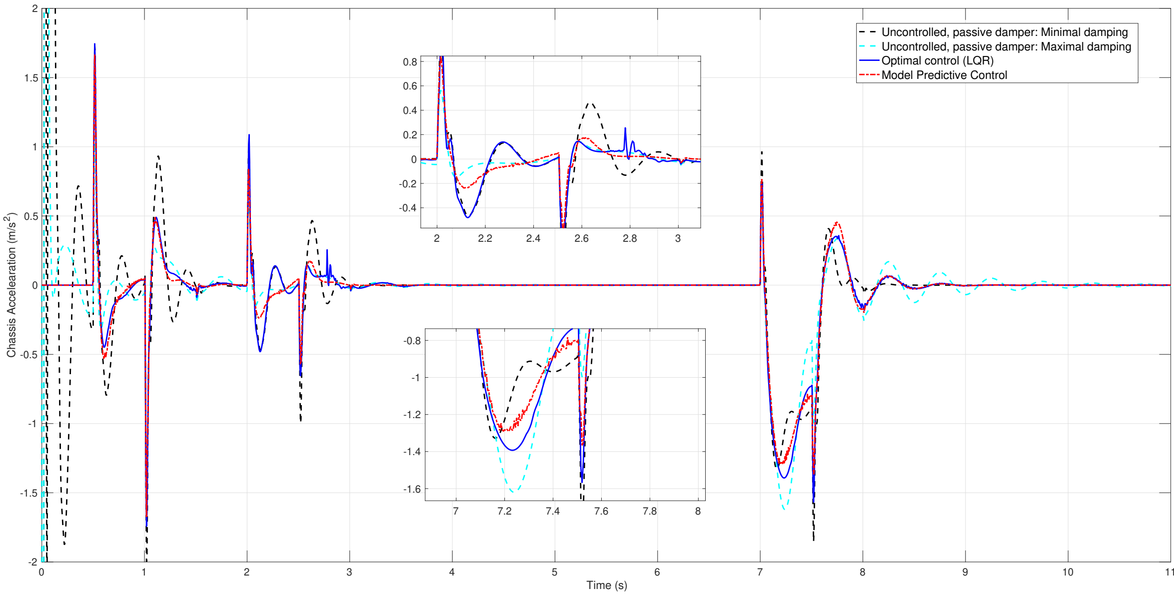


Figure 1.3: Example 1: Chassis' acceleration.

of the obtained results[10] *with regard to the number of active symptomatic infections. In red dashed plots, one can see the expected situation with no social distancing or control whatsoever (e.g. "Open-loop"), while in blue plots, the results obtained with the predictive control scheme. In light blue, the population response to social distancing is presented: this normalised curve represents the percentage of people distanced in the susceptible population set. In sum, the MPC policy acts to mitigate the number of infections, always maintaining this curve below the Intense Care Unit (ICU) threshold.*

MPC also has a significant interest in the context of managing renewable energy systems with multiple sources and carriers. The so-called energy management systems are often synthesised using predictive control algorithms, e.g. [Vergara-Dietrich et al. 2019; Morato et al. 2020e; Morato et al. 2021b]. In the following example, I present a brief insight in how such application operates, considering a real energy plant from Brazil.

Example 3. *Consider the renewable energy generation problem from the state of São Paulo, Brazil, as investigated and described in [Morato et al. 2020b]: three different generation units (microgrids) belong to the same distributer, but are physically apart, see Figure 1.6. These microgrids are powered by solar power and sugarcane biomass residues, which are locally burnt in boilers and generate energy through vapor-engine turbines. The owner sells the generated renewable energy outlet to the state distributer. These microgrids have the same baseline structure, but the biomass production at each site is different, and so are the efficiencies of the local units (boilers, turbines, solar panels, and so on). The energy systems are not so far apart, so one could transport the exceeding biomass from one site to another, if profitable. The main control problem is the following: how to maximise local renewable energy generation and*

[10]These results consider the data available in the early beginning of the COVID-19 pandemic in Brazil, around May, 2020. Thus, they are out of realistic scale.

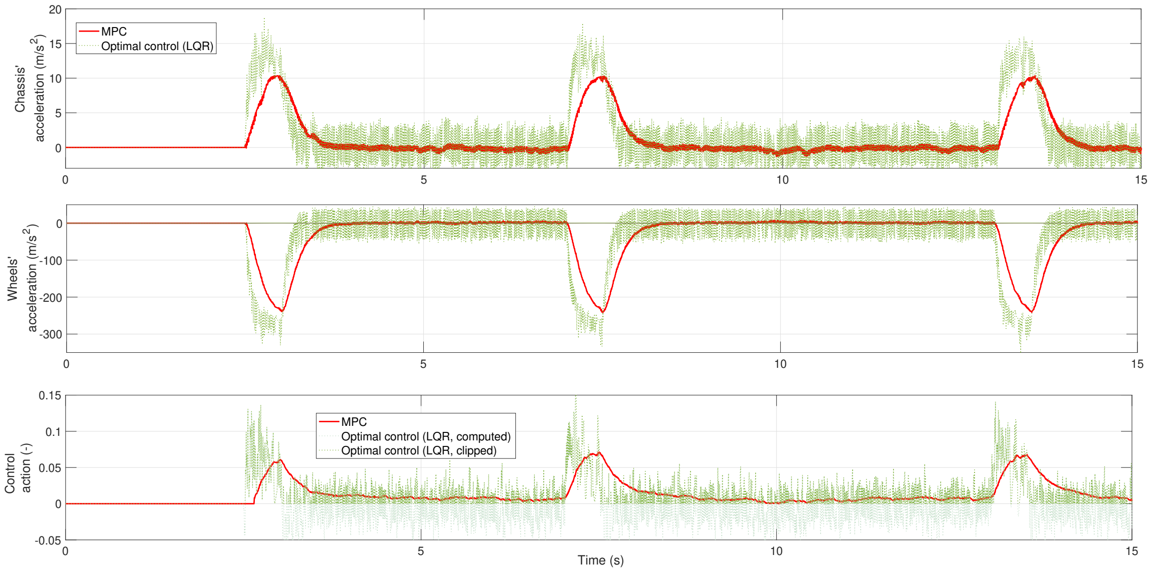


Figure 1.4: Example 1: Chassis' and wheel acceleration, control action.

the total profit from the three systems?

As detailed thoroughly in [Morato et al. 2020b], MPC is well-suited framework to address these kinds of multivariable control problems that economic concerns, feedforward compensation, and constraints. With MPC, there is no need to synthesise individual local controllers to avoid coupling effects, and the economic objective can be directly embedded to the optimisation. Moreover, the use of the future predictions directly embed feedforward to the controller, which acts to compensate future solar irradiance meteorological data that can be directly included in the optimisation. The economic constraints, as well as the physical constraints of each subsystem, can be also expressed directly as convex set operations in the MPC optimisation.

1.1.2 Research frontiers

The core principle of MPC relies on having an adequate process model in order to predict the future behaviour of state (or output) variables. If a trustworthy model is not available, the derived control law may simply be unrealistic and thereby the controller may be insufficiently robust to counter-act the uncertainties caused by the prediction mismatches (even stability may be lost, in some dramatic settings).

Therefore, the research on MPC has consistently debated what happens when imperfect process models are used in the optimisation. Accordingly, the theoretical properties closed-loop MPC schemes have been throughly investigated over the last decades. Next, I recap some of the recent research advances in the context of nonlinear and robust MPC syntheses.

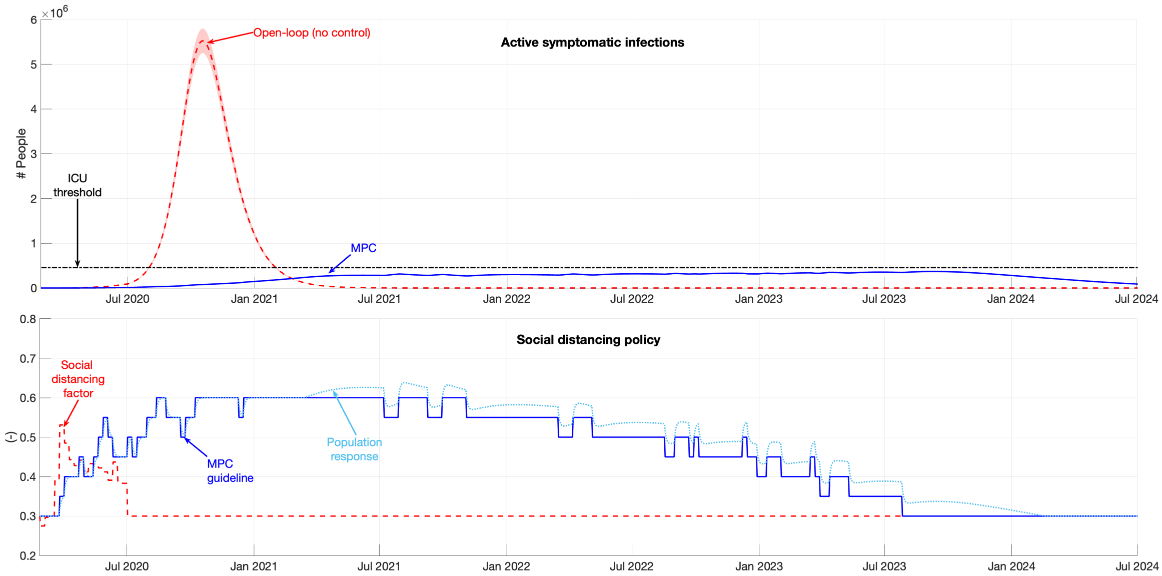


Figure 1.5: Example 2: MPC for the control of the COVID-19 pandemic in Brazil.

Nonlinear MPC

Nonlinear Model Predictive Control (NMPC) algorithms are of great relevance when nonlinear systems are controlled over larger operating conditions or when the process responses heavily depend on external parameters. However, the inclusion of nonlinear model predictions is not trivial and increases the resulting algorithm's complexity, see [Allgöwer and Zheng 2012]. The increased numerical burden becomes an impediment for some real-time applications. In order to exposit this complication, I present a stimulating illustration:

Example 4. *Recall the suspension system from Example 1.2, in which an MPC is tuned to control a semi-active damper in such a way to reduce vibration from the vehicle chassis and thus provide a more comfortable ride for the onboard passengers. In this system, due to the coupling effects and load transfers between vehicle corners, the complete vertical dynamics can only be accurately represented with a seven degrees-of-freedom nonlinear model, as elaborates [Nguyen 2016]. Complementary, the damper dynamics also present polynomial models with an adjustable number of damping coefficients*[11]*. Furthermore, these systems operate under very fast sampling periods, usually below* 5 *or* 10 ms *threshold.*

Thence, this control problem becomes quite intricate in practice, since the available time to compute the control law is reduced and the model is highly complex. For the application of MPC, these issues are even more elaborate given that the numerical complexity of the optimisation problem grows exponentially with the number of system states and the control horizon. In the application paper [Morato et al. 2018a], the MPC solution could only be achieved with a linearisation of the optimisation problem at each sampling instant. If the full nonlinear predictions were consider, the T_s *threshold was rapidly violated. As indicates [Nguyen 2016],*

[11] With more coefficients, the damper dynamics are better represented.

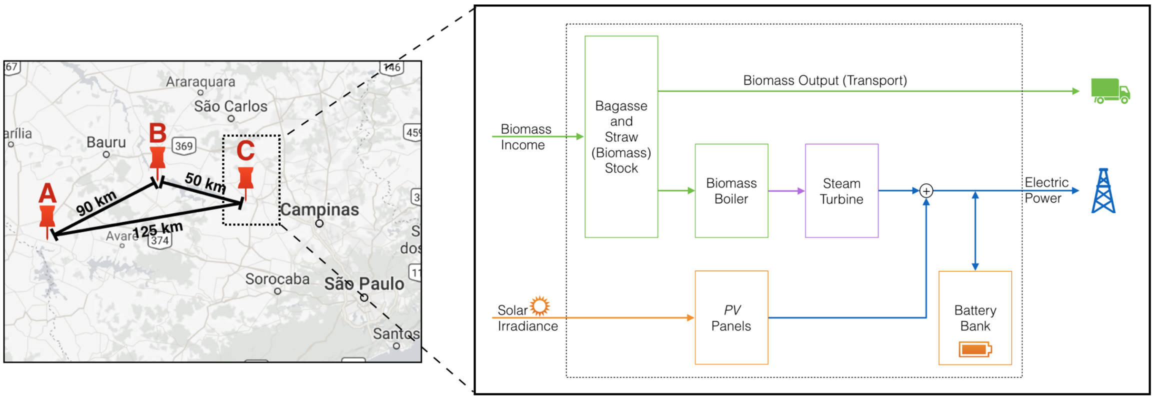


Figure 1.6: Example 3: Renewable microgrids in the state of São Paulo, Brazil.

considering over six damping coefficients in the damper model already led to an "explosion" of the mean computational time required for the MPC solution in an embedded vehicle test-bench, due to mixed integer constraints.

In order to alleviate the computational complexity of NMPC, the research on fast formulations and their theoretical guarantees has gained pace (and concreteness) over the last few years. Refer, for instance, to the survey [Gros et al. 2020]. One of the main aims of these investigation threads has been to develop solver-based solutions, which approximate the resulting nonlinear program by simpler programs (while maintaining the accuracy of the resulting control law). This is made viable mainly through real-time iteration methods, such as ACADO and CasADi [Houska, Ferreau, and Diehl 2011; Quirynen et al. 2015; Andersson et al. 2019], and Lagrange and gradient-based solutions, such as GRAMPC [Richter, Jones, and Morari 2011; Käpernick and Graichen 2014; Englert et al. 2019]. The main interest of these algorithms is that they allow for real-time implementation, thus serving for the control of nonlinear processes with (very) fast sampling rates.

The focus of this thesis is not the development of solver-based NMPC scheme as detailed the prior, but I mention them as benchmark references for the algorithms which I develop. The main advantage of these implementation-oriented approaches is that they are able to recover embedded MPC laws (for systems with relatively large order) within the millisecond range, which is rather impressive. Furthermore, they result in near-optimal performance with constraint satisfaction, close to what would be obtained with a "full-blown" NMPC implementation[12].

[12]Through the sequel, I refer to "full-blown" NMPC as nonlinear predictive control algorithms that consider full-order nonlinear predictions, without any kind of approximation. In this case, the resulting optimisation is a nonlinear program, whose complexity exponentially grows with the horizon size, the number of states, and the number of control inputs.

Robust MPC

Yet with great practical value, the standard (linear and nonlinear) MPC design lacks guarantees of recursive feasibility or closed-loop stability at the presence of disturbances. Therefore, in parallel to the research on fast NMPC, there has also been a large enthusiasm on robust MPC schemes with performance certificates, e.g. [Köhler et al. 2020; Santos and Cunha 2021].

Since the 00's, there has been a growth of research body on robust MPC techniques, i.e. those with focus on the achievement of optimal closed-loop performances, feasibility, and constraints satisfaction despite the effects caused on the outputs by unmeasurable disturbances or model uncertainties. The basic premise for the design of robust MPC schemes is that the uncertainties should be bounded[13] (and the bounds are known or, at least, estimated). Thus the performance certificates can be provided by robustifying the MPC optimisation with regard to the uncertainties' bounds. This can be done via min-max optimisation [Limón et al. 2006a; Löfberg 2012], constraints tightening [Köhler, Müller, and Allgöwer 2018; Santos et al. 2019], tube-based disturbance propagation [Yu et al. 2013; Abbas et al. 2019], and so forth[14].

Along this thesis, I provide developments on novel robust MPC schemes (presented in Part III), considering the use of linear parameter varying models. Synthetically, the main concept is that an adapted optimisation problem is solved, which is irrespective of the uncertainty terms, but rather written in terms of its bounds, thus ensuring certificates. Next, I provide a brief example to illustrate this concept[15].

Example 5. *Consider an LTI system whose behaviour is given by: $y(k+1) = ay(k) + bu(k) + w(k)$, being u the control input, y the controlled output, and w a bounded disturbance term such that $\|w(k)\| \leq \overline{w}, \forall k$. Then, consider the application of an unconstrained MPC, whose control law is found by solving $min_{U_k} \overbrace{\sum_{i=1}^{N_p} \|y(k+i) - y_r\|}^{J_k}$, where y_r is an output set-point target. By replacing the model in J_k, one obtains $\min_{U_k} \sum_{j=0}^{N_p-1} \|ay(k+j)+bu(k+j)+w(k+j)-y_r\|$, which cannot be solved since $w(k+j)$ is unknown. Yet, since this term is bounded, a robust solution can benefit from the known bounds. Consider that $J_k \leq \overline{J}_k$, being $\overline{J}_k = \max_{\|w(k+j)\| \leq \overline{w}} J_k$. Thus, it follows that the minimiser $U_k^\star = \arg\min_{U_k} \overline{J}_k$ is also a minimiser to J_k for all $\|w(k+j)\| \leq \overline{w}$. By exploiting this property, the performance certificates are maintained.*

[13]Here, uncertainties encompass any kind of model discrepancy as well as load disturbances. Boundedness is a quite bland assumption, since, in the majority of real processes, bounds can be directly obtained from physical characteristics.

[14]There are other kinds of robust MPC approaches, but these are some of the most interesting ones, due their implementation simplicity and academic acceptance.

[15]In the following example, I provide an illustration of an MPC scheme which is robust against (bounded) disturbances, and not model uncertainties. In many cases, the term robustness is used to refer to the ability of a given controller to tolerate model uncertainties.

1.2 A brief background on Linear Parameter Varying systems

Next, I bring focus to the second main topic of this thesis: Linear Parameter Varying (LPV) systems. In parallel to the establishment of modern MPC schemes, there grew a concerted scrutiny on robust control, including structured robustness analysis and the generalisation of the Small Gain Theorem, e.g. [Doyle, Wall, and Stein 1982; Scherer 2006]. Accordingly, from the fabric of robust control theory, the LPV system framework was developed [Mohammadpour and Scherer 2012; Sename, Gaspar, and Bokor 2013]. This analysis and control toolkit has become popular and widely used to handle processes with complex (nonlinear, time-varying) dynamics, as debated in [Tóth 2010; Hoffmann and Werner 2014].

In synthesis, one can understand LPV systems[16] as a **special** class of nonlinear processes, which are well suited for the control of dynamics with scheduled parameter variations. In some sense, LPV systems are found somewhere *in between* the nonlinear and LTI formalisms[17], since they are linear in the state space, while nonlinear in the parameter space. Differently than in the LTI case, the state and output transitions depend on time-varying parameters, which are called the **scheduling variables**, denoted ρ. These parameters are assumed to be **known**[18] and **bounded**, and thus they can be used for control.

Consider a generic system Σ, whose behaviour is given by:

$$\left\{ \begin{array}{rcl} x(k+1) & = & A(\rho)x(k) + B(\rho)u(k)\,, \\ y(k) & = & C(\rho)x(k) + D(\rho)u(k)\,, \end{array} \right. \tag{1.2}$$

where x represents the vector of system states, u gives the vector of control inputs, and y stands for the vector of controlled outputs. We stress that $k \geq 0$ denotes the discrete-time stamp and that $A(\cdot)$, $B(\cdot)$, $C(\cdot)$ and $D(\cdot)$ matrix functions given with respect to the vector of scheduling parameters ρ.

According to the nature of the model parameters ρ and the dependency of the matrices, Eq. (1.2) may represent different kinds of dynamics: (a) if the parameters are static (constant and time-invariant), then model is LTI; (b) if ρ is a function of time, e.g. $\rho(k)$, then the model stands for Linear Time Varying (LTV) dynamics; (c) when the parameters are time-varying, yet **measured** and **bounded**, the model becomes LPV. In the LPV philosophy, the designer is not interested in the explicit dependency of the state matrices on time, but rather on the dependency of these matrices on the scheduling parameters themselves. Then, the LPV analysis and design methodology arises by exploiting the form of these bounded parametric-dependencies.

[16] In some references, LPV systems are referred to as "Takagi-Sugeno" systems [Takagi and Sugeno 1985]. In fact, there are some analogies and discrepancies between these categories. A debate on this topic is presented in [Rotondo, Puig, and Nejjari 2016; Rotondo 2017]. For simplicity, in this thesis, I use only "LPV" as nomenclature.

[17] As a major advantage, I stress that the theoretical analyses of LPV systems often fall under the scope of an LTI-alike toolkit. Of course, there is more complexity involved in these analyses than in the classical LTI case, but with adequate transformations and parameter-dependent operations, the toughness is alleviated, e.g. [Szabó and Bokor 2018]. Again, I recall that thorough theoretical details are presented in Chapter 2.

[18] Their current values $\rho(k)$ should be either measured online, or estimated with precision. Note that, generally, their future behaviour $\rho(k+j)$, $j \geq 1$ is not known.

Remark 1. *In many references, a model is named LPV whenever (c) is true [Shamma 2012]. Nevertheless, along this thesis, there is an explicitly differentiation between "pure" LPV models to quasi-LPV (qLPV) ones: (i) in the qLPV setting, it is considered that the parameters are* ***endogenous****, i.e. the scheduling variables are function of system variables, e.g.* $\rho(k) = f_\rho(x(k), u(k))$ *(this function is named the "scheduling proxy"). In this case, if the scheduling proxy is re-introduced to the qLPV model, a full-blown nonlinear model is obtained; and (ii) "pure" LPV models are those with* ***exogenous*** *parameters, which are, a priori, standard functions of time (and independent of the process variables).*

In general lines, the most relevant feature of LPV techniques, in the context of control synthesis, is that they provide a systematic design procedure for self-scheduled multivariable controllers, benefiting from the availability of the scheduling parameters from the process. The LPV methodology, therefore, allows performance and robustness to be integrated into an unified framework, and this is why the topic is of great scholastic relevance.

1.3 Bridging the gap

The LPV toolkit can indeed be used to represent nonlinear dynamics with trustworthiness. Thus, the control synthesis and analyses of nonlinear features become easier when exploiting the availability of known scheduling variables and the corresponding LPV dynamics. Nowadays, LPV control synthesis is standard using fractional transforms [Casella and Lovera 2008], H_2 and H_∞ performance goals [Sename, Gaspar, and Bokor 2013; Emedi and Karimi 2016], as well as for tracking and rejection [Scorletti, Fromion, and De Hillerin 2015]. Nevertheless, the case is definitely not true for predictive control applications. The study of LPV MPC schemes properly began in the mid-00's. Anyhow, there are still some intricate gaps to be further investigated on this topic, as pointed out in [Bachnas et al. 2014] and in our dedicated survey [Morato, Normey-Rico, and Sename 2020a].

Since nonlinear mappings can be re-cast as LPV models, it seems natural to develop NMPC algorithms by exploiting LPV realisations. It is especially interesting to use LPV models in the synthesis of MPC because these representations retain the linearity property along the inputs-outputs channels, which means that computationally efficient design procedures can be rendered. Conversely, this means that the drawbacks of full-blown NMPC algorithms are avoided (the use of nonlinear programs), without any need to approximate the solution of the optimisation problem (as do the most modern fast NMPC solutions, such as real-time iteration and gradient-based methods, for instance).

In the context of MPC, a full-horizon prediction model is required (i.e. to describe the system variables along the future N_p steps ahead of each sample). Nevertheless, when an LPV prediction model is used, this problem depends not solely on the future inputs (to be determined by the optimisation), but also on the future scheduling parameters $\rho(k+i), \forall i \in \mathbb{N}_{[0,N_p-1]}$, which are assumably **unknown**.

Consider an LPV model as in Eq. (1.2), with an initial condition $x(k)$(current state), under the assumption that the states are measured at each time step k. When this model is plugged to an MPC algorithm, the optimisation has to internally elaborate the following sequence of predictions:

1. $x(k+1|k) = A(\rho(k))x(k) + B(\rho(k))u(k)$,
2. $x(k+2|k) = A(\rho(k+1))A(\rho(k))x(k)$
   $+A(\rho(k+1))B(\rho(k))u(k|k)B(\rho(k+1))u(k+1|k)$,

and so forth, up to the N_p-th prediction. Notably, these predictions implicitly require the knowledge of so-called scheduling trajectory ($\rho(k)$, $\rho(k+1)$, up to $\rho(k+N_p-1)$). However, these variables are unavailable at each instant k, when only $\rho(k)$ is known. Due to this unavailability issue, the design of MPC using LPV models get especially complicated, essentially because recursive feasibility and closed-loop stability of the strategy theoretically require the MPC to tolerate the uncertainties that arise due to the unavailability of the future scheduling trajectory.

Problem statement: Thus, with respect to the discussed context, the main problem investigated in this thesis is:

How to conceive (accurate) predictive control algorithms for LPV systems without the actual knowledge of the future scheduling trajectory?

1.4 State-of-the-art: Available techniques

There has been some recent works on MPC design for LPV systems. In our review paper [Morato, Normey-Rico, and Sename 2020a], a detailed overview of the state-of-the-art is presented. Here, for brevity, I recap some of the most relevant techniques, which can be categorised into two main groups:

- Robust methods, e.g. [Jungers, Oliveira, and Peres 2011; Rakovic et al. 2012; Bumroongsri 2014; Hanema, Lazar, and Tóth 2020], which consider the worst-case closed-loop performances implied by the unknown future scheduling parameters. Accordingly, the optimisation is re-written in order to take into account the bounds of all possible future parameter variations, which can render usually conservative results.
- Gain-scheduling methods, e.g. [Ayala et al. 2011; Brunner, Lazar, and Allgöwer 2013; Mate et al. 2019; Alcalá, Puig, and Quevedo 2019]. In these papers, the LPV model is replaced, at each sampling instant, by an LTI model (or a sequence of LTI models) based on a guess for the scheduling trajectory. In many cases, this guess is simply an assumption that the scheduling parameters will remain constant along the prediction horizon. While these methods operate quite fast (they exhibit reduced numerical burden), sub-optimality may be implied. Nevertheless, when the scheduling trajectory is

accurate (as seen for the qLPV case in [Cisneros, Voss, and Werner 2016; Cisneros and Werner 2017b; Cisneros and Werner 2019]), an exact nonlinear MPC solution is obtained by the means of quadratic optimisation programs, thus rendering a solution comparable to state-of-the-art solver-based NMPC solutions (such as ACADO and CasADi).

1.4.1 Investigation gaps related to the problem

Although there exist, nowadays, generalised NMPC formulations, the synthesis of MPC algorithms using the LPV formalism is of utter academic and practical interest, given the fact that the linearity property can be exploited in such a way that the resulting algorithm has relieved numerical complexity. In consonance with the works that have already been proposed regarding this topic, as recapitulated in the prequel, there are still some investigation threads available for further research:

- Firstly, I stress that LPV formulations in the input-output (IO) form, although being supported by strong theoretical holds in the sense of identification, e.g. [Bachnas et al. 2014], have only a handful of control synthesis counterparts. The vast majority of LPV control synthesis (including MPC) is settled for state-space descriptions. Nevertheless, industry is much more prone[19] to accept IO formulations, as discusses [Froisy 2006]. Therefore, the bridge between theory and industrial applications for the case of LPV MPC design will be further sustained when a theoretical research body IO formulations becomes available. Despite this topic being fundamental and promising, there exist rather few papers that elaborate upon it, e.g. [Abbas et al. 2015; Abbas et al. 2016], thus lacking further assessments.
- In the context of gain-scheduled LPV MPC methods, there exists a lack on further exploitation of performance certificates of such algorithms. The use of the LPV parameteric-dependency on the stability arguments and its implications have only briefly evaluated, i.e. in [Cisneros and Werner 2017b; Cisneros and Werner 2020].
- As argued, gain-scheduled LPV MPC algorithms with accurate scheduling trajectory estimate are real-time capable and altogether comparable to the modern NMPC techniques. Nevertheless, the exploitation of the scheduling function (of qLPV models) to generate scheduling trajectory estimates with accuracy (and thus synthesise an LPV MPC with optimality close to that of an NMPC) has only been briefly evaluated, lacking further proofs in the sense of theoretical holds[20].
- Even though the application of robust MPC schemes for LPV systems is rather settled, recent works, e.g. [Cisneros and Werner 2018], have pointed out how dissipativity theory

[19] Just to give an example: the original GPC algorithm [Clarke, Mohtadi, and Tuffs 1987] has wide acceptance in many real-world applications due to the use of a simple IO prediction model. This issue is explicitly mentioned in [Darby and Nikolaou 2012], for instance.

[20] The majority of papers, e.g. [Cisneros, Voss, and Werner 2016; Cisneros, Voss, and Werner 2016; Cisneros and Werner 2019], only provide empirical and practical results, deficient stronger theoretical holds on the accuracy of the scheduling trajectory estimates and the corresponding convergence. The only theoretical assessments on these properties is seen in [Hespe and Werner 2021].

can be used to smooth the online toughness of the resulting optimisation. Yet, a more thorough debate on this matter, with corresponding application results, can still be provided.

1.5 Thesis' objectives

Over the last decades, there has been a growing interest in the development of fast MPC algorithms for nonlinear systems. Accordingly, the linear parameter varying toolkit arises as an interesting alternative to address this topic without the necessity of any solver-based solution, since embeddings or scheduled realisations can be used to describe nonlinear dynamics with exactitude (or good precision).

Nevertheless, the application LPV MPC schemes requires the knowledge of the future scheduling trajectory, which is typically not available. The state-of-the-art, and the corresponding research gaps, point out to two alternatives to handle this matter: (*i*) robust synthesis; and (*ii*) gain-scheduled design based on trajectory estimates. Each one of these branches has certain limitations and advantages, but they are all of philosophical and theoretical enthusiasm.

Therefore, bearing in mind this thesis' problem statement (Section 1.3), and considering the available state-of-the-art (Section 1.4.1), the objectives of this thesis coincide, in majority, with the available investigation threads within this novel field. For the sake of presentation thoroughness, I consider three main objectives, which are corroborated by some specific goals. Below, I list and discuss them in detail, being the items marked in roman numbering the primary objectives, and the ones marked with capital letters, the specific ones:

(i) **Provide new algorithms in order to estimate the future LPV scheduling trajectories, from the viewpoint of each sampling instant, envisioning accurate predictions for the MPC.**

  (i.A) Ensure relieved computational burden of these tools, making them simpler to be implemented than methods based on the scheduling proxy (sequential QPs), i.e. [Cisneros and Werner 2020].

  (i.B) Demonstrate convergence properties of the developed methods with regard to the true scheduling sequence.

(ii) **Develop novel gain-scheduled predictive control algorithms with recursive feasibility and stability certificates.**

  (ii.A) Consider the solution using a state-feedback formulation, corroborated by contracting terminal constraints.

  (ii.B) Provide the corresponding formulation for the dynamic output-feedback case, when state measurements are not available, considering input-output descriptions.

(iii) **Propose robust predictive control algorithms for LPV systems, with fast computation during the implementation, capable for embedded, real-time applications.**

(iii.A) Consider tracking formulations with robust constraint satisfaction certificates for piece-wise constant reference signals (with possibly unreachable values).

(iii.B) Consider robust formulations based on dissipativity arguments, thus not requiring the use of terminal ingredients and relieving the overall computational stress of the resulting online implementation.

1.6 Methodology and applications

The scientific contributions that derived from this thesis, which relate to the aforementioned objectives, are listed in Appendix A. Furthermore, I stress that this doctoral was performed under rigorous scientific, bibliographical, and documental research methodology in order to address the thesis' goals. For each one of the main (and corresponding specific) objectives listed in the prequel, looping phases of specifications, design and empirical essays were conducted.

Complementary, since one cannot disassociate the conducted research from its corresponding researcher, a person with individual subjectivities, I ponder that, due to personal and peculiar reason, and also due their scientific and societal relevance, many of the proposed MPC algorithms were focused on the control of two specific classes of complex systems: (i) renewable energy generation, and (ii) urban mobility technologies. The two main issues that appear with the application of the predictive control for these processes is how to handle the model nonlinearities and the increased digital complexity of the resulting control law, which becomes numerically too though to be implemented in real-time, under strict sampling period thresholds. To address these concerns, the LPV formalism provides a well-suited set of tools, as previously debated.

Since renewable energy processes, e.g. [Pipino et al. 2020b; Bernardi et al. 2021], are, in general, nonlinear systems. Thus, they can be directly represented by LPV structures using appropriate embeddings. The major control concern is to maximise renewable energy efficiency, and thus reduce greenhouse gas emissions and the use of fossil fuels. For this, LPV predictive control algorithms are adequate solutions, because they are able to incorporate meteorological data in the predictions and adequately enhance the resulting performances.

Autonomous vehicles are at the center of societal concerns, given the importance of the topic of modern urban mobility. These systems can often be represented using LPV models, specially when considering vertical dynamics, as detailed in [Savaresi et al. 2010; Morato et al. 2018a; Morato et al. 2019b; Morato, Normey-Rico, and Sename 2021c]. Moreover, they are controlled by the means of embedded microcontrollers that generate new control actions each $1 - 10$ milliseconds, which makes "full-blown" nonlinear predictive control, in much, impractical. In this sense, the LPV MPC approach arises as a computationally affordable alternative.

I highlight that these two topics are at the heart of the 2030 Agenda of the United Nations [United Nations General Assembly 2015; United Nations 2018], which proposes the Sustainable Development Goals (SDGs) in order to prevent a social-environmental catastrophe of severe magnitude and global scale [Löwy 2015]. I will not address these topic with much focus, since it is discussed in other side works, e.g. [Morato et al. 2018b]. Nevertheless, I stress that SDG 7 concerns clean and affordable energy generation [Nathwani and Kammen 2019], while SDG 11 addresses sustainable cities and communities with cohesive and universal mobility [Hermelin and Henriksson 2022]. Therefore, it seems evident to me how these topics are of social relevance and, thus, also enlarge the possible significance of this doctoral work.

As a last comment regarding the considered applications[21], I note that results were provided considering other kinds of systems as well, such as industrial processes (e.g. continuous stir tank reactor, Chapter 3), robotic systems (e.g. an inverted pendulum, Chapter 3), toy, education-oriented benchmark systems (e.g. cascaded tanks, Chapter 5), and solar-thermal collector plants (temperature tracking, Chapter 6).

1.7 Outline

In order to conclude this introductory debate, I present the overall structure of this thesis: I organised this document in a rather particular manner[22], dividing it into four parts. I have chosen to partition it in this way in order to address each of the primary objectives listed in Section 1.5 in an individual part. The first part comprises the introduction, preliminaries, and basic developments, the two intermediate parts are concerned with the primary objectives, while the last part recalls and discusses all the work provided herein.

The Reader should not necessarily read this thesis according to the progressive arch. Although there is a crescent in unfolding (especially from Part II to III), I have sought to establish each Chapter in such a way that it can be read individually and out of order, without having its meaning lost from the others, bearing in bind the discussions presented in this Introduction and the overall preliminaries presented in the next Chapter[23]. In Figure 1.7, a suggested reading scheme is presented, where the arrows indicate the possible order alternatives.

[21] I specifically note that the vehicle essays in simulation are generated using the full nonlinear models from [Savaresi et al. 2010]; solar collector plate dynamics are represented with the nonlinear partial-differential equations from [Torrico et al. 2010]; cascaded tank systems are represented with the adjustable nonlinear model from [Johansson 2000]. In the sequel, when other specific models are used, I explicitly mention it. All simulation results presented in along this thesis were conceived using the mathematical software Matlab, in consonance with toolbox Yalmip. The majority of the results were obtained from a 2.4 GHz, 8 GB RAM Macintosh computer, unless mentioned otherwise. The optimisation solvers were Gurobi for quadratic problems, fmincon for nonlinear programs, and SDPT3 for the solution of linear matrix inequalities.

[22] The Reader can understand these parts as "big chapters", since all Chapters that comprise a Part basically discuss the same main overall topic. My aim with this partition is to present this thesis' work as a "geometric" progression. In such a way, I pursue the scientific arch into a somewhat progressive order.

[23] For Readers previously aware of the relevant results on MPC and the advances on LPV systems, I recommend skipping Chapter 2, which basically recalls the background concepts and definitions, and thus moving directly on to Part II.

I further detail the partitioning of this thesis and content of each Chapter:

- Part I: "Preamble":
  - In the rest of the Preamble, i.e. Chapter 2, I discuss the preliminaries of all that follows, establishing the theoretical background of this thesis, recalling the basic concepts, definitions and results from the literature. I begin this Chapter by presenting an unified LPV representation, characterising basic notions, such as stability, system classes and so forth. I also detail the corresponding used norms and signal spaces, including some handy LMI synthesis procedures, operations and transforms. Then, I detail the embedding procedures that can be used to generate LPV and qLPV models (using differential inclusions), and discuss some properties of this representation feature. Concerning the application of MPC for these systems, I give further details on the relevance of synthesising such algorithms, discussing the main complications that arise. Accordingly, I recap the keystone results on recursive feasibility and stability of the resulting closed-loop, which serve to generate the so-called optimisation "terminal ingredients", as well as presenting the basic notions of dissipativity.
- Part II: "Gain-scheduled formulations". In this portion, I address the topic of gain-scheduled predictive control approaches, targeting Objectives (i) and (ii) of this thesis. Specifically, the following topics are debated:
  - As previously debated, the application of LPV MPC algorithms becomes rather complicated because the scheduling trajectory, along the MPC prediction horizon, is *a priori* unknown. Thus, in Chapter 3, I discuss the available strategies that can be used to generate estimates for the scheduling trajectory, and the advantages (and disadvantages) that arise with each approach, within the context of MPC. Furthermore, I present a novel extrapolation algorithm that estimates the future values of these qLPV scheduling parameters for a fixed prediction horizon of N_p steps. The method is derived from a simple Taylor expansion, and sufficient conditions are presented for convergent estimates (thus addressing Objective (i)). Moreover, using different simulation benchmark examples from the literature, I illustrate and discuss the how the different estimate approaches can be applied in practice and compare them.
  - In Chapter 4, I develop two different gain-scheduled predictive control algorithms formalised using state-feedback and output-feedback structures, which concerns Objective (ii):
    - First, I propose a control scheme for the enhancement of the comfort of on-board passengers in a vehicle with semi-active suspensions, within the MPC formalism. For such, the vertical dynamics of the car modelled in a qLPV setting and, thus, the controlled arises from the solution of a set-constrained optimisation, which embeds a comfort performance indexes. The method is sub-optimal because the synthesis considers a frozen scheduling trajectory approach. Anyhow, assuming bounds on the variation rates of the scheduling

parameters, the method enables a replacement of the original complex nonlinear optimisation by a much simpler quadratic program, which comprises a Lyapunov-decreasing cost and set-based terminal ingredients. Successful realistic nonlinear simulations of a one-fifth-scaled car with electro-rheological suspensions are presented, for which the proposed method is tested and compared with other optimal controllers. The results illustrate the overall good operation of the vehicle; the comfort of the passengers is substantially improved, as measured through time- and frequency-domain indexes.

∗ Secondly, I develop an output-feedback gain-scheduled approach for nonlinear systems represented under input-output qLPV models. For such, I detail how the input-output model can generate a prediction based on a future scheduling trajectory guess. Then, I demonstrate the asymptotic stability of the closed-loop system (and corresponding output tracking establishment). The method includes integral action for each input-output channel, thus ensuring tracking with null steady-state error. Using a numeric simulation benchmark, I demonstrate the effectiveness of the solution.

- Part III: "Robust synthesis". In this portion of the thesis, my major concern is to present novel robust predictive control algorithms for LPV systems, i.e. Objective (iii). Accordingly, I consider formulations with both terminal ingredients and dissipativity arguments used to ensure recursive feasibility and input-to-state stability (the formulations are under state-feedback representation). The issue of tracking piece-wise constant reference signals is also addressed. Specifically, these topics are structured as follows:

  – In Chapter 5, I present a Tracking NMPC formulation for piece-wise constant reference signals using qLPV embedding and scheduling trajectory extrapolation. The proposed framework is able to avoid feasibility losses due to large set-point variations, which are tracked thanks to an artificial feasible target variable, whose distance to the real set-point is minimised through an additional offset cost. At each sampling period, an optimisation problem is solved based on linear (scheduled) predictions; the average numerical toughness is comparable to a quadratic program. Robust constraint satisfaction is achieved with zonotopes that propagate the uncertainty. These sets are computed with respect to the one-step-ahead bounds of the qLPV scheduling sequence estimation error, offering reduced conservatism. Closed-loop stability and recursive feasibility are provided with robust parameter-dependent terminal ingredients. In order to illustrate the performances of the method, I provide a benchmark example, which demonstrates that the algorithm is indeed able to ensure reference tracking with reduced numerical demand when compared to state-of-the-art techniques.

  – Benefiting from the recursive extrapolation algorithm from Chapter 3 and dissipativity theory, in Chapter 6, I provide a robust MPC method that is able to fasten the sluggish performances achieved with the robust schemes from the literature by incorporating the bounds on the estimation errors of the scheduling parameters. Complementary, dissipativity arguments are used in order to demonstrate recursive feasibility and stability of the closed-loop system, demonstrated through the solution of a linear matrix inequality remedy, which determines the zone of attraction for which input-to-state stability is ensured. I consider the nonlinear temperature regulation problem of a flat solar collector as case study. By the means of realistic numerical simulation, I compare the proposed method to other robust algorithms from the literature, demonstrating it as an interesting alternative, with fast computation and relieved numerical burden.

- Part IV: "Closure":
  - In Chapter 7, I present an overall panorama of my doctoral work, where the objectives are recalled and compared to the presented advances and obtained results. A general overview of all that was developed in this work is recalled and the developments are individually analysed in terms of advantages and limitations. In this last portion of the manuscript, I also tease on the perspectives of the open investigation threads mention in this Introduction, shining a light on paths that are yet to be pursued in future research.

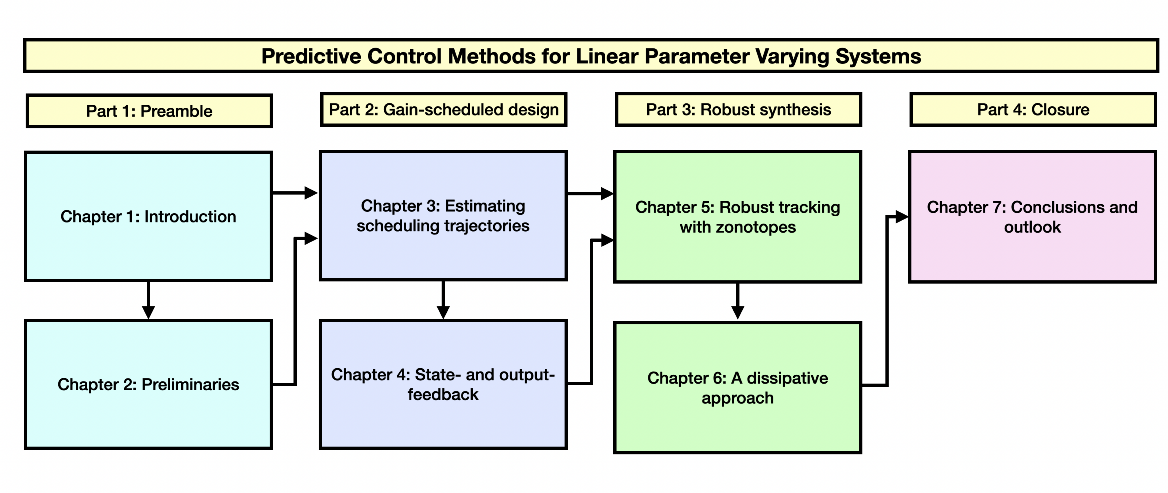


Figure 1.7: Thesis structure.

1.8 Notation

The basic notation used along this thesis is emphasised next. Along the remainder of this work, at the moments that specific complementary notation is used, I explicitly mention it.

- The j-sized identity matrix is I_j, while I_j; $I_{j,\{i\}}$ gives its i-th row. $\mathbf{1}_{n\times m}$ stands for the $n \times m$ vector of unit entries. The convolution product is represented using $\circledast$.
- col$\{\cdot\}$ denotes the column vectorisation (collection) of the entries and diag$\{v\}$ denotes the diagonal matrix generated with the line vector v. $v(k+j|k)$ gives the predicted

value for $v(k+j)$ computed at instant k. An interval matrix $\mathbf{J} \in \mathbb{I}^{n\times m}$ has mid($\mathbf{J}$) and rad($\mathbf{J}$) denoting its middle point and radius, respectively.

- The set of nonnegative real numbers is denoted by $\mathbb{R}_+$, whilst the set of nonnegative integers (including zero) is denoted $\mathbb{N}$. The index set $\mathbb{N}_{[a,b]}$ represents $\{i \in \mathbb{N} \,|\, a \leq i \leq b\}$, with $0 \leq a \leq b$. The set of real compact intervals is given by $\mathbb{I} = \{[a,b], a, b \in \mathbb{R}, a \leq b\}$. For sets defined within normed vector spaces, with define the Minkowski set addition as follows: $A \oplus B := \{a + b \,|\, a \in A\,,\, b \in B\}$, being A and B two sets. Accordingly, the Pontryagin set difference is given by: $A \ominus B := \{a \,|\, a \oplus B \subseteq A\}$. Furthermore, a linear mapping is given by : $R\mathcal{A} = \{y \in \mathbb{R}^n : y = Ra, a \in \mathcal{A}\}$, while the Cartesian product holds as $\mathcal{A} \times \mathcal{C} = \{z \in \mathbb{R}^{n+m} : z = (a^T \; c^T)^T, a \in \mathcal{A}, c \in \mathcal{C}\}$. I call $\mathcal{B}^m_\infty = \{\xi \in \mathbb{R}^m : \|\xi\|_\infty \leq 1\}$ the unitary m-dimensional box.

1.9 Performance indexes

Along this work, in order to quantitatively assess performances of the detailed control algorithms, I use the following indexes:

(i) Integral of the absolute error (IAE), i.e. $\sum_k \|e(k)\|$;

(ii) Root mean square (RMS), i.e. $\sqrt{\frac{1}{k_N}\sum_{k=0}^{k_N}(e(k))^2}$;

(iii) Normalised RMS (NRMS), i.e. $\frac{\text{RMS}\{e(k)\}}{\max_k \|e(k)\|}$;

(iv) Total control variance (TV), i.e. $\sum_k \|\delta u(k)\| = \sum_k \|u(k+1) - u(k)\|$.

Note that smaller IAE, RMS and NMRS indexes indicate better regulation (or tracking). Furthermore, smaller TV values mean better (smoother) control laws.

Chapter 2

Preliminaries

The background concepts that anchor the developments of this thesis are laid out in the sequel. Since this work is concerned with the study of predictive control algorithms applied for linear parameter varying systems, we provide the basic framework for the analysis and exploitation of these processes, as well as the tools required for the corresponding control synthesis. This Chapter focuses on a presentation for uninitiated readers, considering the key concepts that are referred to along this thesis.

As per referencing, we stress that all concepts related to dynamic systems can be found in [Khalil and Grizzle 2002], those related to robust stability in [Skogestad and Postlethwaite 2007], and those related to optimisation and LMIs in [Boyd et al. 1994; Scherer and Weiland 2000]. The notions and background on MPC are extensively discussed in [Allgöwer and Zheng 2012; Camacho and Bordons 2013], and also in the following Ph.D. thesis, which focus on stability issues in MPC: [Marruedo 2002], [Ferramosca 2011], [Santos 2011], and [Köhler 2021]. The main notions regarding LPV systems and the correlated stability issues are presented in [Mohammadpour and Scherer 2012; Sename, Gaspar, and Bokor 2013]; further details can also be found in the following Ph.D. thesis: [Briat 2008] and [Rotondo 2016].

This Chapter is organised as follows: first, the concept of LPV systems is re-introduced, with details given regarding representation forms and classes. Then, we discuss how to infer on the stability of these systems, via Lyapunov arguments and dissipativity analyses. Consequently, we detail how invariant control sets can be computed and generated for nonlinear and LPV processes. In the sequel, we get back to the topic of MPC, presenting how these control algorithms can be generated to induce closed-loop stability and a recursively feasible optimisation. Finally, robust stability certificates for MPC are detailed.

Remark 2. *Before any detail is presented, the Reader is invited to refer to Appendix B, where the core mathematical background is presented, in the sense of the used vector spaces, Algebrae, norms and (particular kinds of) functions. The background on Linear Matrix Inequalities (LMIs), along with useful lemmas for their application, are also presented therein.*

2.1 (Re)-Introducing LPV systems

In this work, we consider different design and analysis methods for discrete-time Linear Parameter Varying systems. Seeking notation consistency, throughout the sequel, we only take into account discrete-time dynamics due to the fact that this research is devoted to the study of discrete-time MPC laws applied to LPV systems, and these control laws are given in discrete-time.

In synthesis, LPV models are used to represent dynamical systems with linear dependencies on parameters with time-varying behaviours. As previously explained, we call these variables the **scheduling parameters**, and they are considered to be bounded and measurable.

Definition 2.1
A given causal system, with discrete-time dynamics marked by sampled time stamps $k \in \mathbb{N}$, is said LPV if its dynamics are linearly dependent on scheduling parameters $\rho(k) \in \mathcal{P}$, $\forall k \geq 0$. We call $\mathcal{P} \subseteq \mathbb{R}^{n_\rho}$ the scheduling set (which is usually convex).

In state-space realisation, a generic LPV system Σ has its dynamics described by the following set of equations:

$$\begin{cases} x(k+1) &= A(\rho(k))x(k) + B(\rho(k))u(k)\,, \\ y(k) &= C(\rho(k))x(k) + D(\rho(k))u(k)\,, \end{cases} \tag{2.1}$$

where $x \in \mathbb{R}^{n_x}$ represents the system states, $u \in \mathbb{R}^{n_u}$ the control inputs, and $y \in \mathbb{R}^{n_y}$ the process variables (outputs). Notice how the scheduling parameter ρ acts on Eq. (2.1) by internally modifying its structure along time. Due to the time-varying dependency of the system matrices $(A(\cdot), B(\cdot), C(\cdot), D(\cdot))$ on the scheduling parameter $\rho(k)$, the input-output behaviour of Σ also becomes time-varying (yet, linear).

As argues [Briat 2008], we stress that the spirit of the LPV framework is very related to that of robust analysis and synthesis, with different, more specialised tools being used according to the type of parameter trajectories and variations. In general, either we consider that the scheduling parameters have arbitrarily fast variations over time, i.e. $(\rho(k+1) - \rho(k))$ unbounded, or we assume that these parameters vary slowly enough and thus that $(\rho(k+1) - \rho(k)) \in \delta\mathcal{P}$, being $\delta\mathcal{P}$ a known set. These are the types of parameter trajectories exploited in this monograph[1].

Remark 3. *With regard to Eq. 2.1, we specifically consider the class of affine LPV systems in this thesis, for which the model matrices show affine dependency on the scheduling parameters, that is, for $\rho \in \mathcal{P} \subseteq \mathbb{R}^{n_\rho}$, we obtain:*

$$\begin{cases} A(\rho) &= A_0 + A_1\rho_1 + A_2\rho_2 + \cdots + A_{n_\rho}\rho_{n_\rho}\,, \\ B(\rho) &= B_0 + B_1\rho_1 + B_2\rho_2 + \cdots + B_{n_\rho}\rho_{n_\rho}\,, \\ C(\rho) &= C_0 + C_1\rho_1 + C_2\rho_2 + \cdots + C_{n_\rho}\rho_{n_\rho}\,, \\ D(\rho) &= D_0 + D_1\rho_1 + D_2\rho_2 + \cdots + D_{n_\rho}\rho_{n_\rho}\,, \end{cases}$$

In [Briat 2008], one can find a full details on all possible classes of LPV systems, with affine, polynomial, rational, polytopic and lifted, interconnected structures. In general, the

[1] Specialised tools for specific trajectories, such as piece-wise constant and periodical ones, are available in the literature, e.g. [Mohammadpour and Scherer 2012], but not debated in this work.

only criteria that should hold for all classes is set upon the form of the system matrices: they should continuous and bounded functions on ρ*, that is:* $A : \mathcal{P} \rightarrow \mathfrak{A} \subseteq \mathbb{R}^{n_x \times n_x}$, $B : \mathcal{P} \rightarrow \mathfrak{B} \subseteq \mathbb{R}^{n_x \times n_u}$, $C : \mathcal{P} \rightarrow \mathfrak{C} \subseteq \mathbb{R}^{n_y \times n_x}$*, and* $D : \mathcal{P} \rightarrow \mathfrak{D} \subseteq \mathbb{R}^{n_y \times n_u}$*, being* $\mathfrak{A}$*,* $\mathfrak{B}$*,* $\mathfrak{C}$*, and* $\mathfrak{D}$ *are bounded sets. In some cases, it is required that the matrices posses continuous and existing derivatives with respect to the parameter, i.e.* $A, B, C, D \in \mathcal{C}^1$.

Remark 4. *In the case of polytopic dependency, Eq. 2.1 can be re-stated as follows:*

$$\begin{cases} A(\rho) &= \sum_{j=1}^{L_\rho} \lambda_j(\rho) A_j \, , \\ B(\rho) &= \sum_{j=1}^{L_\rho} \lambda_j(\rho) B_j \, , \\ C(\rho) &= \sum_{j=1}^{L_\rho} \lambda_j(\rho) C_j \, , \\ D(\rho) &= \sum_{j=1}^{L_\rho} \lambda_j(\rho) D_j \, . \end{cases} \tag{2.2}$$

the number of polytopic vertices is $L_\rho = 2^{n_\rho}$*, and the inclusion is certified if* $\sum_{j=1}^{L_\rho} \lambda_j(\rho) = 1$*, where each term* $0 \leq \lambda_j(\rho) \leq 1 \, , \forall j \in \mathbb{N}_{[1,L_\rho]}$.

2.1.1 Scheduling parameters

As detailed in the Introduction, we can argue that the LPV system representation falls somewhere in between the classical dual landscape of linear time-invariant and nonlinear systems. With correct parameter-varying encapsulations, LPV models as in Eq. (2.1) are able to describe both time-varying and nonlinear dynamics. Specifically, we consider two kinds LPV systems[2], with distinct types of scheduling parameters:

(1) Pure LPV systems, for which $\rho(k)$ is an exogenous functions, *a priori*, unrelated to the system variables:

- The use of LPV systems with exogenous, extrinsic scheduling parameters (pure LPV kind) is often related to underlying control objectives, when these parameters are included in order to schedule the controller in a certain manner, seeking to reshape the closed-loop dynamics according to performance criteria. In these cases, usually, a higher-level hierarchical system is used to adapt the scheduling signal online according to such criteria. As an example, we refer to [Medero et al. 2022], where an optimal scheme is tuned to generate design-related scheduling parameters for adaptive LPV controllers. The referred work shows the application of a pure LPV control system as an advanced driver assistance system, demonstrating how it can enhance performances of semi-autonomous vehicles, when compared to state-of-the-art approaches;
- Another simple illustration of the interest of LPV controller is on how to regulate the closed-loop bandwidth of linear systems. Consider the following SISO

[2]This is the same classification as presented in [Shamma 2012] and it is interesting to use it due to the strong philosophical differences between qLPV and purely LPV systems. We note that in the qLPV approach, the scheduling parameter is known and generated online, by the means of the scheduling proxy, while in pure LPV schemes $\rho(k)$ is a signal which is fed externally to the controller.

input-output dynamics: $y(k+1) = ay(k) + bu(k)$, where $u(k) = -\rho(k)y(k) + b^{-1}(1 - a + b\rho(k))\, r(k)$, where $r(k)$ is an auxiliary input. Note how, in closed-loop, we obtain $y(k+1) = (a - b\rho(k))\, y(k) + (1 - a + b\rho(k))\, r(k)$, which means that $\rho(k)$ can be used to tune the structure of the direct loop from $r(k)$ to $y(k)$. As long as $(a - b\rho(k))$ is given within the unit circle, for all fixed values of $\rho(k)$, it is direct to see that the closed-loop dynamics are asymptotically stable, and that the bandwidth can be changed by the means of $\rho(k)$. Moreover, we obtain a steady-state regime, with $\lim_{k\to+\infty} \rho(k) \to \rho_r$, ruled by $\lim_{k\to+\infty} y(k) \to \lim_{k\to+\infty} r(k)$.

(2) Quasi-LPV systems, for which $\rho(k)$ is an endogenous variable, given as a function of the system variables by the means of a known scheduling proxy, i.e. $f_\rho(x(k), u(k), y(k))$:

- This formalism can be used to approximate, or even exactly describe nonlinear dynamics (and, thus, is of special interest to this work). The procedure of obtaining an LPV model from a nonlinear one is often referred to as *qLPV embedding*. In Example 6, presented in the next section, we provide a brief illustration of this concept, where SS and IO qLPV realisations are detailed for a simple nonlinear system.
- We note, nevertheless, that analysing the stability of a system by the means of a qLPV model can lead to excessively conservative results, considering that the embedding accounted for the worst-case performances implied by the chosen varying parameter. Further discussion on LPV stability is presented in Section 2.5.

2.2 Differential inclusion

As previously disclosed, the focus of this thesis is the control of systems described by LPV models. With regard to this matter, we recall that there is a specific property which allows us obtain reliable qLPV descriptions of nonlinear systems: differential inclusion. We note that there are many types of "embeddings" (i.e. inclusions) that can be used in order to generate qLPV models[3] in order to represent nonlinear dynamics, such as linear, convex-concave, and convex differential inclusions, as argues [Sala 2017]. The different embeddings may generate what we call "over-bounding", which refers to a qLPV representation with excessive conservativeness (sometimes, quite common in practice, e.g. [Hoffmann and Werner 2014]).

Specifically, we detail how **exact** embeddings can be generated by the means of Linear Differential Inclusion (LDI) [Boyd et al. 1994]. If a given nonlinear system Σ satisfies the LDI property, meaning that every trajectory of the nonlinear system is also a trajectory of the LDI. Then, we can prove that if every trajectory of the corresponding LDI exhibits some property (let's say, stability, for instance), then, *a fortiori*, we demonstrate that every trajectory of the original nonlinear system also exhibits this property. For rationale purposes, consider, then, a generic time-invariant discrete-time nonlinear system Σ, whose behaviour is defined by the

[3]In [Abbas et al. 2014], a thorough overview of different qLPV embeddings is presented, for a wide variety of applications, with corresponding identification and validation results.

following set of SS equations:

$$\left\{ \begin{array}{rcl} x(k+1) & = & f\left(x(k),u(k)\right) \, , \\ y(k) & = & h\left(x(k),u(k)\right) \, , \end{array} \right. \tag{2.3}$$

where $x \in \mathbb{R}^{n_x}$ are the system states, $y \in \mathbb{R}^{n_y}$ its outputs, and $u \in \mathbb{R}^{n_x}$ are inputs.

Definition 2.2
Consider the nonlinear system Σ as described by Eq. (2.3). Suppose that, for each state x and entry u, there exists a matrix $G(x,u) \in \Omega$ such that:

$$\left[\begin{array}{c} f(x,u) \\ h(x,u) \end{array} \right] \;=\; G(x,u) \left[\begin{array}{c} x \\ u \end{array} \right] , \tag{2.4}$$

where $\Omega \in \mathbb{R}^{(n_x+n_y)\times(n_x+n_u)}$. Thence, we say that the nonlinear system Σ agrees with an exact LDI. As long as LDI is satisfied, we can thus re-cast Σ into the form of the following qLPV realisation[4]*, considering the scheduling set $\mathcal{P} \subseteq \mathbb{R}^{n_\rho}$:*

$$\left\{ \begin{array}{rcl} x(k+1) & = & A(\rho(k))x(k) + B(\rho(k))u(k) \, , \\ y(k) & = & C(\rho(k))x(k) + D(\rho(k))u(k) \, , \\ \rho(k) & = & f_\rho(x(k),u(k)) \, . \end{array} \right. \tag{2.5}$$

In the following example, we elucidate how linear differential inclusion features can be used in order to generate qLPV models from nonlinear dynamics, in form of state-space and input-output realisations:

Example 6. *Consider a cascade of two cylindrical water tanks, with equal base areas A_b, as illustrates Figure 2.1. The upper tank is flooded by an open-valve inlet flow at its top and deflated by a regular open hole. The second tank is flooded by the outlet of the first tank, while also having a regular open hole at its bottom. The inlet flow of water to the upper tank is given by u, while its outlet flow is given by $\sqrt{2gh_1}$, being h_1 the level of the water in this tank and g the gravitational constant. Equivalently, the outlet flow of the bottom tank is given by $\sqrt{2gh_2}$, being h_2 its water level. Assume that both levels are constantly measured; the controlled output is $y = h_2$. Then, assuming an Euler discretisation*[5] *with a T_s sampling period, being $t = kT_s$, the discrete-time phenomenological dynamics of these cascaded tanks are given by:*

$$\left\{ \begin{array}{rcl} h_1(k+1) & = & h_1(k) - \frac{T_s}{A_b}\sqrt{2g}\sqrt{h_1(k)} + \frac{T_s}{A_b}u(k) \, , \\ h_2(k+1) & = & h_2(k) - \frac{T_s}{A_b}\sqrt{2g}\sqrt{h_2(k)} + \frac{T_s}{A_b}\sqrt{2g}\sqrt{h_1(k)} \, , \end{array} \right. \tag{2.6}$$

where $h = [h_1\, h_2]^T$ denotes the level variables and $y = h_2$ gives the main output, which is the level of the lower tank. Assume that exact differential inclusion is satisfied. Thus, we can re-write the nonlinear model from Eq. (2.6) in both SS and IO qLPV formulations, as follows:

[4] Note that, in many cases, qLPV models with only state-dependent functions are used. In these cases, when $\rho(k) = f_\rho(x(k))$, it follows that the original nonlinear functions $f(x,u)$ and $h(x,u)$ should be only nonlinear with regard to x, that is: $f(x,u) = f_1(x) + f_2(x)u$ and $h(x,u) = h_1(x) + h_2(x)u$.

[5] In some cases of strongly nonlinear dynamics between samples, Euler discretisation might fail to be trustworthy. Then, in order for the discretized model to be reliable, we should resort to trapezoidal or Tustin discretisation methods. For simplicity, we will explicitly mention if non-Euler discretisation methods are required. Otherwise, we assume that Euler discretisation "holds".

- ***SS formulation:*** *Consider* $\rho(k) = [\rho_1(k), \rho_2(k)]^T$, *with* $\rho_1(k) = \left(\frac{T_s\sqrt{2g}}{A_b}\frac{\sqrt{h_1(k)}}{h_1(k)}\right)$ *and* $\rho_2(k) = \left(\frac{T_s\sqrt{2g}}{A_b}\frac{\sqrt{h_2(k)}}{h_2(k)}\right)$. *Then, taking* $x(k) = h(k)$, *we obtain:* $x(k+1) = A(\rho(k))x(k) + Bu(k)$, $y(k) = x_2(k)$, *with affine model matrices given by:*

$$A(\rho) = \begin{bmatrix} 1-\rho_1 & 0 \\ \rho_1 & 1-\rho_2(k) \end{bmatrix}, \; B = \begin{bmatrix} \frac{T_s}{A_b} & 0 \end{bmatrix}^T. \tag{2.7}$$

- ***IO formulation:*** *Consider the same scheduling variables, we use* $\left(1 + a_1(\rho(k))z^{-1} + a_2(\rho(k))\right) y(k) = b_2(\rho(k))z^{-2}u(k)$, *with the following model coefficients:*

$$\begin{cases} a_1(\rho) &= 2 - \rho_1 - \rho_2\,, \\ a_2(\rho) &= 1 - \rho_1 - \rho_2 + \rho_1\rho_2\,, \\ b_2(\rho) &= \frac{T_s}{A_b}\rho_1\,. \end{cases} \tag{2.8}$$

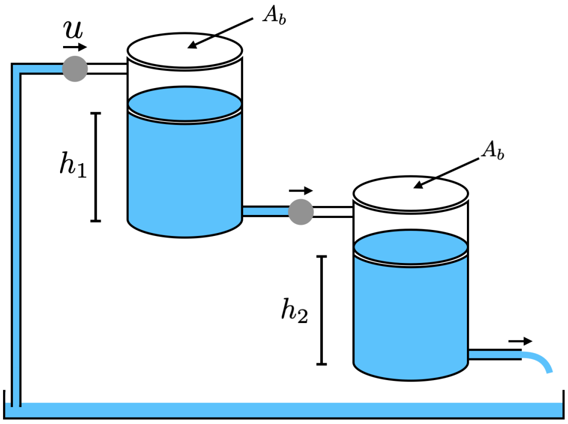


Figure 2.1: Example 6: Level control application.

2.2.1 Overview

All in all, we can organise the wide variety of time-varying nonlinear systems of finite dimensions according to the set diagram illustration in Figure[6] 2.2. The class of qLPV systems is a subset of the set of generic nonlinear time-varying processes, i.e. only those that allow differential inclusions. Furthermore, the class of LPV systems can be understood as a subset of the qLPV ones, when the scheduling is not dependent on the system variables, but more generically understood as exogenous signals. Finally, LTI plants are contained within the LPV class, since these can always be given as LPV ones with constant parameters.

[6]Figure inspired by the Venn diagram presented in [Briat 2008].

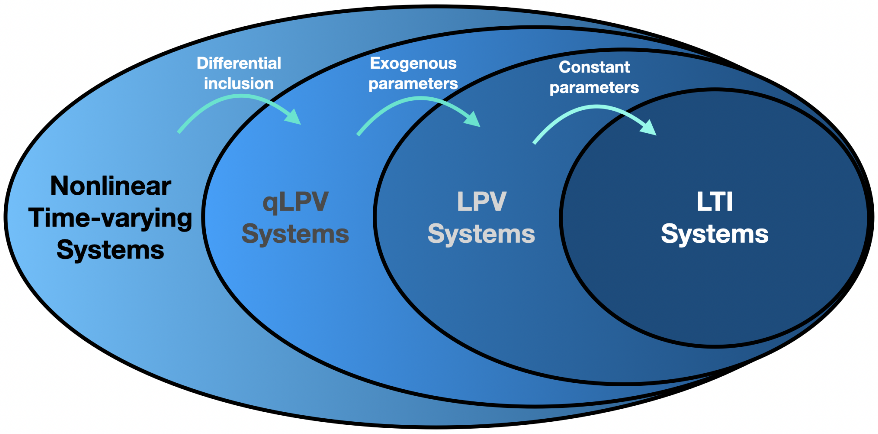


Figure 2.2: Overview of the considered systems with finite-dimensions.

2.3 General notions of stability

Before detailing stability notions for LPV systems, we recall some standard results on the stability of dynamical systems. For such, consider the generic nonlinear plant Σ, as described by Eq. (2.3), assuming no control inputs, i.e. an autonomous system. The following notions can be directly extended to the closed-loop case.

Definition 2.3
Consider the nonlinear system in Eq. (2.3). It follows that x_r defines an equilibrium point of Σ, if and only if the following equality rule holds: $x_r = f(x_r, \cdot)$.

Definition 2.4
Let $x_r \in \mathcal{X} \subseteq \mathbb{R}^{n_x}$ be an equilibrium point of the nonlinear system in Eq. (2.3). Then, it follows that:

1. *x_r is a (Lyapunov) **stable** equilibrium if, given scalar $\epsilon > 0$, there exists a corresponding scalar $\delta(\epsilon) > 0$ such that $\|x(0) - x_r\|_2 < \delta \Rightarrow \|x(k) - x_r\|_2 < \epsilon$ for all $k \geq 0$;*

2. *x_r is said to be **unstable** when it is not (Lyapunov) stable;*

3. *x_r is **attractive** if there exists a scalar $\delta > 0$ such that $\|x(0) - x_r\| \leq \delta \Rightarrow \lim_{k\to+\infty} \|x(k) - x_r\|_2 \to 0$, for all $k \geq 0$;*

4. *x_r is a **locally asymptotically stable** equilibrium if it is (Lyapunov) stable and attractive;*

5. *x_r is an **exponentially stable** equilibrium if there exists scalars δ, $\alpha > 0$ and $\beta \geq 1$ such that $\|x(0) - x_r\|_2 \leq \delta \Rightarrow \|x_r - x(k)\|_2 \leq \beta e^{-\alpha k}\|x(0)\|_2$;*

6. *x_r is said to be **globally asymptotically stable** when it is stable and $\lim_{k\to+\infty} x(k) = x_r$, for all $x(0) \in \mathbb{R}^{n_x}$.*

2.3.1 Complementary practical stability metrics

In many cases, we assume that states of a given system Σ are measurable. Thus, it becomes of interest to analyse the stability of Σ by the means of the input-to-state transfer. Accordingly, we analyse input-to-state stability (ISS), which is defined as follows:

Definition 2.5 (Input-to-state stability)
Consider the system Σ, as given in Eq. (2.3), and assume that the states $x(k)$ are measurable for all sampling instants $k \geq 0$. Σ is input-to-state stable if $u \in \mathcal{L}_2$ and $\|u\| \leq \overline{u}$ and there exists a pair of $\mathcal{K}$-functions $\beta(\cdot, \cdot)$ and $\sigma(\cdot)$ such that the following inequality holds:

$$\|x(k)\| \quad \leq \quad \beta(x(0), k) + \sigma(\overline{u})\,. \tag{2.9}$$

Assume that the system Σ, as given in Eq. (2.3), has a closed-loop state-feedback policy $u(k) = \kappa(x(k))$. Furthermore, assume that the closed-loop dynamics are affected by an additional disturbance variable $w(k)$, i.e.:

$$x(k+1) \quad = \quad f_{\text{CL}}\left(x(k), w(k)\right)\,. \tag{2.10}$$

Then, the analysis of closed-loop ISS is enabled by the means of the following definitions:

Definition 2.6 (Closed-loop input-to-state stability [Jiang and Wang 2001])
Consider a generalised discrete-time nonlinear plant Σ, whose dynamics, under a state-feedback closed-loop policy, are given by Eq. (2.10), where $x(k)$ are measurable states and $w(k)$ is a load disturbance variable such that $\|w(k)\| \leq \overline{w}\,\forall\,,\, k \geq 0$. Then, Σ is said to be input-to-state stable in closed-loop if there exists a pair of $\mathcal{K}$-functions $\beta(\cdot, \cdot)$ and $\sigma(\cdot)$ such that the following inequality holds:

$$\|x(k)\| \quad \leq \quad \beta(x(0), k) + \sigma(\overline{w})\,. \tag{2.11}$$

Even though the analysis of ISS is very useful, it cannot be considered in the situations when Σ is subject to persistent disturbance effects. In this case, the notion of input-to-state practical stability is required:

Definition 2.7 (Input-to-state practical stability (ISpS))
Consider the system Σ, as given in Eq. (2.3), and assume that the states $x(k)$ are measurable for all sampling instants $k \geq 0$. Σ is said to be ISpS within $\mathcal{X}_{ISpS} \subseteq \mathbb{R}^{n_x}$ if there exists a $\mathcal{KL}$-function β_r, a $\mathcal{K}$-function σ_r and a scalar $d_r \in \mathbb{R}_+$ such that, for each $x_0 \in \mathcal{X}_{ISpS}$, and all $u \in \mathcal{L}_2$ such that $\|u\| \leq \overline{u}$, it holds that the corresponding state trajectory satisfies the following inequality holds:

$$\|x(k)\| \quad \leq \quad \beta_r(x(0), k) + \sigma_r(\overline{u}) + d_r\,. \tag{2.12}$$

For a given nonlinear system Σ, if the origin lies in the interior of $\mathcal{X}_{\text{ISpS}}$ and Ineq. (2.12) also holds for $d_r = 0$, Σ is said to be ISS within $\mathcal{X}_{\text{ISpS}}$. Note how the presence of a non-null term d_r in Ineq. (2.12) implies in a conservative ISpS criterion, i.e. for greater values of $\|d_r\|$, we obtain a smaller ISpS region. Thus, lastly, we also recall the notions of closed-loop regional ISpS and regional ISS, used for closed-loop dynamics subject to non-fading disturbances.

Definition 2.8 (Regional ISpS (ISS) [Magni, Raimondo, and Scattolini 2006])
Consider the system Σ in closed-loop with a state-feedback, as given in Eq. (2.10). Assume that the states $x(k)$ are measurable and that $w(k)$ is a load disturbance variable such that $\|w(k)\| \leq \overline{w} \, \forall \, k \geq 0$. Then, Σ is said to be ISpS (ISS) within $\mathcal{X}_{RISpS} \subseteq \mathbb{R}^{n_x}$ if there exists a $\mathcal{KL}$-function β_r, a $\mathcal{K}$-function σ_r and a scalar $d_r \in \mathbb{R}_+$ such that, for each $x_0 \in \mathcal{X}_{RISpS}$, it follows that the corresponding state trajectory satisfies the following inequality:

$$\|x(k)\| \leq \beta_r(x(0), k) + \sigma_r(\overline{w}) + d_r \, , \, \forall k \in \mathbb{N} \, . \tag{2.13}$$

Furthermore, if the origin lies in the interior of $\mathcal{X}_{RISpS}$ and the inequality also holds for $d_r = 0$, the system is said to be ISS in $\mathcal{X}_{RISpS}$.

Note that the presence of a non-null d_r in the Ineq. (2.13) is, again, a conservative solution, since for greater values of $\|d_r\|$, we obtain a smaller ISpS (ISS) region $\mathcal{X}_{\text{RISpS}}$.

2.4 Standard stability tools (Lyapunov theory)

In order to investigate the stability properties of a given system, with regard to the definitions presented in Sec. 2.3, mature theoretical frameworks are available. In this section, we revisit the main results on Lyapunov theory. First of all, we note that there exists a specific (Lyapunov) theorem associated to each kind of stability (asymptotic, exponential, etc). For brevity and simplicity, we provide a synthesis of the main results in the sequel, considering the autonomous nonlinear dynamics of Σ in Eq. (2.3), i.e. taking $u(k) = 0$.

- Consider a real positive defined Lyapunov function $V : \mathbb{R}^{n_x} \rightarrow \mathbb{R}_+$ such that there exists two corresponding $\mathcal{K}$-class functions $\alpha_1(\cdot)$ and $\alpha_2(\cdot)$ such that $\alpha_1(\|x\|) \leq V(x) \leq \alpha_2(\|x\|)$, for all $x \in \mathbf{B}_r$, being the closed n_x ball defined as $\mathbf{B}_r := \{x \in \mathbb{R}^{n_x} \mid \|x\| \leq r\}$. Then, if the following condition is satisfied for all $x \in \mathbf{B}_r$, the origin is a stable equilibrium point of Σ:

$$V(f(x)) - V(x) \leq 0 \, . \tag{2.14}$$

- If the previous conditions are valid for the whole set of real-valued states, i.e. for $\mathbf{B}_r = \mathbb{R}^{n_x}$, then the origin is globally stable.

- Consider another real positive defined Lyapunov function $V : \mathbb{R}^{n_x} \rightarrow \mathbb{R}_+$ such that there exists three corresponding $\mathcal{K}$-class functions $\beta_1(\cdot)$, $\beta_2(\cdot)$ and $\beta_3(\cdot)$ such that $\beta_1(\|x\|) \leq V(x) \leq \beta_2(\|x\|)$, for all $x \in \mathbf{B}_r$, Then, if the following inequality holds:

$$V(f(x)) - V(x) \leq -\beta_3(\|x\|) \, , \tag{2.15}$$

  the origin is an asymptotically stable equilibrium point of Σ.

- Again, if the previous conditions are valid for the whole set of real-valued states, i.e. for $\mathbf{B}_r = \mathbb{R}^{n_x}$, then the origin is a globally asymptotically stable equilibrium.
- Again, consider a real positive defined Lyapunov function $V : \mathbb{R}^{n_x} \to \mathbb{R}_+$. Furthermore, consider that there exits two real positive scalars a and b, and scalar $c \geq 1$ such that $a\|x\|^c \leq V(x) \leq b\|x\|^c$, for all $x \in \mathbf{B}_r$, Then, if there exists a positive scalar d such that the following inequality holds:

$$V(f(x)) - V(x) \quad \leq \quad d\|x\|^d \,, \tag{2.16}$$

  the origin is an exponentially stable equilibrium point of Σ. Furthermore, if the previous conditions are valid for the whole set of real-valued states, i.e. for $\mathbf{B}_r = \mathbb{R}^{n_x}$, then the origin is a globally exponentially stable equilibrium.

The concept of stable autonomous system Σ (i.e. Eq. (2.3) with $u(k) = 0$) can be translated and generalised to the closed-loop case (to non-autonomous systems). Thus, stability becomes stabilisability, meaning that, if Σ is stable for $u(k) = 0$, there exists a control law $u(k) \neq 0$ which makes the closed-loop interconnection stable. Thereby, all previous notions of stability can be extended directly to the closed-loop case. Furthermore, we stress that all conditions presented in the prequel are merely **sufficient**, and can be used to infer if Σ has the origin as an (asymptotically, exponentially) stable equilibrium. Anyhow, there exists also **sufficient and necessary** equivalencies, valid under certain hypothesis, as detailed in [Khalil and Grizzle 2002]. These stronger conditions are not recalled here, for brevity.

2.4.1 Invariant set theory

The concept of stability, and the corresponding analysis using the Lyapunov framework, is closely related to the notions of invariant sets. In practice, such sets are used to guarantee that MPCs ensure closed-loop stability.

Definition 2.9 (Positively invariant set [Blanchini and Miani 2008])
A set $\mathbf{X}_f \subset \mathbb{R}^{n_x}$ is said to be a positively invariant set for a given system Σ, as in Eq. (2.3)*, if for any $x(k) \in \mathbf{X}_f$, it follows that $x(k+1) \in \mathbf{X}_f$, $\forall k \geq 0$. This is, the whole state trajectory always lies inside this set $\mathbf{X}_f$, which is thus called invariant.*

Basically, if a system trajectory reaches a positively invariant set $\mathbf{X}_f$, i.e. the states are found inside this set, the remainder of the state trajectories will also be contained therein. Any system that has an associated positive invariant set is a stabilisable system. That is, if a given system Σ has the origin as a stable equilibrium point, it follows that $V(f(x)) - V(x)$ is negative for all $x \in \mathbf{B}_r$ and, thus, $V(x(k))$ decays as k evolves for any trajectory that departs from $\mathbb{R}^{n_x} \ominus \mathbf{B}_r$, i.e. $V(x(k)) \leq V(x(0))$, $\forall k \geq 0$. Consider a $\mathcal{K}$-class function $\alpha_1(\cdot)$. Thence, it follows that $\mathbf{X}_f := \{x \in \mathbb{R}^{n_x} \,|\, V(x) \leq \alpha_1(r)\}$ is subset of $\mathbf{B}_r$, which implies that for all $x \in \mathbf{X}_f$, it follows that:

$$\alpha_1(\|x(k)\|) \quad \leq \quad V(x(k)) \;\leq\; \alpha_1(r) \,, \, \forall k \geq 0 \,. \tag{2.17}$$

Alternatively, we obtain $\|x(k)\| \leq r$, $\forall k \geq 0$ and, thus, that $\mathbf{X}_f \subseteq \mathbf{B}_r$ is a postively invariant set for Σ. Note that for any $x(0)$ such that $V(x(0)) \leq \alpha_1(r)$, we obtain $V(x(k)) \leq V(x(0)) \leq \alpha_1(r)$, since $x(k) \in \mathbf{X}_f$. Therefore, the origin is stable equilibrium for Σ. In brief, if the negativeness of $V(f(x)) - V(x)$ is verified within a given set $\mathbf{X}_f$, for a system Σ, it follows directly that $\mathbf{X}_f$ is a positively invariant set for Σ, because the system trajectories departing from $\mathbf{X}_f$ will be contained therein.

The previous discussion resided in debating how positive invariant sets can be used to ensure stability. Closely related to these sets are the so-called *control invariant set*, which can be understood as a generalisation of positively invariant sets used specifically for control.

Definition 2.10 (Control invariant set [Fiacchini, Alamo, and Camacho 2010])
Consider the nonlinear system Σ in Eq. (2.3). A set $\mathbf{X}_f \subset \mathbb{R}^{n_x}$ is said to be a control invariant set for Σ if for all $x(0) \in \mathbf{X}_f$, there exists a state-feedback control law $u(k) = \kappa(x(k))$ such that $x(k) \in \mathbf{X}_f$ and that $u(k) \in \mathcal{U}$, $\forall k \geq 0$, being $\mathcal{U} \subset \mathbb{R}^{n_u}$ a set that defines an admissible control input.

Based on this definition, we can conclude that if there exists a control invariant set $\mathbf{X}_f$ for a given system Σ, we are able to synthesise an admissible state-feedback controller $u(k) = \kappa(x(k))$ which stabilises the origin in closed-loop. This is of specific interest when designing MPC laws that enable closed-loop stability. Furthermore, the notion of control Lyapunov functions is also of interest:

Definition 2.11 (Control Lyapunov function [Primbs, Nevistić, and Doyle 1999])
A function $V : \mathbb{R}^{n_x} \to \mathbb{R}_+$ is a control Lyapunov function associated to the nonlinear system Σ in Eq. (2.3) if it is positive defined and satisfies:

$$\min_{u \in \mathcal{U}} \left(V\left(f(x,u)\right) - V\left(x\right)\right) \leq 0\,, \tag{2.18}$$

for all $x \in \mathbf{B}_r$. Again, $\mathcal{U} \subset \mathbb{R}^{n_u}$ is a set that defines an admissible control input.

Synthetically, if there exists a control Lyapunov function $V(\cdot)$, we can generate an admissible state-feedback policy $u(k) = \kappa(x(k)) \in \mathcal{U}$, $\forall k \geq 0$ that ensures that the closed-loop dynamics satisfy

$$V\left(f(x, \kappa(x))\right) \leq V(x),\, \forall x \in \mathbf{B}_r$$

and, thus, that the system is stabilised to the origin. The existence of a control Lyapunov function implies that the corresponding system is stabilisable, we can thus generate a control invariant set for this system.

Remark 5. *Along this thesis, in many applications, we consider ellipsoidal invariant sets. These sets, generically stated as $\mathcal{E}_{Ellipsoid} : \left\{x \in \mathbb{R}^{n_x} \,|\, x^T P x \leq 1\right\}$, when used in the context of MPC as terminal constraints, render the problem non-quadratic (note that $x(k+N_p|k)^T P x(k+N_p|k) \leq 1$ is a nonlinear constraint). Nevertheless, there are many alternatives that can be pursued in order to find polyhedron sets in the form of $\mathcal{E}_{Polyhedron} : \{x \in \mathbb{R}^{n_x} \,|\, H_x x \leq w_x\}$, such that $\mathcal{E}_{Polyhedron} \subseteq \mathcal{E}_{Ellipsoid}$. Thus, using $H_w x(k+N_p|k) \leq w_x$*

as a terminal constraint, in the context of MPC, the same properties as when using the original ellipsoidal set are maintained (control invariant, recursive feasibility, etc), whereas the optimisation remains a QP. Thereof, we say that MPC algorithms (with linear models and quadratic constraints) with ellipsoidal terminal constraints render an optimisation in the form of a QP, assuming that a polyhedron replaces the ellipsoid. Refer to [Bitsoris 1988; Pluymers et al. 2005] for more details on such polyhedra. [Alessio et al. 2007] gives a practical algorithm able to generate a polyhedron subset from a given ellipsoid.

2.4.2 Dissipativity

Another way to verify the stability of Σ is to determine if an associated energy function $V(\cdot)$ dissipates over time. Mathematically, we define system dissipativity as follows:

Definition 2.12 (System dissipativity [Scherer 2022])
A generic system $x(k+1) = f(x(k), u(k))$, with $u(k) \in \mathcal{U}$ and $x(k) \in \mathcal{X}, \forall k \geq 0$ is dissipative with respect to the supply rate $s : (\mathcal{U} \times \mathcal{X}) \to \mathbb{R}$ if there exists a storage function $V : \mathcal{X} \to \mathbb{R}$ such that the dissipation inequality

$$V(x(k_2)) \leq V(x(k_1)) + \sum_{i=k_1}^{k_2-1} s\left(u(i), x(i)\right), \tag{2.19}$$

holds for all admissible trajectories of the system and all time instances $k_1, k_2 \in \mathbb{N}$ such that $k_1 \leq k_2$ and $\left(\sum_{i=k_1}^{k_2-1} s\left(u(i), x(i)\right)\right)$ is non-negative.

Remark 6. *The particular choice of the supply rate $s(u, x) = u^T x$ implies in what is called passitivity.*

We stress that the notion of dissipativity has an underlying physical interpretation: if the combination of forces that operate upon a given system are dissipative (and thus there exists a corresponding non-negative supply rate), we say that the system is dissipative (the stored energy decays); otherwise, we call it conservative.

Example 7. *Consider a vehicle subject to a variable horizontal acceleration force $u(k)$ and to friction due to the contact of the tires with the road. The dynamics of the car's velocity $y(k)$ are, by Newton's second law of motion, as follows:*

$$y(k+1) = y(k) + bu(k) - cy^2(k), \tag{2.20}$$

Take the following associated energy map $V(k) = V(y(k)) = y^T(k)Py(k)$. Accordingly, in order to verify that this system is indeed dissipative, we must ensure that $V(k+1) - V(k) < 0$, for all $k \geq 0$, with an initial positive energy $V(0) > 0$. For simplicity, let us consider u null. Complementary, note that as long as $c > 0$, we have $y(k+1) = y(k)(1 - cy(k)) = a(k)y(k)$. Then, $V(k+1) - V(k) < 0$ is valid as long as $a^T(k)Pa(k) - P \leq 0$. Taking $P = I$, we obtain $(1 - cy(k))^T(1 - cy(k)) - 1 \leq 0$ and, thus, $0 \leq cy(k) \leq 2, \forall k$. Since we seek a dissipative *system, it follows that $y(k) < y(0)$ and thus as long as $c \leq \frac{2}{y(0)}$, we can ensure that V will dissipate.*

In many settings, it becomes quite hard to prove dissipative arguments due to nonlinearities and complicated inequalities. In such cases, an interesting option is to use an abstraction to describe the input-output behaviour of given systems (or uncertainties): Integral Quadratic Constraints (IQCs). With respect to the last example, this would be equivalent to detach the nonlinear dependency of $a(k)$ from the analyses. A time-domain IQC is defined next:

Definition 2.13
Let $\psi \in \mathbb{R}\mathcal{H}_\infty^{n_z \times (n_u+n_y)}$ and $M : \mathcal{P} \to \mathbb{R}^{n_z}$ (symmetric). A system operator $y =: \Sigma u$ satisfies the IQC denoted $\mathcal{I}(\Psi, M)$ if the following inequality holds for all $u \in \mathcal{L}_2$:

$$\sum_{i=k_1}^{k_2} z(i)^T M z(i) \quad \geq \quad 0 \, . \tag{2.21}$$

Basically, if we can employ an IQC abstraction in order to describe the input-output behaviour of a system, a filter, or an uncertainty. Then, instead of analysing the system's dissipativity, we can infer if its dissipative by determining an M that satisfies Ineq. (2.21). This property can be directly generalised to systems with bounded uncertainty interconnections described via IQCs. In Chapter 6, we provide a typical application of this framework in the context of MPC, where an IQC is used to describe the predictive controller.

2.5 Stability of LPV systems

Stability and robustness of LPV systems deserve special attention in this thesis. Since LPV models exhibit time-varying parameters, the stability analysis becomes more complex than in the LTI case. Although a LPV model can be verified stable for all frozen parameter values within its scheduling set, it can be unstable due to the variation of these parameters along the samples, which can be understood as a switching law between different LTI systems. Accordingly, the stability methods described in the prior should be adapted to the LPV context, thus accounting for parametrical variations between samples. In this thesis, we consider two different kinds of stability analysis for LPV systems: quadratic stability and robust stability.

Definition 2.14 (Quadratic LPV stability)
Consider an autonomous LPV system Σ, as in Eq. (2.1) with $u = 0$. Σ is said to be quadratically stable if there exists a positive Lyapunov storage function $V : \mathbb{R}^{n_x} \to \mathbb{R}_+$ such that $V(x) = x^T P x > 0$ for every $x \neq 0$ and $V(0) = 0$ such that:

$$\begin{aligned} \Delta V(x, \rho) &= V(A(\rho)x) - V(x) \\ &= x^T A^T(\rho) P A(\rho) x - x^T P x \; < \; 0 \, , \end{aligned} \tag{2.22}$$

for all $\rho \in \mathcal{P}$.

Synthetically, an LPV system is quadratically as long as there exists a constant matrix $P \in \mathbb{R}^{n_x \times n_x}$ such that Ineq. (2.22) holds. Note, nevertheless, that quadratic stability is ensured for a Lyapunov matrix P which is parameter-independent, yet Ineq. (2.22) should hold for all scheduling parameter values (i.e. for all $\rho \in \mathcal{P}$). In many cases, the analysis of quadratic stability becomes quite conservative, with possibly infeasible solutions (inexistent matrix P). In order to avoid such issue, the notion of robust LPV stability is typically used.

Definition 2.15 (Robust LPV stability)
Consider an autonomous LPV system Σ, as in Eq. (2.1) with $u = 0$. Σ is said to be robustly stable if there exists a positive parameter-dependent Lyapunov storage function $V : \mathbb{R}^{n_x} \times \mathcal{P} \to \mathbb{R}_+$ such that $V(x, \rho) = x^T P(\rho)x > 0$ for every $x \neq 0$ and $V(0, \rho) = 0$ such that:

$$\begin{aligned} \Delta V(x,\rho) &= V(A(\rho)x,\rho) - V(x,\rho) \\ &+ \underbrace{\left(\sum_{i=1}^{n_\rho} \delta\rho_i \frac{\partial V(x,\rho)}{\partial \rho_i} \right)}_{\frac{dP(\rho)}{d\rho}} , \end{aligned} \tag{2.23}$$

$$\begin{aligned} \Delta V(x,\rho) &= V(A(\rho)x,\rho) - V(x,\rho) \\ &+ x^T A^T(\rho) \left(P(\rho + \delta\rho) - P(\rho) \right) A(\rho) x \, , \end{aligned} \tag{2.24}$$

$$\Delta V(x,\rho) = x^T A^T(\rho) P(\rho + \delta\rho) A(\rho) x - x^T P(\rho) x < 0 \, , \tag{2.25}$$

for all $\rho \in \mathcal{P}$, and all $\delta\rho_i \in \delta\mathcal{P}_i$.

The derivative term of the storage candidate $V(x, \rho)$ with regard to the scheduling parameters, i.e. $\frac{dP(\rho)}{d\rho}$, appears due to the fact that Lyapunov theory requires one to evaluate the decay of this function over time. Since $P(\rho)$ is parameter-dependent, the decay of $V(\cdot, \cdot)$ is directly affected by the variation of the scheduling variables between samples, denoted here as $\delta\rho(k) = (\rho(k) - \rho(k-1)) \in \delta\mathcal{P}$.

The term $\frac{dP(\rho)}{d\rho}$ in Ineq. (2.25) illustrates the importance of the rate of variation of the scheduling parameters in the stability of LPV systems. When the bounds on these rates (i.e the sets $\delta\mathcal{P}_i$) are unknown, the designer must assume that the parameters vary arbitrarily between samples, and thus that $\frac{dP(\rho)}{d\rho}$ is *a priori* unbounded. Thereby, robust LPV stability cannot be inferred, but rather taking $\frac{dP(\rho)}{d\rho} = 0$ which implies that $P(\rho) = P_0$ (parameter-independent matrix) and thus that only quadratic LPV stability can be inferred. In practical situations, $\frac{dP(\rho)}{d\rho}$ can be very small, whenever the variations of the scheduling parameters are too subtle and thus we can use $\frac{dP(\rho)}{d\rho} \approx 0$ in Ineq. (2.25) as an approximated argument.

For the vast majority of *practical* LPV applications, the scheduling parameters indeed exhibit bounded variation rates over samples. That is, $\delta\rho(k)$ is bounded for all $k \geq 1$. This property is often exploited in order to synthesise less conservative controllers, which benefit from the robust LPV stability framework (and thus use parameter-dependent stabilisation arguments). We note that, along this thesis, the majority of systems account for bounded scheduling parameter variations. A typical example of this fact is observed in physical systems with scheduling-state dependencies, that is, represented in the form of $x(k+1) = A(\rho(k))x(k)$, with $\rho(k) = f_\rho(x(k))$. In these processes, since the states $x(k)$ are bounded due to physical properties, we can easily infer on bounds for $\delta\rho(k) = f_\rho(x(k)) - f_\rho(x(k-1))$ using interval algebra or optimisation tools.

Remark 7. *Quadratic LPV stability is a particular case of parameter-dependent stability, as argued above. Therefore, quadratic stability implies in robust stability, and it can also be used as a sufficient condition for stability (i.e. if Σ is quadratically stable, it is stable).*

2.6 (Re)-Introducing predictive control

Model Predictive Control, sometimes referred to as sliding-horizon control, is an optimal control approach, for which the control action is generated with regard to some performance criteria. As argued in [Ferramosca 2011; Santos 2011], we recall that MPC is of interest specially because it can deal with coupled, multivariable dynamical processos, while explicitly handling state, input, and output constraints. Moreover, MPC is able to conceptually handle nonlinearities with ease, and, at the same time, express optimality concerns under a systematic design procedure. In synthesis, MPC consists in the feedback implementation of optimal control on the basis of a finite, sliding prediction horizon and the online computation of an optimisation problem. As indicated in [Alamir 2013], one can find over 5800 successful industrial applications of this control method, in many different areas.

2.6.1 Basic ideas on MPC

The basic concepts of MPC are the following: (*i*) It makes explicit use of a model to make predictions of the process behaviour along a prediction horizon. (*ii*) A receding approach is used, which means that, at each discrete-time instant, the prediction horizon is displaced towards the future (rolls forward). Thus, at each discrete-time step, a new window of future predictions of fixed size is taken into account. (*iii*) A full-horizon control sequence is generated at each discrete-time step, from which only the first entry is applied. In more details, we can describe the main characteristics of MPC as follows:

1. At each instant k, the future N_p states are previewed, i.e. $x(k+1|k), x(k+2|k), \dots, x(k+N_p|k)$. These predicted variables $x(k+i|k), \forall i \in \mathbb{N}_{[1,N_p]}$ depend on the known variables (past inputs $u(k+j-1|k)$ and states $x(k+j|k), \forall j \in \mathbb{N}_{[1,i]}$) and on the future control signals ($u(k+j-1|k), \forall j \in \mathbb{N}_{[i,N_p]}$)), which are to be sent to the system and to be calculated.
2. The sequence of future control signals U_k is calculated, at each instant, by the means of an optimisation procedure (Eq. (2.27)). In most MPC applications, the optimisation cost takes a quadratic form, weighting states and inputs. Note that an explicit solution to the optimisation can be obtained if there are no constraints[7].

[7]Remark that, in some cases, even if there are active constraints, explicit solutions can also be found, although they are usually not linear nor trivial [Besselmann, Lofberg, and Morari 2012].

3. The control signal $u^\star(k|k)$ is sent to the process at each instant k, whilst the next control signals calculated are neglected $u^\star(k+i|k), \forall i \in \mathbb{N}_{[1,N_p-1]}$ from $U_k^\star$ are neglected. This repeated solution of the optimisation problem is used because, at the following sampling instant, new state measurements are available, i.e. $x(k+1|k+1)$, and the control input $u(k|k)$ has already been applied. Thus, the optimisation is re-iterated based on the new process variables, and all prediction sequences are brought up to date. Accordingly, the control law $u(k+1) = u^\star(k+1|k+1)$ is calculated at instant $k+1$ based on the measurements $x(k+1)$. Note that, in principle, $u^\star(k+1|k+1)$ is different from $u^\star(k+1|k)$ due to the fact that new information is included in the optimisation. This rolling-horizon paradigm implies that the whole control sequence solution at a given instant k does not necessarily correspond to the real sequence of inputs. The first entry of each sequence $U_k^\star$ is the one that composes the real control signal, while the following do not correspond to the actual control that is applied in the following sample, since new measurements update and correct the previewed inputs, i.e. $u^\star(k+1|k)$ is probably different than $u^\star(k+1|k+1)$, and so forth.

2.6.2 Setting up an MPC algorithm

MPC algorithms three main components in order to operate: (*i*) an accurate prediction model; (*ii*) a performance-related cost function (usually quadratic); and (*iii*) the set of operational constraints that should be respected. We detail these elements individually:

(i) **The prediction model**: MPC requires a model in order to predict the system variables along the future horizon. A prediction model is an analytical mathematical expression which describes the expected behaviour of the process in a given sampling instant. This model can be linear, nonlinear, LPV, variant or invariant in time, written in terms of state variables or in the input-output form. In general, MPC prediction models are written in discrete-time, since this control method is inherently discrete. As of this, only discrete-time models are in used along this thesis.

- For any known process model $x(k+1) = f(x(k), u(k))$, and $y(k) = f_y(x(k), y(k))$, being x, u and y the states, inputs and outputs, respectively, we use $x(k+j|k)$, $u(k+j-1|k)$ and $y(k+j-1|k)$, $\forall j \in \mathbb{N}_{[1,N_p]}$ in order to denote the *predicted* values for the states, inputs and outputs, when computed at instant k. It is implied, under this notation, that initial state prediction is given by a known measurement, i.e., $x(k|k) = x(k)$, whereas all remaining variables are stances along the prediction horizon of N_p steps.
- In general, when disturbances are present in the system, we can use a so-called *disturbance model* in order to account for their behaviours along time. That is, assume $x(k+1) = f(x(k), u(k), w(k))$, where w stands for the disturbances. Then, we can use a known relationship, such as $w(k+j+1|k) = f_w(w(k+1)$ (in this case, an auto-regressive equation), in order to compute the future disturbance values along the prediction horizon. In many practical situations, future data can be used

to compensate the term $w(k+j+1|k)$ in the prediction model[8].

- In the context of MPC, we denote the **terminal** state and the **terminal** local control as $x(k+N_p|k)$ and $u(k+N_p-1|k)$, respectively. These two variables are of special interest in order to synthesise stabilising predictive control schemes (this topic is further detailed by the end of this Chapter).

(ii) **The cost function**: In predictive control, a cost function is required. This function indicates the criterion the optimal control law is tuned to enhance. In general, an MPC cost function J_k is positive defined and expressed in terms of the cost associated to a predetermined system evolution along the prediction horizon. J_k is usually written in of the sampled state measurement $x(k)$ and the corresponding sequence of control inputs U_k, in such a way that $\min_{U_k} J(x(k), U_k)$ gives the envisioned criteria enhancement. For simplicity, the following kind of cost function is used along this thesis:

$$
\begin{aligned}
J_k \;=\; J(x(k), U_k) \;:=\; & \sum_{i=1}^{N_p-1} \ell\left(x(k+i|k), u(k+i-1|k)\right) \\
+ \; & V(x(k+N_p|k)) \,,
\end{aligned}
\tag{2.26}
$$

where $\ell(\cdot,\cdot)$ is named the **stage cost**, related to the system behaviour at the predicted instant $k+i$, and $V(\cdot)$ is called the terminal cost. Henceforth, any stage and terminal cost is assumed positive defined for coherence, in such a way that J_k is also implied positive defined and thus its minimal related to the sought performance criteria.

- Different criteria can be included into J_k, such as a regulation objective of the states, or the tracking goal of certain outputs[9]. A stage cost $\ell(x,u) := \|x\|_Q^2 + \|u\|_R^2$ implies the regulation of the system states to the origin, with weighting criteria Q and R over the state and input trajectories, respectively[10]. A cost $\ell(y) := \|y - y_r\|$ implies the tracking goal the system output y with regard to a known reference target y_r.

(iii) **The constraints**: As argues [Normey-Rico and Camacho 2007], one of the main advantages of MPC is that constraints can be explicitly considered in the design procedure. These indicate the limits within which the process variables should evolve. In practice, all variables of controlled processes are related to real characteristics, which must abide to physical rules. Moreover, constraints can be used to enact safety conditions, or even feasibility limits of actuators and sensors. Constraints can also be used to imply economic criteria to the control law, allowing the system to be maintained near a pre-determined working region.

- In mathematical terms, constraints are given as bounds in the amplitude and in the slew rate of the control signal, states and outputs. Considering constraints on states

[8] In the case of renewable energy systems, for instance, the disturbances are the renewable sources (i.e. solar irradiance data, wind speed profiles, and so forth) and, thus, meteorological forecasts can be used within the prediction model in order to embed a feedforward action.

[9] We name regulation control if the strategy is synthesed in such a way that the states are steered to the origin, whereas tracking refers to steering outputs towards a (usually piece-wise constant) reference signal

[10] We stress that, in general, quadratic costs are used. Therefore, we highlight the use of the 2-norm. Nonetheless, we emphasise that alternative formulations, using other norms,are also possible.

and inputs, for instance, we use $x(k+i|k) \in \mathcal{X}$ and $u(k+i-1|k) \in \mathcal{U}$, $\forall i \in \mathbb{N}_{[1,N_p]}$, being $\mathcal{X}$ and $\mathcal{U}$ known sets which defined admissibility. Furthermore, we note that it is usual to require that the terminal state belongs to a given region, which is implied by a **terminal** constraints, i.e. $x(k+N_p|k) \in \mathbf{X}_f$.

2.6.3 Implementation

Taking into account the previous discussions, we can thus define an MPC application as the solution of a corresponding optimisation problem $\mathfrak{P}_k$, which is solved at each sampling instant k. In general lines, we can state such optimisation problem, composed of a prediction model, a cost function, and constraints, as follows[11]:

Problem 1.

$$\begin{aligned}
\min_{U_k} \quad & \overbrace{\sum_{i=0}^{N_p-1} \ell\left(x(k+i|k), y(k+i|k), u(k+i-1|k)\right)}^{J_k} \qquad (2.27) \\
& + V(x(k+N_p|k))\,, \\
s.t. \quad & x(k+i|k) = f(x(k+i-1|k), u(k+i-1|k))\,, \\
& y(k+i|k) \;=\; f_y(x(k+i|k), u(k+i|k))\,, \\
& u(k+i-1|k) \;\in\; \mathcal{U}\,, \forall i \in \mathbb{N}_{[1,N_p]}\,, \\
& x(k+i|k) \;\in\; \mathcal{X}\,, \forall i \in \mathbb{N}_{[1,N_p]}\,, \\
& y(k+i|k) \;\in\; \mathcal{Y}\,, \forall i \in \mathbb{N}_{[1,N_p]}\,, \\
& x(k+N_p|k) \;\in\; \mathbf{X}_f\,.
\end{aligned}$$

In this problem, it is implied that $U_k = \mathrm{col}\{u(k|k)\,, \dots\,, u(k+N_p-1|k)\}$ gives the sequence of actions inside the (sliding) prediction horizon N_p. Often, a different sliding horizon is chosen for the control signal (named a control horizon N_c, where usually $1 \leq N_c \leq N_p$). In general, this optimisation also embeds a terminal control law constraint, i.e. $u(k+j|k) = \kappa x(k+j|k)$, for all $j \in [N_c, N_p-1]$, without necessarily emphasising it in its formulation, that is, this terminal control law constraint is implicit. Here, we examplified using a state-feedback terminal law, but this could be replaced by a a full-information policy, e.g. $\kappa_t x(k+N_p|k) + \kappa_w w(k+N_p|k)$, in the case of known disturbances, or even by an augmented-state feedback, e.g. $\kappa_a \nu(k+N_p|k)$, in the case of augmented state representations ν. Henceforth, we opt to implicitly consider that a local state-feedback terminal control law is enacted.

The optimisation problem $\mathfrak{P}_k$ in Eq. (2.27) has an internal prediction model constraint. In this setting, this prediction is nonlinear since we use the original nonlinear model to describe the process. This renders $\mathfrak{P}_k$ as a nonlinear programming problem (refer to Appendix B). Nevertheless, as previously detailed, this work focuses on processes described by qLPV and

[11]Henceforth, we denote X_k as the collected sequence of predicted state variables, i.e. $\mathrm{col}\{x(k|k)\,, x(k+1|k)\,, \dots\,, x(k+N_p|k)\}$.

LPV models. In this case, unknown terms appear from the second state prediction onwards, as further assessed in Chapter 3. That is, we obtain $x(k+2|k) = A(\rho(k+1))A(\rho(k))x(k) + A(\rho(k+1))B(\rho(k))u(k|k) + B(\rho(k+1))u(k+1|k)$, and so forth. Note that $\rho(k+1)$ is not known at the sampling instant k.

In general, MPC applications are set in the "state-feedback" form, which means that the first entry of solution of the optimisation $\mathfrak{P}_k$, namely the minimiser $U_k^\star$, whose first entry is $u^\star(k|k)$, can be written as a feedback of the current state variables, i.e. $\kappa^\star\left(x(k)\right)$. Nevertheless, state-feedback MPC is only viable if the state variables are measurable. In the case they are not, we must adapt the MPC problem so that the control input can be expressed in terms of the measured outputs, i.e $u^\star(k) := \kappa^\star\left(y(k)\right)$, and thus "output-feedback". For this kind of formulation, as detailed in Chapter 4, either the process model must be written in the form of input-output realisations, or state observers are required so that an estimate of $x(k)$ can be generated.

2.6.4 Final notes

Predictive control schemes have had considerable interest in theory and practice. This is mainly due to its flexible formulation, written in time-domain, open and quite intuitive. Moreover, MPC is able to handle any kind of process (linear, nonlinear, static, multi-variable, etc), under the same control formalism, while respecting optimality criteria and constraints. Nevertheless, we stress that, albeit having many advantages, there are some implementation drawbacks in MPC which should be emphasised:

- MPC is model-based and, thus, it requires the knowledge of a prediction model which must be sufficiently precise.
- An optimisation problem must be solved online, which requires computational power. Even tough nowadays QPs are easily handled by embedded micro-controllers, when nonlinear prediction models are used, NPs are generated with tough numerical burden. Thereof, in real-time settings, nonlinear MPC applications become rather complicated.
- Stability of the corresponding closed-loop can only be ensured under some conditions, which usually relate to computing, before the implementation, some stabilising ingredients that are embedded to the MPC optimsaiton. This topic is detailed in the sequel.

2.7 Optimality, stability, and MPC

The application of MPC consists in the online solution of an optimisation problem, written with regard to the future closed-loop dynamics. Usually, the cost function is written in order to weight the difference between the states trajectories to a given equilibrium, and the same goes for the control effort. Nevertheless, as stated and demonstrated in many seminal papers, e.g. [Scokaert, Mayne, and Rawlings 1999; Mayne et al. 2000; Limón et al. 2006b; Besselmann, Lofberg, and Morari 2012], even though MPC applies an optimal control action at each instant, there are no implications that this leads to stability of the resulting closed-loop. That is: the **optimality of MPC does not imply in stability**, at least *a priori* (in the general case).

In order to complete this discussion, let us recall some of the arguments provided in [Marruedo 2002]: take the broader case of an infinite-horizon MPC, using a state-feedback formulation. That is, consider $u(k) = \kappa x(k)$, where the feedback gain κ derives from:

$$\begin{aligned} \kappa \quad = \quad & \arg\min_{K} \overbrace{\left(\sum_{i=0}^{+\infty} \ell\left(x(k+i|k), Kx(k+i|k)\right) \right)}^{J_\infty(x(k))} , \\ \text{s.t} \quad & x(k+i+1|k) = f(x(k+i|k), Kx(k+i|k)), \forall i \geq 0 , \\ & Kx(k+i|k) \in \mathcal{U}, \forall i \geq 0 , \\ & x(k+i|k) \in \mathcal{X}, \forall i \geq 0 . \end{aligned} \tag{2.28}$$

Under regular, usual conditions[12], the resulting feedback implies in an asymptotically stable closed-loop for all states $x \in \mathcal{X}$ that posses an existing associated cost $J_\infty(x)$ that is bounded. Therefore, for any asymptotically stable equilibrium point x, there exists a corresponding infinite-horizon stabilising MPC. Nevertheless, although applying MPC via Eq. (2.28) seems like a reasonable option, it must be noticed that this optimal problem can no longer be formulated under a mathematical programming problem, due to the infinite horizon. Thus, the feedback can either be generated under Hamiltonian-Jacobi-Bellman or Euler-Lagrange equations, which are not trivial and change solution according to each function $f(x, u)$, i.e. according to each controlled process (model). In practice, solving the infinite-horizon problem does becomes of only particular interest, since the obtained solution will not be generic nor reproducible for other systems.

Therefore, when using the general concept of MPC with a receding (fixed-size) prediction horizon, stability-related tools must be included in such a way that the closed-loop becomes stable just as in the infinite-horizon case. For such, extensions of Lyapunov theory and optimal costs have been translated along the past decades to the context of MPC. In the following sections, the general conditions for stability under MPC schemes are recalled.

2.7.1 MPC with terminal ingredients

The main formulations of stabilising MPC proposed in the 90's can be summarised by the results provided in the seminal paper by Mayne, Rawlings, Rao and Scokaert [Mayne et al. 2000]. In that work, the authors establish how MPC algorithms with terminal costs and terminal constraints can, under certain conditions, stabilise any nonlinear system subject to constraints. Moreover, sufficient conditions are provided regarding the terminal cost and the terminal region in order for closed-loop stability to be rendered. Along this thesis, this is main philosophy that is used in order to imply closed-loop stability of the developed MPC

[12] Convex compact sets $\mathcal{X}$ and $\mathcal{U}$, observability of the stage cost $\ell(\cdot,\cdot)$, etc, refer to [Marruedo 2002].

algorithms. The set of tools enabled since [Mayne et al. 2000] is often referred to as the approach of MPC with "terminal ingredients". Basically, it is required that:

- The MPC should operate with a terminal constraint $x(k+N_p|k) \in \mathbf{X}_f$, being $\mathbf{X}_f$ an admissible **positive invariant set** for the system dynamics. That is, there must exists a local control law $u = \kappa(x)$ that stabilises the state dynamics within $\mathbf{X}_f$, with admissible state and input trajectories.
- The MPC must contain a terminal cost $V(x(k+N_p|k))$ and this cost must represent a Lyapunov function associated to such local control law, that is:

$$V\left(f(x,\kappa(x)) - V(x) \quad \leq \quad -\ell(x,\kappa(x))\right), \tag{2.29}$$

  for all $x \in \mathbf{X}_f$.
- The terminal control input $u(k+N_p|k)$ should be locally stabilising within $\mathbf{X}_f$, that is: $u(k+N_p|k) \;=\; \kappa(x(k+N_p|k))$.

Remark 8. *Note that Ineq.* (2.29) *can be generated by imposing that the total optimal cost function exhibits monotonicity. Consider the cost function* $J_k = J\left(x(k), U_k\right)$ *as in Eq.* (2.26)*, denoting* $\mathcal{V}_{N_p}(x(k)) \;=\; J^\star\left(x(k), U_k^\star\right)$ *the optimal value of this function with regard to the minimiser* $U_k^\star$*. Furthermore, consider that there exists a sequence of admissible inputs* U_{k+1} *that is based on the optimal solution obtained in the previous instant, i.e.* $U_k^\star$*. Note that this feasible sequence* U_{k+1} *is nothing but the* $N_p - 1$ *terms that haven't been used from the previous sequence, i.e.* $u^\star(k+1|k), \dots, u^\star(k+N_p-1|k)$*, coupled to the last term, the local control law* $\kappa(x^\star(k+N_p|k))$*. Thus, we can ensure that the difference between the MPC cost at instant* $k+1$ *and the previous optimal cost, at instant* k*, is given by:*

$$\begin{aligned} J\left(x(k+1), U_{k+1}\right) - J^\star\left(x(k), U_k^\star\right) \;&=\; -\ell(x(k), u^\star(k|k)) \\ &+\ell\left(x^\star(k+N_p|k), \kappa(x^\star(k+N_p|k))\right) \\ &+V\left(f\left(x^\star(k+N_p|k), \kappa(x^\star(k+N_p|k))\right)\right) \\ &-V\left(x^\star(k+N_p|k)\right) \; . \end{aligned} \tag{2.30}$$

The terms in the second, third and forth line of Eq. (2.30)*, when summed, become negative, assuming that* $V(\cdot)$ *and* $\ell(\cdot)$ *are positive defined. In order to illustrate this fact, denote* $x^\star = x^\star(k+N_p|k)$ *and* $u^\star = \kappa(x^\star)$*. Assume there exists a control invariant terminal set for the controlled system. Then, there must exits a corresponding control Lyapunov function* $V(\cdot)$*, which in turn ensures that*[13] $\ell\left(x^\star, u^\star\right) + V\left(f(x^\star, u^\star)\right) \;\leq\; V\left(x^\star\right)$*. Consequently, by exploiting the use of a terminal cost, we can infer that any generated feasible input sequence* U_k *will generate a following sequence that has a smaller related optimal cost. Thereby, we obtain:*

$$J^\star\left(x(k+1), U_{k+1}\right) - J^\star\left(x(k), U_k\right) \leq -\ell\left(x(k), \kappa(x(k))\right) \; , \tag{2.31}$$

where $\kappa(\cdot)$ *indicates the state-feedback that is implies by applying the first entry of the control sequence* $U_k^\star$*. In sum, we can conclude that Ineq.* (2.29) *implies in the optimal cost* $J^\star(\cdot,\cdot)$ *is a Lyapunov function that decreased with regard to the closed-loop state evolution.*

[13] For further details, refer to [Marruedo 2002, Hypothesis 3.1].

These generic requirements are recalled next under stronger mathematical rigour. Consider, for such, that the following assumptions are valid:

- The terminal set $\mathbf{X}_f$ is a positively invariant set and an admissible set, i.e. $\mathbf{X}_f \subseteq \mathcal{X}$;
- The stage cost $\ell(\cdot)$ is $\mathcal{K}$-class lower bounded, i.e. there must exist a $\mathcal{K}$-class function $\beta_1(\|x\|)$ such that
$$\ell(x,u) \;\geq\; \beta_1(\|x\|), \forall x \in \mathcal{X}, u \in \mathcal{U}\,;$$
- The terminal cost $V(\cdot)$ is $\mathcal{K}$-class upper bounded, i.e. there must exist a $\mathcal{K}$-class function $\beta_2(\|x\|)$ such that $0 < V(x) \;\leq\; \beta_2(\|x\|), \forall x \in \mathcal{X}$.

Definition 2.16 (Recursive feasibility)
An optimisation algorithm is said to be recursively feasible inside a feasibility set $\mathcal{X}_F$ if, for any starting condition $x_0 \in \mathcal{X}_F$, the optimisation is feasible and remains feasible throughout the following instants.

Theorem 1 (Terminal ingredients)**.** *Consider a nonlinear system whose dynamics are described by $x(k+1) = f(x(k), u(k))$ and that the previous assumptions are valid. Consider a rolling-horizon MPC algorithm with terminal cost and constraint. Suppose there exists a local stabilising law $u = \kappa(x)$, within the terminal state set given by $\mathbf{X}_f$. Then, the MPC ensured closed-loop asymptotical stability if the following conditions hold:*

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

(C2) Any consecutive state to x, in closed-loop given by $f(x, \kappa(x))$ lies within $\mathbf{X}_f$;

(C3) The discrete Lyapunov equation is verified within this invariant set, this is, $\forall\, x \,\in\, \mathbf{X}_f$:

$$V\left(f(x,\kappa(x)) - V\left(x\right) \quad \leq \quad -\ell\left(x, \kappa(x)\right)\,; \tag{2.32}$$

(C4) The image of the nominal feedback lies within the admissible control domain: $\kappa(x) \,\in\, \mathcal{U}$;

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

Then, assuming that the initial solution of the MPC problem is feasible, the optimisation is recursively feasible and the controller stabilises the state origin.

The formulation of MPC algorithms enabled via the solution of the optimisation $\mathfrak{P}_k$ as in Eq. (1), with terminal ingredients that satisfy the requirements of Theorem 1 ensures stability of the closed-loop and is, since the seminal work of Mayne at colleagues [Mayne et al. 2000], well established. Let the main elements and implications of Theorem 1 be further discussed:

- Note that Theorem 1 is based on the fact that set $\mathbf{X}_f$ is invariant. If this terminal region is indeed positive invariant, then the set of feasible states is equivalent to the set of states that are stabilisable within N_p steps. Note that the set of stabilisable states within N_p steps is a subset of those stabilisable within $N_p - 1$ steps, and so forth. By exploiting this property, the factibility of the derived predictive control law is implied for all samples.

- Theorem 1 also indicates that the terminal cost should be a Lyapunov function with regard to the system dynamics (C3). As discussed in Remark 8, this condition implies that the optimal cost is strictly decreasing along discrete-time samples and, thus, a closed-loop Lyapunov map for the controlled system. This condition, in turn, ensures the asymptotical stability of the system in closed-loop when subject to constraints.

- This Theorem is quite omnibus, since all kinds of nonlinear systems can be considered, as long as if the states are measurable. In the case of input-output models, an equivalent state-space can be generated based on previous outputs and inputs (details given in Chapter 4). When state-space descriptions are used, but state measurements are not available, the use of state observers and estimates can also be included in the formulation, e.g. [Köhler, Allgöwer, and Müller 2019; Souza, Efimov, and Raïssi 2021].

Remark 9. *As also discussed in [Marruedo 2002], we make two additional comments with regard to the previous stability analyses of MPC using terminal ingredients:*

- *Even though an MPC algorithm with terminal ingredients that satisfy Theorem 1 implies in closed-loop stability, we cannot conclude that the closed-loop trajectory is optimal, i.e. there may exist another trajectory with a smaller associated total cost, unlike in the infinite-horizon case previously discussed.*

- *Theorem 1 also states that it is the* ***feasibility*** *property of an optimal control sequence* $U_k^\star$ ***that implies in closed-loop stability****, which thus always ensures the decay of the associated MPC cost function. Therefore, even if the MPC may stabilise systems even if a local minimal of the optimisation is found, i.e. there is no need for global optimal minima to ensure stability.*

As a final note, we stress that, along this thesis, a major focus is given to ensure that the MPC optimisation problem $\mathfrak{P}_k$ is recursively feasible. Anyhow, in many situations, we are not only interested in ensuring that an MPC algorithm implies in closed-loop stability and remains recursively feasible, but also in determining the region/basin of attraction of the controller, i.e. the set initial conditions that result in converging trajectories and recursively feasible optimisation procedures $\mathfrak{P}_k$.

Definition 2.17 (Region of attraction)
The set of all initial conditions $x_0 = x(0)$ that result in converging system trajectories and generates recursively feasible solutions to $\mathfrak{P}_k$ is denoted $\mathcal{R}_A \subseteq \mathbb{R}^{n_x}$, and called the region (basin or domain) of attraction in closed-loop of the controller.

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, e.g. [Peaucelle et al. 2012].

Therefore, a simple approximate is a closed $\mathbb{R}^{n_x}$-ball $\mathcal{R}_E$ such that $\mathcal{R}_E \subseteq \mathcal{R}_A$. Accordingly, we can use level sets of the associated candidate Lyapunov terminal cost $V(\cdot)$ in order to compute this estimated region, as done in [Jungers and Castelan 2011]. Thus, we associate the region of attraction of the MPC with the Lyapunov terminal map that ensures dissipativity of the MPC cost J_k. Considering that there exists a generic quadratic Lyapunov candidate function $V(\cdot)$, the associated level set is given by:

$$\mathcal{L}_V(\mu) \;=\; \{x \in \mathbb{R}^{n_x} \,|\, V(x) \leq \mu\} \,. \tag{2.33}$$

2.7.2 Stability in MPC without terminal ingredients

We can also ensure closed-loop stability for systems controlled using MPC without the use of terminal ingredients. A general result on this topic is presented in [Grüne and Pannek 2017, Chapter 6]. The main characteristic of MPC that ensure stability without terminal ingredients is that the stability-verification path requires *a priori* verification, as done in [Cisneros and Werner 2018]. This verification is closely related to the size of the prediction horizon and the form of the MPC cost function and constraints.

For the synthesis of such stabilising predictive controllers, the setup and the algorithm are exactly the same those related to Eq. (2.27); the only difference, obviously, is that the terminal cost and terminal constraints are removed. Then, instead of synthesising the terminal elements that imply stability within an invariant set, the following suitable stabilising conditions are checked regarding the cost controllability and decay:

- Consider that the stage cost $\ell(\cdot)$ is $\mathcal{K}$-class lower bounded, i.e. there must exist a $\mathcal{K}$-class function $\beta_1(\|x\|)$ such that

$$\ell(x,u) \;\geq\; \beta_1(\|x\|), \forall x \in \mathcal{X}, u \in \mathcal{U}\,;$$

- Consider that there exists a scalar $\mu_\kappa \geq 1$ such that for any admissible state $x(k) \in \mathcal{X}$ and any prediction horizon size $N_p \geq 1$, the optimisation program in Eq. (2.27) (without the terminal ingredients) is feasible and the value function[14] $\mathcal{V}_{N_p}(x(k)) := J(x(k), U_k^\star)$ satisfies:

$$\mathcal{V}_{N_p}(x) \;\leq\; \mu_\kappa \ell(x,u), \,\forall x \in \mathcal{X},\, u \in \mathcal{U}\,. \tag{2.34}$$

- Then, consider there exists a positive horizon lower bound $N_0 > 0$ such that, for any prediction horizon $N_p > N_0$, the resulting closed-loop is asymptotically stable, following an admissible input-state trajectory (i.e. $x(k) \in \mathcal{X}$ and $u(k) \in \mathcal{U}$, $\forall k \geq 0$). Moreover, consider that there exists an unit scalar $\lambda_\kappa \in (0\,,\,1]$ such that the following sub-optimality estimate holds:

$$\sum_{k=0}^{+\infty} \ell(x(k),u(k)) \;\leq\; \frac{\mathcal{V}_{N_p}(x(0))}{\lambda_\kappa} \;\leq\; \frac{\mathcal{V}_{+\infty}(x(0))}{\lambda_\kappa}\,.$$

[14]Recall that $U_k^\star$ is the control sequence minimiser to the optimisation, whereas $X_k^\star$ is the corresponding sequence of states (predicted trajectory).

Basically, the prior conditions ensure that the optimal MPC cost (value function) is dissipative and upper bounded, exhibiting an associated Ineq. (2.31) that holds. This result has been exploited thoroughly in [Grüne et al. 2010; Grüne and Palma 2015] and [Grüne, Pirkelmann, and Stieler 2018], with analogous continuous-time conditions presented in [Reble and Allgöwer 2012]. As a final remark, we note that the synthesis of stabilising MPC schemes without terminal ingredients imply in finding a sufficiently large lower bound horizon N_0 such that the sub-optimality estimate holds (which is, in general, a tricky pursuit).

2.8 Some final comments

In this Chapter, the main concepts regarding LPV systems and MPC algorithms have been detailed. Specifically, the classes and types of LPV systems have been discussed, and also how differential inclusion can be exploited in order to generate qLPV models for nonlinear systems. Furthermore, general guidelines on stability and the use of stability-related tools have been prescribed, with specific focus to the case of LPV system. We also explained how to set up and apply a model-based predictive control algorithm, and how these are able to imply in closed-loop stability if adequate terminal ingredients are synthesised. The main content of this thesis begins in the following Chapter, where the application of MPC algorithms for nonlinear systems using LPV models is further discussed.

Part II

Gain-scheduled design

Chapter 3

Estimating scheduling trajectories

As detailed in the Introduction, the application of MPC to LPV systems is of utter interest for the control of nonlinear, time-varying processes. Yet, it comes with an intrinsic complication: the so-called scheduling parameters, which coordinate the LPV dynamics, are *a priori* unknown along a future prediction horizon. In this Chapter, we debate this issue, in the context of gain-scheduled design. Specifically, we detail the four main approaches available that can be used to compute estimates for the future scheduling trajectories (and also how to operate the corresponding MPC algorithm). These methods are:

(i) *Frozen* estimates, when the controller assumes a prediction as if the parameters would remain constant along the horizon, as deployed in [Morato, Sename, and Dugard 2018; Alcalá, Puig, and Quevedo 2019; Morato et al. 2020e];

(ii) Identification-based estimates, for which auto-regressive equations are used model the behaviour of the future scheduling variables, as proposed in [Morato, Normey-Rico, and Sename 2019];

(iii) Iterative rules, which generate estimates based on sequential iterations of the MPC optimisation, per sample, exploiting the known relationship between the scheduling variables and the future system endogenous variables (only possible for the qLPV case). This approach is the current state-of-the-art in qLPV MPC literature, e.g. [Cisneros, Sridharan, and Werner 2018; Cisneros and Werner 2019; Cisneros and Werner 2020];

(iv) Extrapolation schemes, which generate estimates based on a simple Taylor expansion condition (again, only possible for the qLPV case), as proposed originally in [Morato, Normey-Rico, and Sename 2022b] and also applied in [Morato, Normey-Rico, and Sename 2021b; Morato et al. 2023a; Morato 2023].

With regard to these alternatives, we present several different simulation benchmark examples from the literature, in order to illustrate and discuss their main features. These results are also compared to state-of-the-art techniques.

Remark 10. *The developments presented in this Chapter correspond (in parts) the works published in [Morato, Normey-Rico, and Sename 2021c; Morato, Normey-Rico, and Sename 2022a; Morato 2023; Morato et al. 2023a; Morato, Normey-Rico, and Sename 2023a]. Specifically, the extrapolation method (approach (iv)) is one of the main contributions derived from* *this thesis, as formalised in depth in [Morato, Normey-Rico, and Sename 2022b]. This topic comprises, in fact, Objective (i) of this thesis (refer to Chapter 1, Section 1.5). As of this, special attention is given to this approach. We also note that another contribution from this work is the recursive Least-Squares (LS) procedure presented in [Morato, Normey-Rico, and Sename 2019] (approach (i)).*

3.1 Motivations

As discussed in the introduction of this thesis, we recall that, nowadays, there exist several efficient solver-based NMPC algorithms [Zhang, Li, and Liao 2019; Rathai et al. 2018; Gros et al. 2020]. Nevertheless, these methods are coined using approximations of the nonlinear optimisation problem that arises when applying nonlinear MPC. Exact NMPC formulations require the online solution of such NPs, which is typically not solvable within small sampling periods (in the case of fast, embedded control systems). In sum, the application of NMPC for real-time process has an inherent impediment of solving the exact NP during the online implementation, as pointed out by all major systematic reviews on the topic [Camacho and Bordons 2007; Allgöwer and Zheng 2012].

We argue insistently in this work that an elegant approach to resolve this issue (reliving the computational toughness of NMPC) is to replace the nonlinear system model by an LPV one[1]. Thereby, as discussed in [Morato, Normey-Rico, and Sename 2020a], in general, the resulting optimisation is able to be run much faster, since the original NP is replaced by a QP (or an SQP, in some cases). Next, we discuss some complementary aspects:

- For simplicity, let us consider the application of a state-feedback MPC, enable through the solution of the following optimisation[2] i, at each discrete-time sampling instant k:

$$\begin{aligned}
\min_{U_k} \quad & \overbrace{\left(\sum_{i=1}^{N_p-1} \ell\left(x(k+i|k), u(k+i-1|k)\right)\right)}^{J_k = J(x(k),U_k)} \\
& + V\left(x(k+N_p|k)\right), \\
\text{s.t.} \quad & \text{Process model}, \forall i \in \mathbb{N}_{[1,N_p]}, \\
& x(k+i|k) \in \mathcal{X}, \; \forall i \in \mathbb{N}_{[1,N_p]}, \\
& u(k+i-|k) \in \mathcal{U}, \; \forall i \in \mathbb{N}_{[1,N_p]}, \\
& x(k+N_p|k) \in \mathbf{X}_f,
\end{aligned} \tag{3.1}$$

[1] This can be done either by the means of high-fidelity LPV models with coordinated exogenous scheduling [Tóth 2010], or via exact qLPV representations, enabled via differential inclusion [Hoffmann and Werner 2014].

[2] As explained in Chapter 2, Section 2.7, the optimisation cost J_k is composed of a stage cost and a terminal offset cost, for stability-related purposes Furthermore, the terminal constraint $x(k+N_p|k) \in \mathbf{X}_f$ is included for the same purposes (being $\mathbf{X}_f$ a control postively invariant set).

Again, we use $x \in \mathcal{X}$ to denote the state admissibility constraints, $u \in \mathcal{U}$ for input admissibility, $x \in \mathbf{X}_f$ for terminal admissibility, and U_k to denote the vector of predicted control efforts given inside the MPC horizon, i.e.:

$$U_k = \begin{bmatrix} u(k|k)^T & u(k+1|k)^T & \dots & u(k+N_p-1|k)^T \end{bmatrix}^T . \tag{3.2}$$

- Any MPC algorithm, as the prior, takes into account the dynamics of the controlled system over a prediction horizon of N_p (the process model constraint). Thus, in the general nonlinear setting, considering a discrete-time generic nonlinear state dynamic $x(k+1) = f\left(x(k), u(k)\right)$, the following sequence of model-based rules are evaluated (internally by the optimiser):

$$\begin{aligned} x(k+1|k) &= f\left(x(k), u(k|k)\right) , \\ x(k+2|k) &= f\left(f\left(x(k), u(k|k)\right), u(k+1|k)\right), \\ x(k+3|k) &= f\left(f\left(f\left(x(k), u(k|k)\right), u(k+1|k)\right), u(k+2|k)\right), \end{aligned} \tag{3.3}$$

and so on, up to the N_p-th prediction.

- Note that the (original) solution of $\min_{U_k} J_k$ (using a nonlinear model), requires the evaluation of the predictions given in Eq. (3.3), which makes the optimisation a nonlinear program (which may even be non-convex).

- Nevertheless, when we replace the nonlinear model by an LPV one, i.e. by $x(k+1) = A(\rho(k))x(k) + B(\rho(k))u(k)$, the sequence of model-based prediction rules that are evaluated by the optimiser becomes:

$$x(k+1|k) = A(\rho(k))x(k) + B(\rho(k))u(k|k) , \tag{3.4}$$

$$\begin{aligned} x(k+2|k) &= A(\rho(k+1))A(\rho(k))x(k) \\ &+ A(\rho(k+1))B(\rho(k))u(k|k) \\ &+ B(\rho(k+1))u(k+1|k) , \end{aligned} \tag{3.5}$$

$$\begin{aligned} x(k+3|k) &= A(\rho(k+2))A(\rho(k+1))A(\rho(k))x(k) \\ &+ A(\rho(k+2))A(\rho(k+1))B(\rho(k))u(k|k) \\ &+ A(\rho(k+2))B(\rho(k+1))u(k+1|k) \\ &+ B(\rho(k+2))u(k+2|k) , \end{aligned} \tag{3.6}$$

and so forth, up to the N_p-th prediction:

$$\begin{aligned} x(k+N_p|k) &= A(\rho(k+N_p-1)) \dots A(\rho(k))x(k) \\ &+ A(\rho(k+N_p-1)) \dots A(\rho(k+1))B(\rho(k))u(k|k) \\ &+ A(\rho(k+N_p-1)) \dots A(\rho(k+2))B(\rho(k+1))u(k+1|k) + \dots \\ &+ B(\rho(k+N_p-1))u(k+N_p-1|k) . \end{aligned} \tag{3.7}$$

Notice that, in the LPV case, these predictions require the values of the future scheduling variables, i.e. $\rho(k+1)$, $\rho(k+2)$, and so forth. We consider that the future "scheduling trajectory" can be compacted as follows:

$$P_k = \begin{bmatrix} \rho(k)^T & \rho(k+1)^T & \dots & \rho(k+N_p-1)^T \end{bmatrix}^T ,$$

where only $\rho(k)$ is, in practice, known.

- Yet, the full scheduling trajectory vector P_k can be used to analytically express the complete vector of state predictions, i.e. $X_k = [x(k+1|k)^T, \ldots, x(k+N_p|k)^T]^T$. As shown in [Cisneros and Werner 2020], it follows that:

$$X_k = \mathcal{A}(P_k)x(k) + \mathcal{B}(P_k)U_k\,, \tag{3.8}$$

where $\mathcal{A}(P_k) \in \mathbb{R}^{(n_x N_p)\times n_x}$ and $\mathcal{B}(P_k) \in \mathbb{R}^{(n_x N_p)\times(n_u N_p)}$ are given in Eqs. (3.9)-(3.10). In practice, these matrices maintain form at each sample, and thus can be computed rapidly. Thereby, the MPC law is enabled through the online solution of $\min_{U_k} J_k$ subject to Eq. (3.8) and constraints, which is a QP, as long as P_k is known[3].

- Overall, since when we replace the nonlinear model by an LPV one, the state predictions, computed through Eq. (3.11), becomes dependent on the future scheduling trajectory P_k, as gives Eq. (3.8). Thus, we can use estimates of P_k in order to solve the MPC problem efficiently.

$$\mathcal{A}(P_k) = \begin{bmatrix} A(\rho(k)) \\ A(\rho(k+1))A(\rho(k)) \\ \vdots \\ A(\rho(k+N_p-1))A(\rho(k+N_p-2))\ldots A(\rho(k)) \end{bmatrix}, \tag{3.9}$$

$$\mathcal{B}(P_k) = \begin{bmatrix} B(\rho(k)) & 0 & \ldots \\ A(\rho(k+1))B(\rho(k)) & B(\rho(k+1)) & \ldots \\ & \vdots & \\ A(\rho(k+N_p-1))\ldots A(\rho(k+1))B(\rho(k)) & A(\rho(k+N_p-1))\ldots A(\rho(k+2))B(\rho(k+1)) & \ldots \end{bmatrix}. \tag{3.10}$$

Based on the previous context, in this Chapter we discuss how can these scheduling trajectory guesses be efficiently formulated, and which mechanisms can be used. The main elements of the sequel are the following:

1. We discuss the four available alternatives for scheduling trajectory estimates, for both LPV and qLPV models. Accordingly, we discuss their main features, required assumptions, and properties.
2. More details are given regarding the extrapolation method for qLPV scheduling parameters trajectories. Specifically, we present sufficient conditions that enable a convergent extrapolation scheme, conceived with regard to the form and class of the scheduling function and the robustness of the corresponding MPC algorithm.

[3] Note that we can easily re-write the cost function J_k in terms of X_k. For instance, $\sum_{i=0}^{N_p-1} \ell(x(k+i+1|k), u(k+i|k))$, with $\ell(x,u) = \|x\|_Q + \|u\|_R$, becomes $X_k^T Q X_k + U_k^T R U_k$. Furthermore, due to Eq. (3.8), this cost is equivalent to $U_k^T\left((\mathcal{B}(P_k))^T Q \mathcal{B}(P_k) + R\right)U_k + 2\left((\mathcal{A}(P_k)x(k))^T Q\mathcal{B}(P_k)\right)U_k + \left((\mathcal{A}(P_k)x(k))^T Q(\mathcal{A}(P_k)x(k))\right)$, which is clearly quadratic on the decision vector U_k.

3. Illustrative examples, considering benchmark models from the literature, are presented in order to demonstrate the effectiveness and main characteristics of each mechanism.

3.2 Preliminaries

Before detailing the available methods to estimate P_k, we first give more concrete details on the investigated problem. In this Chapter, we specifically consider the class of affine discrete-time LPV systems:

$$x(k+1) \;=\; A(\rho(k))x(k) + B(\rho(k))u(k)\,, \tag{3.11}$$

where $x \in \mathbb{R}^{n_x}$, $u \in \mathbb{R}^{n_u}$, $\rho \in \mathbb{R}^{n_p}$ and $A : \mathbb{R}^{n_p} \to \mathbb{R}^{n_x} \times \mathbb{R}^{n_x}$, $B : \mathbb{R}^{n_p} \to \mathbb{R}^{n_x} \times \mathbb{R}^{n_u}$ are continuous affine maps. The states are measurable for all sampling instants and the scheduling parameters belong to a convex scheduling set $\mathcal{P} \subset \mathbb{R}^{n_p}$. Also, in the qLPV case, we assume that[4] $\rho(k) = f_\rho\left(x(k)\right)$. We also consider that the following set of assumptions is verified with regard to the controlled process:

Set of Assumptions 1.

- *The system is asymptotically stable in closed-loop for all $\rho(k) \in \mathcal{P}$;*
- *The admissibility of states, inputs and scheduling parameters are given by known compact sets:*

$$\begin{aligned}
\mathcal{X} &:= \left\{x \in \mathbb{R}^{n_x} \,|\, \underline{x}_j \leq x_j \leq \overline{x}_j\,,\, \forall j \in \mathbb{N}_{[1,n_x]}\right\}\,, \\
\mathcal{U} &:= \left\{u \in \mathbb{R}^{n_u} \,|\, \underline{u}_j \leq u_j \leq \overline{u}_j\,,\, \forall j \in \mathbb{N}_{[1,n_u]}\right\}\,. \\
\mathcal{P} &:= \left\{\rho \in \mathbb{R}^{n_p} \,|\, \underline{\rho}_j \leq \rho_j \leq \overline{\rho}_j\,,\, \forall j \in \mathbb{N}_{[1,n_p]}\right\}\,.
\end{aligned}$$

- *The variation rate of the scheduling parameters $(\rho(k+1) - \rho(k))$ is bounded for all sampling instants $k \geq 0$.*
- *The variables x, u and ρ are component-wise energy-bounded, in the sense of the 2 norm. Thus, it follows that $\|x\|_2 \leq \overline{x}$, $\|u\|_2 \leq \overline{u}$ and $\|\rho\|_2 \leq \rho_{max}$.*
- *In the qLPV case, the scheduling proxy satisfies[5] $f_\rho(\mathcal{X}) \subseteq \mathcal{P}$.*

Remark 11. *The necessity of closed-loop stability is standard. For such, there must exist a feedback $u := \kappa(\cdot)x \in \mathbb{R}^{n_x}$ such that $x^+ = \left(A(\rho) + B(\rho)\kappa(\cdot)\right)x$ is stable for all $\rho \in \mathcal{P}$. The feedback gain $\kappa(\cdot)$ could be either parameter-dependent (i.e. $\kappa(\rho)$) or constant (i.e. κ), depending on the control synthesis. For generality, we henceforth consider the MPC state-feedback as a parameter-dependent gain $\kappa(\rho)$. We also stress that, in some cases, one can*

[4]We note that the following discussions are made with regard to the case of state-dependent scheduling proxies, i.e. $\rho = f_\rho(x)$, yet all methods also apply to the more general case of $\rho = f_\rho(x, u)$.

[5]Here, we use an abusive notation. In fact, we mean that for all $x \in \mathcal{X}$, we obtain $f_\rho(x) \in \mathcal{P}_f \subseteq \mathcal{P}$.

only ensure that the MPC provides a closed-loop asymptotically stable system if the model is open-loop controllable. That is: this hypothesis can also be viewed from the perspective of closed-loop stabilisability. These issues should be taken into account in the control design step, which are not the focus of this Chapter. Refer to [Morato, Normey-Rico, and Sename 2020a; Hanema, Tóth, and Lazar 2021] for further details on the synthesis of stable closed-loops for LPV systems under MPC algorithms. Corresponding discussions have also been presented in Chapter 2, Section 2.5 and in Chapter 5.

Proposition 1. *Assume the constraints on states, inputs and scheduling parameters ($x(k) \in \mathcal{X}, u(k) \in \mathcal{U}$, and $\rho(k) \in \mathcal{P}$, respectively) are known, compact, and valid for all sampling instants. Thus, the state deviation variable $\Delta x(k) := (x(k+1) - x(k))$ is also bounded to a compact set $\Delta\mathcal{X}$.*

Proof. Take $\Delta x(k) := (x(k+1) - x(k)) = (A(\rho(k)) - I_{n_x})\, x(k) + B(\rho(k))u(k)$. Since $(x, u, \rho) \in (\mathcal{X} \times \mathcal{U} \times \mathcal{P})$ for all instants k, it follows that $\Delta\mathcal{X} := \left\{\Delta x \in \mathbb{R}^{n_x} \,|\, \|\Delta x_j\| \leq \overline{\Delta x}_j \,, \forall j \in \mathbb{N}_{[1,n_x]}\right\}$. The component-wise bounds $\overline{\Delta x}_j$ can be determined either by interval arithmetics or optimisation. □

Regarding MPC application, operated via the online solution of Eq. (3.1), where $u^\star(k|k)$ from the solution $U_k^\star$ is applied to the plan, we consider the following main (stage) cost and terminal cost that compose the MPC optimisation cost J_k:

$$\ell\,(x, u) = \|x\|_Q + \|u\|_R = x^T Q x + u^T R u\,, \tag{3.12}$$

$$V(x) = \|x\|_P = x^T P x\,, \tag{3.13}$$

being P, Q and R positive-definite weighting matrices. Note that Q and R are weights that determine the control objective, while P is, in general, defined with regard to stability purposes[6].

3.3 Approach (i): Frozen scheduling trajectories

The first approach seen in the literature to handle the unavailability issue regarding the scheduling trajectory P_k is based on a quite simple idea: from the viewpoint of the MPC operation, at each sampling instant, we assume that the scheduling variables will remain constant (thus, *frozen*). That is, P_k is replaced by $\begin{bmatrix} \rho(k)^T & \dots & \rho(k)^T \end{bmatrix}^T$ to generate the future state predictions X_k. In many practical applications with slowly-varying scheduling parameters, such as the case of renewable-energy systems [Pipino et al. 2020b; Morato et al. 2020e], this method is rather interesting, because the uncertainty propagation along the prediction horizon is negligible.

Yet practically-relevant, the resulting MPC scheme that is generated when using frozen scheduling trajectories may suffer from severe robustness-related issues. Recall that the

[6] In some cases parameter-dependent terminal ingredients ($V(\cdot)$ and $\mathbf{X}_f$) are used. We focus herein in the quadratic formulations. Refer to Chapter 5 for discussions regarding the parameter-dependent case.

scheduling parameters enter the prediction model multiplicatively. Thus, when the MPC is based on a nominal model[7] in the form of $x(k+j|k) = A(\rho_k)x(k+j-1|k) + B(\rho_k)u(k+j-1|k)$, $\forall j \in \mathbb{N}_{[1,N_p]}$. Then, the corresponding model-process prediction uncertainty is, for the one-step ahead prediction, given by:

$$\begin{aligned} \mu(k+1) &= A(\rho(k+1))x(k+1|k) - A(\rho_k)x(k+1|k) \\ &+ B(\rho(k+1))u(k+1|k) - B(\rho_k)u(k+1|k)\,, \end{aligned} \tag{3.14}$$

which, evidently, depends not only on the real future scheduling parameters $\rho(k+1)$, but also on the future state and input trajectories.

Remark 12. *In the qLPV setting, Eq.* (3.14) *exhibits an even harder nonlinear dependency on states and inputs, which turns out far more complicated. That is, taking* $\rho(k+j|k) = f_\rho(x(k+j))$*, we obtain:*

$$\begin{aligned} \mu(k+1) &= A(f_\rho(x(k+1|k)))x(k+1|k) - A(\rho_k)x(k+1|k) \\ &+ B(f_\rho(x(k+1|k)))u(k+1|k) - B(\rho_k)u(k+1|k)\,, \end{aligned} \tag{3.15}$$

which means that the uncertainty propagation carries along the horizon influences from the real and predicted future states, as well as future inputs. Depending on the form of the parameter-dependent matrices $A(\cdot)$ *and* $B(\cdot)$*, this uncertainty may hardly affect the closed-loop performances and rapidly grown in magnitude.*

For the purpose of further discussion, we exploit an affine dependency on the system matrices in order to construct the following one-step-ahead uncertainty:

$$\begin{aligned} \mu(k+1) &= A(\rho(k+1) - \rho_k)x(k+1|k) \\ &+ B(\rho(k+1) - \rho_k)u(k+1|k)\,, \end{aligned} \tag{3.16}$$

where, for the case of bounded scheduling parameters' variations over samples, i.e. $\delta\rho(k) = \rho(k+1) - \rho(k) \in \delta\mathcal{P}$, it follows that:

$$\|\rho(k+j) - \rho_k\| \;\leq\; j\overline{\delta\rho}\,. \tag{3.17}$$

Knowing that the uncertainty grows exponentially with respect to the state predictions, we can concretely use the following lower-bound:

$$\|\mu(k+j)\| \;\geq\; \left(A((N_p-1)\overline{\delta\rho})\overline{x} + B((N_p-1)\overline{\delta\rho})\overline{u}\right)\,. \tag{3.18}$$

We stress that an upper-bound $\overline{\mu} \geq \|\mu(k+j)\|$ can be found using interval algebrae, for instance. In practice, the corresponding MPC, in order to stabilise the controlled system, should robustly tolerate this bounded uncertainty term $\mu(k+j)$. Nevertheless, under relatively simple assumptions, the uncertainty that arises grows not only with regard to all endogenous

[7] Here, we use ρ_k to emphasise that the model is based on a *frozen* scheduling trajectory $\rho(k+j) = \rho(k) = \rho_k$, $\forall j \in \mathbb{N}_{[1,N_p]}$.

variables, but also with regard to the size of the prediction horizon, as gives Eq. (3.18). Quite easily, the bounds on $\mu(k+j)$ may be even larger than the available state space, thus putting to an end any possible performance certificate of an MPC synthesised on the basis of a frozen LPV model.

Synthetically, the so-called frozen LPV MPC design alternative has a major theoretical drawback: for it to ensure closed-loop stability, excessive robustness may have to be implied, which thus shrinks the corresponding region of attraction enabled by the controllers. From a philosophical perspective, in order to design a frozen LPV MPC algorithm which ensures stability, even under standard assumptions, we actually seek to determine a single controller which is able to (quadratically) stablise **all** LTI models in the from of $x(k+1) = A(\rho_k)x(k) + B(\rho_k)u(k)$, generated by fixed values of $\rho_k \in \mathcal{P}$. Moreover, the controller must ensure that stability is maintained even if these model change within samples, which is, of course, a tough problem.

In any case, MPC algorithms for LPV systems described on the basis of frozen scheduling trajectories have great practical value. For the case of LPV systems with slowly-varying scheduling parameters, i.e. for small bounds on $\|\delta\rho(k)\|$, this approach is standard, and widely exploited in the literature[8]. What is done, in many works, is to simply neglect the existence of the uncertainty propagation $\mu(k)$, assuming that the MPC will enable a closed-loop that is robustly stable with regard to this variable.

Dating from an original theoretical paper from 2003, [Casavola, Famularo, and Franzè 2003], many works with practical focus have seen been presented by using this methods of considering frozen LPV models at each sample, e.g. in the following recent papers [Cisneros and Werner 2017b; Alcalá, Puig, and Quevedo 2019; Rodriguez-Guevara et al. 2021; Cavanini, Ippoliti, and Camacho 2021]. We can even argue, from the recent survey on MPC algorithms using LPV models, as presented in [Morato, Normey-Rico, and Sename 2020a], that the frozen, gain-scheduled approach is the **standard** method seen in the literature, with most number of practical applications being registered.

3.3.1 Simulation results

Finally, before moving on to the discussion of the following approach, we present some simulation results in order to illustrate and discuss the characterstics of the frozen LPV MPC approach. We emphasise that the most attractive feature of the approach is that the generated prediction model is basically LTI, from the viewpoint of each sample. Thus, no calculations are required to compute the scheduling trajectory P_k, and the resulting optimisation is a QP by nature, enabling solutions of the MPC algorithm in real-time.

Consider a continuous stirred-tank reactor (CSTR) system, composed of a reactant A that

[8] In Chapter 4, we provide a state-feedback LPV MPC solution based on a frozen scheduling trajectory, where the derived uncertainty bounds are exploited by the means of the bounds on the variations of the scheduling parameters over consecutive samples. As shown therein, we can compensate the uncertainty propagation by shrinking the constraints of the MPC.

becomes the product B by the means of an irreversible and exothermic chemical reaction. The goal of this system is to regulate the concentration of A, C_A, the reactor temperature, T, and the reaction volume, V, by manipulating the output process flow rate, Q_s, and coolant flow rate, Q_c. The reaction takes place in a stirred cylindrical tank, as shown in Figure 3.1, from [Pipino et al. 2020a]. The dynamics of the CSTR system are given by the following set of nonlinear differential equations:

$$\begin{cases} \frac{dV(t)}{dt} &= Q_o - Q_s(t)\,, \\ \frac{dC_A(t)}{dt} &= \frac{Q_o}{V(t)}\left(C_{Ao} - C_A(t)\right) - k_0 e^{\frac{-\frac{E}{R}}{T(t)}} C_A(t)\,, \\ \frac{dT(t)}{dt} &= \frac{Q_o}{V(t)}\left(T_o - T(t)\right) - k_1 e^{\frac{-\frac{E}{R}}{T(t)}} C_A(t) \\ & + \; k_2 \frac{Q_c(t)}{V(t)}\left(1 - e^{\frac{-k_3}{Q_c(t)}}\right)\left(T_{co} - T(t)\right)\,, \end{cases} \tag{3.19}$$

where $V(t)$, $C_A(t)$ and $T(t)$ are the measurable states of the system (respectively, $x_1(t)$, $x_2(t)$ and $x_3(t)$). The output process flow rate is denoted $q_s(t)$, while $q_c(t)$ stands for the cooling flow rate (control signals $u_1(t)$ and $u_2(t)$). Finally, q_o is the process flow rate, C_{Ao} is the feed concentration, k_0 is the reaction rate constant, $\frac{E}{R}$ is the activation energy term, T_o and T_{co} are, respectively, the feed and inlet coolant temperatures, $k_1 = -\Delta H k_0/(\mu C_p)$, $k_2 = \frac{\mu_c C_{pc}}{(\mu C_p)}$ and $k_3 = \frac{hA}{(\mu_c C_{pc})}$, where ΔH is the heat of reaction, μ and μ_c are the liquid densities, C_p and C_{pc} are the specific heats and hA is heat transfer term. The considered system parameters are reported in Table 3.1.

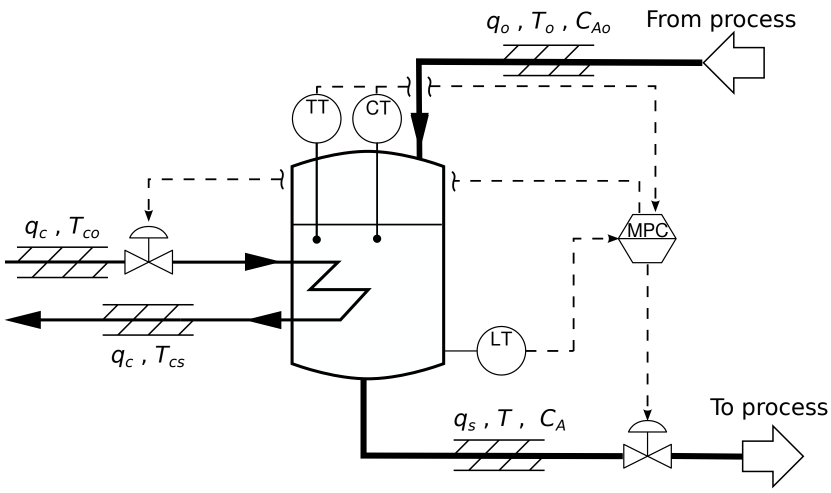


Figure 3.1: CSTR system: Illustrative diagram, from [Pipino et al. 2020a].

Following the lines of [Hoffmann and Werner 2014], an LPV model is for this CSTR system is generated in [Bernardi 2021], by the means of Jacobian linearisation. The obtained LPV model has the form of $x(k+1) = A(\rho(k))x(k) + B(\rho(k))u(k)$, with two scheduling parameters

Table 3.1: CSTR system: Model parameters and constraints.

Parameter	Value	Parameter	Value
Q_o	$100\,\text{l} \ \ \text{min}^{-1}$	C_{Ao}	$1\,\text{mol} \ \ \text{l}^{-1}$
T_o	$350\,\text{K}$	T_{co}	$350\,\text{K}$
k_0	$7.2 \times 10^{10}\,\text{min}^{-1}$	$\frac{E}{R}$	$1 \times 10^4\,\text{K}$
ΔH	$-2 \times 10^5\,\text{cal} \ \ \text{mol}^{-1}$	μC_p	$1000\,\text{cal} \ \ \text{l}^{-1}\,\text{K}^{-1}$
hA	$7 \times 10^5\,\text{cal} \ \ \text{min}^{-1}\,\text{K}^{-1}$	$\mu_c C_{pc}$	$1000\,\text{cal} \ \ \text{l}^{-1}\,\text{K}^{-1}$
k_1	$\Delta H k_0/(\mu C_p)$	k_2	$\mu_c C_{pc}/(\mu C_p)$
k_3	$hA/(\mu_c C_{pc})$	$V(t)$	$\in [80, 110]\,\text{l}$
$C_A(t)$	$\in [0.03, 0.12]\,\text{mol} \ \text{l}^{-1}$	$T(t)$	$\in [440, 460]\,\text{K}$
$Q_s(t)$	$\in [90, 110]\,\text{l} \ \ \text{min}^{-1}$	$Q_c(t)$	$\in [80, 100]\,\text{l} \ \ \text{min}^{-1}$
V_{lin}	$[80, 100, 110]\,\text{l}$	T_{lin}	$[440, 450, 460]\,\text{K}$

$\rho(t) = \begin{bmatrix} V(t) & T(t) \end{bmatrix}^T$, and the following polytopic matrices:

$$\begin{cases} A(\rho(k)) = \sum_{j=1}^{4} \lambda_j(\rho(k)) A_j \,, \\ B(\rho(k)) = \sum_{j=1}^{4} \lambda_j(\rho(k)) B_j \,, \end{cases} \tag{3.20}$$

where $\sum_{j=1}^{4} \lambda_j(\rho(k)) = 1$, $0 \leq \lambda_j(\rho(k)) \leq 1$, $\forall j \in \mathbb{N}_{[1,4]}$ and:

$$\begin{cases} A_j = \begin{bmatrix} 0 & 0 & 0 \\ \frac{Q_o(C_{A_j} - C_{Ao})}{V_j^2} & -\frac{Q_o}{V_j} - k_0 e^{\frac{-E}{RT_j}} & -\frac{E C_{A_j} k_0 e^{\frac{-E}{RT_j}}}{RT_j^2} \\ A_{31} & -k_1 e^{\frac{-E}{RT_j}} & A_{33} \end{bmatrix}, \\ B_j = \begin{bmatrix} -1 & 0 \\ 0 & 0 \\ 0 & B_{32} \end{bmatrix}, \\ C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \end{cases} \tag{3.21}$$

$$\begin{cases} A_{31} = \frac{Q_o(T_j - T_o)}{V_j^2} - \frac{k_2 Q_{c_j} (e^{\frac{-k_3}{Q_{c_j}}} - 1)(T_j - T_{co})}{V_j^2} \,, \\ A_{33} = \frac{k_2 Q_{c_j} (e^{\frac{-k_3}{Q_{c_j}}} - 1)}{V_j} - \frac{Q_o}{V_j} - \frac{E C_{A_j} k_1 e^{\frac{-E}{RT_j}}}{RT_j^2} \,, \\ B_{32} = \frac{k_2 (T_j - T_{co}) \left(e^{\frac{-k_3}{Q_{c_j}}} - 1 \right)}{V_j} + \frac{k_2 k_3 (T_j - T_{co}) e^{\frac{-k_3}{Q_{c_j}}}}{V_j} . \end{cases} \tag{3.22}$$

This LPV model is (Euler) discretised with a sampling period of $T_s = 3$ s. In the sequel, we present the obtained results of a gain-scheduled MPC algorithm based on the frozen iterations of this LPV model. For comparisons, we also include the results obtained with a full-blown NMPC algorithm. Note that the real nonlinear phenomenological process model from Eq. (3.19) is used to simulate the real process behaviour.

The control system is set to track a reference objective of 100 l for the reaction volume and 450 K for the reactor temperature, while guaranteeing the constraints on x and u given in the Table 3.1. The prediction horizon is chosen as $N_p = 5$ steps.

Figure 3.2 exhibits the time evolution for the reaction volume, the reactant concentration and reactor temperature with regard to the applied control signals (given in Figure 3.3). We can be seen how both control algorithms are able to drive the system to the target set-point, as envisioned. Nevertheless, we should pay attention to some aspects of the results, as collected in Table 3.2, which presents the IAE index for the controlled variables:

- Since no nonlinear optimisation is involved, the frozen LPV MPC method is able to achieve overall better performances than the nominal MPC, regarding T and C_A. We stress that in this CSTR system, for the considered operational conditions, the nonlinearities exhibit little influence. Furthermore, the variation of the scheduling parameters is small, and thus the resulting uncertainty does not compromise the performance of the frozen LPV MPC approach.

- Most importantly, we also compare the online computational effort with each method, summarised by the t_c index, given in the percentage of the sampling period. This index gives the average time required by the optimisation in order to evaluate the control solution, showing that the LPV MPC can be evaluated much faster than the nominal, nonlinear MPC.

- We also mention that if real-time applications are considered, the frozen-based LPV MPC solution can indeed be an elegant and efficient approach, when the parameter variations are subtle. In practice, the model uncertainties show very diminished impact in the obtained results.

Table 3.2: CSTR system: Performance indexes.

		Frozen LPV MPC	Nominal (nonlinear) MPC
IAE	V	21.59	**20.54**
	C_A	**0.02**	0.02
	T	**2.02**	2.50
t_c		**0.056**%	0.67%

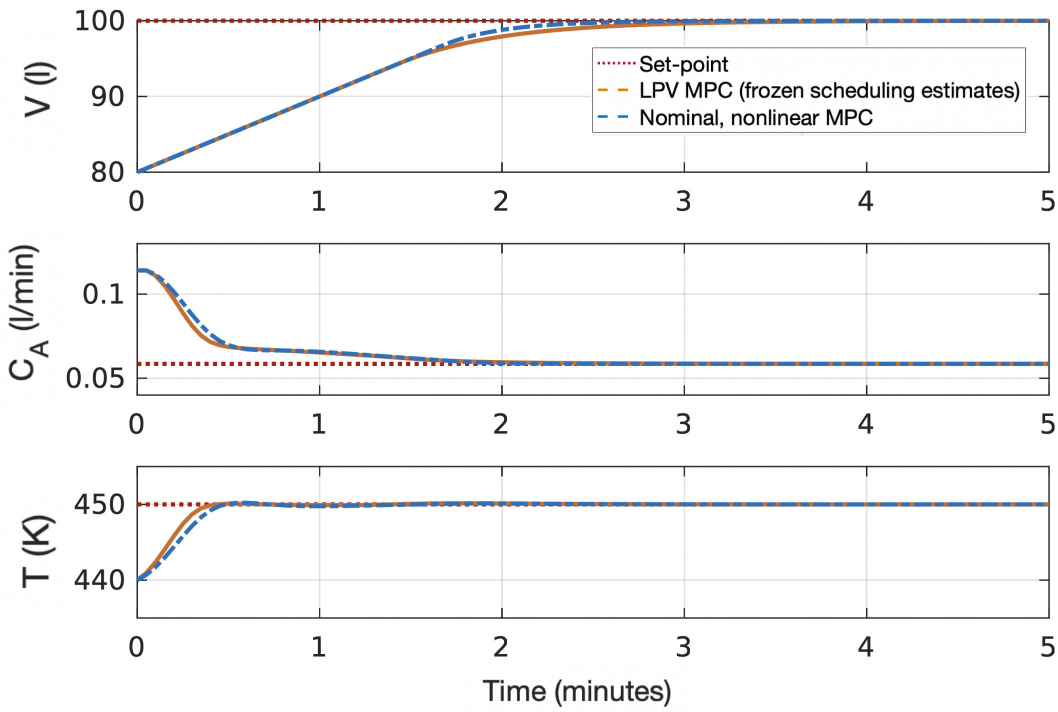


Figure 3.2: CSTR system: Reaction volume, concentration and temperature.

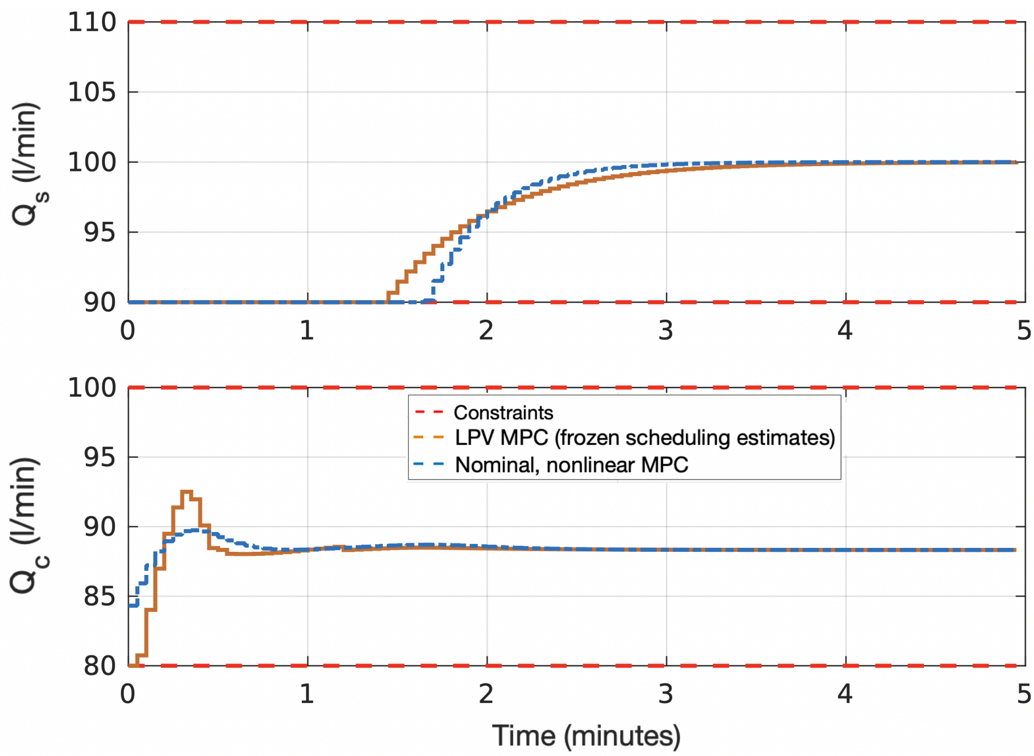


Figure 3.3: CSTR system: Output process and coolant flow rates.

3.4 Approach (ii): Identification-based estimates

The frozen LPV model approach for gain-scheduled MPC synthesis is quite standard in practice. Nevertheless, as evidenced, performances may be compromised when the parameters vary rapidly between sampling periods (and, in the qLPV case, if the controller is too aggressive, enforcing the state trajectories to move too fast and, thus implicitly also making the scheduling parameters vary abruptly). Therefore, we now detail a second alternative to generate estimates for the scheduling trajectories, which is based on an online identification procedure. This method has been proposed within the context of this thesis, and presented in [Morato, Normey-Rico, and Sename 2019]. Its main focus is to provide a better estimate than simply considering the parameters to be constant along the prediction horizon.

Again, we consider that the scheduling variables ρ are only known (or measured) at the current sampling instant k, meaning that the whole future scheduling trajectory behaviour P_k is unknown. Anyhow, we assume that these parameters agree to some linear auto-regressive dynamic behaviour along time. That is, we consider that the signal $\rho(k)$ satisfies the following assumption.

Assumption 1. *There exists a linear auto-regressive model* Π*, in the form of Eq.* (3.23)*, which maps the behaviour of the scheduling parameters* $\rho(k)$ *of the controlled LPV system from Eq.* (3.11)*.*

$$\begin{aligned} \rho(k+N_p) &= a_1\rho(k-(N_p-1)) + \cdots + a_{N_p}\rho(k) \\ &+ b_1\xi(k-(N_p-1)) + \cdots + b_{N_p}\xi(k)\,. \end{aligned} \tag{3.23}$$

In practice, Assumption 1 implies that there exists a discrete-time model with N_p sample delays, which gives the relationship between an input ξ and the scheduling variables ρ, i.e. there exists a linear transfer $\rho := \Pi\xi$, being Π gives the auto-regressive scheduling model. Some comments are presented next with respect to the input variable of this transfer, ξ:

- In the case of pure LPV case, Π is activated by some exogenous variable ξ, such as a coordination signal or an auxiliary entry. When these signals are unavailable, Eq. (3.23) can be converted into a pure auto-regressive model, with no inputs, or written with regard to an input noise given within a known frequency range.
- In the qLPV setting, Assumption 1 is quite reasonable, since the scheduling parameters, at each sampling instant, are forcefully related to the endogenous system variables. Thus, according to which endogenous variables are included in the scheduling proxy, signal ξ is chosen. That is, for state-related proxies $f_\rho(x(k))$, one can simply take $\xi(k) = x(k)$. Equivalently, for the case of output and input related proxies $f_\rho(u(k), y(k))$, we take $\xi(k) = \begin{bmatrix} u(k)^T & y(k)^T \end{bmatrix}^T$, and so forth.

In compact form, denote $\Xi_k = \begin{bmatrix} \xi(k)^T & \dots & \xi(k+N_p-1) \end{bmatrix}^T$. Thence, by exploiting

Assumption 1, we can write the following compact relationship for Π:

$$\rho(k) = \Theta \overbrace{\left[\begin{array}{cc} P^T_{(k-2N_p+1)} & \Xi^T_{(k-2N_p+1)} \end{array} \right]}^{\Psi_{(k-2N_p+1)}} , \tag{3.24}$$

being the auto-regressive model parameters are compacted within

$$\Theta := \left[\begin{array}{cccccc} a_1 & \dots & a_{N_p} & b_1 & \dots & b_{N_p} \end{array} \right] . \tag{3.25}$$

Then, if Θ is known[9], the linear auto-regressive relationship can be exploited in order to span the whole future scheduling trajectories, as follows:

$$\left\{ \begin{array}{rcl} \rho(k+1) & = & a_1\rho(k-2N_p+2) + \dots + a_{N_p}\rho(k-N_p+1) \\ & + & b_1\xi(k-2N_p+2) + \dots + b_{N_p}\xi(k-N_p+1) \\ & = & \Theta\Psi_{(k-2N_p+2)}, \end{array} \right. \tag{3.26}$$

$$\left\{ \begin{array}{rcl} \rho(k+2) & = & a_1\rho(k-2N_p+3) + \dots + a_{N_p}\rho(k-N_p+2) \\ & + & b_1\xi(k-2N_p+3) + \dots + b_{N_p}\xi(k-N_p+2) \\ & = & \Theta\Psi_{(k-2N_p+3)}, \end{array} \right. \tag{3.27}$$

and so forth, up to:

$$\left\{ \begin{array}{rcl} \rho(k+N_p-1) & = & a_1\rho(k-N_p) + \dots + a_{N_p}\rho(k-1) \\ & + & b_1\xi(k-N_p) + \dots + b_{N_p}\xi(k-1) \\ & = & \Theta\Psi_{(k-N_p)}. \end{array} \right. \tag{3.28}$$

In order to consider more generic behaviours, with time-varying dynamics related to the scheduling variables, we imply that the parameters in the auto-regressive model Π vary over time, and, thus, that $\rho(k) = \Theta(k)\Psi_{(k-2N_p+1)}$. Then, the identification-based mechanism is operated as follows:

1. At each sampling instant, one collects the stacked vector of previous scheduling variables and auxiliary inputs, i.e. $\Psi_{(k-2N_p+1)}$;
2. Then, an online recursive Least-Squares minimisation procedure is solved in order to estimate parameters $\Theta(k)$, that is:

$$\Theta(k) = \Theta(k-1) + \lambda Q_\theta \left(\Psi_{(k-2N_p+1)}, \rho(k), \xi(k) \right) , \tag{3.29}$$

   where λ is an update parameter (LS forgetting factors) and Q_θ is an update function[10].
3. The future scheduling parameter trajectory P_k is estimated on the basis of:

$$P_k = \left[\begin{array}{c} \rho(k) \\ 0 \\ \vdots \\ 0 \end{array} \right] + \Theta(k) \left[\begin{array}{c} 0 \\ \Psi_{(k-2N_p+2)} \\ \vdots \\ \Psi_{(k-N_p)} \end{array} \right] . \tag{3.30}$$

[9]Note that all vectors, from $P_{(k-2N_p+1)}$ and $\Xi_{(k-2N_p+1)}$ to $P_{k-(N_p-1)}$ and $\Xi_{k-(N_p-1)}$, are known. This means that an identification procedure can be directly applied.

[10]Since the focus of this thesis is not identification, we invite the Reader to refer to the complete deduction for a recursive LS solution, as presented in [Ljung 1987, Chapter 11.2].

Let us discuss some aspects of this identification-based approach, before presenting simulation results:

- The method is conceived on the basis of Assumption 1, which can be partially false for many systems. In many pure LPV applications, the designer has no access to some activation signal ξ and thus the only alternative is to consider the model Π to be auto-regressive and autonomous, or subject to a noisy inputs;
- In practice, when the parameters are considered time-varying and re-identified online, by the means of a recursive LS procedure, accurate estimates can be formulated;
- The estimates are generated by the means of linear operations, Eqs. (3.29)-(3.30), which enable fast, real-time applications of the corresponding MPC algorithm;
- Yet with great practical value, and with quite satisfactory empiric results (as shown next), the method lacks rigorous proofs of convergence (that is, we cannot ensure that the estimates for $\rho(k+j)$, $\forall j \in \mathbb{N}_{[1,N_p-1]}$ track the correct future scheduling parameters' values), and the estimation error is, *a priori*, unbounded (we have no proper means of how to compute error bounds or to ensure a generic auto-regressive model, since this changes from system to system).

3.4.1 Simulation results

Next, we show brief results of how such LS-based approach can be used in practice. For such, we consider the control problem of the vertical dynamics of a reduced size car (1/5-scale) equipped with four semi-active dampers[11]. Details on modelling are given in [Morato, Normey-Rico, and Sename 2019]. Synthetically, the vertical behaviour of the car are described by a qLPV model, which gives the dynamics of each chassis corner and each wheel (states $x(k)$), due to the road disturbances $w(k)$. The control input is the semi-active damping coefficient $u(k)$, and the scheduling parameter $\rho(k)$ is the suspension deflection velocity (difference between the chassis velocity and the wheel velocity): $x(k+1) = A(\rho(k))x(k) + B_1(\rho(k))u(k) + B_2 w(k)$, where $\rho(k) = f_\rho(x(k))$. This system operates under a sampling period of $T_s = 5$ ms.

The control goal is to minimize both chassis and wheel accelerations and, by doing so, to achieve a smoother and more comfortable drive, while respecting the semi-active damper dissipativity constraints. For such, an MPC is tuned with a prediction horizon N_p of 10 samples. In order to elucidate the effectiveness of the LS-based qLPV MPC algorithms, we compare it to a simpler MPC, which uses a frozen-based prediction model (approach (i)). Furthermore, we consider the following road disturbance scenario, presented in Figure 3.4: a car is 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, which excites bouncing motion.

[11]Results are shown considering the front-left corner of the vehicle; similar results were obtained for the other corners. In Chapter 4, this control problem is studied in more depht. Herein, we focus only on the results regarding the estimation strategy for the future scheduling sequences.

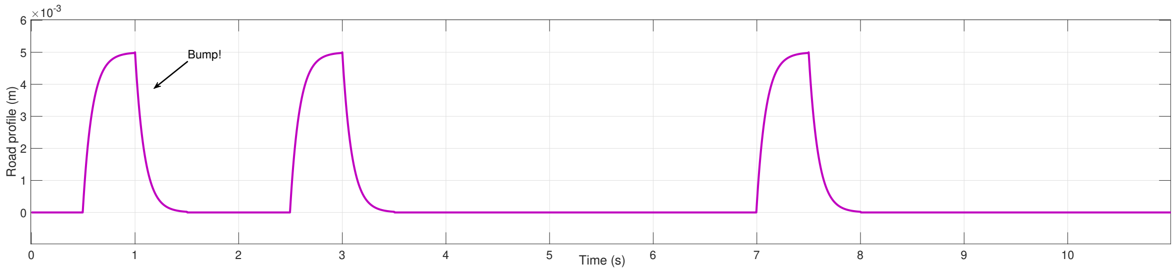


Figure 3.4: Suspension system: Road profile.

Figure 3.5 shows the control variables of interest (acceleration of the chassis axle $\frac{d^2 z_s(t)}{dt^2}$, acceleration of the wheel link $\frac{d^2 z_{us}(t)}{dt^2}$, and total semi-active damper force $F_d(t)$), as obtained with both scheduling trajectory estimation approaches (i) and (ii). Clearly, the LS-based solution is able to provide a more accurate prediction model for the MPC and, thus, the controller is able to further minimize the control objective, while abiding to the semi-active damper dissipativity constraints. In numerical terms, the MPC based on approach (ii) presents a 9.35 % of reduction of the root-mean-square value of the performance objective[12], which would certainly be felt in terms of **passenger comfort**.

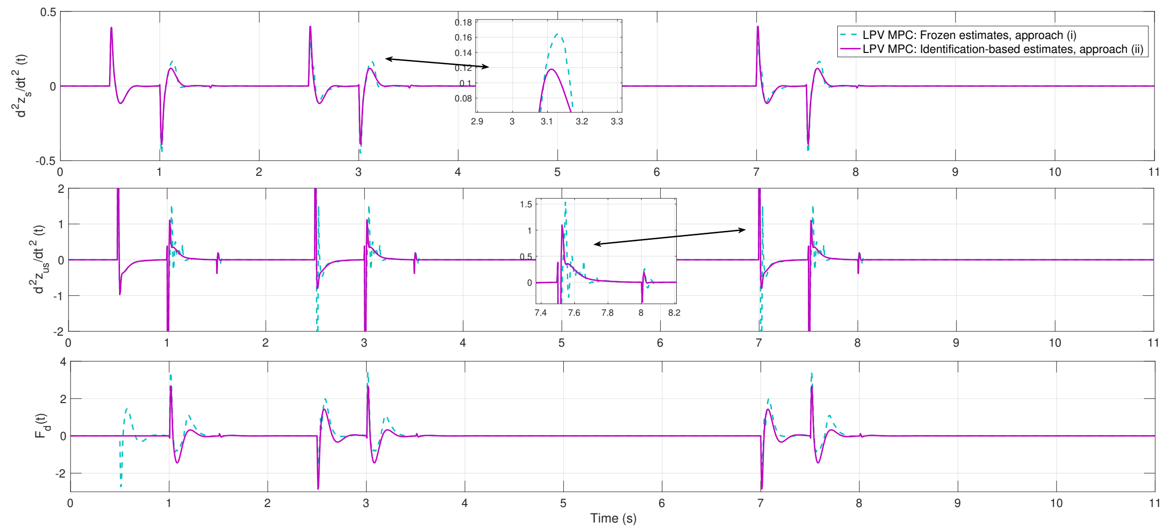


Figure 3.5: Suspension system: Sprung and unsprung accelerations and damper force.

Figure 3.6 shows some snippets of the evolution of the qLPV scheduling parameter ρ, compared with the estimates made by the means of the LS solution, at some sampling instants. Thus, it is clear that, for this case study, the auto-regressive model Π is somewhat valid (Assumption 1), since the estimates are quite close to the real scheduling trajectory. Again, we emphasise that we have no means on how to compute bounds over estimation errors.

[12]Obtained values: 0.21217 (frozen LPV MPC, approach (i)) and 0.19233 (identification-based LPV MPC, approach (ii)).

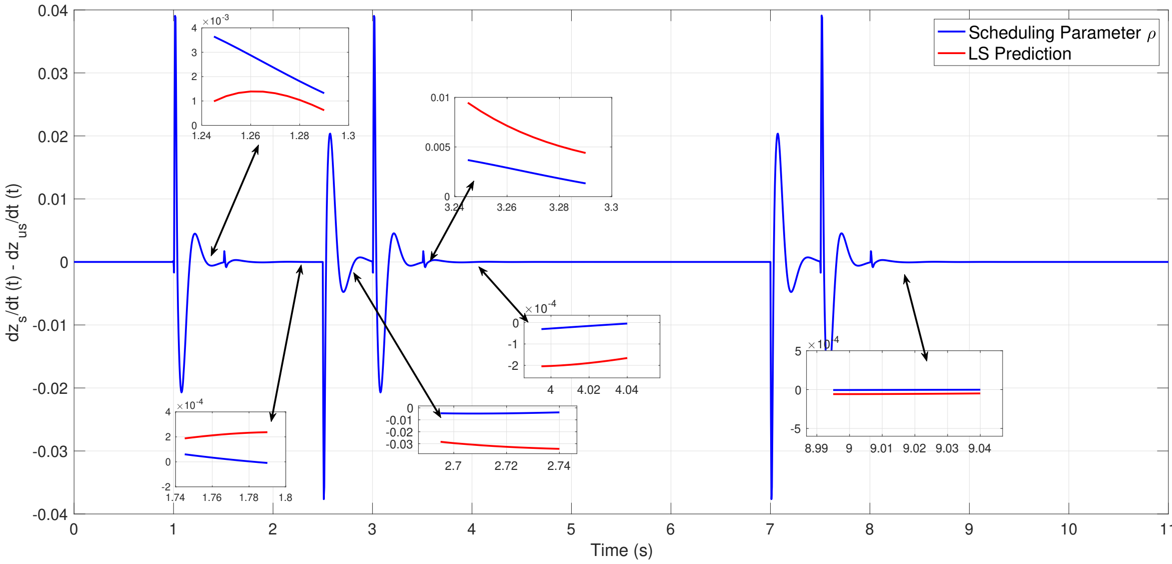


Figure 3.6: Suspension system: Scheduling parameter and LS-based estimates.

Lastly, we discuss the computational complexity of both methods: in average, the LS-based solution takes only 3.09 % longer elapsed time to compute the predictive control policy, yet still remaining under the T_s threshold. Note that this additional time is required to evaluate the linear equations that enable the scheduling trajectory estimate, i.e. Eqs. (3.29)-(3.30). This increase is arguably tolerable, given that it does not violate real-time constraints while providing better overall control performances.

3.5 Approach (iii): Iterative estimation mechanism

The previous methods (frozen approach and auto-regressive LS-based mechanism) represent interesting practical alternatives to solve the issue of the unavailability of the scheduling trajectories, required to compute the MPC predictions. Both these approaches have empirical counter-parts with good results, as registered in the literature. Nevertheless, with the frozen-based method, approach (i), we obtain a correlated uncertainty propagation which may grow significantly, while with the LS-based mechanism, approach (ii), we are unable to quantify the magnitude of the estimation error (and the validity, and existence, of an auto-regressive model).

Next, we detail another widely-employed alternative used to generate P_k online, originally proposed in [Cisneros, Voss, and Werner 2016] and applied for many applications since then, i.e. [Cisneros and Werner 2017a; Cisneros, Sridharan, and Werner 2018; Cisneros and Werner 2019; Abbas et al. 2019; Cisneros and Werner 2020]. The mechanism is based on the iterative operation of the MPC optimisation as a basis to generate the scheduling sequence. The approach requires the model to be quasi-LPV, since the scheduling proxy $f_\rho(\cdot)$ is used to generate the estimates P_k. Since the mechanism operates n_{iter} iterations per sample, the corresponding MPC exhibits a computational complexity of an SQP.

The core idea of the iterative estimation mechanism is as follows:

- At each sample, we require P_k in order to solve the MPC optimisation, which gives as outputs the optimal future control sequence $U_k^\star$ and the predicted state sequence $X_k^\star$;
- Then, instead of solving the MPC optimisation only once per sample, it is solved n_{iter} times: the outputs of the optimisation from one iteration (U_k^l and X_k^l, where the super-index l denotes the iteration instance) are used to generate the scheduling trajectory of the following sample, using: $P_k^l = f_\rho\left((X_k^{l-1})^\perp, U_k^{l-1}\right)$, where the vector

$$(X_k^{l-1})^\perp = \begin{bmatrix} x(k)^T & (X_k^{l-1})^T \end{bmatrix}^T$$

  collects the current state measurement and the state predictions at the l-th iteration of the scheme, with the last entry suppressed[13];
- The iterations continue until convergence is obtained (a certain threshold limit is reached, i.e. $\|P_k^l - P_k^{l-1}\| \leq \xi_P$) or a maximal number of iterations is reached, i.e. $l = n_{\text{iter}}$;
- Note that, if convergence of the scheduling trajectory estimates is indeed reached, i.e. $\lim_{l\to+\infty} P_k^l$ is equal to the **real** scheduling trajectory P_k, then the solution to the corresponding MPC exact (no model prediction mismatches), and thus the same as what would have being obtained with a “full-blown” NMPC (under convexity of the optimisation cost and constraints);
- Figure 3.7 illustrates the mechanism: at each sample k, the MPC optimisation is repeated until one of the stop criteria is met. During the implementation, each predicted scheduling sequence P_k^l is constructed on the basis of the scheduling proxy $f_\rho(\cdot)$, applied over the predicted state and input sequences. The block "Compact" denotes the operation of providing $(X_k^{l-1})^\perp, U_k^{l-1}$ from X_k^{l-1}, U_k^{l-1}, and $x(k)$.

At the first iteration ($l = 1$) of the first sample ($k = 0$), the estimation mechanism begins with a frozen guess:

$$P_0^1 = \begin{bmatrix} \rho(0)^T & \dots & \rho(0)^T \end{bmatrix}^T .$$

Then, at each sample, the first scheduling estimate is taken as $P_k^1 = P_{k-1}^\perp$. Here, $P_{k-1}^\perp$ denotes a shifted and "corrected" vector: at the current instant k, $\rho(k)$ is already measured, so it replaces the prediction made at the previous instant[14]. Furthermore, since P_{k-1} collects

[13]Note that the vector-wise operation $P_k^l = f_\rho\left((X_k^{l-1})^\perp, U_k^{l-1}\right)$ implies on the function application over each vector entry, that is: $\rho(k+j-1) = f_\rho(x(k+j-1), u(k+j-1))$, $\forall j \in \mathbb{N}_{[1,N_p]}$. Furthermore, we stress that we require the values of $\rho(k)$ up until $\rho(k+N_p-1)$, while the state predictions vector X_k comprises $x(k+1|k)$ up until $x(k+N_p-1|k)$. Thereof, we eliminate the last entry of X_k^{l-1}, adding $x(k)$ as a first entry. This new vector is denoted $(X_k^{l-1})^\perp$.

[14]Note that, at instant $k-1$, only $\rho(k-1)$ is known and, thus, P_{k-1} comprises an estimate for $\rho(k)$.

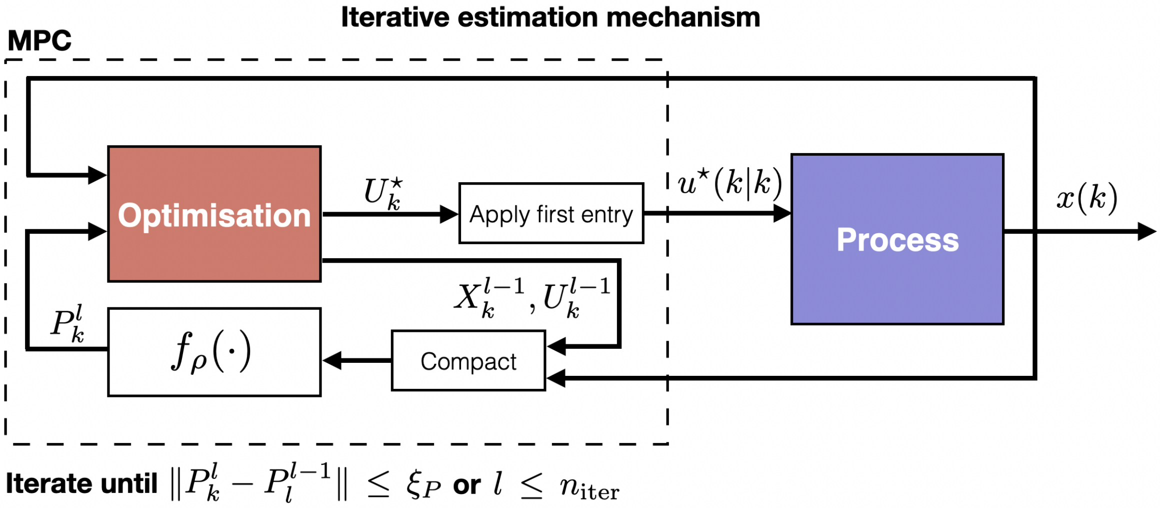


Figure 3.7: Approach (iii): Iterative estimation mechanism.

the scheduling parameter estimates up until $\rho(k + N_p - 2)$, and P_k requires the values until $\rho(k + N_p - 1)$, the last entry of P_{k-1} is repeated in $P_{k-1}^{\perp}$. In synthesis, we obtain:

$$P_k^1 = \left[\overbrace{\rho(k)^T}^{\text{known}} \quad \overbrace{\rho(k+1)^T}^{\text{from } P_{k-1}} \quad \dots \quad \overbrace{\rho(k+N_p-2)^T}^{\text{from } P_{k-1}} \quad \overbrace{\rho(k+N_p-2)^T}^{\text{from } P_{k-1}} \right]^T .$$

As discussed thoroughly in [Cisneros and Werner 2020], empirical evidence shows that this approach guarantees the convergence of the predicted scheduling trajectory P_k, at each sample k, to the true scheduling behaviour. Furthermore, this property is usually achieved within a relatively small number of iterations per sample (around 5-10 inner iterations l), for a wide variety of systems. Practical evidence has been shown in [Cisneros et al. 2019], which discussed how the resulting MPC algorithm becomes a competitive alternative to the implementation of NMPC (operating embedded as fast as ACADO, or CasADi, for many processes). In opposition to what has seen regarding the previous method, these properties have also been verified theoretically. Despite not having any proof of the boundedness of the estimation error, we recall the theoretical proof of convergence from [Hespe and Werner 2021] in the sequel (Sec. 3.5.1).

The main restraint of this scheduling estimate approach is that the internal loop may take several iterations to converge. For fast systems, this is not desirable because the number of inner iterations l needed for convergence (or until the stop criterion $l \leq n_{\text{iter}}$ is reached) may theoretically require more time than the available sampling period threshold (and thus convergence may not be guaranteed). Moreover, depending on the kind of nonlinearity exhibited in the scheduling proxy $f_\rho(\cdot)$, the computation of each entry of P_k^l, using $f_\rho(x(k+j-1|k), u(k+j-1|k))$, $\forall j \in \mathbb{N}_{[1,N_p]}$, may also be numerically expensive.

We stress the main advantages of the mechanism:

- At each sample, the MPC optimisation is formulated as a QP, and thus the computational complexity of the complete MPC algorithm is equivalent to that of an SQP, per sample;

- The application of the mechanism is relatively simple: it requires only vector shifting and the application of the scheduling proxy over N_p vector entries;

- Empirical evidence shows that the method converges in a relatively small number of iterations, for many different systems.

Remark 13. *Unlike the discussions of the previous approaches ((i) and (ii)), simulation results of this scheme are not presented herein, but rather in comparison to approach (iv), presented in Section 3.7, at the end of this Chapter.*

3.5.1 Convergence properties

Next, we briefly recall the convergence analysis of the iterative scheduling estimation procedure, as presented in [Hespe and Werner 2021]. For such, we interpret the corresponding MPC application as the iterative application of an inexact Newton step procedure for root determination.

Accordingly, we re-write the corresponding MPC from Eq. (3.1) in compact, shortened notation, using $\mathbf{s}^l := \left(X_k^l, U_k^l\right)$, where the super-index l denotes the iteration index of the method. By this, we obtain the following optimisation:

$$\begin{aligned} \min_{\mathbf{s}^l} \quad & \|\mathbf{s}^l\|_{\mathcal{Q}}^2 \\ \text{s.t.} \quad & \left\{ \begin{array}{rcl} \mathcal{G}(P_k^l)\mathbf{s}^l + \mathcal{C}x(k) & = & 0\,, \\ A_{\text{in}}\mathbf{s}^l & \leq & b_{\text{in}}\,. \end{array} \right. \end{aligned} \tag{3.31}$$

Note that in Eq. (3.31), the equality constraint appears in order to enforce that the first entry of X_k^l is equal to $x(k)$ (measured state). The inequality constraint imposes $x(k+j|k) \in \mathcal{X}$ and $u(k+j-1|k) \in \mathcal{U}$, $\forall j \in \mathbb{N}_{[1,N_p]}$. Considering $V\left(x(k+N_p|k)\right) := \|x(k+N_p|k)\|_P$ and $\ell(x,u) := \|x\|_Q^2 + \|u\|_R^2$, the compact weighting matrix is given by $\mathcal{Q} := \text{diag}\left(I_{N_p} \otimes Q, P, I_{N_p} \otimes R\right)$, while $\mathcal{C} := \left[\begin{array}{cccc} -I_{n_x} & 0 & \dots & 0 \end{array}\right]^T$.

Next, we consider the first-order necessary conditions of Eq. (3.31), i.e. if $\mathbf{s}^l$ is indeed a solution to Eq. (3.31), then there exists a real vector λ of coherent dimension such that the following equality constraints hold:

$$F\left(\overbrace{\left[\begin{array}{cc} \mathbf{s}^l & \lambda \end{array}\right]}^{z^l(k)}\right) := \left[\begin{array}{c} 2\mathcal{Q}\mathbf{s}^l + \mathcal{G}^T(P_k^l)\lambda \\ \mathcal{G}(P_k^l)\mathbf{s}^l + \mathcal{C}x(k) \end{array}\right] = \left[\begin{array}{c} 0 \\ 0 \end{array}\right]. \tag{3.32}$$

Then, we compute the Jacobian of $F(z^l(k))$ using the following approximation[15]:

$$\begin{aligned}\frac{dF(z^l)}{dz^l} &:= \begin{bmatrix} 2\mathcal{Q} + \frac{\partial(\mathcal{G}^T(P_k^l)\lambda)}{\partial \mathbf{s}^l} & \mathcal{G}^T(P_k) \\ \frac{\partial(\mathcal{G}(P_k^l)\mathbf{s}^l)}{\partial \mathbf{s}^l} & 0 \end{bmatrix} \\ &\approx \underbrace{\begin{bmatrix} 2\mathcal{Q} & \mathcal{G}^T(P_k^l) \\ \mathcal{G}^T(P_k^l) & 0 \end{bmatrix}}_{\mathcal{J}(z^l(k))},\end{aligned} \tag{3.33}$$

which means that solving the (unconstrained version of the) optimisation in Eq. (3.31) is equivalent to finding the solution to the following equality rule:

$$F(z^l(k)) + \mathcal{J}(z^l(k))\left(z(k) - z^l(k)\right) = 0\,, \tag{3.34}$$

which exhibits the structure of an approximate Newton step (note that here we use the approximate of $\frac{dF(z^l)}{dz^l}$, i.e. $\mathcal{J}(z^l)$. Thus, sequentially iterating the QP in Eq. (3.31) is analogous to the use of a Newton-based scheme to determine the root of the problem posed in Eq. (3.34). As argues [Hespe and Werner 2021], there exist a considerable body of research devoted to the analysis of Newton-based algorithms, which can be exploited in the detailed context to provide sufficient conditions for the convergence of Eq. (3.31). Therefore, we apply these conditions to demonstrate the convergence of the iteratively-estimated scheduling sequences with the method from [Cisneros and Werner 2020].

The main arguments from [Hespe and Werner 2021] are recalled. These conditions are sufficient in order to demonstrate that the iteration convergence, based on the local contraction over one iteration. For such, we first assume that[16] the following conditions are satisfied:

(i) The maps $\mathcal{G}(P_k^l)$ and $f_\rho(\cdot)$ are twice continuously differentiable;

(ii) $\mathcal{G}(P_k^l)$ has full row rank and $\mathcal{Q}$ is positive definite within the kernel of $\mathcal{G}(P_k^l)$, being P_k^l an admissible existing solution point to $F(P_k^l) = 0$;

(iii) There exists two real scalar $\lambda_w < +\infty$ and $\kappa_w < 1$, an open ball $\mathcal{B}$ around P_k such that $\mathcal{J}(z^l)$ is non-singular and that:

$$\begin{aligned}\|\mathcal{J}^{-1}(z^l(k))\left(\frac{dF(z^l(k))}{dz} - \frac{dF(z^{l+1}(k))}{dz}\right)\| &\leq \lambda_w \|z^l(k) - z^{l+1}(k)\|\,, \\ \|\mathcal{J}^{-1}(z^l(k))\left(\frac{dF(z^l(k))}{dz} - \mathcal{J}(z^l(k))\right)\| &\leq \kappa_w\,,\end{aligned} \tag{3.35}$$

holds for all $z^l(k)\,,\, z^{l+1}(k) \in \mathcal{B}$.

[15]Here, we drop the discrete-time dependency seeking a shortened notation; that is: z^l represents $z^l(k)$.

[16]In [Hespe and Werner 2021], one can find full discussions on why each of these hypothesis are necessary and how simple are they to hold (i.e. under which settings). Herein, for brevity, we present them rapidly, only in order to formulate the convergence proof.

Then, as long as these conditions hold, the sequence of estimates $\left(z^l(k)\,,\, z^{l+1}(k),\; z^{l+2}(k)\,,\dots\right)$ generated by the solution of Eq. (3.34) converge towards the real value $z(k)$ with the following contraction rate:

$$\|z^{l+1}(k) - z(k)\| \quad \leq \quad \kappa_w \|z^l(k) - z(k)\| + \frac{\lambda_w}{2}\|z^(k) - z(k)\|^2 \,.$$

for all $z^0(k) \in \mathcal{B}$ such that $\|z^0(k) - z(k)\| \;<\; 2\frac{1-\kappa_w}{\lambda_w}$.

From this fact, since the contraction rate is lower bounded, we can conclude there exists a non-empty region of attraction such that Eq. (3.34) has a solution and converges. Equivalently, it follows that, within this region, Eq. (3.31) also converges. For a full demonstration of this convergence rate, refer to [Hespe and Werner 2021].

3.6 Approach (iv): Taylor-based extrapolation scheme

Next, we present the last approach discussed in this Chapter, which is, in fact, one of the main contributions of this thesis, discussed in depth in [Morato, Normey-Rico, and Sename 2022b]. All requirements, preliminary assumptions and settings for the following developments are those presented in Section 3.2. This method is, as the previous one, only possible for the qLPV setting, since the scheduling proxy[17] $f_\rho(\cdot)$ is used to generate the scheduling trajectory estimate.

The proposed approach is **recursive**, in the sense that the scheduling trajectories are generated by the means of the following law:

$$\hat{P}_k \quad = \quad \Phi\left(\hat{P}_{k-1}, \rho(k), x(k)\right) \,. \tag{3.36}$$

Thus, in the sequel, we discuss how the operator $\Phi(\cdot)$ can be linear, and how the estimated vectors converge to the correct scheduling trajectory behaviour, in a finite amount of samples. Note that the procedure in Eq. (3.36) generates a new extrapolation for the scheduling trajectories P_k, at instant k, based on the prior extrapolation and the new dataset available ($\rho(k)$ and $x(k)$). For such, the proposed recursive approach is conceived from the basis of following assumption:

Assumption 2. *The static map $f_\rho(x)$ can be approximated by the following first order Taylor expansion around $\overline{x}$:*

$$f_\rho(x) \quad \approx \quad f_\rho(x)|_{\overline{x}} + \left.\frac{\partial f_\rho}{\partial x}\right|_{\overline{x}} (x - \overline{x})\,, \tag{3.37}$$

being $\overline{x}$ an arbitrary linearisation point. The actual function can be analytically expressed by the sum of this approximation to a residual signal ξ_ρ, which inherits the discrepancy between the real static map and its Taylor approximate:

$$f_\rho(x) \quad = \quad f_\rho(x)|_{\overline{x}} + \left.\frac{\partial f_\rho(x)}{\partial x}\right|_{\overline{x}} (x - \overline{x}) + \xi_\rho \,. \tag{3.38}$$

[17] Next, we take a state-dependent scheduling proxy $\rho = f_\rho(x)$. The method can be easily applied also for input and output-dependent proxies, without loss of generality.

Consider Assumption 2 holds. Then, the following expression is valid, considering the linearisation at a given instant $k+j-1$ and the increment along x to the following instant $k+j$, namely $\Delta x(k+j-1)$:

$$\begin{aligned} f_\rho(x(k+j)) &= f_\rho(x(k+j-1)) + \xi_\rho(k+j-1) \\ &+ \left.\frac{\partial f_\rho(x)}{\partial x}\right|_{x(k+j-1)} \Delta x(k+j-1)\,. \end{aligned} \tag{3.39}$$

We henceforth denote $f_\rho^\partial(k+j-1) = \left.\frac{\partial f_\rho}{\partial x}\right|_{x(k+j-1)}$. Expanding the expression in Eq. (3.39) along the fixed prediction horizon of N_p steps and embedding it to the scheduling proxy $\rho(k) = f_\rho\left(x(k)\right)$ yields:

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

As of the qLPV model in Eq. (3.11), $\rho(k)$ and $\Delta x(k)$ are known, whereas $f_\rho^\partial(k)$ can be numerically evaluated on the basis of the current state measurement $x(k)$. Nevertheless, in practice, $f_\rho^\partial(k+j)$ for $j \in \mathbb{N}_{[1,N_p-2]}$ is unknown, which requires a second assumption:

Assumption 3. *For simplicity, at each sampling instant k, it is assumed that the partial derivative $f_\rho^\partial(k)$ stays constant along the prediction horizon, i.e. $f_\rho^\partial(k+j) = f_\rho^\partial(k)\,, \forall\, j \in \mathbb{N}_{[1,N_p-2]}$.*

The partial derivatives terms $f_\rho^\partial(k+j)$ could be computed on the basis of the state trajectory prediction X_k (generated by the MPC algorithm). Nevertheless, this is numerically costly. Thus, we exploit Assumption 3 in order make our extrapolation procedure fast and numerically cheap, thus taking $f_\rho^\partial(k+j) = f_\rho^\partial(k)$. In the sequel, we show that even by using such approximation, convergence is still ensured.

Note that the expansions along the prediction horizon can be given in terms of the previous scheduling parameter value and a correction term, as follows:

$$\begin{aligned} \rho(k+j|k) &= \rho(k+j-1|k-1) + f_\rho^\partial(k)\Delta x(k+j-1) \\ &+ \xi_\rho(k+j|k)\,, \end{aligned} \tag{3.40}$$

Therefore, with a slight abuse of notation[18], we use the following vector representation:

$$\hat{P}_k = \hat{P}_{k-1}^\star + f_\rho^\partial(k)\Delta X_k^\star + \Xi_k\,, \tag{3.41}$$

[18] Along the sequel, the term $f_\rho^\partial(k)$ in the vector form denotes the diagonal matrix $\mathrm{diag}\{f_\rho^\partial(k), f_\rho^\partial(k), \ldots, f_\rho^\partial(k)\}$.

which is a recursive estimation law of the fashion in Eq. (3.36).

Notice that $\hat{P}^\star_{k-1}$ stands for the previous scheduling trajectory estimation with the first term corrected with the known value $\rho(k)$ (known data), while $\Delta X^\star_k$ represents the state deviations along the horizon (also corrected with the known value $\Delta x(k)$). Since $\Delta X^\star_k$ represents the difference of the states over time k, this vector is computed by adapting Eq. (3.8). For such, we shift the control sequence $U^\star_{k-1}$ as a basis for U_k, with the last entry kept constant. This is, we take the following vector of dimension $n_u \times N_p$:

$$\breve{U}_k = \begin{bmatrix} u^\star(k|k-1) \\ \vdots \\ u^\star(k+N_p-2|k-1) \\ u^\star(k+N_p-2|k-1) \end{bmatrix} . \tag{3.42}$$

Accordingly, we obtain $\Delta\hat{X}_{k-1} = \mathcal{A}(P^\star_{k-1})\Delta x(k) + \mathcal{B}(P^\star_{k-1})\breve{U}_k$. Lastly, we stress that Ξ_k is a bias residual vector, which "corrupts" the extrapolation. Since this vector is unknown, we disregard it in the recursive estimation procedure, which means that Eq. (3.41) is replaced by an approximation, as gives Eq. (3.43). Here, the $\hat{\cdot}$ indicates a prediction (estimate) for the referred variable, i.e. $\hat{P}_k = [\rho(k)^T, \rho(k+1|k)^T, \dots, \rho(k+N_p-1|k)^T]^T$. We stress that Eq. (3.41) is an exact representation for any non-null (or vanishing) residual vector term Ξ_k.

$$\hat{P}_k = \hat{P}^\star_{k-1} + f^\partial_\rho(k)\Delta\hat{X}^\star_k , \tag{3.43}$$

Figure 3.8 illustrates the concept behind Eq. (3.43) and how the scheduling trajectory estimate from the last sample $\hat{P}_{k-1}$ is updated and thus used to generate the current estimate $\hat{P}_k$. We note that the corrections on $\hat{P}_{k-1}$ and $\Delta\hat{X}_k$ are given by:

$$\hat{P}^\star_{k-1} = \lambda\hat{P}_{k-1} + \nu\rho(k) , \tag{3.44}$$
$$\Delta\hat{X}^\star_k = \lambda\Delta\hat{X}_{k-1} + \nu\Delta x(k) , \tag{3.45}$$

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

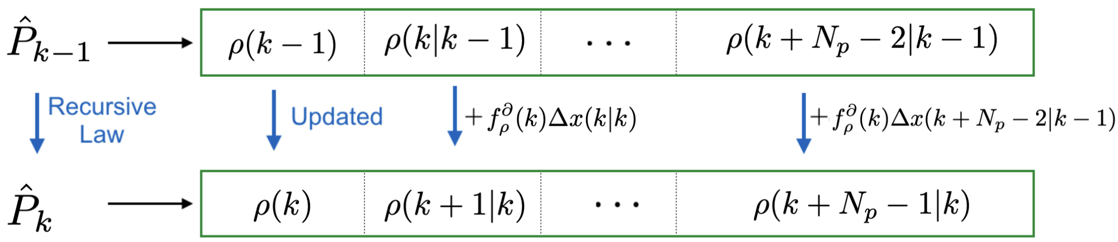


Figure 3.8: Approach (iv): Vector shifting.

Remark 14. *The dimensions of λ and ν in Eqs. (3.44)-(3.45) should be in accordance with n_p and n_x. We note that recursive extrapolation mechanism from Eq. (3.43) is not able to ensure* *that each entry of the scheduling trajectory estimate vector (i.e.* $\rho(k+j)\,,\,\forall j \in \mathbb{N}_{[1,n_p]}$*) abides to the admissibility constraints (each* $\rho(k+j) \in \mathcal{P}$*). Thus, in order to provide "coherent" extrapolated parameters* $\rho(k+j|k)$*, the extrapolation vector* $\hat{P}_k$ *is "clipped".*

Remark 15. *An additional forgetting factor can be included to Eqs.* (3.44)-(3.45)*, replacing the identity matrices in* λ *by exponentially decaying terms, i.e.* $Ie^{-k/k_{max}}$*. This can serve to attenuate the amount of mistaken information passed from one estimate* P_k *to the following* P_{k+1}*.*

Note that Assumption 3 is an approximation while the convergence property of the extrapolation mechanism has not yet been established (further details in Lemma 3.6.2), since $f_\rho^\partial(k+j) \neq f_\rho^\partial(k)$. Nevertheless, as the system stabilizes and the extrapolation converges, it follows that $f_\rho^\partial(k+j) \approx f_\rho^\partial(k)$ becomes a very reasonable approximation. This approximation may be violated more easily with larger horizons N_p, since the discrepancies between the real scheduling variable $\rho(k+j)$ and its extrapolated estimate $\rho(k+j|k)$, grow along $j \in \mathbb{N}_{[1,N_p-2]}$. Yet, the sliding-horizon mechanism of the MPC ensures that this imprecision has minor effects on the closed-loop performances, as illustrated by the simulation results presented in Section 3.7. The complete recursive estimation procedure is presented in Algorithm 1.

Algorithm 1 Recursive extrapolation approach

For every sampling instant k, loop:

- Measure the system states $x(k)$;
- Compute $\rho(k) = f_\rho\left(x(k)\right)$;
- From Eq. (3.11) compute $\Delta x(k)$;
- Correct $\hat{P}_{k-1} \rightarrow \hat{P}^\star_{k-1}$ using Eq. (3.44);
- Compute the state deviations' estimate vector $\Delta\hat{X}_k$ based on $\hat{P}^\star_{k-1}$ and $\breve{U}_k$, using Eq. (3.8);
- Update this vector $\Delta\hat{X}_k \rightarrow \Delta\hat{X}^\star_k$ using Eq. (3.45);
- Compute the static derivative term $f_\rho^\partial(k)$;
- Compute $\hat{P}_k$ using Eq. (3.43);
- Clip each entry of the extrapolation $\hat{P}_k$, i.e. each $\rho(k+j|k)$ is replaced by $\max(\underline{\rho}\,,\,\min(\rho(k+j|k)\,,\,\overline{\rho}))$.

Remark 16. *The initialisation of the scheduling trajectory guess* P_0 *is of particular interest. At the initial point, only* $x(0)$ *and* $\rho(0)$ *are known; assumably, no control input has yet been generated, since* $u(0)$ *is computed on the basis of the optimisation, which requires* $\hat{P}_0$*. Therefore, since the MPC is ensured to stabilize the system for all* $\rho \in \mathcal{P}$ *(Lemma 3.6.1), we simply* *use $\hat{P}_0 = col\{\rho(k)\}$, i.e. N_p repeated entries of $\rho(k)$. This is "the best possible candidate" for the scheduling trajectory at the initial sampling instant, and it is refined progressively as the recursive extrapolation convergences.*

3.6.1 Sufficient conditions for convergence

With respect to the extrapolation algorithm, we now present sufficient conditions for its convergence (that is, for the estimate scheduling trajectories to converge to the true behaviour). For such, the following rationale is used: if the MPC controller, based on an approximated scheduling sequence estimation, still ensures closed-loop stability, then the extrapolation $\hat{P}_k$ will converge (i.e. $\hat{P}_k \to P_k$). In the sequel, we present a Lemma regarding the form of the MPC state-feedback gain, and another Lemma which provides five sufficient conditions for convergence.

Lemma 3.6.1. *There exists a* terminal *state-feedback policy, implied by the MPC, such that $u(k+N_p) = \kappa(\rho(k+N_p))x(k+N_p)$ and an upper gain κ_∞ such that $\|\kappa(\rho(k))\| \leq \kappa(\|\rho(k)\|) = \kappa_\infty, \forall \rho(k) \in \mathcal{P}$.*

Proof. Consider an MPC application for a process with constant time-invariant parameters (LTI), i.e. $x(k+1) = Ax(k) + Bu(k)$. In this case, consider a terminal control law, implied by the MPC, and given as $u(k+N_p) = \kappa x(k+N_p)$, where the state-feedback gain depends on the system model parameters (A, B), tuning weights (P, Q, R). The closed-loop stability can be shown through classical Lyapunov arguments, which generate the nominal terminal explicit feedback gain κ. Since in our study we consider a time-varying system expressed through the qLPV model in Eq. (3.11), the MPC can be scheduled with respect to the known scheduling parameters, since the model matrices are dependent on this variable. Therefore, the resulting terminal state-feedback predictive control law has the form of $u(k+N_p) = \kappa(\rho(k+N_p))x(k+N_p)$, where the feedback gain is parameter dependent, as shown in [Cisneros and Werner 2020]. Since the parameters are expressed within a bounded set $\mathcal{P}$, we can benefit from the polytopic representation of Eq. (3.11) to determine a parameter-dependent state-feedback gain $\kappa_\infty \geq \|\kappa(\rho)\|, \forall \rho \in \mathcal{P}$ by evaluating $\kappa(\rho)$ at the vertices of the embedding polytope. Such state-feedback gain $\kappa(\rho(k))$ can be explicitly computed. Assume there exist a stage cost $\ell(x, u)$ and a terminal $V(x)$, as in Eqs. (3.12)-(3.13), with positive-definite weights $P(\rho)$, Q, and R. Let $Y(\rho) = (P(\rho))^{-1}$ and take $\kappa(\rho) = W(\rho)Y(\rho)$. Assume the closed-loop is stable and the MPC is recursively feasible. Then, it follows that a Lyapunov argument holds for the system, meaning that the finite-horizon MPC cost is decreasing, i.e. $J_{k+N_p+1} - J_{k+N_p} \leq 0, \forall k$. As detailed in [Jungers et al. 2009], [Cisneros and Werner 2020], and in Chapter 2, this inequality implies that $V\left((A(\rho) + B(\rho)\kappa(\rho))x\right) - V(x) \leq -\ell(x, \kappa(\rho)x), \forall x \in \mathcal{X}$. This condition can be re-stated in a parameter-dependent LMI form (see [Morato, Normey-Rico, and Sename 2022b, Theorem 1]), which is solvable in a grid of points over $\rho \in \mathcal{P}$. This LMI solution provides the parameter-dependent matrices $P(\rho)$ and $W(\rho)$, which are used to generate the terminal feedback gain $\kappa(\rho)$. This ends the proof. □

Remark 17. *The considered MPC should be robustly stable, despite the model-process un-*

certainties derived using all possible "wrong" (non-ideal) scheduling sequences $\check{P}_k$. *As in any control method, robustness comes at the expense of performance deterioration. Anyhow, that the proposed method provides scheduling estimates which are rather accurate, with bounded residuals (refer to (C5) in Lemma 3.6.2, the simulation results in Sec. 3.7, and further discussions from Chapters 4, 5 and 6). Accordingly, less conservative controllers are enabled. Moreover, we note that the method consists basically of linear vector-wise operations, which is computationally simple. More discussions on this matter are presented in [Morato, Normey-Rico, and Sename 2022b], omitted in this thesis for brevity.*

Lemma 3.6.2. *Consider the following sufficient conditions:*

(C1) The static map $f_\rho(\cdot)$ *is, at least, class* $\mathcal{C}^1$*, i.e. first-order differentiable with respect to* x*, for all* $x \in \mathcal{X}$*;*

(C2) The differentiation function $f_\rho^\partial(k)$ *is energy-bounded (2-norm sense) for all* k*;*

(C3) The state deviation term $\Delta x(k+j)$ *is energy-bounded (2-norm sense) for all* k*;*

(C4) The qLPV system is stable in closed-loop;

(C5) The nominal closed-loop dynamics remain stable in the sense of 2-norm even if the MPC is generated with a biased scheduling variable, i.e. $\rho(k) + \xi_\rho(k)$*, for any bounded* $\|\xi_\rho(k)\|_2 \leq \xi_\rho^{bound}$*.*

Then, as long as these conditions hold, the recursively proposed extrapolation tool from Eq. (3.43) *is convergent, meaning that the residual term in Eq.* (3.41) *satisfies* $\lim_{k\to+\infty} \Xi_k \to 0$.

Definition 3.1 (Little-o notation)
A given function $f(k)$ *can be expressed as* $f(k) = o(g(k))$ *as* $k \to \infty$ *if, for every positive constant* ϵ*, there exists another constant* β *such that* $|f(x)| \leq \epsilon g(x)$*, for all* $k \geq \beta$*.*

Proof. We proceed by detailing each of these five sufficient conditions individually:

- (C1): $f_\rho(x)$ must be at least class $\mathcal{C}^1$, so that its derivative $f_\rho^\partial(k)$ exists for all $x \in \mathcal{X}$. The derivative term is necessary in order for the Taylor approximation of Eq. (3.38) to be valid.
- (C2) and (C3): Since, for simplification purposes, the recursive extrapolation is computed as if $f_\rho^\partial(k+j)$ remained constant as $f_\rho^\partial(k)$ through the prediction horizon, from the viewpoint of each sampling instant k, it must hold that $\|f_\rho^\partial(k)\|_2 \leq \overline{f_\rho^\partial}$ for P_k in Eq. (3.43) to exist. Moreover, in order to construct $\hat{P}_k$ and $\Delta\hat{X}_k^\star$ must be energy-bounded, which conversely implies that each term $\|\Delta x(k+j|k)\|_2 \leq \Delta\overline{x}$. This condition is also necessary for Lemma 3.6.1 to hold.
- (C4): For the MPC to stabilize the system, there must exist a nominal feedback $u = \kappa(\rho)x$ such that the closed-loop dynamics are exponentially stable for all $\rho \in \mathcal{P}$, as argued

in Remark 11. This is ensured if Lyapunov conditions are satisfied through adequate terminal ingredients of the MPC optimisation, computed offline, as those presented in [Cisneros and Werner 2020].

- (C5): In order to demonstrate this condition, we verify that the qLPV system is stable in closed-loop, when scheduled by a biased law, i.e. by $\rho(k+j|l) = \rho(k+j|k)+\xi_\rho(k+j|k)$, where $\xi_\rho(k+j|k)$ represents the uncertainty upon the scheduling parameter. In short, we denote that the corresponding MPC input as $u(k) = \kappa\left(\rho(k) + \xi_\rho(k)\right) x(k)$ (assuming that the bias appears already in the first entry of $\hat{P}_k$).

Accordingly, consider the qLPV system is affine (Set of Assumptions 1). Moreover, consider the following uncertainty inputs, given in the z-domain (frequency):

$$\begin{cases} u_{\Delta_x}(z) &= \overbrace{(A_1\xi_\rho(z)) \circledast x(z)}^{\Delta_x(z)x(z)} \,, \\ u_{\Delta_u}(z) &= \underbrace{(B_1\xi_\rho(z)) \circledast u(z)}_{\Delta_u(z)u(z)} \,. \end{cases}$$

Thus, we are able to obtain the following static state-space description of the closed-loop:

$$\begin{cases} x(k+1) &= A_n x(k) + B_n u(k) + \overbrace{(u_{\Delta_x}(k) + u_{\Delta_u}(k))}^{u_\Delta(k)} \,, \\ A_n &= A_0 + A_1\rho(k) \,, \\ B_n &= B_0 + B_1\rho(k) \,. \end{cases} \tag{3.46}$$

In order to verify the stability of this system, we use an $M - \Delta$ analysis framework [Zhou and Doyle 1998]. For such, consider two "uncertainty outputs": $y_{\Delta_x}(k) = x(k)$ and $y_{\Delta_u}(k) = u(k)$, being $y_\Delta(k) = \text{diag}\{y_{\Delta_x}(k)\,,\, y_{\Delta_u}(k)\}$. The corresponding static LTI model is expressed in the z-domain as follows:

$$\begin{cases} x(z) &= G_n(z)u(z) + G_\Delta(z)\left(u_{\Delta_x}(z) + u_{\Delta_u}(z)\right) \,, \\ G_n(z) &= \left(zI_{n_x} - A_n\right)^{-1} B_n \,, \\ G_\Delta(z) &= \left(zI_{n_x} - A_n\right)^{-1} \,. \end{cases} \tag{3.47}$$

Next, we use Lemma 3.6.1 to state the biased MPC policy as $u(k) = \kappa_\infty\left(x(k) + r(k)\right)$, being $r(z)$ a fictive input to demonstrate input-to-state stability, and κ_∞ the upper gain computed with respect to $\rho(k) + \xi_\rho(k)$. Thus, we obtain:

$$\begin{aligned} \|x(z)\|_2 \;\leq\; & \| \overbrace{\left(I_{n_x} - G_n(z)\kappa_\infty\right)^{-1} G_n(z)\kappa_\infty}^{T_n(z)} r(z) \\ + \; & \left(I_{n_x} - G_n(z)\kappa_\infty\right)^{-1} G_\Delta(z) u_{\Delta_x}(z) \\ + \; & \underbrace{\left(I_{n_x} - G_n(z)\kappa_\infty\right)^{-1} G_\Delta(z)}_{T_\Delta(z)} u_{\Delta_u}(z)\|_2 \,. \end{aligned} \tag{3.48}$$

The corresponding LFT of this frozen system satisfies the following inequality:

$$\left\| \begin{bmatrix} y_\Delta(z) \\ \hline x(z) \end{bmatrix} \right\|_2 \;\leq\; \|N(z)\|_\infty \left\| \begin{bmatrix} u_\Delta(z) \\ \hline r(z) \end{bmatrix} \right\|_2 , \tag{3.49}$$

where:

$$N(z) \;=\; \left[\begin{array}{c|c} \overbrace{\begin{pmatrix} T_\Delta(z) & T_\Delta(z) \\ T_\Delta(z)\kappa_\infty & T_\Delta(z)\kappa_\infty \end{pmatrix}}^{M(z)} & \begin{pmatrix} T_n(z) \\ \kappa_\infty + T_n(z)\kappa_\infty \end{pmatrix} \\ \hline \begin{pmatrix} T_\Delta(z) & T_\Delta(z) \end{pmatrix} & T_n(z) \end{array} \right] .$$

Finally, we can write $u_\Delta(k) = \Delta(\cdot) y_\Delta(k)$, with:

$$\left\{ \begin{array}{lcl} \|\Delta_x(\cdot)\|_\infty & = & \frac{\|A_1 \xi_\rho \circledast x\|_2}{\overline{x}} , \\ \|\Delta_u(\cdot)\|_\infty & = & \frac{\|B_1 \xi_\rho \circledast u\|_2}{\overline{u}} . \end{array} \right. \tag{3.50}$$

Since $\|\xi_\rho(k)\|_2 \leq \xi_\rho^{\text{bound}}$ by definition, these convolution products are upper bounded as follows:

$$\left\{ \begin{array}{lcl} \|\xi_\rho \circledast x\|_2 & \leq & \xi_\rho^{\text{bound}} \overline{x} , \\ \|\xi_\rho \circledast u\|_2 & \leq & \xi_\rho^{\text{bound}} \overline{u} , \end{array} \right.$$

which conversely implies that:

$$\begin{array}{rcl} \|\Delta(\cdot)\|_\infty & = & \left\| \begin{matrix} \Delta_x(\cdot) & 0 \\ 0 & \Delta_u(\cdot) \end{matrix} \right\|_\infty \\ & \leq & \left\| \begin{bmatrix} A_1 & 0 \\ 0 & B_1 \end{bmatrix} \right\|_2 \begin{bmatrix} \frac{\|\xi_\rho \circledast x\|_2}{\overline{x}} & 0 \\ 0 & \frac{\|\xi_\rho \circledast u\|_2}{\overline{u}} \end{bmatrix} \\ & \leq & \left\| \begin{bmatrix} A_1 & 0 \\ 0 & B_1 \end{bmatrix} \right\|_2 \xi_\rho^{\text{bound}} . \end{array}$$

Note that we obtain $\|M(z)\|_\infty \;\leq\; \|\kappa_\infty T_\Delta(z)\|_\infty$, which means that $\|M(z)\|_\infty \leq \|\kappa_\infty (I_{n_x} - G_n(z)\kappa_\infty)^{-1} G_\Delta(z)\|_\infty$. Thence, the stability condition is very direct: simply checking the following inequality for all $\rho \in \mathcal{P}$, $x \in \mathcal{X}$ and $\|\xi_\rho(k)\|_2 \leq \xi_\rho^{\text{bound}}$ (being κ_∞ is determined according to Lemma 3.6.1):

$$\|\kappa_\infty T_\Delta(z)\|_\infty \;\leq\; \frac{1}{\|\Delta(\cdot)\|_\infty} . \tag{3.51}$$

As a last remark, we note that the bias term $\xi_\rho(k+j|k)$ is bounded by definition of the Taylor extrapolation mechanism. Note that in Eq. (3.39), we obtain $\xi_\rho(k+j|k) = h_w(x(k+j|k))\Delta x(k+j|k)$. This function can be expressed through the little-o notation, using $\xi_\rho(k+j|k) = o\left(\|\Delta x(k+j|k)\|_2\right)$, which translates to $\|\xi_\rho(k+j|k)\|_2 \leq \epsilon \|\Delta x(k+j|k)\|_2$, holding for any number of discrete-time steps $k \geq \beta$ and a positive real constant ϵ. Since the size of the prediction of the extrapolation is $N_p - 1$, we obtain $\xi_\rho^{\text{bound}} \leq (N_p - 1)\Delta\overline{x}$. This concludes the proof. □

Remark 18. *Condition (C5) in Lemma 3.6.2 is essential. As long as the closed-loop system is robustly stable despite the residual term* $\xi_\rho(k+j|k)$*, we can demonstrate the convergence property by showing that* $\lim_{k\to+\infty}\xi_\rho(k+j|k)\to 0$ *and that* $f_\rho^\partial(k+j)=f_\rho^\partial(k)\,,\,\forall j\in\mathbb{N}_{[1,N_p-2]}$*. Assume the process is stable in closed-loop. Thus, it holds that* $\lim_{k\to\infty}x(k+j|k)=\lim_{k\to\infty}x(k+j-1|k)$*. Then, take* $\xi_\rho(k+j-1|k)=f_\rho(x(k+j))-f_\rho(x(k+j-1|k))-f_\rho^\partial(k+j-1)\Delta x(k+j-1|k)$*. It directly follows from (C5) that* $f_\rho^\partial(k+j-1)=f_\rho^\partial(k)\,,\,\forall j\in\mathbb{N}_{[1,N_p-2]}$*. Thus,* $\lim_{k\to\infty}f_\rho(x(k+j+1))=\lim_{k\to\infty}f_\rho(x(k+j))$ *and* $\lim_{k\to\infty}\Delta x(k+j|k)=0$*. Finally,* $\lim_{k\to\infty}\xi_\rho(k+j|k)=-\lim_{k\to\infty}f_\rho^\partial(k)\Delta x(k+j|k)\to 0$*, which conversely ensures that the scheduling sequence extrapolation* $\hat{P}_k$ *indeed converges to the real scheduling sequence.*

Remark 19. *In order to verify that the proposed method ensures a convergent extrapolation, (C5) can be checked through the Algorithm 2, used to verify Ineq.* (3.51)*. We note that the computation of the compact set* $\Delta\mathcal{X}$ *from Proposition 1 requires optimisation or interval arithmetics, while the feedback gain* κ_∞ *is derived using LMIs (as provided in Lemma 3.6.1 and also expressed in [Cisneros and Werner 2018]).*

Remark 20. *The convergence of Eq.* (3.43) *implies that the residual errors* $\xi_\rho(\cdot|k)$ *turn null as k increases, even if consecutive biases within samples are increasing, i.e.* $\lim_{k\to+\infty}\xi_\rho(\cdot|k)$ *even if* $\xi_\rho(k_1+j|k_1)\geq\xi_\rho(k_1+j+1|k_1)$*, for any* $k_1\geq 0$*.*

Remark 21. *In the following Chapters, we provide a complementary result (Lemma 4.8.2), which gives tighter bounds on the extrapolation error* $\xi_\rho(k+j|k)$*, exploiting a Lipschitz continuity property of the scheduling proxy.*

Algorithm 2 Checking sufficient condition (C5)

For every $\rho\in\mathcal{P}$:

- Compute the feedback gain κ_∞ via Lemma 3.6.1, solving the optimisation for each $x(k)\in\mathcal{X}$ and each $\rho(k)\in\mathcal{P}$;
- Compute the nominal model matrices A_n and B_n;
- Compute the closed-loop nominal model $T_\Delta(z)$;
- Compute the uncertainty bound $\|\Delta(\cdot)\|_\infty$;
- Compute $\|\kappa_\infty T_\Delta(z)\|_\infty$ and verify Ineq. (3.51).

As a final remark regarding approach (iv), we stress that the previous discussions of this tool were presented considering the class of qLPV systems with affine parameter dependency. Nonetheless, other classes could also be considered, such as polynomial forms $A(\rho(k))=A_0+A_1\rho(k)+A_2\rho^2(k)+\dots$ or LFT forms. The only difference would be the computation of the $u_\Delta(k)$ terms in the fifth sufficient condition (C5), which must embed the uncertainty term on the scheduling parameter caused by a bad estimation at a given instant. As an example, or the second-order polynomial case, we would obtain:

$$u_{\Delta_x}(k) \;=\; A_1\xi_\rho(k)+A_2(\xi_\rho(k)+2\rho(k)\xi_\rho(k))\xi_\rho(k)\,. \tag{3.52}$$

3.7 Illustrative results

In order to conclude this Chapter, we provide several results in order to illustrate the different scheduling trajectory estimation approaches. Specifically, we choose to discuss approach (iv) in more details, since this is a major contribution from this thesis. We also include comparisons to approaches (i) and (iii), being these widely used in the literature. Note that approach (ii) is not included in these results, since comparisons and discussions with regard to approach (i) have already been presented in the prequel. Next, we consider three different qLPV case studies are considered: a numeric example, a semi-active suspension system, and a pendubot system. Some aspects regarding these systems are relevant:

1. The first case study is chosen because it exhibits a time-varying derivative behaviour $f_\rho^\partial(k+j)$. Therefore, it serves to demonstrate that even if Assumption 3 is violated, if the five conditions from Lemma 3.6.2 are satisfied, the recursive estimation approach (iv) is still able to ensure convergence.
2. The second and third case studies are chosen from the literature because qLPV MPC techniques have proposed to control these systems (see [Morato, Normey-Rico, and Sename 2019] and [Cisneros and Werner 2019], respectively). The second case presents some load disturbances which meddle with the stabilisation of the process, while the third has a larger number of states (six). Moreover, both these cases have constant or null derivative terms $f_\rho^\partial(k+j)$, differing from the first.
3. For these systems, the corresponding state-feedback MPCs in the form of $u = \kappa(\rho)x$ are synthesised with identity Q and R weights and a parameter-dependent $P(\rho)$.

3.7.1 Numerical benchmark

Consider a qLPV system operating at a sampling rate of 1000 Hz. It is controlled by an MPC loop which operates within $T_s = 1$ ms. The system model is $x(k+1) = A(\rho(k))x(k) + B(\rho(k))u(k)$, with:

$$\left\{\begin{array}{rcl} A(\rho(k)) & = & \begin{bmatrix} -0.5(1+\rho(k)) & 0 \\ 0 & -0.3(1+\rho(k)) \end{bmatrix}, \\ B(\rho(k)) & = & \begin{bmatrix} (1+\rho(k)) \\ 2(1+\rho(k)) \end{bmatrix}, \\ \rho(k) & = & f_\rho(x(k)) = \sin\left(\begin{bmatrix} 1 & 1 \end{bmatrix} x(k)\right), \\ f_\rho^\partial(k+j-1) & = & \begin{bmatrix} 1 & 1 \end{bmatrix} \cos\left(\begin{bmatrix} 1 & 1 \end{bmatrix} x(k+j-1)\right). \end{array}\right. \tag{3.53}$$

This system has box-type constraints as are those in the Set of Assumptions 1, with: $\underline{x} = -[0.5\,0.3]^T$, $\overline{x} = [0.5\,0.4]^T$, $\underline{u} = -0.025$, $\overline{u} = 0.025$, $\underline{\rho} = -1$, $\overline{\rho} = 1$. Both state deviation and scheduling parameter deviation variables are energy-bounded.

With respect to the five sufficient conditions given in Lemma 3.6.2, we note that they are all satisfied:

(Ci) A sine function is class $\mathcal{C}^{\infty}$ with respect to its domain;

(Cii) A cosine is always energy-bounded;

(Ciii) Due to the box-type constraints on x, $\|\Delta x(\cdot)\| \leq 0.8$;

(Civ) ndeed, the closed-loop qLPV system is stable under the corresponding MPC algorithm;

(Cv) Algorithm 2 verifies the inequality from (C5) for all $\rho \in \mathcal{P}$: it holds that $\sup_{\rho(k)\in\mathcal{P}} \|\kappa_{\infty} T_{\Delta}(z)\|_{\infty} = 0.1651$, while $(\|\Delta(\cdot)\|_{\infty})^{-1} = 0.6667$.

The corresponding simulation results are presented in the sequel, which show that the recursive extrapolation (approach (iv)) converges within roughly five samples (i.e. 5 ms). The control horizon is taken as $N_p = 30$ samples. The total simulation run comprises 0.2 s. The system is perturbed by a disturbance signal at $t = 0.1$ s. Figure 3.9 shows the stabilisation of the states and the predictive control policy. Most importantly, Figure 3.10 shows[19] the extrapolation of P_k obtained with approach (iv), at different sampling instants, and the actual qLPV scheduling trajectories, demonstrating that convergence of $\hat{P}_k$ is indeed verified.

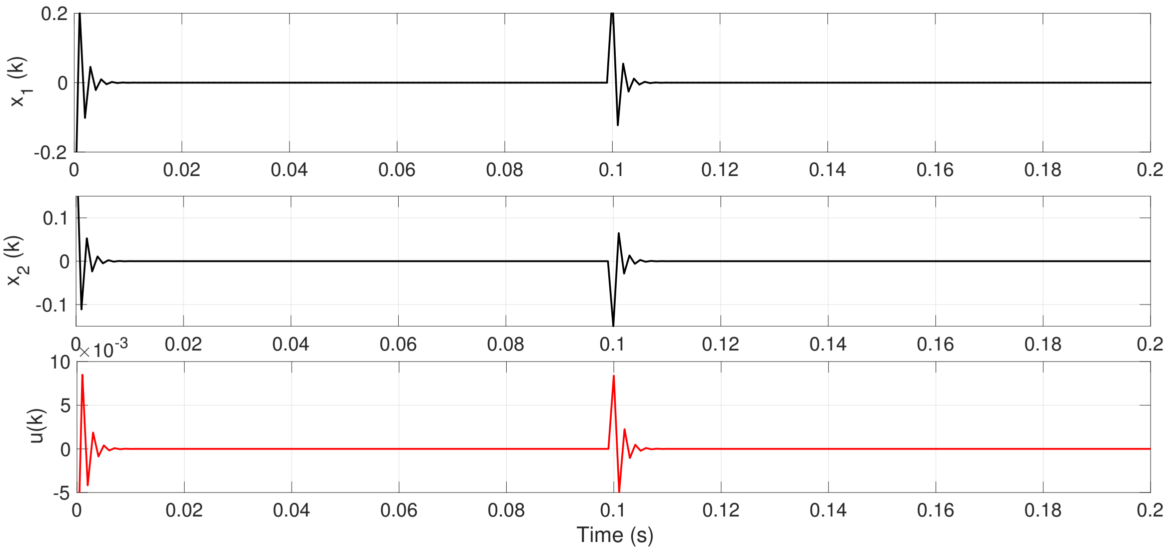


Figure 3.9: Numerical example: State stabilisation (black line) and control signal (red line).

For illustration purposes, we compare approach (iv) to approaches (i) and (iii), since these are widely used in the literature. Approach (i) is name, for simplicity, as “frozen-guess”, and (iii) as "iterative SQPs". These MPC methods are synthesised with the same tuning weights. We compare the obtained performances in terms of NRMS indexes for each state trajectory, which are presented in Table 3.3. As it can be seen, the resulting closed-loop performances are very similar with the proposed method and the one by [Cisneros and Werner 2020], both slightly superior than the frozen guess method (baseline indexes). Nevertheless, we must stress

[19] The real scheduling parameter trajectory is presented in bold black line. The coloured lines represent the extrapolated scheduling sequence at different time instants.

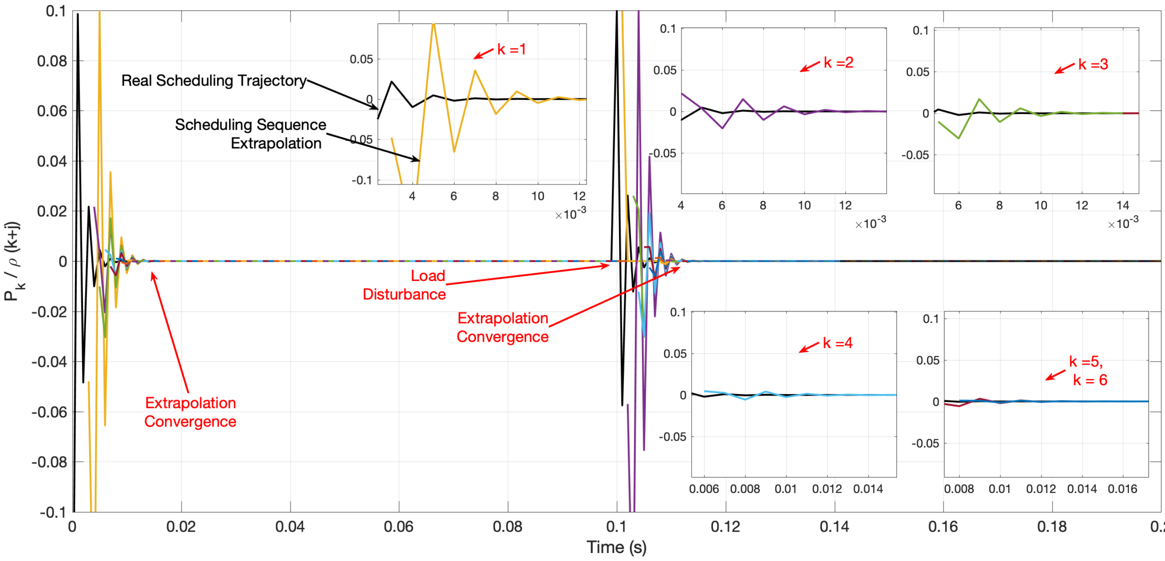


Figure 3.10: Numerical example: Scheduling trajectory extrapolation and convergence.

that the MPC with the proposed extrapolation mechanism can operate four times faster than the state-of-the-art SQP scheme, since it does not require to evaluate any nonlinear vector-wise operation online, neither multiple QPs.

In order to further illustrate the convergence of the scheduling sequence estimates, Figure[20] 3.11 compares the extrapolations derived with the proposed method against the estimated with the iterative mechanism (approach (iii), as detailed in [Cisneros and Werner 2020]). In this Figure, P_k (real trajectory) and $\hat{P}_k$ (estimate) are given for the first six simulation samples. Note that $\hat{P}_0$ is an initial guess. Clearly, both methods converge in roughly the same number of samples. The discrepancy between $\hat{P}_5$ and P_5 are slightly smaller with the iterative approach (iii), but this advantage comes at the expense of more computational cost. Nevertheless, we note that these discrepancies can be reduced with adequate forgetting factors (see Remark 15), such that the prior estimate data $\hat{P}_{k-1}$ has less effects on the following estimate $\hat{P}_k$.

Table 3.3 evidences an important feature of the proposed method, which we highlight. The additional computational time t_c required to solve the recursive extrapolation mechanism is of 0.01 ms (in average), with respect to the "frozen guess" MPC. Even with such minor additional computational load (of roughly 8%, in this case), the closed-loop performances are much enhanced. We stress that the total computational load of an MPC scheme operating together with the proposed extrapolation algorithm is very close to that of a QP. This is due to the fact that the operation of Eq. (3.43) consists only of linear vector-wise operations, whose numerical toughness depends linearly on the size of the prediction horizon N_p and on the number of number of scheduling parameters n_p.

In the case of systems with a higher number of states, the numerical load required by

[20]The real scheduling parameter trajectory is presented in bold black lines, the estimates with the approach (iv) are given in dotted blue lines, and the estimates with approach (iii) are found in dashed blue lines.

proposed estimation law will represent an even smaller ration with respect to the load required by the MPC QP, which grows exponentially with n_x. This is a very relevant advantage of the proposed method, since the state-of-the-art iterative SQP mechanism (approach (iii), from [Cisneros and Werner 2020]) grows exponentially with respect to the number of states n_x and with the prediction horizon size N_p, while also being proportional to the (maximal) number of iterations of the mechanism n_{iter}. Thereby, for systems with an elevated number of states, approach (iii) may easily violate the sampling period threshold of real-time applications ($t_c < T_s$), while the proposed scheme may not, since it will require, basically, the computational time needed for a single QP.

Table 3.3: Numerical example: Performance evaluation.

Method	NRMS$\{x_1\}$	NRMS$\{x_2\}$	t_c
Approach (i): "frozen-guess"	100 %	100, %	0.124 ms
Approach (iii): "iterative SQPs"	84.62 %	89.86 %	0.632 ms
Approach (iv): "recursive extrapolation "	83.84 %	89.86 %	0.134 ms

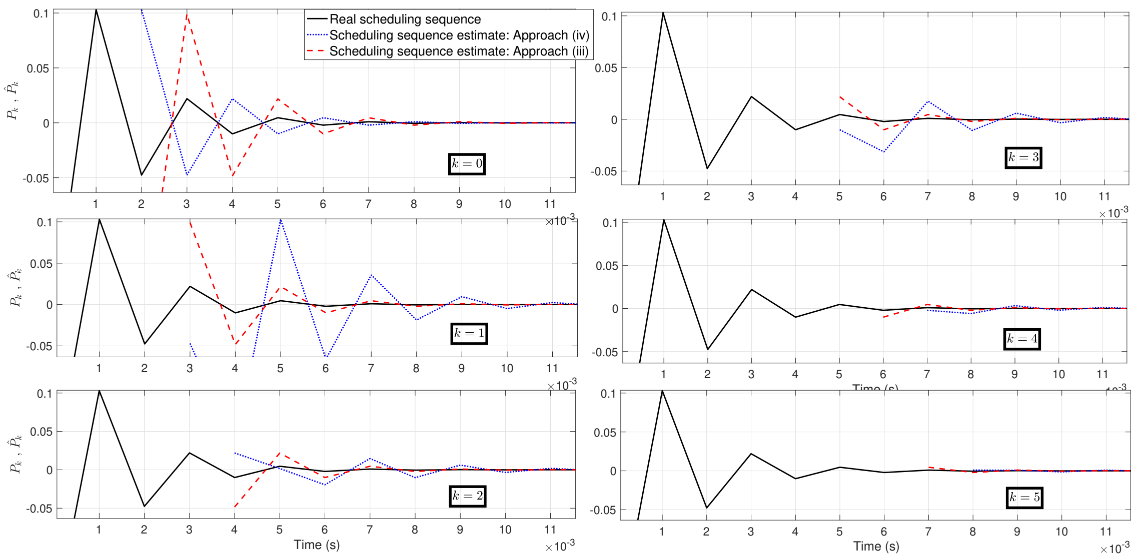


Figure 3.11: Numerical example: scheduling trajectory estimation convergence (approaches (iii) and (iv)).

3.7.2 Semi-Active suspension system

As a second study case, we consider a semi-active suspension system, represented by a quarter-car qLPV benchmark model from the literature, as presented in the prequel (Section 3.4.1). Recall that these dynamics comprise the displacements and velocities of a car's chassis and wheel (four states). The scheduling parameter is the suspension deflection velocity. This system operates with a $T_s = 5$ ms sampling period, and is disrupted by road bumping (load disturbances). The MPC horizon is again chosen as $N_p = 10$.

The model is $x(k+1) = A(\rho(k))x(k) + B(\rho(k))u(k) + B_2 w(k)$, where $w(k)$ are the road disturbances and $\rho(k) = (x_2(k) - x_4(k))$. Recall that the system has box-type constraints on states and inputs. Further details (and all parameter values and constraints) are given in references [Morato et al. 2018a; Morato, Normey-Rico, and Sename 2020b]. Again, all sufficient conditions from Lemma 3.6.2 are satisfied:

(Ci) The difference $(x_2(k) - x_4(k))$ is a linear operator and thus class $\mathcal{C}^\infty$;

(Cii) The derivative term $f_\rho^\partial(k) = [0\,1\,0\, -1]$ is energy-bounded;

(Ciii) The box-type constraints ensure energy bounds on Δx;

(Civ) The qLPV model is stable in closed-loop;

(Cv) Algorithm 2 verifies the inequality from (C5) for all $\rho \in \mathcal{P}$: it holds that $\sup_{\rho(k)\in\mathcal{P}} \|\kappa_\infty T_\Delta(z)\|_\infty = 15.54$, while $(\|\Delta(\cdot)\|_\infty)^{-1} = 66.73$.

We consider a simulation scenario of 10 s with ± 5 mm bump-like road disturbances at three different instants. The extrapolation algorithm converges within 25 samples, i.e. 0.125 ms. The average computational stress is of 0.048 ms. Figure[21] 3.12 shows the stabilisation of the states to the origin, the predictive control policy and the road profile disturbances, while Figure[22] 3.13 shows the real scheduling trajectories P_k and the corresponding estimates $\hat{P}_k$, obtained with approach (iv) at different sampling instants over the simulation run.

In order to corroborate the comparison discussions from the first example, we again compare the results obtained with approaches (i), (iii) and (iv). All MPCs are synthesised with the same tuning weights. In this example, due to larger number of states ($n_x = 4$) than the prior, we compare the obtained performances in terms of the NRMS index for the stage cost trajectory $\ell(x(k), u(k))$, which are presented in Table 3.4. Firstly, we stress that the frozen-guess mechanism (approach (i)) already obtains good driving performances by itself, as argued in [Morato, Normey-Rico, and Sename 2020b]. Nevertheless, as show in Table 3.4, the resulting closed-loop performances are enhanced with the recursive mechanism (approach (iv)) as with the iterative method (approach (iii), both with superior indexes than the "frozen" gain-scheduled baseline result. As indicated in previous discussions, the proposed extrapolation mechanism yield almost negligible computational time with respect to the time required by the QP. Furthermore, the iterative SQPs approach almost violates the sampling time constraint of 5 ms, taking over ten times more than the MPC coupled with the proposed estimation procedure (on average). This occurs since the QPs grow exponentially with the number of states, while the extrapolation does not.

[21] State dynamics are shown in bold, black and dashed, light blue lines, the control input is given in bold, red line, and the road disturbances is shown in bold, blue line.

[22] The real scheduling parameter trajectory is presented in bold black line. The coloured lines represent the extrapolated scheduling sequence at different time instants.

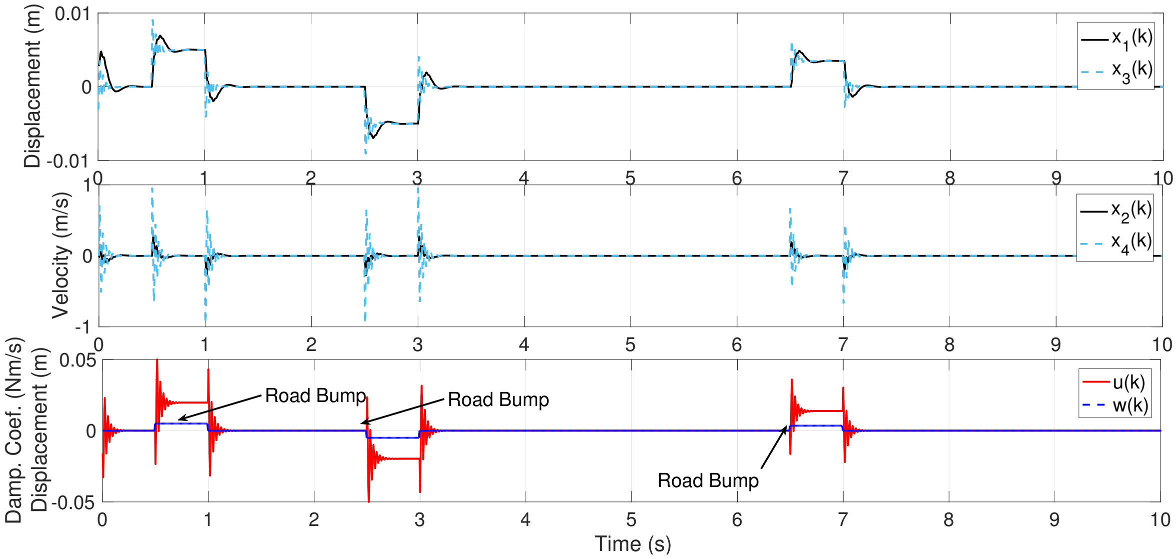


Figure 3.12: Vehicle suspension system: State stabilisation, control input, and road disturbances.

Table 3.4: Semi-Active suspension system: Performance evaluation.

Method	NRMS$\{\ell(x,u)\}$	t_c
Approach (i): "frozen-guess"	100 %	0.370 ms
Approach (iii): "iterative SQPs"	56.85 %	**3.795** ms
Approach (iv): "recursive extrapolation "	68.06 %	0.377 ms

3.7.3 Pendubot

A final simulation example is provided next. Consider the six-states pendubot benchmark system from [Cisneros and Werner 2019]. This inverted pendulum has two arms at rotating angles measured with respect to the vertical axis. A motor is connected to the first arm and acts as the actuator in this system. The control goal is to stabilize the system at a vertical up-up equilibrium (origin). The dynamics of this system may be disturbed by the occurrence of unexpected torques against the rotating arms.

This pendubot is represented by a discrete-time qLPV model in the form of Eq. (3.11), with an additional $w(k)$ term summed to the state transition map, which represents the torque disturbance. The system operates under a sampling period T_s of 10 ms. The model exhibits two state-related scheduling parameters: $\rho(k) = [x_1(k)\,,\, x_3(k)]^T$. This system is controlled by a sub-optimal MPC algorithm with horizon $N_p = 40$ steps. All matrices, parameters and MPC weights (P, Q, and R) are given in [Cisneros and Werner 2019]. Once again, all five sufficient conditions from Lemma 3.6.2 are satisfied. It holds that $\sup_{\rho(k)\in\mathcal{P}} \|\kappa_\infty T_\Delta(z)\|_\infty = 0.0845$, while $(\|\Delta(\cdot)\|_\infty)^{-1} = 0.2140$.

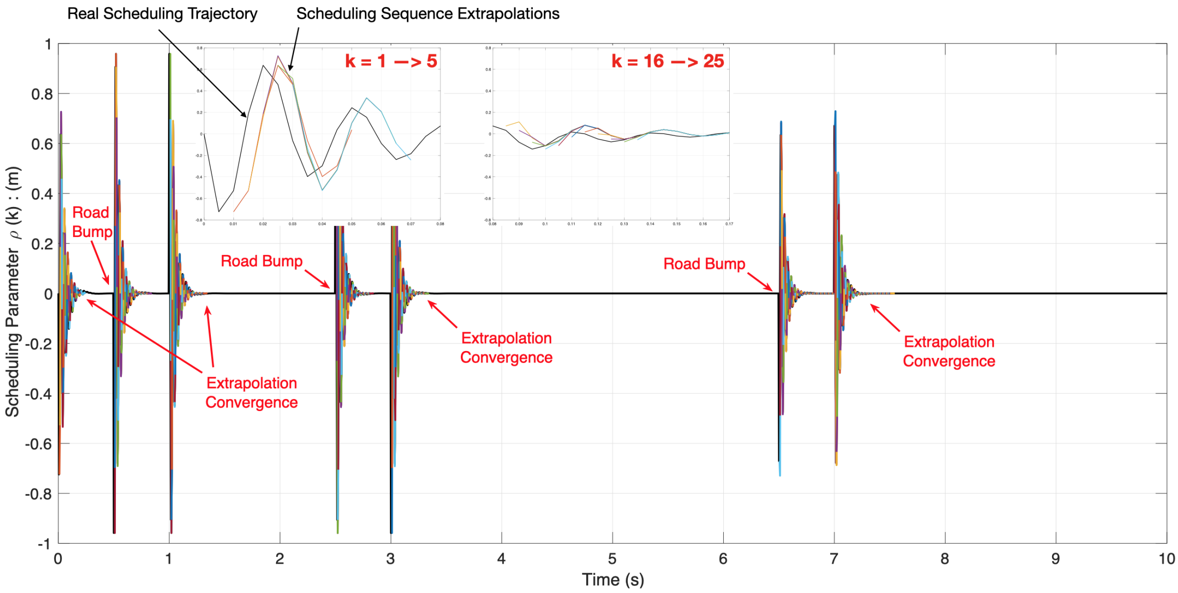


Figure 3.13: Vehicle suspension system: Scheduling trajectory and estimate (approach (iv)).

In order to illustrate this the closed-loop behaviour of this process under the action of an MPC algorithm based on the proposed recursive extrapolation procedure, we consider a simulation run of 3 s. In this scenario, the initial conditions are non-null and a load disturbance torque that occurs at $t = 2$ s, which requires the controller to stabilise the pendubot at the up-up equilibrium (steering the state trajectories to the origin) twice. In Figure[23] 3.14, we show the obtained trajectories of this system, considering the stabilisation of the first four states (positions and velocities, accelerations are suppressed for simplicity), as well as the generated predictive control policy. Complementary, Figure[24] 3.15 presents the real scheduling trajectory and the estimates generated with approach (iv)[25], which clearly convergences rapidly.

The average computational time required to evaluate the recursive extrapolation mechanism is of 2 ms. As of this, the total control law (extrapolation and QP solution) is evaluated within the $T_s = 10$ ms sampling period threshold. We also note that the extrapolation convergence is achieved within 0.5 ms (both due to initial conditions and due to the torque disturbance). The obtained performances are coherent with those presented in [Cisneros and Werner 2019]. Evidently, approach (iv) is suitable for a wide variety of applications.

[23]The state dynamics are given in black, light blue, light purple, and gray lines, while the control signal is shown using a red line.

[24]The real scheduling parameter trajectory is presented in bold black line. The coloured lines represent the extrapolated scheduling sequence at different time instants.

[25]Only one of the scheduling parameters is shown, for simplicity. Similar results were obtained for the other parameters.

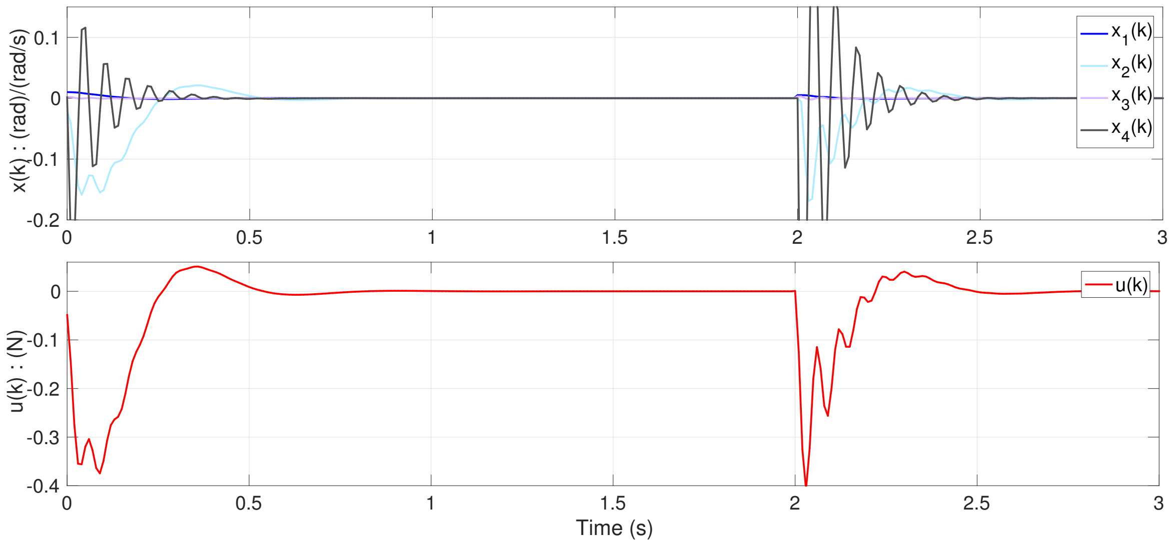


Figure 3.14: Pendubot system: State stabilisation and control signal.

3.8 Discussion

Corroborated by the simulation results presented in the prequel, we provide some comments:

- Approach (i), based on a frozen-guess estimate of the scheduling trajectories, provides sufficient performances, in many cases. It is able to operate rapidly online, since the resulting MPC is a based only on a single QP. Nevertheless, are argued in Section 3.3, the controller must be excessively robust in order to tolerate the prediction uncertainties, which thus may lead to conservative performances.

- Approach (ii) is an alternative to enhance the performances of the frozen-model mechanism. It is more appropriate for the qLPV setting, since the identification mechanism requires, in general, an activation signal for the auto-regressive transfer (such as endogenous variables which affect the scheduling parameters). Anyhow, we are unable to ensure if the method indeed converges, and bounds on the estimation error are also unavailable. The resulting MPC has the complexity of a QP and a LS (linear) recursive solution, being possible for real-time applications.

- Approach (iii) has been widely applied in recent works, including robustified, tube-based MPC variations, e.g. [Abbas et al. 2019; Hanema, Tóth, and Lazar 2021]. The method has many empirical proofs of convergence of the scheduling trajectory estimates, thus having valuable importance. Nevertheless, there is guarantee that the method converges rapidly (within the sampling period threshold), nor there can be generated estimation error bounds. As evidenced in the previous results, and also in [Cisneros and Werner 2020], the resulting MPC is comparable to benchmark state-of-the-art NMPC solutions, while operating with an online complexity of SQPs.

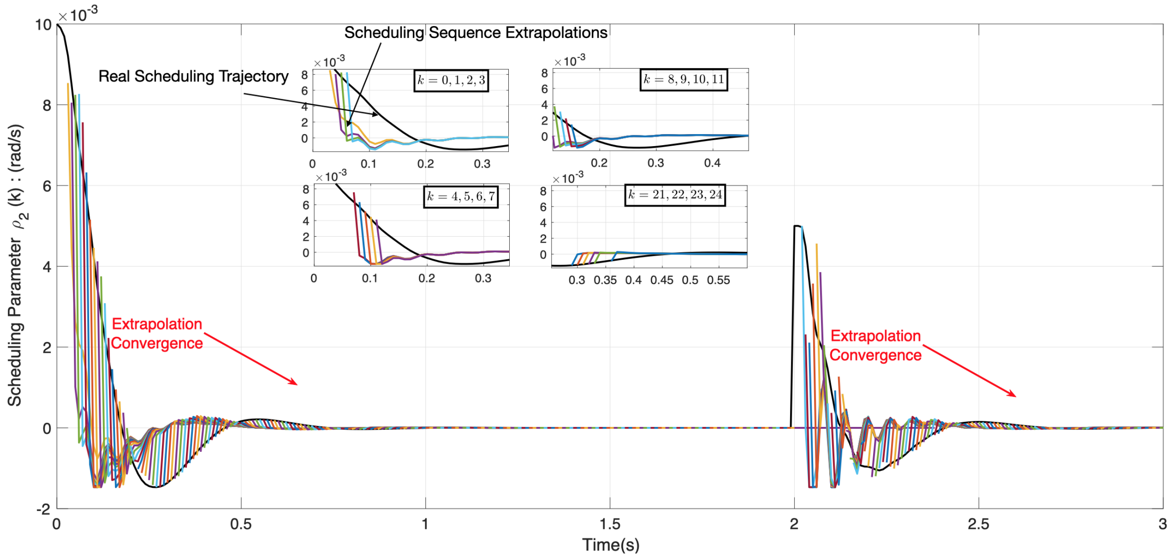


Figure 3.15: Pendubot system: Scheduling trajectory estimate (approach (iv)).

- Approach (iv), one of the main contributions of this thesis, is a recursive estimation procedure based on rather simple Taylor expansion argument. With this method, we are able to ensure convergence of the estimation, as long as the five sufficient conditions from Lemma 3.6.2 are satisfied. Furthermore, we can compute theoretical bounds on the estimation error during the transient behaviour, which can be accounted for in the context of robust MPC design. Furthermore, the method is able to enhanced MPC performances with respect to approach (i), while maintaining computational load close to that of a single QP, thus being faster than approach (iii). The recursive extrapolation mechanism resides on simple linear operators, with numerical toughness growing linearly with the prediction horizon size N_p and the number of scheduling variables n_ρ.

- Since approach (iii) has a numerical toughness that grows exponentially with the prediction horizon size N_p and with the number of system states n_x, in the case of larger order real-time systems, approach (iv) may be much more suitable, since it requires much less computational load.

3.9 Conclusions and perspectives

This Chapter presented an overview of the available method that can be used to estimate the future LPV scheduling parameters, in order to construct prediction laws in the context of MPC. Four methods were detailed, with corresponding simulation results shown and discussed. Most importantly, a novel approach is presented for the case of qLPV systems, residing in a simple and fast recursive law, which only needs the evaluation of a partial derivative computation at each sampling instant. Furthermore, five simple-to-verify sufficient conditions are presented for the convergence of the this mechanism.

As argued in the prequel, the developed approach is compared to the state-of-the-art mechanism of estimating scheduling parameters through SQPs (looping the MPC multiple times), showing equivalent estimates and similar convergence rate. For the avid Reader, and in order to further illustrate the applicability of this method, more comparisons and discussions have been presented in [Morato et al. 2021d; Morato et al. 2022b].

Overall, we note that the described scheduling trajectory estimation schemes can certainly serve for the design of fast LPV MPC algorithms. Also, these can be exploited for the design of NMPC by the means of LPV embeddings: the resulting controller has computational complexity smaller than a nonlinear program, since the nonlinearities from the model prediction constraints are removed in al LPV fashion.

Chapter 4

State- and output-feedback

In this Chapter, we discuss the exploitation of gain-scheduled LPV MPC design under novel state-feedback and output-feedback formulations. Accordingly, the main contributions presented in this Chapter are the following:

- First, we present a comfort enhancement algorithm for passengers in a vehicle with semi-active suspensions, synthesised under a predictive control formulation. The considered MPC application takes into account a qLPV model of the car dynamics and embeds a comfort performance index as its cost function. The proposed method is sub-optimal due to the fact that it is based on a frozen estimation for the scheduling parameters along the horizon. Bounds on the variation rates of the scheduling parameters are taken into account and, thus, the uncertainty propagation is relieved. We handle the uncertainty issue using set-based terminal ingredients. Successful realistic nonlinear simulations of a scaled car are presented, comparing the developed solutions to other optimal controllers. Results illustrate the overall good operation of the vehicle; the comfort of the passengers is substantially improved, as measured through time and frequency domain indexes.
- Secondly, we present an MPC algorithm for qLPV systems represented in the Input-Output (IO) form. The method is based on the recursive Taylor-based extrapolation mechanism from Chapter 3. The main innovation is that, by using an IO description of the system dynamics, state measurements are not necessary, which is interesting from an industrial and practical application perspective (no need for observer design, for instance). In order to ensure offset-free reference tracking, the algorithm includes an explicit integral action formulation, which, coupled with quadratic terminal ingredients, also enable asymptotic IO stability. A numeric benchmark example is used to illustrate the advantages of the proposed method, as well as its real-time capabilities.

Remark 22. *The developments presented in this Chapter correspond (in parts) the works published in [Morato, Normey-Rico, and Sename 2019; Morato, Sename, and Dugard 2019b; Morato, Normey-Rico, and Sename 2020c; Morato, Normey-Rico, and Sename 2021c] (state-feedback formulation) and [Morato, Normey-Rico, and Sename 2022a; Morato 2023] (output-feedback solution).*

4.1 Organisation

Regarding the organisation of this Chapter, we divide it into two main branches:

1. Regarding the state-feedback formulation for vehicle suspension control:
   - In Section 4.2, we motivate the topic of automotive suspension control, presenting the main reasons why MPC is a suitable approach for this (open) control problem.
   - In Section 4.3, we present control-oriented qLPV model for vehicular SA suspension systems. We also detail how model predictions are generated using a frozen scheduling trajectory guess mechanism, and debate the corresponding uncertainties.
   - In Section 4.4, we present the corresponding qLPV MPC algorithm setup, conceived with terminal set constraints generated with regard to the prediction uncertainty propagation.
   - In Section 4.5, the proposed method is successfully applied to high-fidelity vehicle benchmark. Results are debated according to standard performance indexes [Poussot-Vassal et al. 2012], in both time and frequency domains.
2. Then, with respect to the output-feedback MPC:
   - In Section 4.8, we propose an explicit integral action scheme in the context of input-output LPV MPC, which enables offset-free output tracking of piece-wise constant reference trajectories. The scheme includes the future scheduling parameter estimation scheme from Chapter 3, using a Taylor argument to conceive a recursive extrapolation law, herein adapted to the IO case.
   - Benefiting from the stability framework from [Mayne et al. 2000] as expanded to the LPV IO setting in [Abbas et al. 2016], we provide, in Section 4.9, quadratic terminal ingredients that ensure a recursively feasible optimisation, and an asymptotically stable closed-loop. These ingredients are enabled through a sufficient LMI constraint.
   - In Section 4.10, we use a numeric benchmark example from the literature in order to demonstrate the effectiveness of the proposed IO LPV MPC method, as well as its advantages and capabilities of the algorithm for real-time nonlinear applications.

4.2 Automotive suspensions and control

Automotive suspension systems are able to enhance 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 [Savaresi et al. 2010]. SA suspensions are well-performing and energy-efficient, being altogether less expensive than purely active ones [Fischer and Isermann 2004]. In active suspension systems, damping forces can be implied in both the direction of the deflection movement, as well as in its opposition. In SA systems, the damping force can only be provided in the same direction as the deflection movement, benefiting from it and thus consuming less energy (the damping force is dissipative, by nature). The key investigation issue in the SA suspension research body is how to provide real-time laws for the controllable damper, enabling more comfortable rides for the passengers altogether with easier maneuvering of the vehicle for the driver (keeping the car closer to the ground, with less roll angle and less vertical trepidation). The difficulty resides in how to handle the dissipativity constraints of the SA dampers while ensuring good performances. The key references [Poussot-Vassal et al. 2012; Tseng and Hrovat 2015] detail some of the available methods proposed for this goal (see, also, the cite references therein). Some of the most modern techniques have been tested for these systems, such as (clipped) optimal LQRs in [Unger et al. 2013], H_∞ techniques [Nguyen et al. 2015], nonlinear and Linear Parameter Varying (LPV) strategies [Poussot-Vassal et al. 2008].

Of particular interest to this thesis, we can also find results proving the application of MPC to the problem of SA suspension control [Beal and Gerdes 2013; Nguyen et al. 2016; Morato et al. 2018a], which is a topic increasingly sought by the automotive industry, in particular when considering more complex, nonlinear models. MPC is a natural framework to address the issue of SA suspension control, since it facilitates optimal performances of constrained processes and is able to consider input and state constraints in the design process. 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, which makes this kind of problem fall into a saturation paradigm which can be elegantly dealt by MPC.

In the sequel, the focus is given to reduced-order car frameworks (such as quarter-car or 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-related variables is small, as discusses [Nguyen et al. 2016]. Literature has indeed shown some interesting SA suspension control solutions using MPC based on reduced-order models. These approaches include; sub-optimal, clipped (saturated) MPC propositions [Brezas, Smith, and Hoult 2015]; methods based on LPV quarter-car models [Morato, Sename, and Dugard 2018], with no theoretical feasibility guarantees; fast half-car experimentally-tested methods [Beal and Gerdes 2013], which are able to operate within 10 ms, but cannot account for the effect of the road disturbances. [Poussot-Vassal et al. 2012] is highlighted, since it provides a methodology for performance evaluation of SA suspensions under optimal control algorithms (MPC included).

With respect to [Nguyen et al. 2016; Morato et al. 2018a], fast MPC algorithms were developed considering full car models. The input nonlinearity and the dissipativity constraints were handled with the use of a pre-filter, which made the model, from the MPC viewpoint, LTI. This pre-filtering technique, for practical purposes, may cause implementation distress, given that a bilinear term $\dot{z}_{\text{def}}(k)u(k)$ is converted into a linear term $\dot{z}_{\text{def}}(k)u_{nom} + u_f(k)$, which means that a division by $\dot{z}_{\text{def}}(k)$ is necessary and, for situations when this velocity term approaches zero, the pre-filtering may result in deteriorated performances. Note that near-zero piston velocity situations are very common in SA suspensions (as in constant straight road profile, for instance), and, thus, have to be taken into account by the design method. We also note that [Morato et al. 2018a] shows a solution implemented within 5 ms, but residing in sub-optimal computations due to the use of heuristics.

From the above-referenced literature, we can note that the main MPC algorithms applied for automotive SA suspension systems which 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 with respect to other control frameworks. Indeed, most papers show good performance enhancements. Nonetheless, as of today, 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 part of the current Chapter is to present a predictive control algorithm for vehicular SA suspensions that embed the recursive feasibility property.

In general, reduced-order SA suspension models that are able to handle the damper dissipativity constraints exhibit nonlinear characteristics. The pre-filtering method, as discussed previously, is not such a good option concerning real implementation. Nonlinear MPC (NMPC) algorithms are indeed able to handle these dissipativity constraints without using pre-filtering, but they are usually not able to run fast enough (in real-time). Recently, as addressed in the first Chapters of this thesis, works have shown how nonlinear MPC can be generated by exploiting qLPV realisation [Morato, Normey-Rico, and Sename 2020a]. Indeed, SA suspension systems can be formulated by the means of a qLPV description, as illustrated in [Morato et al. 2018a].

Motivated by the prior discussions, our aim is to develop a predictive control policy for SA suspensions that enhances the comfort of the onboard passengers. For such, we extend and adapt the state-feedback qLPV MPC algorithm from [Morato, Normey-Rico, and Sename 2019] to this context, herein also considering bounded rates of the scheduling parameters, as suggested by [Jungers, Oliveira, and Peres 2011]. The motivation of improving comfort performances lies in providing a better ride for the passengers, with less vertical trepidation, which can be much more enjoyable than a "shaky" ride; passengers will be much less prone to feel nausea or queasiness under less vertical acceleration.

4.3 Models, performances and constraints

We consider a vehicular suspension system which comprises four electro-rheological dampers. Accordingly, the control system is composed of four MPC algorithms, one concerned and tuned with regard to the performances and constraints of each one of the vehicle's corners. Indeed, we show next how a qLPV representation can be used to express the corner dynamics of a moving car. Such qLPV model is also able to express the nonlinear dissipativity constraints of the dampers in the form of linear constraints over the control input.

4.3.1 Vehicle testbed

For accurateness and fidelity purposes, all tests are performed using a high-fidelity benchmark system, which is derived using complete nonlinear vehicle dynamics [Morato et al. 2018a], i.e. taking into account couplings, measurement noises, hysteresis, and so forth. The considered benchmark is validated from a diverse set experimental essays, based on a real mechatronic testbed: the *INOVE Soben-Car*, a one-fifth sized vehicle (show in Figure 4.1[1]). The SA dampers in this testbed are electro-rheological, which means that the PWM signal u regulates an electric field which varies the viscosity of a rheological fluid found inside the damper chamber, increasing or decreasing the delivered force.

Figure 4.1: *INOVE Soben-Car* mechatronic testbed.

The real nonlinear behaviour of the SA dampers is shown in Figure 4.2, in terms of force *vs.* deflection speed diagrams. Real experimental data is shown at the left side and fitted data at the right side. The "dissipativity constraints" of these damper are the following: the control action must ensure that the damping force only acts in the same direction as the deflection speed movement. This is conversely expressed as a linear contraint over the control input signal $u(t)$, which should belong to the convex set $\mathcal{U}$.

Remark 23. *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* $5\,ms$*. This sampling rate is realistic and seen in many top-cars [Poussot-Vassal et al. 2012].*

4.3.2 The control-oriented model

As stated in the prequel, we represent the dynamics of the described vehicle test-bench by the means of a control-oriented qLPV (reduced-order) corner model [Morato et al. 2018a]. This model is used for both control design and performance analysis purposes. The state-space description involves the vertical dynamics of the vehicle, at each corner, considering the

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

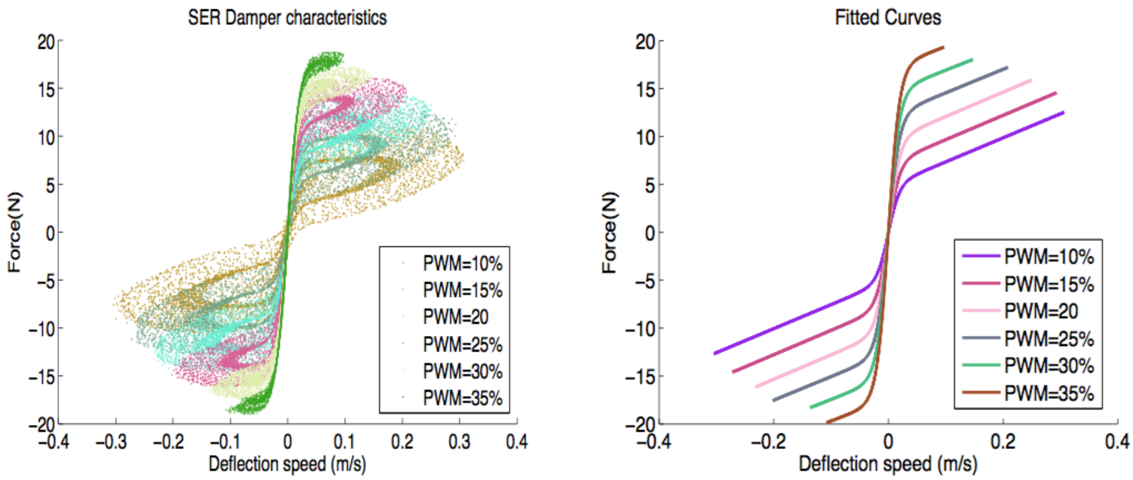


Figure 4.2: Force-speed characteristics of electro-rheological SA dampers.

chassis dynamics (z_s) and the displacements of the wheel link (z_{us}), which are meddled by the road profile disturbances (z_r). Figure 4.3 shows a schematic representation of a vehicle corner, which is governed through the following set of differential equations:

$$\begin{cases} m_s \ddot{z}_s(t) &= -F_s(t) - F_d(t)\,, \\ m_{us} \ddot{z}_{us}(t) &= F_s(t) + F_d(t) - F_t(t)\,, \end{cases} \tag{4.1}$$

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. Table 4.1 gives the parameters values for the considered testbed. The forced involved in the vertical dynamics of the car are further detailed: the spring force and the tire force are given as respectively proportional to the suspension deflection ($z_{\text{def}} = z_s - z_{us}$) and the wheel deflection, as follows:

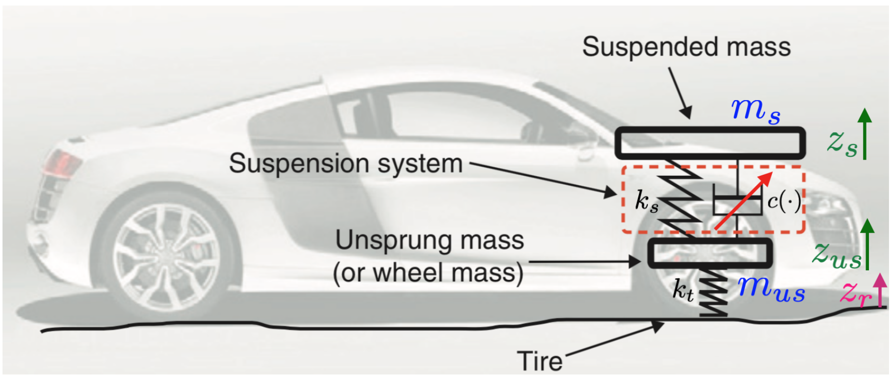


Figure 4.3: Vehicle with SA suspension system.

$$\begin{cases} F_s(t) &= k_s z_{\text{def}}(t)\,, \\ F_t(t) &= k_t\left(z_{us}(t) - z_r(t)\right)\,, \end{cases} \tag{4.2}$$

being k_s and k_t the stiffnesses of the spring and the tire, respectively.

In order to represent the dissipative behaviour of the damper force in qLPV control-oriented framework, it is traced as the static nonlinear map, as suggests [Guo, Yang, and Pan 2006], which can be used for both Magneto-rheological or Electro-rheological dampers (the technologies seen in the majority of SA suspension systems), as follows:

$$F_d(t) \;=\; k_0 z_{\text{def}}(t) + c_0 \dot{z}_{\text{def}}(t) + \rho(t)u(t)\,, \tag{4.3}$$

where the scheduling parameter $\rho(t) = f_c \tanh\left(k_1 z_{\text{def}}(t) + c_1 \dot{z}_{\text{def}}(t)\right)$ directly embeds the hysteresis-like behaviour of the SA damper. The parameters k_0 and c_0 denote the nominal stiffness and damping coefficient of the SA damper. The suspension deflection velocity variable $\dot{z}_{\text{def}}(t)$ 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 time instant. Note that $c(\cdot) = (c_0 + \rho(t))$ stands for the complete semi-active damping coefficient. The dissipativity constraints of the SA damper are set upon $F_d(t)$, which must always lie within a feasibility set whose agrees with the experimental curves of Figure 4.2.

The control input $u(t)$ represents 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 rheological fluid. In practice, it is this PWM signal $u(t)$ that is the control system's manipulated variable for the suspension application.

The dissipativity set bounds the hysteresis behaviour of the hyperbolic-tangent function present in the scheduling parameter ρ. The SA damper force can only be given in the same direction of the deflection speed, being dissipative, in consonance with the experimental characteristics presented in Figure 4.2. Accordingly, the relationship between the damper force $F_d(t)$ and the deflection velocity $\dot{z}_{\text{def}}(t)$ can be also expressed through a time-varying proportional function, $F_d(t) = c(\cdot)\dot{z}_{\text{def}}(t)$, where the variation of $c(\cdot)$ derives from the control input $u(t)$, as discusses [Savaresi et al. 2010]. With regard to this representation, the dissipativity constraints can be represented through the following inequalities:

$$\overline{F}_d \;\le\; F_d(t) \;\le\; \overline{F}_d\,, \tag{4.4}$$

$$0 \;\le\; \underline{c} \le c(\cdot) \;\le\; \bar{c}\,. \tag{4.5}$$

Remark 24. *Since the nominal dissipativity parameter is constant (i.e. given by c_0), the constraints from Eqs.* (4.4)-(4.5) *can be expressed directly in terms of the control input variable $u(t)$, this is: $0 \le u(t) \le 1$. When $u = 0$, it follows that $c(\cdot) = \underline{c} = c_0$, and, for $u = 1$, $c(\cdot) \le c_0 + f_c = \bar{c}$.*

Remark 25. *As displayed in many papers from the literature with experimental validation included, e.g. [Unger et al. 2013; Ren et al. 2016; Morato et al. 2019b], observers can be used, using acceleration variables, to estimate the states of SA suspensions, considering corner models. Therefore, we assume that the system states are available (computed by some accurate observer). The issues of state measurements/estimation unavailability will be discussed, in the context of LPV MPC synthesis, at the end of this Chapter, considering IO model realisations.*

For the reasons discussed in the prequel, the two acceleration variables from Eq. (4.1) are considered as available process output measurements, that is: $y(t) = \begin{bmatrix} \ddot{z}_s(t) & \ddot{z}_{us}(t) \end{bmatrix}^T$.

Then the qLPV state-space realisation of the described suspension system is derived by directly applying Newton's second law of motion, i.e. Eq. (4.1), considering the system states as $x(t) = [z_s(t)\,, \dot{z}_s(t)\,, z_{us}(t)\,, \dot{z}_{us}(t)]^T$, which gives:

$$\begin{cases} \dot{x}(t) &= A_c x(t) + B_{c_1}(\rho(t))u(t) + B_{c_2} z_r(t)\,, \\ y(t) &= C_c x(t) + D_{c_1}(\rho(t))u(t) + D_{c_2} z_r(t)\,. \end{cases} \tag{4.6}$$

The matrices for the continuous-time qLPV model from Eq. (4.6) are:

$$\begin{cases} A_c = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{-(k_s+k_0)}{m_s} & \frac{-c_0}{m_s} & \frac{(k_s+k_0)}{m_s} & \frac{c_0}{m_s} \\ 0 & 0 & 0 & 1 \\ \frac{(k_s+k_0)}{m_{us}} & \frac{c_0}{m_{us}} & \frac{-(k_s+k_0+k_t)}{m_{us}} & \frac{-c_0}{m_{us}} \end{bmatrix}, \\ B_{c_1}(\rho(t)) = \begin{bmatrix} 0 & \frac{-\rho}{m_s} & 0 & \frac{\rho}{m_{us}} \end{bmatrix}^T, \\ B_{c_2} = \begin{bmatrix} 0 & 0 & 0 & \frac{k_t}{m_{us}} \end{bmatrix}^T, \\ C_c = \begin{bmatrix} \frac{-(k_s+k_0)}{m_s} & \frac{-c_0}{m_s} & \frac{(k_s+k_0)}{m_s} & \frac{c_0}{m_s} \\ \frac{(k_s+k_0)}{m_{us}} & \frac{c_0}{m_{us}} & \frac{-(k_s+k_0+k_t)}{m_{us}} & \frac{-c_0}{m_{us}} \end{bmatrix}, \\ D_{c_1}(\rho(t)) = \begin{bmatrix} \frac{-\rho}{m_s} & \frac{\rho}{m_{us}} \end{bmatrix}^T, \\ D_{c_2} = \begin{bmatrix} 0 & \frac{k_t}{m_{us}} \end{bmatrix}^T. \end{cases} \tag{4.7}$$

Remark 26. *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. A thorough discussion on this matter is available in [Morato et al. 2019b].*

We obtain a discrete-time realisation by using an Euler discretisation method with $T_s = 5\,\text{ms}$, which yields[2]:

$$\begin{cases} x(k+1) &= Ax(k) + B_1(\rho(k))u(k) + B_2 w(k)\,, \\ y(k) &= Cx(k) + D_1(\rho(k))u(k) + D_2 w(k)\,, \\ \rho(k) &= f_c \tanh\left(A_\rho x(k)\right)\,, \end{cases} \tag{4.8}$$

where the load disturbance variable is the vertical road profile, i.e. $w(k) = z_r(k)$, and the discrete-time matrices are:

$$\begin{cases} A_\rho &= \begin{bmatrix} k_1 & c_1 & -k_1 & -c_1 \end{bmatrix}, \\ A &= I_{n_x} + T_s A_c\,, \\ B_1(\rho(k)) &= T_s B_{c_1}(\rho(k))\,, \\ B_2 &= T_s B_{c_2}\,, \\ C &= C_c\,, \\ D_1 &= D_{c_1}\,, \\ D_2 &= D_{c_2}\,. \end{cases} \tag{4.9}$$

[2]The continuous-time matrices A_c, B_{c_1}, and so onm are derived from Newton's second law of motion, as gives Eq. (4.1). Parameter values are presented in Table 4.1.

Remark 27. *Due to physical limits of the SA suspension, constraints are also set upon the system states, considering $n_x = 4$ and $n_u = 1$:*

$$x(k) \in \mathcal{X} \ := \ \{x_j \in \mathbb{R}^{n_x} \,|\, \underline{x}_j \leq x_j \leq \overline{x}_j\}\,. \tag{4.10}$$

The input constraints are:

$$u(k) \in \mathcal{U} \ := \ \{u \in \mathbb{R}^{n_u} \,|\, 0 \leq u \leq 1\}\,. \tag{4.11}$$

Conversely, the dissipativity constraints are:

$$F_d(k) \in \mathcal{D} \ := \ \{F_d \in \mathbb{R}^{n_u} \,|\, \underline{F}_d \leq F_d \leq \overline{F}_d\}\,, \tag{4.12}$$

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

Table 4.1: SA suspension system 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	Ns/m
c_1	Hysteresis velocity coefficient	21	Ns/m
f_c	Dynamic yield force of the fluid	28.07	N

4.3.3 Performances indexes

The main problem faced when designing SA suspension controllers is how to determine a feasible control law that is able to physically isolate the vehicle body from the disturbances implied by the road through which the car is driven [Savaresi et al. 2010]. Vehicles with passive suspensions, which do not have controlled dampers, present drives for which the chassis vertically accelerates almost together with the road. This means that bumps on the road, for instance, are proportionally translated to the chassis. When SA dampers are used, the damping force can be used to counter-act the influence of the road and, thereby, the car becomes easier to maneuver. Anyhow, at the same time that the chassis should be isolated from the road, the comfort of the onboard passengers should also be enhanced, so that the ride becomes more enjoyable. Humans are prone to motion sickness and queasiness under harsh vertical acceleration.

The objectives of enhancing passenger comfort and providing easier maneuvering are physically conflicting. Stiffdamping enhances passenger comfort by reducing the chassis body acceleration, while smooth damping enables easier road holding by reducing the vibration of the wheels, e.g. [Poussot-Vassal et al. 2012]. For simplicity, we focus singularly in enhancing the first objective, of isolating the vehicle body from road trepidations and reducing vertical acceleration of the chassis. This is coherent with the considered platform application, for which the sole objective is minimising chassis trepidations. Multi-objective performances have been considered in [Nguyen et al. 2016; Morato et al. 2018a], for instance.

As proposed in [Poussot-Vassal et al. 2012], 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 $\ddot{z}_s$. The vertical chassis acceleration $\ddot{z}_s$ response to the road disturbances w can be evaluated within the range from 0 to 20 Hz, considering comfort specifications [Fischer and Isermann 2004]. Specifically, we employ two different criteria used to evaluate the comfort of the passengers, as suggested by [Poussot-Vassal et al. 2012]:

1. A comfort performance index in the time-domain:

$$J^t_{\text{comfort}} = \int_0^{\tau} \ddot{z}_s^2(t)dt\,, \tag{4.13}$$

   where τ represents a given period of time. Note that this is an integral index by definition. Its interpretation in discrete time is related to the sum of discrete samples for the chassis' acceleration variable, i.e.:

$$J^t_{\text{comfort}} \approx \sum_{k=0}^{\frac{\tau}{T_s}} T_s \ddot{z}_s^2(k)\,. \tag{4.14}$$

2. The most general criterion for comfort performances in the frequency-domain is the standard ISO 2631 norm, found in [International Organization for Standardization 2016]. This normalisation index can be applied not only to vehicles, but to all sorts of vibrating environments and systems. This standard defines the body vibration exposure limits within the frequency range of [1 , 80] Hz, wherein reduced comfort, decreased proficiency and preservation of healthy concerns are evaluated. According to this ISO rule, the human being is more sensible to vertical accelerations within $4 - 8$ Hz. The ISO 2631 criterion is applied as a filter with the following continuous-time transfer function from [Zuo and Nayfeh 2003]:

$$W_f(s) = \frac{0.1456s^4 + 0.2331s^3 + 13.75s^2 + 1.705s + 0.3596}{s^5 + 7.757s^4 + 19.06s^3 + 28.37s^2 + 18.52s + 7.23}\,,$$

   where $\ddot{z}_s^{\text{filtered}}(s) := W_f(s)\ddot{z}_s(s)$, being the later the car's COG acceleration. The frequency-domain results index follows:

$$J^f_{\text{comfort}} = \mathcal{C}(f\{\ddot{z}_s^{\text{filtered}}\}, 0, 20)\,, \tag{4.15}$$

   where $f\{\cdot\}$ 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, as thoroughly discussed in [Poussot-Vassal et al. 2012].

4.3.4 MPC design

MPC is a very elegant option for constrained processes and thus can be suitably incorporated to the context of suspension control. Recall that one of the objective of this Chapter is to propose a state-feedback LPV predictive control algorithm. Thus, we specifically orient this control scheme for the context of SA suspensions, in order to enhance the comfort performances of the onboard passengers. Such control algorithm must be realisable and run within the 5 ms sampling period of the considered SA suspension system. Moreover, the control policy must ensure input and state constraints are respected. Concerning the comfort performances of the vehicle and the discussion in Section 4.3.3, the MPC algorithm is designed by embedding the time-domain comfort index as a cost that is minimised along a rolling horizon. For prediction purposes, we use a *frozen* scheduling trajectory guess mechanism (i.e. Chapter 3, approach (i)). That is, we consider the following gain-scheduled state-feedback MPC problem[3]:

Problem 2. *State-feedback MPC*

$$
\begin{aligned}
U_k^\star \;&=\; \min_{U_k} \overbrace{\sum_{i=1}^{N_p} \ell\left((k+i|k), u(k+i-1|k)\right) + V(x(k+N_p|k))}^{J_k = J(x(k),U_k)} && (4.16)\\
s.t. \quad & x(k+i|k) = Ax(k+i-1|k) + B_1 A(\rho_k) u(k+i-1|k) && (4.17)\\
& \qquad\qquad + B_2 w(k+i-1)\,,\ \forall i \in \mathbb{N}_{[1,N_p]}\,,\\
& y(k+i|k) = Cx(k+i|k) + D_1 A(\rho_k) u(k+i|k) && (4.18)\\
& \qquad\qquad + D_2 w(k+i-1)\,,\ \forall i \in \mathbb{N}_{[1,N_p-1]}\,,\\
& u(k+i|k) \in \mathcal{U}\,,\ \forall i \in \mathbb{N}_{[0,N_p]}\,, && (4.19)\\
& y(k+i|k) \in \mathcal{Y}\,,\ \forall i \in \mathbb{N}_{[1,N_p-1]}\,, && (4.20)\\
& x(k+i|k) \in \mathcal{X}\,,\ \forall i \in \mathbb{N}_{[1,N_p-1]}\,, && (4.21)\\
& x(k+N_p|k) \in \mathbf{X}_f\,. && (4.22)
\end{aligned}
$$

Recall that U_k denotes the vector that comprises the sequence of actions previewed within the prediction horizon N_p. The output constraint in Eq. (4.20) is included for generality of argument. The complete MPC optimisation cost J_k is comprised of the sum of a stage cost $\ell(\cdot)$ along the horizon and of a terminal stage value $V(x(k+N_p|k))$. We will later show that J_k is implied as Lyapunov function that decreases over the discrete-time samples k, which thus ensures recursive feasibility of the optimisation (and stability of the closed-loop). The set $\mathbf{X}_f$ defines a terminal constraint for the MPC algorithm.

For the considered application, the future load disturbances $w(k+i)$, $\forall i \in \mathbb{N}_{[1,N_p-1]}$ (road profile) are required. Thus, through the sequel, we assume that $w(k) \in \mathbb{R}^{n_w}$ are assumed to be known at each instant k, for the corresponding next N_p steps. This means that this knowledge can be directly included to the model predictions in Eq. (4.17). We stress that, in the case of automotive applications, this is a quite reasonable assumption. Such road preview

[3]Here, we mark $\rho_k = \rho(k)$ in order to emphasise the, at each instant k, the prediction model is the frozen version of the LPV model.

information can be pursued with different schemes from the literature, such as frequency-based mechanisms, adaptative estimation schemes or even extended observers that estimate the road together with the states. Some options for these algorithms are available in [Unger et al. 2013; Tudón-Martínez et al. 2015; Morato et al. 2019b].

As argued in Chapter 3, when we use a frozen scheduling trajectory prediction, i.e. fixing $\rho(k+i|k) = \rho(k)$, $\forall i \in \mathbb{N}_{[1,N_p-1]}$ in Eq. (4.16), we should take into account the corresponding uncertainty propagation. We recall that using such frozen estimation mechanism is of interest since Problem 2 becomes a QP. Nevertheless, the numerical complexity relief comes at the expense of yielding sub-optimal performances due to model-process mismatches. In order to account for these mismatches, the following hypothesis is required:

Assumption 4. *The scheduling variable exhibits a bounded variation rate, this is, considering* $\delta\rho(k+1) = (\rho(k+1) - \rho(k))$, $\forall k \geq 0$, *it is implied that* $\delta\rho(k) \in \delta\mathcal{P} := [\underline{\delta\rho}, \overline{\delta\rho}]$.

Remark 28. *Considering the SA suspension application, we can ensure that, in practice, the Assumption 4 holds, since* $\rho(k) = f_c \tanh(A_\rho x(k))$ *is always bounded for all* $k \geq 0$, *due to the fact that* z_{def} *is bounded by construction. Thereof,* $\delta\rho(k+1) = (f_c \tanh(A_\rho x(k+1)) - f_c \tanh(A_\rho x(k)))$ *is also inherently bounded.*

Then, from the viewpoint of each sampling k, the minimal and maximal possible scheduling trajectories are given, respectively, by:

$$\underline{P_k} = \begin{bmatrix} \rho(k) & \rho(k) + \underline{\delta\rho} & \dots & \rho(k) + (N_p - 1)\underline{\delta\rho} \end{bmatrix}^T , \tag{4.23}$$

$$\overline{P_k} = \begin{bmatrix} \rho(k) & \rho(k) + \overline{\delta\rho} & \dots & \rho(k) + (N_p - 1)\overline{\delta\rho} \end{bmatrix}^T . \tag{4.24}$$

Figure 4.4 illustrates how these minimal and maximal scheduling trajectories bound the real one. Therefore, it is implied that $P_k, \hat{P}_k \in [\underline{P_k}, \overline{P_k}]$. Here, $\hat{P}_k$ gives the frozen-guess estimate, i.e. $\hat{P}_k = [\rho(k), \dots, \rho(k)]^T$. In the likes of Eq. (3.8), which gives the full-horizon state prediction vector in terms of the scheduling trajectory P_k, we can obtain the following representation for the j-th state prediction:

$$x(k+j|k) = \overbrace{\prod_{n=0}^{j-1} A(\rho(k+n))}^{\mathcal{A}_j(P_k)} x(k) + \sum_{m=1-k}^{j-k} \underbrace{\mathcal{B}_{(m,q)}(P_k) u(j-m)}_{\mathcal{B}_j(P_k)U_k} , \tag{4.25}$$

$$\mathcal{B}_{(m,q)}(P_k) = \left(\prod_{q=k+1}^{m-1} A(\rho(q)) \right) B(\rho(j-m)) . \tag{4.26}$$

Note that, here, $\coprod$ stands for the left-side matrix product. With a slight abuse of notation, we use $\mathcal{A}_j(\hat{P}_k) - \mathcal{A}_j(P_k) = \mathcal{A}_j(\hat{P}_k - P_k)$ and $\mathcal{B}_j(\hat{P}_k) - \mathcal{B}_j(P_k) = \mathcal{B}_j(\hat{P}_k - P_k)$. Furthermore,

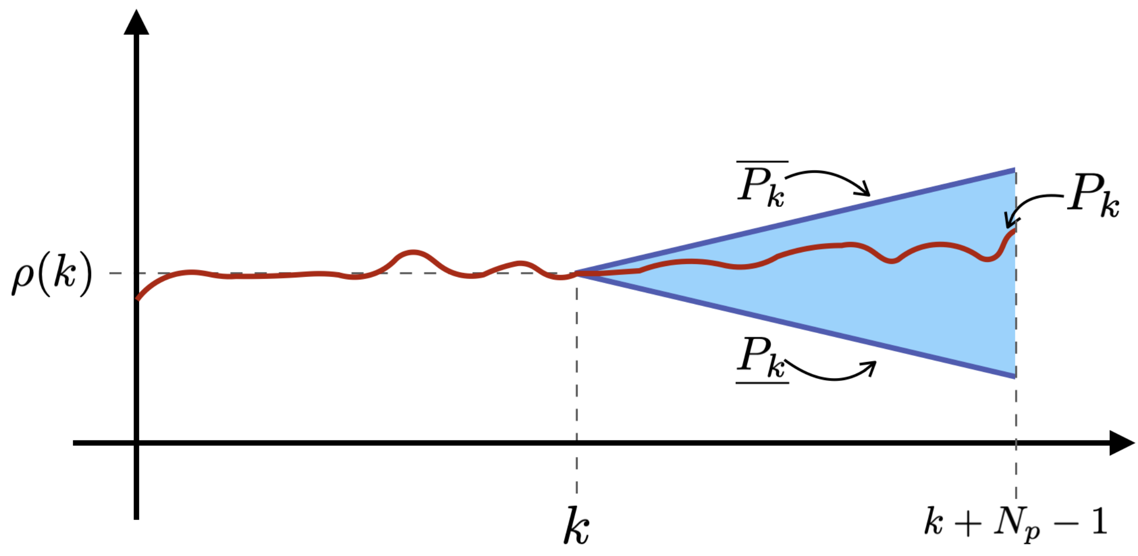


Figure 4.4: Minimal and maximal scheduling trajectories.

with regard to Eq. (3.8), it follows that:

$$\begin{cases} \mathcal{A}(P_k) = \begin{bmatrix} \mathcal{A}_1(P_k) \\ \vdots \\ \mathcal{A}_{N_p-1}(P_k) \end{bmatrix}, \\ \mathcal{B}(P_k) = \begin{bmatrix} \mathcal{B}_1(P_k) \\ \vdots \\ \mathcal{B}_{N_p-1}(P_k) \end{bmatrix} = \begin{bmatrix} \mathcal{B}_{(1,1)}(P_k) & \dots & \dots \\ \vdots & \ddots & \vdots \\ \mathcal{B}_{(N_p-1,1)}(P_k) & \dots & \mathcal{B}_{(N_p-1,N_p-1)}(P_k) \end{bmatrix}. \end{cases} \tag{4.27}$$

Since the frozen-based scheduling trajectory estimate is always limited within the bounds given by $\underline{P_k}$ and $\overline{P_k}$, the model-process mismatches that arise by using a frozen model are also bounded and can be directly computed. Accordingly, consider $\mu(k+j)$ as the mismatch related to the j-th state prediction, computed as the difference between the real state value $x(k+j)$, which is a function of P_k, and the predicted state $x(k+j|k)$, which is a function of the frozen scheduling guess $\hat{P}_k$, as gives:

$$\begin{aligned} \mu(k+j) &= x(k+j|k) - x(k+j) \\ &= \left(\mathcal{A}_j(\hat{P}_k - P_k)\right) x_k + \left(\mathcal{B}_j(\hat{P}_k - P_k)\right) U_k \,. \end{aligned} \tag{4.28}$$

At each sampling instant k, the model-process mismatch for the next j steps $\mu(k+j) = x(k+j|k) - x(k+j)$ depends on the initial state value $x(k)$ and the control sequence U_k within these j steps. Anyhow, this uncertainty propagation variable is inherently bounded, which yields:

$$\begin{aligned} \underline{\mu}_j &\leq \mu(k+j) \leq \overline{\mu}_j \,, \\ \underline{\mu}_j &= \mathcal{A}_j(\hat{P}_k - \underline{P}_k)x_k + \mathcal{B}_j(\hat{P}_k - \underline{P}_k)\underline{U} \,, \\ \overline{\mu}_j &= \mathcal{A}_j(\hat{P}_k - \overline{P}_k)x_k + \mathcal{B}_j(\hat{P}_k - \overline{P}_k)\overline{U} \,, \end{aligned} \tag{4.29}$$

where $\overline{U}$ and $\underline{U}$ represent, respectively, a N_p-sized stacked vector of the maximal and minimal control input values, respectively.

Synthetically, these mismatches are bounded by the saturation conditions implied by the input constraints and a sequence of minimal or maximal scheduling parameter variations. Remark that $\mu(k+j)$ increases along with the prediction horizon N_p, departing from $\mu(k) = 0, \forall k \geq 0$. This fact 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 diminish over time.

4.4 State-feedback procedure

Next, we detail the elements of the proposed gain-scheduled qLPV MPC algorithm for passenger comfort enhancement is SA suspension systems. For performance satisfaction, the design of this MPC integrates tools that enable the uncertainty propagation variable $\mu(k+j)$ (that grows along the horizon) to be tolerated in closed-loop. Essentially, we adapt the MPC constraints and cost in such a way that, even though sub-optimal results are obtain, the algorithm will remain recursively feasible and stabilise the system. Specifically, we make use of robust set invariant[4] and thus ensure that the MPC cost J_k is a quadratic Lyapunov function for the closed-loop dynamics, and thus decreases over time. The referred tool is presented in details in [Limon et al. 2018; Köhler, Müller, and Allgöwer 2018; Santos et al. 2019; Cunha and Santos 2021]. Seeking objectivity, the proofs are abbreviated; complete counterparts are available in [Morato, Normey-Rico, and Sename 2020c].

Assumption 5. *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 stabilising state-feedback control gain. Consider that the discrete-time model from Eq.* (4.8) *is controllable. 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 $\rho_k \in \mathcal{P}$.*

Proposition 2. *Consider Assumption 5 holds. Then, 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 J_k, to optimise and enhance the comfort of onboard passengers, with respect to nominal situations (those with a passive damper).*

[4]We stress that robust constraint satisfaction and robust performances in the context of LPV MPC are addressed in thorough details in Chapter 6, where we propose a robust controller for tracking with disturbance propagation using zonotopes.

Proof. Indeed, using 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 minimise the primary control objective $\ell(\cdot,\cdot)$ (stage cost along the control horizon). In order to do so, the time-domain index given in Eq. (4.13) is embedded to the stage cost $\ell(\cdot,\cdot)$ by the means of adequate weights Q and R. Thus, adapting the discrete-time formulation from Eq. (4.14) to the horizon-long prediction yields:

$$J_{\text{comfort}}^{N_p} = \sum_{j=0}^{N_p} T_s \ddot{z}_s^2(k+j|k)\,. \tag{4.30}$$

which is re-written as follows:

$$\sum_{j=0}^{N_p} \ddot{z}_s^2(k+j|k)T_s = \sum_{j=0}^{N_p} \|x(k+j|k)\|_Q^2 + \|u(k+j-1|k)\|_R^2\,. \tag{4.31}$$

From Eq. (4.8), we obtain[5]:

$$\begin{aligned} \ddot{z}_s^2(k+j|k)T_s &= (C\{1,:\}x(k+j-1|k) \\ &+ D_1\{1,:\}(\rho(k+j-1|k))u(k+j-1|k))^2\, T_s\,. \end{aligned} \tag{4.32}$$

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

$$Q = \left(C\{1,:\}\right)^T T_s \left(C\{1,:\}\right)\,, \tag{4.33}$$
$$R = \left(D\{1,:\}(\rho_k)\right)^T T_s \left(D\{1,:\}(\rho_k)\right)\,, \tag{4.34}$$

the MPC policy with stage cost $\ell(\cdot,\cdot)$ will act to minimise $\ddot{z}_s^2(k)$ and, thus, enhance comfort performances. □

Assumption 6. *Consider there exists an admissible robust control invariant set* $\mathbf{X}_f \subset \mathcal{X} \subset \mathbb{R}^{n_x}$ *such that for all* $x(k) \in \mathbf{X}_f$*, we obtain* $u(k) = \kappa x(k) \in \mathcal{U}$ *and* $x(k+1) = (A + B_1(\rho_k)\kappa)x(k)) + \mu(k) \in \mathbf{X}_f$, $\forall \mu(k) \in [\underline{\mu}_{N_p-1}, \overline{\mu}_{N_p-1}], \rho \in \mathcal{P}$.

Remark 29. *Assumption 6 implies in the existence of a control invariant set which is* ***robust*** *against the* ***worst-case*** *uncertainty propagation values. Note that the bounds* $\underline{\mu}_{N_p-1}$ *and* $\overline{\mu}_{N_p-1}$ *grow with* N_p*, and thus the solution can be quite conservative. When such robust terminal set is coupled to the MPC optimisation procedure, it becomes implied that the corresponding closed-loop domain of attraction tolerates the uncertainty that arises with the biased prediction model, which also gives holds on (quadratic) stability and recursive feasibility of the optmisation. The computation of such set is detailed in [Morato, Normey-Rico, and Sename 2020c], along with the theoretical certificates for the mentioned properties. This development is suppressed herein, for brevity. We note that in Chapter 5, a zonotopic-based solution for robust constraints satisfaction is developed in depth.*

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

In sum, we provided a state-feedback MPC design procedure for the control of SA suspension systems, aiming to enhance passenger comfort performances. This procedure converts the nonlinear optimisation problem for the original qLPV model into a gain-scheduled algorithm wit QP complexity. To make sure the simplifications of using a scheduling trajectory guess do not compromise the control performances, a robust control invariant set is used such that recursive feasibility is maintained. 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 disturbances $w(k)$ are known for the future N_p steps. 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 3.

Algorithm 3 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_p steps;
2. Compute the LTI model that approximates the process along the horizon, based on the scheduling evolution guess $\hat{P}_k$ for $j = 1, \dots, N_p$;
3. Compute the robust terminal $\mathbf{X}_f$ that satisfies Assumption 6;
4. Loop from $k = 0$:
   (a) Measure (or estimate) the system states $x(k)$;
   (b) Compute scheduling variable $\rho(k)$ and corresponding frozen scheduling trajectory vector $\hat{P}_k$;
   (c) Solve the the QP in Eq. (4.16);
   (d) From the solution $U_k^\star$, take the first entry $u^\star(k|k) = \kappa x(k)$ and apply it to the process.

4.5 State-feedback control results

In this Section, numerical simulation results are presented in order to illustrate the performances of a SA suspension system regulated under the proposed qLPV MPC algorithm. The simulation is performed with a realistic, validated, full vehicle nonlinear model of the experimental testbed of a vehicle equipped with four SA dampers. Recall that the control input for the SA suspension system is the PWM signal, which varies the damping coefficient of the electro-rheological dampers by changing the electric field applied over them, which varies the amount of force that is delivered. Furthermore, the primary control objective $\ell(\cdot)$ is taken in order to minimise chassis accelerations, to ensure that a smoother ride is provided and the comfort of the passengers is enhanced. The indexes provided in Section 4.3.3 are here used in order to evaluate the enhancement provided by the proposed scheme.

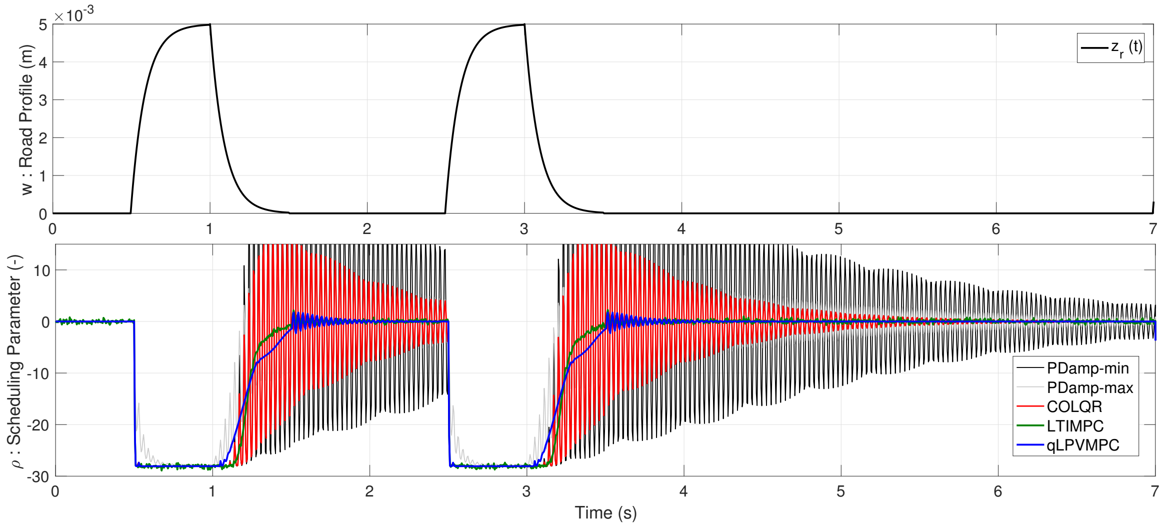


Figure 4.5: Road profile and scheduling parameters.

In the sequel, we compare the proposed method to other optimal suspension control methodologies: the proposed method is denoted "qLPVMPC"; "COLQR" denotes a clipped optimal LQR, based on a linearised (LTI) model of the process, using the same synthesis weights Q, R and P [Morato, Sename, and Dugard 2019a]; "LTIMPC" denotes an optimal control method based on an LTI[6] model [Morato et al. 2018a], using the same stage cost and weights; finally, the results obtained with a purely passive damper strategy are marked as "PDamp-min" and "PDamp-max" (tuned using fixed damping coefficients, respectively given by $c(\cdot) = \underline{c}$ and $c(\cdot) = \overline{c}$).

Following the tuning rules presented in [Morato et al. 2018a], we choose a prediction horizon N_p of 10 samples. In order to evaluate 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 are constructed offline.

The following results consider the SA suspension at the front-left corner of the vehicle. We stress that similar results were obtained for the other three corners (and thus not presented herein). 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 ($t' = 0.5\,\text{s}$) a sequence of two 5 mm bumps on all its wheels, exciting the bounce motion, which must be counteracted by the suspension controller. This simulation scenario comprises 7 s. Figure 4.5 shows these bumps and the scheduling parameters along the simulation.

Figure 4.6 depicts the evolution of both controlled outputs ($\ddot{z}_s, \ddot{z}_{us}$) along time and, accordingly, the delivered damping force (as well as the dissipativity constraints $\mathcal{D}$). From this Figure, the comfort performances can be evaluated. Smaller/smoother vertical acceleration

[6] In fact, [Morato et al. 2018a] consideres an LPV model for simulation, which is pre-filtered an becomes LTI from the control viewpoint, at each sampling instant. The method provides no theoretical guarantees on recursive feasibility and stability.

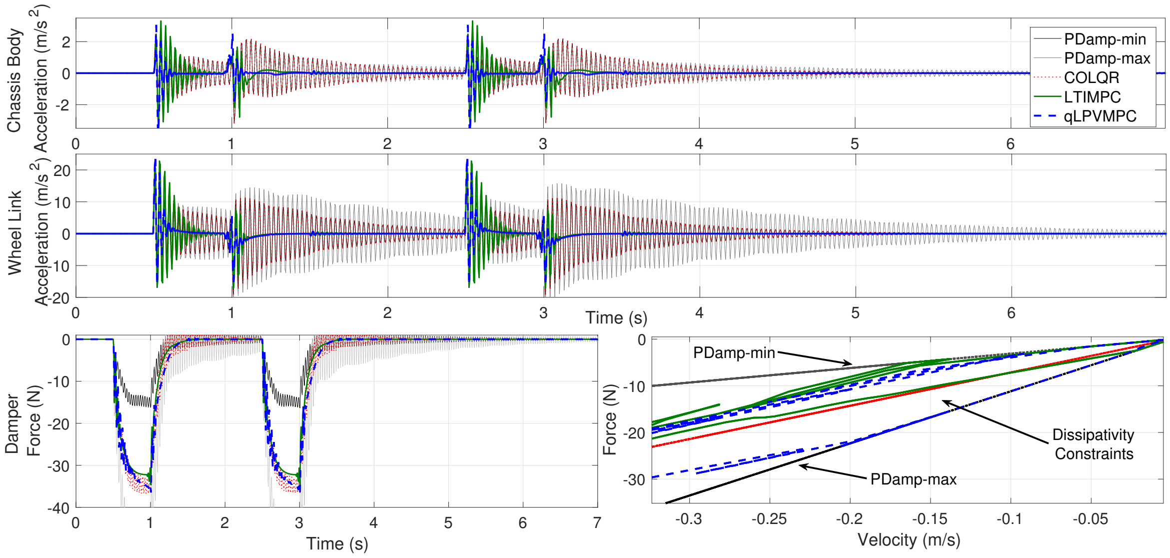


Figure 4.6: Controlled outputs and control signal.

variables mean better comfort for the onboard passengers. Recall that the proposed MPC is tuned in order to minimised these variables. Therefore, it becomes evident that the proposed predictive controller yields smoother results than the other strategies, further minimising the control objective $\ell(\cdot)$. The LTIMPC method also achieved good results, but not as smooth as the ones with the proposed scheme (due to the over-approximation of the model by pre-filtering and "hiding" of the nonlinearity). Due to saturation effects, the LQR strategy achieves results almost equivalent to those with a passive damper (as in open-loop conditions).

In order to quantitatively assess these results, the time index J^t_{comfort} is computed through a normalised root-mean-square (RMS) function of the acceleration variables; note that smaller RMS values denote better performances. Table 4.2 shows the RMS indexes obtained for both $\ddot{z}_s(t)$ and $\ddot{z}_{us}(t)$ variables, and the enhancements achieved with each control method with respect to the passive condition. The advantage of the proposed method resides in upgrading the comfort of passengers in over 14 %, with respect to a passive damper condition, in terms of chassis body acceleration. Recall that stiff damping (PDamp-max) provides better comfort, related to $\ddot{z}_s$, while smooth (PDamp-min) damping provides better roll safety, related to $\ddot{z}_{us}$. In terms of wheel link acceleration, the performances are enhanced in over 45 %. Both upgrades[7] are quite significant.

Furthermore, in order to demonstrate the benefits of the proposed method in frequency terms, the index J^f_{comfort} is provided. This index is computed with respect to the chassis acceleration curves $\ddot{z}_s$ filtered by ISO 2631-1 human vibration sensitivity weightings [International Organization for Standardization 2016]. This fifth-order filter $W_f(s)$, as given in [Zuo and Nayfeh 2003], selects the frequency-range of the signal that concerns human comfort under whole-body vibration (motion sickness, etc). Figure 4.7 shows the FFT results of the

[7]The performance index J^t_{comfort} is computed as detailed in Sec. 4.4. The upgrade percentages are given with respect to the least performant strategy.

Table 4.2: Performance enhancement.

RMS	Method	Value	Enhancement
Regulation	**PDamp-min**	0.4650	0 %
	COLQR	0.4646	0.08 %
$\ddot{z}_s(t) \to 0$	**PDamp-max**	0.4572	1.67 %
	LTIMPC	0.4116	11.48 %
	qLPVMPC	0.3984	14.32 %
Regulation	**PDamp-max**	4.826	0 %
	PDamp-min	3.008	37.67 %
$\ddot{z}_{us}(t) \to 0$	**COLQR**	3.005	37.73 %
	LTIMPC	2.742	43.15 %
	qLPVMPC	2.611	45.89 %

ISO-filtered $\ddot{z}_s(t)$ curves under the $0 - 20$ Hz frequency range. Maximal damping is expected to yield better comfort performances, as discussed by [Poussot-Vassal et al. 2012]. Note that 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 22.56 % smaller than the one with the passive (stiff) damper, which demonstrates furthermore the enhancement provided to the passengers. Note that the enhancement with the other methods (COLQR, LTIMPC) are smaller that the one with the proposed method. Respectively, they yielded 4.76 and 20.03 % peak reduction.

Remark 30. *The considered system is inherently complex and enhancing passenger comfort is a difficult task for SA suspensions, since the damping force can only be applied in the same direction as the deflection velocity movement, i.e. there is a limited working space for the task, regarding F_d, as illustrates Figure 4.2. Even though the provided enhancements are not so expressive in numerical terms, their conversion provided to the comfort of passengers is strong. Note, furthermore, that the perfromance enhancements provided with the proposed method, in both time and frequency domain, are quite relevant considering that these results are derived from the application of the algorithm to a scaled vehicle model. Note that the recent real-time LTIMPC method from [Morato et al. 2018a] had already been demonstrated to yield much better comfort and roll performances (over 50%) than other available techniques in the literature. The avantages with the proposed methods, with respect to to [Morato et al. 2018a], is that it can provide better performances and, at the same time, theoretically ensure recursive feasibility and stability, which was not ensured by the prior.*

We stress that the proposed method with its sets computations can be performed using standard SDP optimisation tools. In fact, the control calculation step (Step 4(c) of Algorithm 3) allows to achieve reasonably reduced numerical effort (elapsed within 3.5 ms $< T_s$, in average). 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.

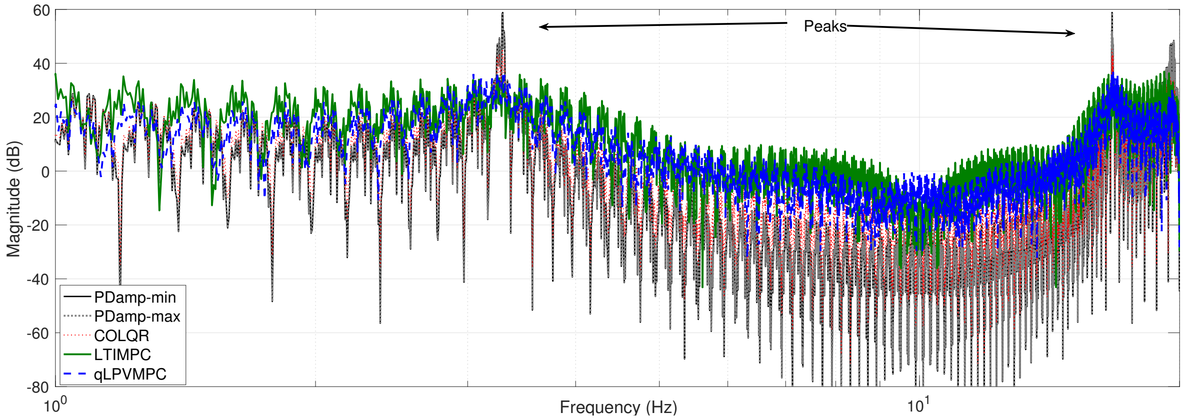


Figure 4.7: FFT: J^f_{comfort}.

4.6 State-feedback MPC: Some remarks

Before presenting the IO technique, we highlight some final remarks on the proposed state-feedback algorithm:

- The proposed MPC represents a novel scheme 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 embeds 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. A terminal robust positive invariant control set is used in the context of MPC so that it is able to maintain (quadratic) stability and recursive feasibility, despite model-process mismatches (in the regulation case). These properties are analytically demonstrated. The optimisation 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 SA suspension system via realistic simulation, achieving good results compared to existing control optimal control methods.
- The main limitation of the method resides in accounting for the frozen estimate, and the complexity of computing the set-based ingredients. Also, the method uses state variables, which are typically not available in practical situations.

4.7 Why use output-feedback?

As elaborated and debated along the Introduction of this thesis, we can concretely affirm that Model Predictive Control is nowadays a widely used technique, with practical relevance and theoretical interest. MPC schemes using state-space process models have had considerable research focus over the last years, with many results registered. As shown in the prequel, MPC based on SS models can be elaborated to satisfy rigorous performance requirements. Moreover, under bland assumptions, these algorithms enable optimal closed-loop performance.

Yet with great theoretical value, the standard SS MPC design requires the availability of state measurements in real time. In turn, state variables are often difficult to measure or estimate with precision. Moreover, state estimation schemes can significantly deteriorate closed-loop performances of MPC in the presence on disturbances and constraints. Due to these issues, SS realisations fall short of industrial expectations, which are seldom anchored in input-output (IO) process descriptions. Since we are concerned in this work with the exploitation of MPC algorithms for LPV systems, we also highlight that, nowadays, powerful LPV IO identification tools exist for a great variety of applications and system classes, e.g. [Bachnas et al. 2014].

For the reasons given, we focus henceforth on IO MPC schemes for LPV systems. Obtaining reliable SS descriptions of an LPV IO model is numerically very tough[8], since the dynamic-dependency problem hinders such translation, as argued in [Tóth, Abbas, and Werner 2011]. Therefore, it is desirable to design MPC schemes directly using IO LPV models, without the need for any IO-SS conversion. As of today, there are only a few papers which propose MPC schemes for LPV systems described in the IO form: [Abbas et al. 2015], [Abbas et al. 2016], and [Abbas et al. 2018]. Although these works provide closed-loop stability assurances, they assume that the future scheduling behaviour is known, which is false for the vast majority of applications, or resort to the worst-case solution, robustifying the MPC by considering the scheduling variables as bounded uncertainties, which often leads to excessive conservativeness.

Motivated by the fact that recent works have proposed (numerically-cheap) recursive linear schemes that are able to extrapolate the future scheduling trajectories with accurateness, e.g. [Morato, Normey-Rico, and Sename 2019; Morato, Normey-Rico, and Sename 2021b], and the results revisited in Chapter 3, approach (iv). Next, we formulate a novel MPC scheme for IO LPV systems using such parameter extrapolation laws.

4.8 Input-Output LPV setup

In this Section, we provide the preliminaries of using an input-output representation in the context of LPV systems, the corresponding fictive state-space description used for stability analysis, and the translation of the scheduling parameter extrapolation scheme to this context.

[8]This issue is exploited in details in recent key papers on the topic, e.g. [Tóth, Abbas, and Werner 2011; Tóth et al. 2011].

4.8.1 Preliminaries: IO LPV model

We consider the following discrete-time multi-input multi-output IO LPV system:

$$\left(I_{n_y} + \sum_{i=1}^{n_a} a_i(\rho(k))z^{-i}\right) y(k) \quad = \quad \sum_{j=1}^{n_b} b_j(\rho(k))z^{-j}u(k)\,, \tag{4.35}$$

where $u(k) := \delta(k) + u(k-1) \in \mathbb{R}^{n_u}$ is vector of control inputs, being $\delta(k)$ the corresponding control increments, $y(k) \in \mathbb{R}^{n_y}$ is the process outputs vector, z^{-1} is the one-sample backward shift operator, $n_a, n_b \geq 0$, while $a_i \in \mathbb{R}^{n_y \times n_y}$ and $b_i \in \mathbb{R}^{n_y \times n_u}$ are coefficient functions. The incremental control representation implies in an implicit integral action of the controller that defines the signal $\delta(k)$.

We also consider that the system in Eq. (4.35) is, in fact, quasi-LPV, being scheduled by the vector of output-dependent time-varying parameter $\rho(k) = f_\rho(y(k-1)) \in \mathcal{P} \subseteq \mathbb{R}^{n_\rho}$, being bounded and measured online (known by definition). The nonlinear scheduling map $f_\rho(y)$ is algebraic, class $\mathcal{C}^1$ for all y; $\mathcal{P} := \{\rho \in \mathbb{R}^{n_\rho} \colon \underline{\rho}_j \leq \rho_j \leq \overline{\rho}_j,\ j \in \mathbb{N}_{[1,n_\rho]}\}$ defines a compact, convex scheduling set. By definition, the future scheduling parameters $\rho(k+j),\ \forall j \in \mathbb{N}_{[1,\infty]}$ are unknown at time instant k.

Remark 31. *We recall that many nonlinear systems can be embedded to a qLPV realisations in the IO form such as gives Eq.* (4.35)*. For such, a coherent differential inclusion that maps* $\rho(k) = f_\rho(y(k-1))$ *should be generated. More details on this topic have been presented in Chapter 2; experimental examples are available in [Hoffmann and Werner 2014].*

4.8.2 Impulse response

The IO realisation from Eq. (4.35) has an equivalent infinite impulse response form:

$$y(k) \quad = \quad \sum_{i=0}^{+\infty} h_i(\rho(k), \ldots, \rho(k-i))u(k-i)\,, \tag{4.36}$$

where $h_i(\cdot) \in \mathbb{R}^{n_y \times n_u}$ are known as the *Markov* coefficients of the LPV system. These coefficients can be computed recursively as follows, where $h_i(k) := h_i(\rho(k), \ldots, \rho(k-i))$:

$$h_i(k) \quad = \quad \begin{cases} b_i(\rho(k) - \sum_{j=1}^{\min(i,n_a)} a_j(\rho(k))h_{i-j}(k-j), & i \leq n_b, \\ -\sum_{j=1}^{\min(i,n_a)} a_j(\rho(k))h_{i-j}(k-j), & \text{else.} \end{cases}$$

4.8.3 SS representation for analyses

Consider the corresponding non-minimal (fictive) SS realisation of Eq. (4.35), used for stability analysis:

$$\begin{aligned} x(k+1) &= A(\rho(k))x(k) + B(\rho(k))\delta(k)\,, \\ y(k) &= C(\rho(k))x(k) + D(\rho(k))\delta(k)\,, \end{aligned} \tag{4.37}$$

where $x(k) = [y(k-1)^T\,, \ldots\,, y(k-n_a)^T\,, u(k-1)^T\,, \ldots\,, u(k-n_b)^T] \in \mathbb{R}^{n_x}$ defines the state vector with $n_x = n_a n_y + n_b n_u$. It is important to note that in this SS representation, all states variables are known (current and past values of the inputs and outputs). Matrices $\left[\begin{array}{c|c} A(\cdot) & B(\cdot) \\ \hline C(\cdot) & D(\cdot) \end{array}\right]$ are respectively given by:

$$\left[\begin{array}{ccccccc|c} -a_1(\cdot) & \ldots & -a_{n_a}(\cdot) & (b_0(\cdot)+b_1(\cdot)) & \ldots & b_{n_b}(\cdot) & & b_0 \\ I_{n_y} & \ldots & 0 & 0 & \ldots & 0 & & 0 \\ & \vdots & & & \vdots & & & \vdots \\ 0 & \ldots & 0 & I_{n_u} & \ldots & 0 & & I_{n_u} \\ 0 & \ldots & 0 & I_{n_u} & \ldots & 0 & & 0 \\ & \vdots & & & \vdots & & & \vdots \\ \hline -a_1(\cdot) & \ldots & -a_{n_a}(\cdot) & (b_0(\cdot)+b_1(\cdot)) & \ldots & b_{n_b}(\cdot) & & b_0(\cdot) \end{array}\right],$$

As shown in Section 4.9, these matrices are used to compute stability-related tools for the MPC, in an offline synthesis step. During the implementation, the MPC requires only an IO prediction equation (no state measurements!), as detailed next.

4.8.4 Full-horizon IO predictor

The IO qLPV model from Eq. (4.35) is used by an MPC algorithm and, thus, a full-horizon prediction equation is required to compute control law at each sample. For such, consider that the scheduling trajectory $P_k = \text{col}\{\rho(k+j|k)^T\}^T, \forall j \in \mathbb{N}_{[0,N_p-1]}$ is at hand, being N_p the prediction horizon. Thus, it follows that:

$$Y_k = H(P_k)\Delta(k|k) + \Theta(P_k)x(k)\,, \tag{4.38}$$

where $Y_k \in \mathbb{R}^{n_y N_p}$ collects the output predictions $y(k+j|k), \forall j \in \mathbb{N}_{[0,N_p-1]}$ in a column-stacked vector and $\Delta(k|k) \in \mathbb{R}^{n_u N_p}$ comprises the future inputs $\delta(k+j|k), \forall j \in \mathbb{N}_{[0,N_p-1]}$ in another column-stacked vector.

In MPC theory, $H(P_k)\Delta(k|k)$ is usually named the "forced response" of the system, giving the dynamics implied by the control effort, where $\Theta(P_k)x(k)$ is called the "free response", which gives the behaviour in the absence of future control inputs. The prediction matrices are:

$$\left\{\begin{array}{rcl} H(P(k|k)) & = & \left[\begin{array}{ccc} h_0(k) & \ldots & 0 \\ h_0(k+1)+h_1(k+1) & \ldots & 0 \\ \vdots & \ddots & \vdots \\ \sum_0^{N_p-1} h_i(k+N_p-1) & \ldots & h_0(k+N_p-1) \end{array}\right], \\ \Theta(P(k|k)) & = & \left[\begin{array}{ccc} \tilde{\theta}(k+1)^T & \ldots & \tilde{\theta}(k+N_p)^T \end{array}\right]^T. \end{array}\right. \tag{4.39}$$

The elements of $\Theta(P_k)$ are found using $\theta(k+j) = -\sum_{i=1}^{\min(j,n_a)} a_i(\rho(k+j))\theta(k+j-1) + \overrightarrow{I^j}\breve{\theta}(k+j)\,, \forall j \in \mathbb{N}_{[1,N_p]}$, with $\breve{\theta}(k+j) = [-a_1(\rho(k+j)) \ldots - a_{n_a}(\rho(k+j))b_1(\rho(k+$

$j)) \dots b_{n_b}(\rho(k+j))]^T$ and $\overrightarrow{I^j} = \text{diag}\{\overrightarrow{I^j_{n_a}}, \overrightarrow{I^j_{n_b}}\}$. Finally, each $\tilde{\theta}(k+j)$ is given by $\theta(k+j)$ with its (n_a+1)-th element added to a correction bias $\sum_{i=1}^{j} h_i(k+j)$.

4.8.5 Process constraints

We consider that the qLPV system in Eq. (4.35) is subject to hard compact polyhedral constraints on outputs and inputs, which define an admissible operation. Specifically, we use $y(k) \in \mathcal{Y}$ and $u(k) \in \mathcal{U}, \forall k \geq 0$, where[9]:

$$\begin{cases} \mathcal{Y} & := & \{y \in \mathbb{R}^{n_y} : |y_j| \leq \overline{y}_j,\ \forall j \in \mathbb{N}_{[1,n_y]}\}\,, \\ \mathcal{U} & := & \{u \in \mathbb{R}^{n_u} : |u_j| \leq \overline{u}_j,\ \forall j \in \mathbb{N}_{[1,n_u]}\}\,. \end{cases} \tag{4.40}$$

Due to the explicit integral description of the control input, the following constraint is also used: $\delta(k) \in \mathcal{D} := \{\delta \in \mathbb{R}^{n_u} : |\delta_j| \leq \overline{\delta}_j, \forall j \in \mathbb{N}_{[1,n_u]}\}\, \forall k \geq 0$. Furthermore, we assume that the scheduling proxy is locally Lipschitz whenever these constraints are satisfied, i.e.:

Assumption 7. *The nonlinear scheduling parameter map $f_\rho : \mathcal{Y} \to \mathcal{P}$ agrees to a local Lipschitz condition around any arbitrary point $y \in \mathcal{Y}$, i.e. $\|f_\rho(y) - f_\rho(\hat{y})\| \leq \gamma_\rho \|(y - \hat{y})\|\,, \forall\, y, \hat{y} \in \mathcal{Y}$, where the smallest constant γ_ρ is known as the Lipschitz constant for $f_\rho(\cdot)$.*

4.8.6 Scheduling parameter extrapolation

The concept of MPC is based on spanning a prediction of the process variables along a future horizon window. For such, we use the IO qLPV prediction Eq. (4.38). As previously stated, the future scheduling sequence P_k is required. In previous works on the studied topic, i.e. [Abbas et al. 2015; Abbas et al. 2018], the scheduling trajectory P_k was treated as a bounded uncertainty variable (frozen-guess approach, as done in the state-feedback formulation), or assumed to be known (which is false, and thus performance certificates are lost). For this, we benefit from the extrapolation framework presented in Chapter 3 (approach (iv)), which is based on a first-order Taylor expansion of the scheduling proxy $f_\rho(y(k-1))$. As previously demonstrated, the main advantage of this extrapolation procedure is that it guarantees a convergent guess with a small, bounded residual error, while only resorting to linear operators. We henceforth adapt the formulation for the IO case.

Denote $\delta y(k) = y(k-1) - y(k-2)$ as the incremental output deviation. By definition, δy is bounded to a compact and convex box-type set $\delta\mathcal{Y} := \{\delta y \in \mathbb{R}^{n_y} : |\delta y_j| \leq \overline{\delta y}_j, \forall j \in \mathbb{N}_{[1,n_y]}\}$. Recall that the method is as follows: consider that the static scheduling map $f_\rho(y)$ can be approximated by $f_\rho(y) = f_\rho(y)|_{\breve{y}} + \left.\frac{\partial f_\rho(y)}{\partial y}\right|_{\breve{y}} (y - \breve{y}) + \xi_\rho$, being $\breve{y}$ the expansion point and ξ_ρ a residual which inherits the discrepancy between the real static map and its approximate. Since $f_\rho(y)$ is assumed class $\mathcal{C}^1$, it is direct that the partial derivatives $\left.\frac{\partial f_\rho(y)}{\partial y}\right|_{\breve{y}}$ are ultimately

[9] Here, $\overline{y}_j$ and $\overline{u}_j$ define box-type upper bounds for the corresponding j-th output and input, respectively.

bounded for all $\breve{y} \in \mathcal{Y}$. From this development, we obtain:

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

Note that $\rho(k|k) = \rho(k)$ and $\delta y(k)$ are known variables at each instant k, whereas $f_\rho^\partial(k)$ can be numerically evaluated. In practice, $f_\rho^\partial(k+j)$ is unknown for $j \in \mathbb{N}_{[1,N_p-2]}$, but it can be replaced by $f_\rho^\partial(k+j) = f_\rho^\partial(k)\,, \forall\, j \in \mathbb{N}_{[1,N_p-2]}$ where $f_\rho^\partial(k)$ denotes the partial derivative evaluated at instant k (refer to the discussion in Chapter 3 and in [Morato, Normey-Rico, and Sename 2023d]). By doing so, it is implied that $\rho(k+j|k) \approx \rho(k+j-1|k)+f_\rho^\partial(k)\delta y(k+j-1|k)$. Therefore, the estimate for the future scheduling variables can be written as the sum of the estimate from the previous sample corrected with an adjustment term $f_\rho^\partial(k)\delta y(k+j-1|k)$. Accordingly, we can write the vector-wise IO qLPV scheduling trajectory extrapolation in a recursive fashion:

$$\hat{P}_k = \hat{P}^\star_{k-1} + f_\rho^\partial(k)\delta\hat{Y}_k\,, \tag{4.41}$$

where the sequence of output increments is given by $\delta\hat{Y}_k = \mathrm{col}\{\delta y(k+j|k)^T\}^T, \forall j \in \mathbb{N}_{[0,N_p-2]}$.

Lemma 4.8.1. *Assume that $f_\rho(\cdot)$ is class $\mathcal{C}^1$ and that $f_\rho^\partial(k)$ is ultimately bounded. Then, the recursive extrapolation algorithm in Eq.* (4.41) *converges if the closed-loop is stable.*

Proof. This proof is reduced for brevity, full details are given in Chapter 3. The proof is herein briefly sketched for the IO case: consider that the residual terms $\xi_\rho(\cdot|k)$ should turn null. Thus, use $\lim_{k\to\infty} y(k) = y_e$ holds (stability) and take $\xi_\rho(k+j|k) = f_\rho(y(k+j|k)) - f_\rho(y(k+j-1|k)) - f_\rho^\partial(k)\delta y(k+j|k)$. Due to the stabilisation[10], it directly follows that $\lim_{k\to\infty} f_\rho(y(k+j|k)) = \lim_{k\to\infty} f_\rho(y_e)$ and $\lim_{k\to\infty} \delta y(k|k) \to 0$, which implies in $\lim_{k\to\infty} \xi_\rho(\cdot|k) = -\lim_{k\to\infty} f_\rho^\partial \delta y(\cdot|k) \to 0$. This concludes the proof. □

Lemma 4.8.2. *The residual error is ultimately bounded:* $\|\xi_\rho(\cdot|k)\| \leq \overbrace{\left(\gamma_\rho + \overline{f_\rho^\partial}\right)\overline{\delta y}}^{\xi_\rho^{bound}}, \forall k \geq 0.$

Proof. The residual term in the extrapolation law is given by $\xi_\rho(k+j+1|k) = f_\rho(y(k+j+1|k)) - f_\rho(y(k+j|k)) - f_\rho^\partial(k)\delta y(k+j|k)$. Using a triangular inequality, we obtain $\|\xi_\rho(k+j+1|k)\| \leq \|f_\rho(y(k+j+1|k)) - f_\rho(y(k+j|k))\| + \|f_\rho^\partial(k)\delta y(k+j|k)\|$. Finally, due to Assumption 7, we we state that $\|\xi_\rho(k+j+1|k)\| \leq \gamma_\rho\|\delta y(k+j)|k\| + \|f_\rho^\partial(k)\delta y(k+j)\|$. Since $f_\rho^\partial(k)$ is ultimately bounded, it follows that: $\|\xi_\rho(k+j|k)\| \leq \left(\gamma_\rho + \overline{f_\rho^\partial}\right)\overline{\delta y}$. This concludes the proof. □

[10]Note that the MPC will be verified to quadratically stabilise the LPV process for any scheduling variable value (Sec. 4.10). Even if the future scheduling prediction is biased, the closed-loop remains stable.

Remark 32. *The extrapolation procedure presented in this section is completely detailed in Chapter 3 and in published paper [Morato, Normey-Rico, and Sename 2023d]. The Hessian-based estimates from [Hanema, Tóth, and Lazar 2021] and the iteratively refined estimated from [Cisneros and Werner 2017a] are comparable methods to the one considered herein. Nevertheless, the main advantage of the proposed scheme is that converging estimates with reduced error bounds, formulated by the means of linear laws. Thereof, the corresponding model-process uncertainties that are derived when applying MPC became very reduced, enabling less conservative control synthesis.*

4.8.7 The MPC design

As previously stated, we consider an MPC design for LPV processes with IO descriptions. Accordingly, we use the recursive extrapolation procedure in Eq. (4.41) to generate the scheduling sequence estimate $\hat{P}_k$, used to generate the future output predictions Y_k via Eq. (4.38).

Remark 33. *The residual errors $\xi_\rho(k+j|k)$ are bounded (Lemma 4.8.2) and, in practice (Sec. 4.10), we observe that these bounds are very small (thus, negligible) and that the residuals dissipate within a few samples. Thus, for the MPC formulation, we assume $\xi_\rho(k+j|k)$ as null (i.e. that the scheduling trajectory estimates via Eq.* (4.41) *are accurate). In Chapters 5 and 6, we provide theoretical assessments for a robustified MPCs (and in [Morato 2023], for the IO case).*

Consider the following cost function:

$$J_k \;=\; J\left(x(k), r(k), P_k\right) \;=\; \sum_{i=0}^{N_p-1} \ell\left(e(k+i|k), \delta(k+i|k)\right) + V(x(k+N_p|k) - x_r)\,,$$

where the stage cost $\ell(e,\delta) := \|e\|_Q + \|v\|_R$ is given with respect to the output tracking error $e(k) = r(k) - y(k)$. The terminal cost $V(x - x_r)$ requires x, which is the fictive state description given through Eq. (4.37), and x_r, the envisioned state reference target, defined in terms of $r(k)$; this cost is used to penalise the distance of non-minimal states at the end of the prediction horizon to a given target, whereas $\ell(\cdot,\cdot)$ weights the performance along the horizon. Consider Q and R as positive definite weighting matrices, used to imply the envisioned trade-off between control effort and output reference tracking.

Taking J_k into account, the proposed MPC resides in solving the following optimisation problem at each instant k:

$$\begin{aligned}
\min_{\Delta(k|k)} \quad & \textstyle\sum_{i=0}^{N_p-1} \ell\left(e(k+i|k), \delta(k+i|k)\right) + V(x(k+N_p|k) - x_r)\,, \\
\text{s.t.}: \quad & Y_k = H(P_k)\Delta(k|k) + \Theta(P_k)x(k), \\
& y(k+j|k) \in \mathcal{Y}\,, \\
& u(k+j|k) \in \mathcal{U}\,, \\
& \delta(k+j|k) \in \mathcal{D}\,, \\
& (x(k+N_p|k) - x_r) \in \mathbf{X}_f\,,
\end{aligned} \tag{4.42}$$

where $\mathbf{X}_f$ is a terminal invariant set for the controlled IO LPV system. Let $J^\star(x(0), r(k), P_k)$ be the optimal solution of the optimisation in Eq. (4.42), from which $\Delta^\star(k|k)$ is the optimal sequence of control inputs. Then, the MPC law at time instant k considers in apply the first entry of $\Delta(k|k)^\star$, i.e. $\delta^\star(k|k)$, to the process using $u(k) = u(k-1) + \delta^\star(k|k)$.

4.9 Stability and recursive feasibility in the IO form

We proceed by how the MPC terminal ingredients $V(\cdot)$ and $\mathbf{X}_f$ should be generated in order to render an asymptotically stable closed-loop, as well as a recursively feasible optimisation. First, let us define the admissible steady-state targets for the IO LPV system in Eq. (4.35). We say that $r \in \mathcal{R}$ is an **admissible**[11] output reference target if and only if there exists a control input $u_r \in \mathcal{U}$ such that $\left(I_{n_y} + \sum_{i=1}^{n_a} a_i(f_\rho(r))\right) r = \left(\sum_{i=1}^{n_b} b_i(f_\rho(r))\right) u_r$. Accordingly, we introduce the set of all admissible state-input targets (x_r, u_r), that is: $\mathcal{X}_r := \{x_r \in \mathbb{R}^{n_x} \,|\, x_r = [1_{n_a} r, 1_{n_b} u_r], \forall (r, u_r) \in \mathcal{R} \times \mathcal{U}\}$. Due to the box-type constraints over y and u, $\mathcal{X}_r$ can be equivalently described as $\{x \in \mathbb{R}^{n_x} \,|\, |x_j| \leq \overline{x}_j, \forall j \in \mathbb{N}_{[1,n_x]}\}$.

Assumption 8. *The reference set-point is admissible, i.e. $r(k) \in \mathcal{R}$.*

Assumption 9. *The scheduling trajectory P_k is known, with each entry $\rho(k+j|k) \in \mathcal{P}, \forall j \in \mathbb{N}_{[0,N_p-1]}$.*

Assumption 10. *The scheduling parameters take a constant value ρ_r in steady-state, i.e. $\rho = \rho_r$, $\forall\, (x - x_r) \in \mathbf{X}_f$.*

Assumption 11. *The stage cost function is positive definite and uniformly continuous such that $\ell(e,\delta) \geq \alpha_\ell(\|e\|)$ and $|\ell(e_1,\delta_1) - \ell(e_2,\delta_2)| \leq \lambda_e(\|e_1 - e_2\|) + \lambda_\delta(\|\delta_1 - \delta_2\|)$, where α_ℓ, λ_e and λ_δ are $\mathcal{K}$-functions. It is implied that $\ell(0,0) = 0$.*

Set of Assumptions 2.

1. *There exists an admissible terminal feedback law $u(k-1) + \kappa_t(x(k), r(k), P_k) \in \mathcal{U}$.*
2. *The terminal set $\mathbf{X}_f$ is closed, contains the origin, and represents admissible positive invariant set.*
3. *The terminal cost $V(x - x_r)$ is continuous and positive for all $x - x_r \in \mathcal{X}_r$. Moreover $V(\cdot)$ represents a control Lyapunov function for the unconstrained LPV system in Eq. (4.35), meaning that there exist constants $b > 0$ and $\sigma > 1$ such that $V(x - x_r) \leq b|x - x_r|^\sigma$. It is implied, thus, that $V(A(f_\rho(r))(x - x_r) + B(\rho)\kappa_t(\cdot)) - V(x - x_r) \leq \ell(y - r, \kappa_t(\cdot))$, for all $r \in \mathcal{R}$and $V(x_1 - x_r) - V(x_2 - x_r) \leq \alpha_r(|x_1 - x_2|)$ (i.e. $V(\cdot)$ is a $\mathcal{K}$ function).*

Next, we provide a sufficient condition to compute the terminal elements that enable the error dynamics $e(k) = r(k) - y(k)$ to converge to the origin, which conversely means that

[11] Here, it is implied that the reference signal $r(k)$ is reachable. For further generality, artificial reference variables could be included, as in [Limon et al. 2018].

offset-free reference tracking is ensured, while maintaining recursive feasibility of the MPC optimisation.

In order to satisfy the conditions required by Theorem 1, we choose the following quadratic terminal cost function $V(x - x_r) = (x - x_r)^T P(x - x_r)$, where $P = P^T$ is a positive definite weight. Accordingly, the terminal set $\mathbf{X}_f$ is taken as a sub-level set of the terminal cost, i.e.: $\mathbf{X}_f := \{x \in \mathbb{R}^{n_x} \,|\, x^T P x \leq 1\}$. By definition, $\mathbf{X}_f$ is an ellipsoidal set constraint, which should be positively invariant for the terminal feedback $\kappa_t(\cdot)$. Thus, the following Theorem gives a numerically solvable sufficient solution that can be used to generate the terminal ingredients.

Theorem 2. *Terminal ingredients*
Consider that Assumptions 8-2 hold. Then, conditions (C1)-(C5) from Theorem 1 are satisfied if there exist a symmetric positive definite matrix $P \in \mathbb{R}^{n_x \times n_x}$ and a rectangular matrix $W \in \mathbb{R}^{n_u \times n_x}$ such that $Y = P^{-1} > 0$, $W = KY$ and that LMIs (4.43)-(4.45) *hold under the minimisation of* $\log det\{Y\}$ *for all $\rho \in \mathcal{P}$. The terminal feedback is then given by $\kappa_t(\cdot) = K(x - x_r)$.*

$$\left[\begin{array}{cc|cc} Y & \star & \star & \star \\ (A(\rho)Y + B(\rho)W) & Y & \star & \star \\ Y & 0 & \tilde{Q}^{-1} & \star \\ W & 0 & 0 & R^{-1} \end{array}\right] \geq 0 \,, \tag{4.43}$$

$$\left[\begin{array}{c|c} \overline{x}_j^2 & I_{\{j\}}Y \\ \hline I_{\{j\}}Y^T & Y \end{array}\right] \geq 0, \, j \in \mathbb{N}_{[1,n_x]} \,, \tag{4.44}$$

$$\left[\begin{array}{c|c} \overline{\delta}_i^2 & I_{\{i\}}W \\ \hline \star & Y \end{array}\right] \geq 0, \, i \in \mathbb{N}_{[1,n_u]} \,. \tag{4.45}$$

Proof. This proof is reduced for brevity, a full demonstrative counter-part is available in [Morato 2023]. Consider $r(k)$ is piece-wise constant, thus satisfying Assumption 8. Note that $\ell(e, v) = \|e\|_Q + \|v\|_R$ is equivalent to $\|x - x_r\|_{\tilde{Q}} + \|v\|_R$, using $\tilde{Q} = \text{diag}\{QI_{n_y n_a}, 0_{n_u n_b}\}$. Consider $r(k)$ is piece-wise constant, thus satisfying Assumption 8. Consider P_k is known due to the extrapolation procedure from Eq. (4.41). Since $\rho(k) = f_\rho(y(k-1))$, it follows that $\rho(k) = \rho_r$ in steady-state, iff $\lim_{k\to+\infty} y(k-1) = r$. Then, by applying a Schur complement to LMI (4.43), we obtain condition (C3), which suffices for (C2). By definition, an ellipsoid ensures (C1). (C4) and (C5) are respectively satisfied by applying Schur complements to LMI (4.44) and (4.45). In turn, the terminal feedback $\kappa_t(\cdot) = K(x - x_r)$ ensures that the SS representation in Eq. (4.37) is asymptotically stable, which conversely implies in the output tracking of $r(k)$. This concludes the proof. □

Remark 34. *The terminal ingredients provided through Theo. 2 ensure recursive feasibility and asymptotic stability of the tracking error trajectories (refer to Propositions 3-4). Note that LMI* (4.43) *is infinite-dimensional, having to hold $\forall \rho \in \mathcal{P}$. In practice, the solution can be found by enforcing the inequalities over a sufficiently dense grid of points (ρ) along $\mathcal{P}$, then verifying it for a denser grid.*

Remark 35. *An alternative formulation to drop the parameter-dependency of the LMI* (4.43) *is to use the full-block $\mathcal{S}$-procedure over an LFT of the LPV system [Scherer 2001], which results in similar LMIs to those presented in [Abbas et al. 2018].*

Proposition 3 (Recursive feasibility)**.** *Let there exist a solution Y to Theorem 2. Then, given any $x \in \mathbf{X}_f$, $x_r \in \mathcal{X}_r$, $r \in \mathcal{R}$ and $\delta = \kappa_t(x, r, \cdot)$, we have $x^+ = Ax + BK(x - x_r) + BKu_r \in \mathbf{X}_f$. Consider an optimal sequence $\Delta^\star = (\delta_0^\star, \delta_1^\star, \ldots, \delta_{N_p-1}^\star)$ and an reference target r. Then $\hat{\delta}^c = (\delta_1^\star, \ldots, \delta_{N_p-1}^\star, \kappa_t(x, r, \cdot))$ define feasible (candidate) solution of the MPC problem in Eq.* (4.42) *for any $r \in \mathcal{R}$, which means that the optimisation is recursively feasible.*

Proof. Let Assumptions 8-2 hold. Consider there exists a solution Y to Theorem 2. Then, from conditions (C1), (C2), (C4), and (C5) from Theorem 1, we can infer that the generated control signal provides recursively feasible solutions to the MPC optimisation in (4.42). Take $\kappa(\cdot) = u_r + K(x - x_r)$ and $\rho_r = f_\rho(r)$. Then, the if the initial condition $x(0)$ generates a feasible input sequence $\Delta^\star$, all future iterations of the optimisation will also be feasible: the generated control law control is admissible (condition (C4)) and all state variables $x \in \mathbf{X}_f$ generate successor state variables x^+ which are also inside $\mathbf{X}_f$ (condition (C2)), which contains the origin (condition (C1), terminal condition for $(x - x_r)$, and $\mathbf{X}_f$ being sub-set of $\mathcal{X}_r$ (condition (C5), which ensures that x, x^+ are admissible). This concludes the proof. □

Proposition 4 (Asymptotic stability)**.** *Let there exist a solution Y to Theorem 2. Then, the LPV system* (4.35) *in closed loop with the MPC input $\kappa(\cdot)$ exhibits an asymptotic stable tracking error dynamics. That is, for any feasible initial condition x_0 and constant set-point $r \in \mathcal{R}$, it is implied that $\|x(k) - x_r\| \leq \beta(\|x(0)\|, k)$, where β is a $\mathcal{K}$-function which passes through the origin.*

Proof. Let there be a terminal stage cost $V(\cdot)$ such that Assumption 11 holds. Let Assumption 2 also hold and Proposition 3 be verified. Note that since $\ell(\cdot, \cdot)$ is a quadratic stage cost, α_ℓ, λ_e and λ_u indeed exists. Consider there exists a solution Y to Theorem 2. Then, the SS closed-loop is stable due to (C3) of Theorem 1, which conversely ensures that $V(x(k) - x_r) - V(x(k-1) - x_r)) \leq -\|y(k) - r\|_Q$. Thus, we obtain $\|y(k) - r\|_Q \leq \beta(\|(x(0) - x_r)\|, k)$. Since $Q > 0$ (positive definite), asymptotic stability is established. This concludes the proof. □

4.10 Output-feedback control results

In this Section, we demonstrate the effectiveness of the proposed method for controlling qLPV systems in the IO form, without any state measure or estimation procedure involved. For such, we consider an adapted version of the unstable second order system from [Abbas et al. 2015], with additional nonlinearities; the dynamics are:

$$
\begin{aligned}
y(k) &= \overbrace{(0.2 - 0.7\frac{y(k-1)^2}{10})}^{-a_1(\rho(k))} y(k-1) + \overbrace{(-0.7 - 0.4\frac{y(k-1)^2}{10})}^{-a_2(\rho(k))} y(k-2) \\
&+ \overbrace{(3.4 - 1.2\frac{y(k-1)^2}{10})}^{b_1(\rho(k))} u(k-1) + \overbrace{(1.6 - 2.8\frac{y(k-1)^2}{10})}^{b_2(\rho(k))} u(k-2)\,,
\end{aligned} \tag{4.46}
$$

where the qLPV scheduling parameter is endogenous and given by $\rho(k) = \frac{y(k-1)^2}{10}$.

This system should be controlled such that the output trajectory $y(k)$ tracks a given piece-wise reference signal, whilst the following constraints are respected: $u(k) \in \mathcal{U} := [-1, 1]$, $\delta(k) \in \mathcal{D} := [-0.04, 0.04]$, $y(k) \in \mathcal{Y} := [0, 5], \forall k \geq 0$. We note that these constraints imply that $\rho(k) \in \mathcal{P} := [0\,, 2.5], \forall k \geq 0$. Furthermore, we assume the system operates under a sampling rate of 40 Hz. The proposed MPC is tuned with a prediction horizon of $N_p = 10$ discrete-time steps, thus previewing a window of 250 ms.

First of all, we show that there indeed exists a solution to Theo. 2, which enables quadratic terminal ingredients computed using:

$$
\left\{
\begin{aligned}
P^{-1} = Y &= \begin{bmatrix} 15.92 & 1.28 & 1.68 & 2.51 \\ \star & 14.94 & 2.31 & 1.46 \\ \star & \star & 1 & 0.45 \\ \star & \star & \star & 1 \end{bmatrix}, \\
W &= \begin{bmatrix} 0.10 & 0.10 & 0.03 & 0.03 \end{bmatrix}.
\end{aligned}
\right. \tag{4.47}
$$

Thereof, we can concritely affirm that $\kappa_t(x - x_r) = K(x - x_r)$, with $K = WP$, $V(x - x_r) = (x - x_r)^T P(x - x_r)$, and $\mathbf{X}_f := \{x - x_r \in \mathbb{R}^{n_x} \,|\, (x - x_r)^T P(x - x_r) \leq 1\}$ are suitable terminal ingredients for the MPC optimisation in Eq. (4.42), ensuring recursive feasibility and asymptotic stability (and, in turn, offset-free reference tracking). We stress that these ingredients are generated offline, with respect to the tuning parameters $Q = 0.8I_{n_y}$ and $R = 0.2I_{n_y}$ and the known sets $\mathcal{U}$, $\mathcal{D}$, $\mathcal{Y}$, and $\mathcal{P}$.

Taking into account that stability is ensured by design (and, thus, Lemma 4.8.1 holds), we now demonstrate how the scheduling parameter extrapolation procedure operates. During the implementation, the recursive estimates are generated through the linear operator given in Eq. (4.41). As shows Figure 4.8, the generated estimates P_k are very accurate, thus passing to the MPC precise information regarding the future behaviour of ρ. In this Figure, we can also see that the estimation error ξ_ρ is indeed small (refer to Lemma 4.8.2) and converges rapidly, which means that it can be neglected for simplicity (treating this uncertainty robustly directly in the MPC synthesis step is a topic for future works).

Next, we show the obtained performances with the proposed method. Figure 4.9 shows the system trajectories being steered to the terminal set $\mathbf{X}_f$, as ensured by the MPC. As one can see, the tracking error dynamics are repeatedly steered to the origin (for each new

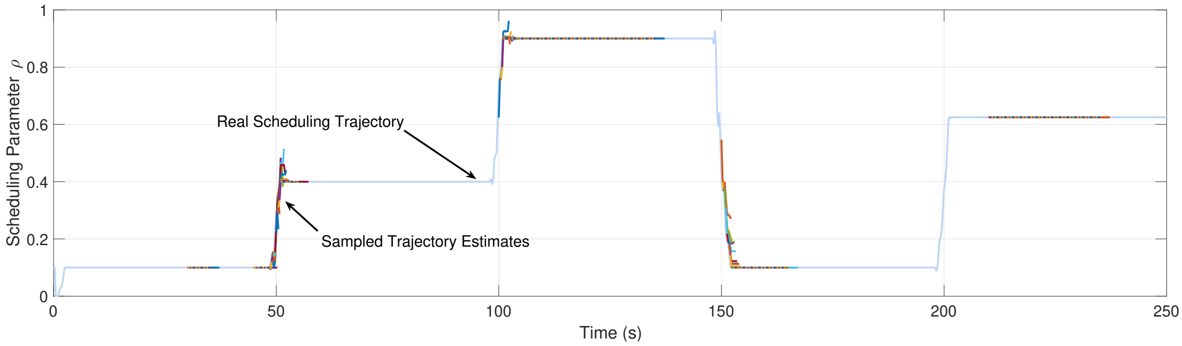


Figure 4.8: Scheduling trajectory and Taylor-based extrapolation estimated (at different samples).

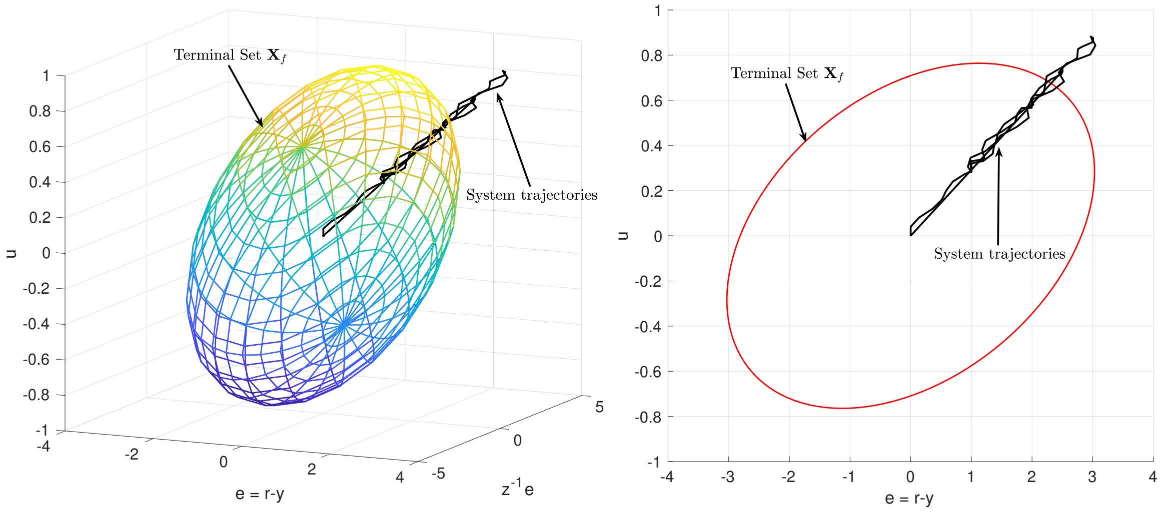


Figure 4.9: IO LPV MPC: Stability and terminal invariance.

reference goal). Figure 4.10 gives the control input trajectories (and the control increment), altogether with the piece-wise constant reference target signal $r(k)$ and the output. Evidently, the integral-embedded MPC ensures offset-free tracking, which is significant.

Quantitatively, we stress that the RMS index of the tracking error is of 0.28. The MPC requires, in average, 6.5 ms to solve the optimisation procedure, while the Taylor-based scheduling extrapolation takes only 0.07 ms, in average. This clearly indicates the relieved numeric burden of the proposed technique (6.57 ms, in average), which is ready for real-time embedded applications. We recall, once again, that no state measures or observers are necessary, making the proposed method coherent with industrial practices.

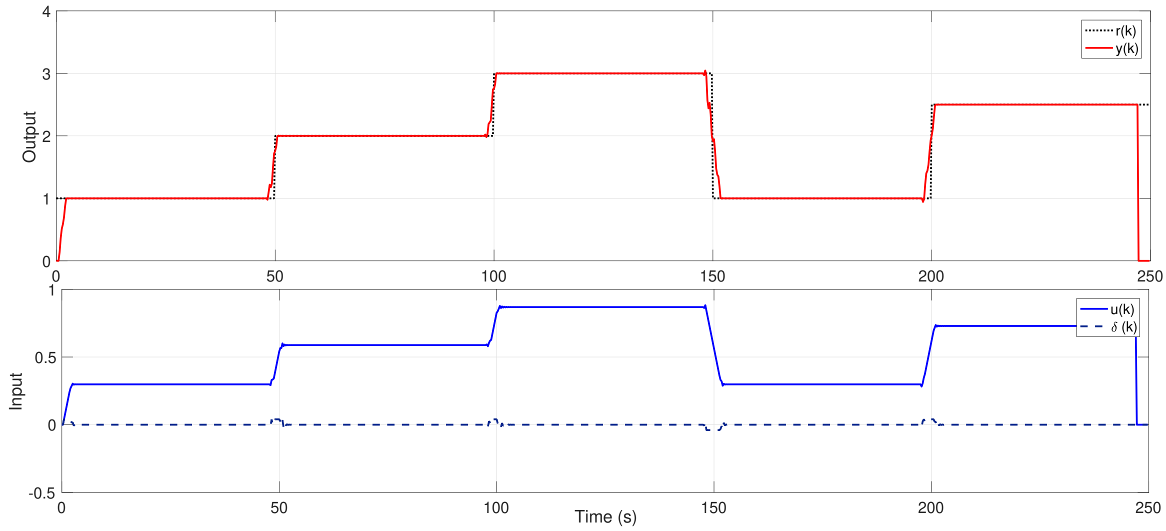


Figure 4.10: IO LPV MPC: System performances.

4.11 Some remarks on output-feedback

In the prequel, a novel MPC algorithm for IO LPV systems is proposed. The future LPV scheduling parameters are extrapolated using a recursive Taylor expansion law, which generates the MPC prediction matrices at each sampling period. Terminal ingredients are offered through an LMI-solvable remedy, which ensures Lyapunov properties of the closed-loop. The method is able to ensure asymptotic offset-free reference tracking, also thanks to an explicit integral action and to these optimisation ingredients. In order to demonstrate the effectiveness of the method, it is to a numeric benchmark system, exhibiting good performances. We highlight the key findings:

- The proposed IO MPC method possesses real-time capabilities, since its online implementation is only related to a linear operator (scheduling parameter extrapolation) and to the solution of a single QP. As shown in the examples, the control law can be generated in the range of milliseconds with standard solvers.

- The method is able to tackle the tracking control problem of highly nonlinear systems, as long as if an IO qLPV model can be generated. Moreover, the method does not require any additional reference tracking tool (such as the artificial reference variables).

- The offered terminal ingredients, which ensure recursive feasibility and asymptotic output stability, are enabled through quadratic LMI-solvable remedies and thus can be easily computed with standard solvers.

4.12 Overall remarks

In this Chapter, we provided two distinct gain-scheduled MPC solutions for LPV systems:

- The first is a state-feedback procedure based on frozen scheduling trajectories. The method included adapted, robustified state constraints in order to handle the uncertainty propagation due to the biased prediction model. The method is applied for the control of a semi-active suspension system, with success.
- The second is an input-output formulation, which does not require state observers or estimation. The method is fully formulated by the means of IO LPV models, with an implicit integral action that ensures tracking. The method also includes quadratic terminal ingredients in order to ensure asymptotical stability and recursive feasibility of the optimisation

Part III

Robust synthesis

Chapter 5

Reference tracking with zonotopes

In this Chapter, we present an NMPC for tracking for piece-wise constant reference signals. This scheme is computationally efficient due to the use of a qLPV-embedding realisation to describe the nonlinear dynamics. Accordingly, we benefit from the Taylor-based extrapolation procedure (Chapter 3, approach (iv)) in order to estimate the future behaviour of the scheduling parameters with bounded estimation error. At each sampling period, the optimisation problem uses linear predictions and thus exhibits relieved numerical toughness (comparable to a QP). Benefiting from artificial target variables, the method is also able to avoid feasibility losses due to large set-point variations. Robust constraint satisfaction, closed-loop stability, and recursive feasibility certificates are provided, thanks to uncertainty propagation zonotopes and parameter-dependent terminal ingredients. Finally, a benchmark example is used to illustrate the effectiveness of the method, which is compared to state-of-the-art techniques.

Remark 36. *The developments presented in this Chapter correspond to those presented in [Morato et al. 2021e] (regulation) and [Morato et al. 2023a] (tracking).*

5.1 Introduction

As thoroughly discussed in previous Chapters, we recall that the application of robust nonlinear MPC is not trivial and comes with increased numerical burden, which may be an impediment for real-time applications. The majority of stabilizing NMPC schemes ensures regulation of the closed-loop dynamics to a fixed target [Boccia, Grüne, and Worthmann 2014]. Accordingly, asymptotic stability and constraints satisfaction are usually guaranteed with terminal ingredients, which verify invariance conditions in the neighborhood of the operation target. This design method is not valid for set-point changes, since feasibility may be lost.

Therefore, there has been an increasing focus on “Tracking” NMPC schemes, considering time-varying set-points. Specifically, we highlight two of the main frameworks towards this matter, as debated in recent years: (*i*) artificial reference variables, as proposed in [Limon et al. 2018], which allow for less conservative terminal constraints and thus ensure feasibility is not lost for sudden set-point changes; and (*ii*) terminal equality constraints and optimised terminal sets, for the case of periodic reference signals, as developed in [Köhler, Müller, and Allgöwer 2020].

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

1. The use of online NPs, which are numerically expensive and not viable for time-critical applications[1];
2. No model uncertainties nor disturbances are considered, which should be included for any realistic application.

As done all along this thesis, we benefit from the LPV toolkit in order to enhance the theoretical framework of tracking NMPC schemes, under a (partially) linear setting. By doing so, we enable fast implementation of the resulting control scheme. With regard to the detailed context[2], we provide a novel Tracking NMPC algorithm based on nominal predictions generated via qLPV embedding. Following the final suggestions from [Köhler, Müller, and Allgöwer 2020], the proposed solution also incorporates robustness features against model uncertainties and additive load disturbances (with known bounds), using a constraint tightening framework [Chisci, Falugi, and Zappa 2003; Köhler, Müller, and Allgöwer 2018; Santos et al. 2019; Cunha and Santos 2021]. Based on the one-step-ahead disturbance propagation, we enforce the satisfaction of the performance requirements by binding the prediction error within zonotope extensions. Additionally, we benefit from the recent recursive extrapolation method from Chapter 3, approach (iv) [Morato, Normey-Rico, and Sename 2022b] in order to estimate the future trajectory of the qLPV scheduling parameters. Specifically, we exploit the boundedness of the prediction error generated by the extrapolation scheme.

The main novelties of this Chapter are summarised next:

1. The proposed Tracking NMPC is based on qLPV embeddings, which enable linear model predictions. The model is exploited by an extrapolation mechanism that provides the complete sequence of future scheduling parameters, at each sampling instant. Accordingly, we compute simple bounds on the prediction error from these qLPV realisations. Furthermore, we propose zonotopes that bound the corresponding uncertainty propagation[3], which are then used for robust constraints satisfaction.
2. We offer robust parameter-dependent terminal ingredients for the proposed NMPC. These tools ensure recursive feasibility of the optimisation procedure and stability of the tracking error dynamics, considering any set-point value within a predefined set. Furthermore, we propose an additional optimisation for the choice of the artificial reference variable, with relieved complexity.

[1] We stress that the application results provided in [Köhler, Müller, and Allgöwer 2020] solve such NPs only very fast, but relying on a solver-based solution (CaSaDi), which internally approximates the solution of the problem.

[2] Up to our best knowledge, NMPC algorithms based on qLPV embeddings have only been formalised for regulation purposes, refer to the survey [Morato, Normey-Rico, and Sename 2020a] and also to [Morato, Normey-Rico, and Sename 2019; Cisneros and Werner 2020; Morato, Normey-Rico, and Sename 2020b; Hanema, Tóth, and Lazar 2021].

[3] The proposed method also considers bounded process disturbances. The uncertainty propagation zonotopes offer a direct extension for such case.

3. Finally, using a benchmark example, we demonstrate that the numerical complexity of the proposed online algorithm is, on average, comparable to QP, being faster than the NP solutions from the literature.

5.2 Problem statement

The focus of this Chapter is the development of an MPC algorithm for the class of nonlinear systems that can be represented by qLPV-embeddings. For a start, consider the following discrete-time nonlinear system:

$$\begin{cases} x(k+1) &= f(x(k),u(k)) + w(k)\,, \\ y(k) &= h(x(k),u(k))\,, \end{cases} \tag{5.1}$$

where $x \in \mathbb{R}^{n_x}$ represent the states, $u \in \mathbb{R}^{n_u}$ the inputs, and $y \in \mathbb{R}^{n_y}$ the outputs. The additive disturbance $w \in \mathbb{R}^{n_x}$ is bounded to a compact set with the origin at its interior, in such a way that $w(k) \in \mathcal{W} \subseteq \mathbb{R}^{n_x} \implies \|w(k)\| \leq \overline{w}$. Throughout the sequel, it is implied that the states are measurable for all sampling instants $k \geq 0$. Consequentially, we design a state-feedback NMPC.

This system is said *admissibly operated* if the following hard constraints are satisfied: $(x(k),u(k)) \in \mathcal{Z} = \mathcal{X} \times \mathcal{U}$, where:

$$\mathcal{X} := \{x \in \mathbb{R}^{n_x} : H_x x \leq h_x\}\,, \text{ and } \mathcal{U} := \{u \in \mathbb{R}^{n_u} : H_u u \leq h_u\}\,. \tag{5.2}$$

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

Next, the nonlinear model from Eq. (5.1) is re-written under the following **exact** qLPV representation:

$$\begin{cases} x(k+1) &= A(\rho(k))x(k) + B(\rho(k))u(k) + w(k)\,, \\ y(k) &= C(\rho(k))x(k) + D(\rho(k))u(k)\,, \\ \rho(k) &= f_\rho(x(k)) \in \mathcal{P}\,. \end{cases} \tag{5.3}$$

This qLPV model is scheduled by the state-dependent time-varying parameter $\rho(k) = f_\rho(x(k)) \in \mathcal{P} \subseteq \mathbb{R}^{n_\rho}$, which is implied to be bounded[5] for all $k \geq 0$. We stress that this parameter is known by definition, since $x(k)$ is measured and $f_\rho(\cdot)$ is a **known** function. Yet, we stress that the future scheduling trajectory $\rho(k+j)$, $\forall j \in \mathbb{N}_{[1,\infty]}$ is unknown. We consider the scheduling set as compact and convex: $\mathcal{P} := \{\rho \in \mathbb{R}^{n_\rho} : \underline{\rho}_j \leq \rho_j \leq \overline{\rho}_j,\ j \in \mathbb{N}_{[1,n_\rho]}\}$.

Assumption 12. *The nonlinear scheduling map $f_\rho : \mathcal{X} \to \mathcal{P}$ is algebraic and class $\mathcal{C}^1$ for all $x \in \mathcal{X}$. Moreover, it agrees to a local Lipschitz condition around any arbitrary point $x \in \mathcal{X}$,*

[4] The outer box-type bounds $\overline{x}_j$ and $\overline{u}_j$ can be found from Eqs. (5.1)-(5.2) using linear programming.

[5] Note that the boundedness of the scheduling variable ρ is related to the boundedness of the states.

this is:

$$\|f_\rho(x) - f_\rho(\hat{x})\| \leq \gamma_\rho \|(x - \hat{x})\| \,, \forall\, x, \hat{x} \in \mathcal{X} \,, \tag{5.4}$$

where the smallest constant γ_ρ that satisfies Eq. (5.4) *is called its Lipschitz constant. Furthermore, the scheduling variables exhibit a bounded rate of variation over samples, this is: $\delta\rho(k+1) = (\rho(k+1) - \rho(k)) \in \delta\mathcal{P} \, \forall k \geq 0$, where:*

$$\delta\mathcal{P} := \left\{ \delta\rho_j \in \mathbb{R} : \underline{\delta\rho}_j \leq \delta\rho_j \leq \overline{\delta\rho}_j, \forall j \in \mathbb{N}_{[1,n_p]} \right\} . \tag{5.5}$$

Remark 37. *As explained in Chapter 2, the validity of the qLPV realisation requires that the satisfaction of a differential inclusion. With regard to the nonlinear model in Eq.* (5.1)*, we require that the nonlinear maps $f(x, u)$ and $h(x, u)$ to be re-written, respectively, as $f_1(x)x + f_2(x)u$ and $h_1(x)x + h_2(x)u$, with $f_1(x)$, $f_2(x)$, $h_1(x)$ and $h_2(x)$ being bounded and known for all $x \in \mathcal{X}$.*

Remark 38. *The requirement of Lipschitz continuity for $f_\rho(x)$ is often very possible, since the selection of this function is a design choice. The qLPV realisation from Eq.* (5.3) *only requires that a corresponding differential inclusion exists and that $f_\rho(x)$ is bounded for all states $x \in \mathcal{X}$. Note, anyhow, that the bounds on scheduling parameters' variations is **not** a design choice. These bounds, which describe the scheduling variation set $\delta\mathcal{P}$, naturally appear due to the discrete-time characteristic of the system. Note that $\delta\rho(k+1) = f_\rho(x(k+1)) - f_\rho(x(k))$ and, since $x(k+1), x(k) \in \mathcal{X}, \forall k \geq 0$, $\delta\rho(k+1) \in \delta\mathcal{P}$. Considering the nonlinear system to be ultimately bounded, we can obtain the scheduling variation bounds by minimising[6] $\underline{\delta\rho}$ and $-\overline{\delta\rho}$ such that $\underline{\delta\rho} \leq f_\rho(A(f_\rho(x))x + B(f_\rho(x))u + w) - f_\rho(x) \leq \overline{\delta\rho}$ with $(x, u) \in \mathcal{Z}$, $(A(f_\rho(x))x + B(f_\rho(x))u + w, u) \in \mathcal{Z}$ and $w \in \mathcal{W}$.*

5.2.1 Tracking objective

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

Following the lines of previous works on Tracking MPC, i.e. [Limon et al. 2018; Köhler, Müller, and Allgöwer 2019], we consider that there exists an unique combination of the states and inputs which ensures that $\lim_{k\to+\infty} y(k) = y_r$ (stabilisation at the set-point target). For such, the following additional hypothesis is required:

Assumption 13. *For all admissible steady-state targets $y_r \in \mathcal{Y}$, there exits an unique admissible steady-state state-input pair $z_r = (x_r^T\,,\, u_r^T)^T \in \mathcal{Z}$ such that the following inequality*

[6] The solution to this minimisation problem can be found either by interval arithmetic or optimisation.

holds:

$$\begin{bmatrix} x_r - (A(f_\rho(x_r))x_r - B(f_\rho(x_r))u_r) \\ C(f_\rho(x_r))x_r + D(f_\rho(x_r))u_r \end{bmatrix} = \begin{bmatrix} 0_{n_x} \\ y_r \end{bmatrix}. \tag{5.6}$$

From Assumption 13, we denote $\mathcal{Y}_T \subset \mathcal{Y}$ as the set of admissible steady-state tracking outputs, i.e. those generated by an admissible state-input pair $z_r \in \mathcal{Z}$ which satisfies Eq. (5.6). Specifically, we define this set as follows: $\mathcal{Y}_T := \{y \in \mathcal{Y} \,|\, y = C(f_\rho(x_r))x_r + D(f_\rho(x_r))u_r \,|\, \left((A(f_\rho(x_r))x_r + B(f_\rho(x_r))u_r)^T, u_r^T\right)^T + e \in \mathcal{Z}, \forall e \in \epsilon\mathcal{B}_\infty\}$. Note that, in this definition, ϵ is an arbitrarily small constant included so that the frontier of $\mathcal{Z}$ is excluded.

Following the lines of [Limon et al. 2018], we exploit the unique steady-state state-input pair to tracking output set-point correspondence in terms of locally Lipschitz continuous functions. That is, Assumption 13 implies that there exists locally Lipschitz continuous maps $g_x : \mathcal{Y} \to \mathcal{X}$ and $g_u : \mathcal{Y} \to \mathcal{U}$ such that $x_r = g_x(y_r)$ and $u_r = g_u(y_r)$ for all $y_r \in \mathcal{Y}_T$. As debated in [Limon et al. 2018], for these functions to exists, the Jacobian matrix of the left hand-side in Eq. (5.6) must be square and non-singular for all $z_r \in \mathcal{Z}$. Complementary, we also consider a composed locally Lipschitz[7] continuous function $g_\rho : \mathcal{Y} \to \mathcal{P}$ that maps the equilibrium scheduling parameter, i.e. $\rho_r = g_\rho(y_r) = f_\rho(g_x(y_r), g_u(y_r))$.

5.2.2 Control and disturbance propagation

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

$$u(k) \quad = \quad v(k) + K_\pi x(k)\,, \tag{5.7}$$

where $v(k)$ is a virtual input, determined by the MPC, and K_π a feedback gain, used to locally stabilise the process and attenuate the propagation of disturbances. Accordingly, from Eqs. (5.3)-(5.7), we obtain the following closed-loop dynamics:

$$x(k+1) \quad = \quad \overbrace{(A(\rho(k)) + B(\rho(k))K_\pi)}^{A_\pi(\rho(k))}\, x(k) + B(\rho(k))v(k) + w(k)\,. \tag{5.8}$$

Then, from Eqs. (5.2)-(5.8), we re-write the admissibility constraints $(x(k), u(k)) \in \mathcal{Z}$ in terms of the virtual input as follows: $(x(k), v(k)) \in \mathcal{Z}_\pi$, where:

$$\mathcal{Z}_\pi \quad := \quad \left\{ z \in \mathbb{R}^{n_x+n_u} \;:\; \begin{pmatrix} H_x & 0 \\ H_u K_\pi & H_u \end{pmatrix} z \leq \begin{pmatrix} h_x \\ h_u \end{pmatrix} \right\}. \tag{5.9}$$

Regarding the application of the Tracking MPC policy, we distinguish *nominal predictions* of the system dynamics from the *real system trajectories*. Note that the MPC has no knowledge

[7]Note that $g_\rho(\cdot)$ is locally Lipschitz continuous since it is a composition of locally Lipschitz continuous functions, due to Assumption 12.

of the disturbance variables $w(k+j)$, $\forall j \geq 0$, and also does not know the future scheduling trajectory $\rho(k+j)\,\forall j \geq 1$, which thus makes the nominal predictions differ from the real system trajectories. In terms of the scheduling trajectory, we assume that there exists an estimate for its behaviour:

Assumption 14. *Consider a prediction horizon of N_p samples. Then, at each sample k, there exists a scheduling trajectory guess $\hat{P}_k = \left[\begin{array}{ccc} \hat{\rho}^T(k|k) & \dots & \hat{\rho}^T(k+N_p-1|k) \end{array}\right]^T$ with bounded mismatch towards the real scheduling trajectory $P_k = \left[\begin{array}{ccc} \rho^T(k) & \dots & \rho^T(k+N_p-1) \end{array}\right]^T$. That is: $\|\xi_\rho(k+j|k)\| = \|\left(\rho(k+j) - \hat{\rho}(k+j|k)\right)\| \leq \xi_\rho^{bound} < +\infty$.*

Then, by expanding Eq. (5.8) forward, along the following N_p steps, the **real system trajectories** are given, from an initial condition $x(k) \in \mathbb{R}^{n_x}$, by:

$$X_k = \left[\begin{array}{c} x(k+1) \\ x(k+2) \\ \vdots \\ x(k+N_p) \end{array}\right] := \phi_{N_p}\left(x(k), V_k, W_k, P_k\right) , \tag{5.10}$$

where $V_k := \left[\begin{array}{ccc} v^T(k) & \dots & v^T(k+N_p-1) \end{array}\right]^T$ is the vector of future inputs and $W_k = \left[\begin{array}{ccc} w^T(k) & \dots & w^T(k+N_p-1) \end{array}\right]^T$ the vector of future disturbances.

From Assumption 14, the **nominal predictions** are then given, from an initial condition $x(k) \in \mathbb{R}^{n_x}$, by:

$$\hat{X}_k = \left[\begin{array}{c} x(k+1|k) \\ x(k+2|k) \\ \vdots \\ x(k+N_p|k) \end{array}\right] := \phi_{N_p}\left(x(k), V_k, \mathbf{0}, \hat{P}_k\right) . \tag{5.11}$$

Note that, in these predictions, the future disturbances are presumably nil, i.e. Eq. (5.10) is equivalent to using: $x(k+1|k) = A_\pi(\hat{\rho}(k|k))x(k) + B(\hat{\rho}(k))v(k)$, $x(k+2|k) = A_\pi(\hat{\rho}(k+1|k))x(k+1|k) + B(\hat{\rho}(k+1|k))v(k+1)$, and so forth. We stress that Eq. (5.11) is analogous to Eq. (3.8), yet including the closed-loop propagation (i.e. here $v(k)$ is the input).

Throughout the sequel, we use scheduling trajectory estimates $\hat{P}_k$ which verify Assumption 14 as provided by the scheme detailed in Chapter 3, approach (iv). This is, at each discrete-time sample k, these trajectories are estimated by an extrapolation method based on first-order Taylor expansions of the scheduling proxy $f_\rho(\cdot)$. This scheme has been thoroughly detailed in the prior, and thus we opt not to repeat the proofs and propositions of the boundedness condition and convergence property herein (refer to Lemma 3.6.2). We recall that, as long as the baseline hypothesis are satisfied, we obtain an estimation error $\xi_\rho(k+j|k)$ (upon each scheduling parameter estimate) which is ultimately bounded $\forall j \in \mathbb{N}_{[1,N_p-1]}$, $k \geq 0$ to the convex set $\mathcal{Q} := \{\xi_\rho \in \mathbb{R}^{n_\rho} \mid \|\xi_\rho\| \leq \left(\gamma_\rho + \overline{f_\rho^\partial}\right)\overline{\delta x}\}$ (refer to Lemma 4.8.2).

5.3 Disturbance propagation using zonotopes

Next, we detail how to conceive zonotopes that bound the disturbance propagation that arises between the nominal qLPV prediction model from Eq. (5.11) and the real system trajectories from Eq. (5.10). This topic is one of the contributions of this Chapter, as previously detailed.

Consider a prediction horizon of N_p steps and a group of generic compact sets $\mathcal{E}(j)$, $j \in \mathbb{N}_{[0,N_p-1]}$. These sets bound the difference between the scheduling parameter estimates generated at samples k and $k+1$, using the Taylor-based extrapolation approach detailed in the previous section (i.e. via Eq. (3.43)). That is, we consider that the scheduling parameter estimation error between samples $(\rho(k+j+1|k+1) - \rho(k+j+1|k))$ belongs to the compact set $\mathcal{E}(j)$, $\forall j \in \mathbb{N}_{[0,N_p-1]}$.

By leveraging from Lemma 4.8.2, we can rapidly conclude that $\| (\rho(k+j+1|k+1) - \rho(k+j+1|k)) \| \leq \|\xi_\rho(k+j+1|k+1)\| \leq \xi_\rho^{\text{bound}}$, i.e. $(\rho(k+j+1|k+1) - \rho(k+j+1|k)) \in \mathcal{Q}$. Seeking simplicity, we use $\mathcal{E}(j) = \mathcal{Q}, \forall j$. Usually, this is not at all conservative since these residuals have considerably small bounds in many applications (as shown in the validations results provided in Sec. 5.5). In any case, if one considers a known decay rate of these residuals $\xi_\rho(k+j|k)$ as the predictions span along the horizon $j \in \mathbb{N}_{[0,N_p-1]}$, j-decaying sets could replace $\mathcal{Q}$ over the prediction samples $j \in \mathbb{N}_{[1,N_p]}$.

Remark 39. *We recall, once again, that this estimation error boundedness property stands for a major advantage of the extrapolation method. In general, when the so-called frozen scheduling estimates are used (often done in practice, e.g. as seen in [Morato, Sename, and Dugard 2018; Alcalá, Puig, and Quevedo 2019]), the resulting scheduling estimation error is bounded as follows: $\|\xi_\rho(k+j|k)\| \leq (N_p-1)\overline{\delta\rho}$. Therefore, more conservative control laws would have be generated, since the corresponding uncertainty propagation is far larger (in such case, we would have to consider $\mathcal{E}(j) = \delta\mathcal{P}, \forall j$). We stress that the error obtained with the proposed extrapolation mechanism does not depend on the horizon size, but only on characteristics of the scheduling proxy and the bounds on the state deviations. Thus, conservatism of the corresponding disturbance propagation zonotopes is much smaller (as detailed in the sequel).*

In order to bound the deviance from the nominal state trajectories $x(k+j|k)$ and the real ones $x(k+j)$, we consider **one-step-ahead** propagation sets, which will be later used provide performance certificates for our proposed MPC (in Sec. 5.4):

Definition 5.1
One-step-ahead disturbance propagation sets $\mathcal{S}(j)$, $j \in \mathbb{N}_{[0,N_p]}$ are compact sets that satisfy the following conditions:

1. *The initial set $\mathcal{S}(0)$ bounds the load disturbances $w(k)$, i.e. $\mathcal{W} \subseteq \mathcal{S}(0)$;*
2. *Consider states $x(k+j|k), x(k+j) \in \mathbb{R}^{n_x}$ (nominal prediction, real value), control input $v(k+j) \in \mathbb{R}^{n_u}$, and scheduling parameters $\rho(k+j|k), \rho(k+j) \in \mathbb{R}^{n_p}$*

(extrapolated estimate, real value). Then, if following tightened bounds are satisfied: $(x(k+j|k), v(k+j)) \in \mathcal{Z}_\pi \ominus (\mathcal{S}(j-1) \times \{0\})$, $(x(k+j) - x(k+j|k)) \in \mathcal{S}(j-1)$, *and* $(\rho(k+j) - \rho(k+j|k)) \in \mathcal{E}(j-1)$, *it follows that:*

$$\begin{aligned} &\underbrace{(A_\pi(\rho(k+j))x(k+j) + B(\rho(k+j))v(k+j))}_{x(k+j+1)} \\ -&\underbrace{(A_\pi(\rho(k+j|k))x(k+j|k) + B(\rho(k+j|k))v(k+j))}_{x(k+j+1|k)} \;\in\; \mathcal{S}(j)\,, \forall j \in \mathbb{N}_{[1,N_p]}\,, \end{aligned} \tag{5.12}$$

i.e. the one-step-ahead deviation. from the nominal state prediction and the real state value is bounded.

In sum, Definition 5.1 implies that $x(k+j|k+1) \in x(k+j|k) \oplus \mathcal{S}(j-1)$, $\forall j \in \mathbb{N}_{[1,N_p+1]}$, for any admissible sequence of inputs and scheduling parameter predictions. These one-step-ahead disturbance propagation sets $\mathcal{S}(j)$, thus, bound the difference between the predictions made in k and $k+1$, and therefore can be used to guarantee recursive feasibility and constraint satisfaction of the MPC based on nominal predictions (Sec. 5.4).

Next, we detail an exact structure in order to compute such disturbance propagation sets. Specifically, in the following Theorem, we provide zonotope reachable sets that satisfy Definition 5.1, based on Lemma 5.3.1.

Lemma 5.3.1 (Zonotopic extensions, adapted from [Alamo, Bravo, and Camacho 2005; Cunha and Santos 2021])**.** *Consider a centered zonotope* $X = M\mathcal{B}_\infty^{n_g} \subseteq \mathbb{R}^m$ *and an interval matrix* $\mathbf{J} \in \mathbb{I}^{n \times m}$. *The columns of* $M \in \mathbb{R}^{m \times n_g}$ *are named the generators of the zonotope* X *Then, the zonotopic inclusion of the product of the zonotope and interval matrix is defined by:*

$$\diamond(\mathbf{J}X) \;:=\; mid(\mathbf{J})X \oplus T\mathcal{B}_\infty^n\,, \tag{5.13}$$

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

$$T_{ii} \;=\; \sum_{j=1}^{n_g}\sum_{k=1}^{m} rad(\mathbf{J})_{ik}|M_{kj}|, \quad \forall i \in \mathbb{N}_{[1,n]}\,. \tag{5.14}$$

Thus, it holds that $JX \subseteq \diamond(\mathbf{Z})$, *for all* $J \in \mathbf{J}$.

Proof. Follows directly from Theorem 3 of [Alamo, Bravo, and Camacho 2005], by taking $p = 0$ and $\mathbf{M} = \mathbf{J}M$. □

Theorem 3. *Consider two zonotopes* $Z \in \mathbb{R}^{n_x+n_u}$ *and* $\mathcal{S}_0 \in \mathbb{R}^{n_x}$, *an interval matrix* $\mathbf{A} \in \mathbb{I}^{n_x \times n_x}$, *and two sets of interval matrices* $\mathbf{\Delta}_A(j) \in \mathbb{I}^{n_x \times n_x}$ *and* $\mathbf{\Delta}_B(j) \in \mathbb{I}^{n_x \times n_u}$, $\forall j \in \mathbb{N}_{[0,N_p]}$. *Furthermore, consider that* $\mathcal{S}_0$ *bounds the load disturbances, i.e.* $\mathcal{W} \subseteq \mathcal{S}_0$, *and that* Z *contains the admissibility bounds of the nonlinear system in Eq.* (5.1), *i.e.* $\mathcal{Z}_\pi \subseteq Z$. *Also, consider that:*

1. $A_\pi(\rho) \in \mathbf{A}, \forall \rho \in \mathcal{P}$;
2. $A_\pi(\rho(k+j)) - A_\pi(\rho(k+j|k)) \in \mathbf{\Delta}_A(j), \forall \rho(k+j), \rho(k+j|k) \in \mathcal{P}$, *under* $(\rho(k+j) - \rho(k+j|k)) \in \mathcal{E}(j), \forall j \in \mathbb{N}_{[0,N_p-1]}$;
3. $B(\rho(k+j)) - B(\rho(k+j|k)) \in \mathbf{\Delta}_B(j), \forall \rho(k+j), \rho(k+j|k) \in \mathcal{P}$, *under* $(\rho(k+j) - \rho(k+j|k)) \in \mathcal{E}(j), \forall j \in \mathbb{N}_{[0,N_p-1]}$.

Then, for $\mathcal{V}(j) := \diamond\left(\left(\mathbf{\Delta}_A(j) \quad \mathbf{\Delta}_B(j)\right) Z\right)$*, the following set of zonotopes* $\mathcal{S}(j)$, $\forall j \in \mathbb{N}_{[0,N_p]}$*, satisfy Definition 5.1:*

$$\mathcal{S}(j) := \begin{cases} \mathcal{S}_0, & j = 0, \\ \mathcal{V}(j) \oplus \diamond(\mathbf{A}\mathcal{S}(j-1)), & j \in \mathbb{N}_{[1,N_p]}. \end{cases} \tag{5.15}$$

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

$$\begin{aligned} \Delta(k+j|k) &= (A_\pi(\rho(k+j)) - A_\pi(\rho(k+j|k)))x(k+j|k) && (5.16) \\ &+ A_\pi(\rho(k+j))(x(k+j) - x(k+j|k)) + (B(\rho(k+j)) - B(\rho(k+j|k)))v(k+j) \\ &\in \left(\mathbf{\Delta}_A(j) \quad \mathbf{\Delta}_B(j)\right) Z \oplus \mathbf{A}\mathcal{S}(j-1) \subseteq \mathcal{V}(j) \oplus \diamond(\mathbf{A}\mathcal{S}(j-1)) = \mathcal{S}(j). && (5.17) \end{aligned}$$

Therefore, the sets $\mathcal{S}(j)$ satisfy Definition 5.1, which concludes this proof. □

The main idea behind Theorem 3 is that is offers a direct and rather simple way on how to compute one-step-ahead disturbance propagation sets (i.e. satisfying Definition 5.1). In order to generate these zonotopes, one only needs the interval matrices $\mathbf{A}, \mathbf{\Delta}_A$, and $\mathbf{\Delta}_B$, which can be found using interval algebra. Then, by using the zonotopic extensions enabled by Lemma 5.3.1, the computation of each $\mathcal{S}(j)$ is direct from Eq. (5.15), using Minkowski set addition operators. The main interest behind Theorem 3 is that these zonotopes are numerically cheap to compute, and can be directly used in the design of robust MPC algorithms

The zonotopes $\mathcal{S}(j)$ obtained via Theorem 3 grow with regard to the disturbance propagation term $\Delta(k+j|k)$ from Eq. (5.16), which measures the one-step deviance from the real system trajectories and the nominal ones, which consider the estimated scheduling parameters from Eq. (3.43). Accordingly, these sets depend on the original admissibility bounds of the system $\mathcal{Z}_\pi$ and also on the estimation errors $\xi_\rho(k+j|k)$, which appears as a multiplicative uncertainty. Thanks to Lemma 4.8.2, we are able to replace these errors by their worst-case bounds $\xi_\rho^{\text{bound}} \geq \|\xi_\rho(k+j|k)\|$. Regarding the conservatism implied with such zonotopes, we note that the worst-case bounds ξ_ρ^{bound} from Lemma 4.8.2 are quite reduced, which thus make these sets not so large with respect to the original constraints $\mathcal{Z}$ (as shown in practice, in Sec. 5.5).

The use of zonotopes that bound disturbance propagation has been recurrently seen in recent robust MPC literature, e.g. [Santos et al. 2019; Cunha and Santos 2021], even for the LPV case, e.g. [Morato, Normey-Rico, and Sename 2019; Alcalá, Puig, and Quevedo 2019]. An alternative approach would be to compute disturbance propagation tubes, as done in [Hanema, Tóth, and Lazar 2017]. We also note that the proposed zonotopes tend to be slightly larger than the ones proposed in [Cunha and Santos 2021], due to the addition of the term $\mathcal{V}(j)$. This addition, however, is due to the use of linear estimations to describe nonlinear dynamics, since the online update of the qLPV model must be taken into account in the disturbance propagation. Nonlinear predictions can result in less conservative sets $\mathcal{S}(j)$, but also result in a higher online computational burden. This represents a trade-off between larger uncertainty propagation and lower online computational cost brought by the linear predictions.

Remark 40. *In the case of qLPV systems with $A_\pi(\rho)$ and $B(\rho)$ affine*[8] *on ρ, it follows that $A_\pi(\rho(k+j)) - A_\pi(\rho(k+j|k)) = \overline{A}_\pi(\rho(k+j) - \rho(k+j|k))$ and $B(\rho(k+j)) - B(\rho(k+j|k)) = \overline{B}(\rho(k+j) - \rho(k+j|k))$, with $\overline{A}_\pi(\cdot)$ and $\overline{B}(\cdot)$ being linear maps. Then, the interval matrices $\mathbf{\Delta}_A(j)$ and $\mathbf{\Delta}_B(j)$ can be computed directly from $\mathcal{E}(j)$. In the case of non-affine models, interval arithmetic or optimisation can be used to obtain the interval matrices $\mathbf{\Delta}_A(j)$ and $\mathbf{\Delta}_B(j)$ from $\mathcal{P}$ and $\mathcal{E}(j)$.*

Remark 41. *Being zonotopes a symmetric class of sets, the disturbance propagation given by Theorem 3 may be overly conservative if the uncertainty distribution is highly asymmetrical. This problem can be mitigated by considering constrained zonotopes, which do not suffer from this source of conservatism. For this end, Theorem 3 can possibly be adapted based on the mean-value extension of constrained zonotopes proposed in [Rego et al. 2020]. This discussion is out of the scope of this Chapter and thus not extended herein.*

5.4 The novel robust NMPC

In this Section, we present the main contribution of this Chapter, which is the novel robust NMPC scheme for Tracking. Although the proposed strategy holds similarities to the tracking NMPC algorithms from [Limon et al. 2018; Köhler, Müller, and Allgöwer 2020], it differs significantly due to the fact that it is based on a qLPV prediction model of the system, i.e. Eq. (5.11). By using such qLPV nominal predictions, the proposed method is able to operate much faster than the prior, since the resulting numerical complexity becomes much closer to that of a QP (rather than an NP). Complementary, we employ constraint tightening, terminal cost and terminal constraints in order to ensure recursive feasibility and stability properties of the resulting closed-loop, as done in [Santos et al. 2019].

5.4.1 Motivation: a generic NMPC

Firstly, we motivate the debate by detailing how NMPC algorithms can be tuned for the case of possibly unreachable output reference signals, as shown in the literature [Limon et al. 2018;

[8] In this case, the feedback gain K_π must be parameter-independent such that $A_\pi(\rho) = A(\rho) + B(\rho)K_\pi$ becomes affine.

Köhler, Müller, and Allgöwer 2018]. In general, in order to potentially increase the closed-loop domain of attraction (and avoid feasibility losses due to set-point changes), artificial (virtual) reference variables are included to the optimisation, e.g. [Limon et al. 2018; Skibik et al. 2021].

The concept is as follows: instead of ensuring that the output variable tracks the time-varying output set-point y_r, the MPC is tuned so that the output alternatively tracks a new decision variable y_a. Furthermore, an additional offset cost $V_O(y_a - y_r)$ is included to the optimisation, which ensured that the deviation between artificial reference y_a and the real set-point y_r is minimised, while y_a stays within the set of admissible output targets reached within N_p steps (horizon of the MPC). For a correct implementation, this artificial variable tool must also be converted into related steady-state state and input variables. That is, the MPC must choose a reachable and admissible artificial reference $y_a \in \mathcal{Y}_a$ and, then, convert it into state and input coordinates through $x_a = g_x(y_a)$ and $u_a = g_u(y_a)$. These (nonlinear) constraints are, thus, included to the optimisation problem.

Then, in the general nonlinear case, for time-varying piece-wise constant output reference signal $y_r \in \mathcal{Y}_T$, the NMPC for Tracking algorithm (from [Limon et al. 2018]) is as given by the solution of the following NP at each sampling instant:

$$\min_{V_k, y_a} \overbrace{\sum_{j=0}^{N_p-1} \ell(x(k+j|k) - x_a, u(k+j|k) - u_a) + V(x(k+N_p|k) - x_a)}^{J} \qquad (5.18)$$
$$+V_O(y_a - y_r)\,,$$
$$s.t.: \begin{cases} x(k+j+1|k) = f\left(x(k+j|k), v(k+j|k) + K_\pi x(k+j|k)\right), j \in \mathbb{N}_{[0,N_p-1]}, \\ u(k+j|k) = v(k+j|k) + K_\pi x(k+j|k), j \in \mathbb{N}_{[0,N_p-1]}, \\ (x(k+j|k), v(k+j|k)) \in \mathcal{Z}_\pi, j \in \mathbb{N}_{[0,N_p-1]}, \\ x_a = g_x(y_a),\; u_a = g_u(y_a) \\ (x(k+N_p|k), y_a) \in \Gamma \\ y_a \in \mathcal{Y}_a \end{cases}$$

where, once again, $\ell(\cdot, \cdot)$ is a quadratic stage cost, $V(\cdot)$ is a terminal cost, and Γ is tracking positive invariant set. From the optimal solution of this NP, i.e. $V_k^\star$, the first entry $v^\star(k|k)$ is applied to the process according to Eq. (5.7), which implicitly defines $u(k)$.

5.4.2 The new formulation

In opposition to the NP in Eq. (5.18), we propose herein a novel robust formulation, which makes use of the qLPV model and removes the nonlinear constraints from the prior. We make reference to Remark 9 from [Köhler, Müller, and Allgöwer 2020]: the NMPC propositions in previous references [Limon et al. 2018; Köhler, Müller, and Allgöwer 2020] provide exponential closed-loop stability and recursive feasibility guarantees in the case of no model mismatch. Nevertheless, this is rarely the case in any practical application. Thus, one of the main features of the proposed mechanism is that it includes constraint tightening tools in order to robustly tackled the issue of disturbances and model mismatches.

When the qLPV predictions from Eq. (5.11) are used, considering the scheduling trajectory estimates from Eq. (3.43), model uncertainties inherently emerge, and thus should be accounted for. For such, as previously debated, the synthesis of our NMPC benefits from the zonotopes detailed in Sec. 5.3. Through the sequel, we assume that robust constraint satisfaction is guaranteed thanks to the constraint tightening and the corresponding disturbance propagation zonotopes. The contracted constraints that ensure robustness are detailed: Consider an initial constraint set $\mathcal{Z}_\pi(0) = \mathcal{Z}_\pi$. Then, the following sets, along the prediction horizon (i.e. for $j \in \mathbb{N}_{[1,N_p]}$), are iteratively given by:

$$\mathcal{Z}_\pi(j+1) \quad = \quad \mathcal{Z}_\pi(j) \ominus (\mathcal{S}(j) \times \{0\}) \ , \tag{5.19}$$

where $\mathcal{S}(j)$ stands for the zonotopes derived by the means of Theorem 3, which propagate the uncertainty along the prediction horizon.

Remark 42. *Note that the disturbance propagation sets $\mathcal{S}(j)$ increase along the horizon and thus the sets $\mathcal{Z}_\pi(j)$ from Eq. (5.19) shrink as j increases. For correctness of the NMPC application, these sets must be non-empty for all steps within the horizon $j \in \mathbb{N}_{[1,N_p]}$.*

Furthermore, the terminal constraint in Eq. (5.18) is also adjusted. Specifically, we consider a parameter-dependent tracking robust positive invariant[9] (TRPI) set:

Definition 5.2 (Parameter-dependent TRPI Set)
Consider a set $\Gamma(\rho) \subseteq \mathbb{R}^{n_x+n_y}$ and a terminal control law $u_t = \kappa_t(x, y_r) - K_\pi x$. $\Gamma(\rho)$ is a TRPI set the qLPV system in Eq. (5.3) if, for all $(x, y_r) \in \Gamma(\rho)$ and $w \in \mathcal{S}(N_p)$, it follows that $(A(\rho)x + B(\rho)\kappa_t(x, y_r)) + w, y_r) \in \Gamma(\rho)$.

Remark 43. *The TRPI sets differ from the Tracking Positive Invariant (TPI) sets from [Limon et al. 2018] due to the robustness properties. The synthesis of TPI sets implicitly considers that the real system trajectories and the nominal predictions are identical.*

Remark 44. *The definition of a TRPI set implies that once the states x and the virtual reference y_a are found inside such set, terminal control law $\kappa_t \colon \mathbb{R}^{n_x+n_y} \to \mathbb{R}^{n_u}$ ensures that any subsequent state, for the same output reference, also stays inside this set, regardless of the bounded load disturbance $w \in \mathcal{W}$.*

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

[9]The requirement of a tracking positive invariant set is not at all restrictive for piece-wise constant reference signals, and equally used in other references, e.g. [Limon et al. 2018]. Note that this set can be partitioned in multiple regions, as shown in Sec. 5.5.

(3.43), and solve the following optimisation problem:

$$\min_{V_k, y_a} \quad \sum_{j=0}^{N_p-1} \ell(x(k+j|k) - x_a, u(k+j|k) - u_a) \qquad (5.20)$$
$$+V(x(k+N_p|k) - x_a, \hat{\rho}(k+N_p-1|k)) + V_O(y_a - y_r)\,,$$
$$s.t.: \begin{cases} x(k+j+1|k) = A_\pi(\hat{\rho}(k+j|k))x(k+j|k) + B(\hat{\rho}(k+j|k))v(k+j|k), j \in \mathbb{N}_{[0,N_p-1]}, \\ u(k+j|k) = v(k+j|k) + K_\pi x(k+j|k), j \in \mathbb{N}_{[0,N_p-1]}, \\ (x(k+j|k), v(k+j|k)) \in \mathcal{Z}_\pi(j), j \in \mathbb{N}_{[0,N_p-1]}, \\ x_a = g_x(y_a),\ u_a = g_u(y_a) \\ (x(k+N_p|k), y_a) \in \Gamma(\hat{\rho}(k+N_p-1)) \\ y_a \in \mathcal{Y}_a \end{cases}.$$

The main changes from the NP in Eq. 5.18 (NMPC for Tracking from [Limon et al. 2018]) to the proposed approach in Eq. (5.20) are:

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

The set of states $x(k) \in \mathcal{X}_a(N_p)$ such that Eq. (5.20) has a feasible solution is called the domain of attraction of the proposed controller[10].

5.4.3 Artificial reference choice

The pair of constraints $x_a = g_x(y_a)$ and $u_a = g_u(y_a)$ in Eq. (5.20) still render it an NP. Then, in order to further alleviate its resulting numerical burden, we proceed by providing a final adjustment to the proposed MPC. Note that the presence of the artificial reference variable $y_a \in \mathcal{Y}_a$ in the NMPC optimisation serves to prevent the possible loss feasibility due to abrupt changes in the set-point (or non-admissible set-point value). Nevertheless, the constraints related to this variable (i.e. $x_a = g_x(y_a)$ and $u_a = g_u(y_a)$) significantly increase the computational complexity of the optimisation, since they are associated to (most possibly) nonlinear functions $g_x(\cdot)$ and $g_u(\cdot)$.

Therefore, we replace these constraints by solving the optimisation in two steps:

1. First, we determine the artificial reference variable y_a via a separate optimisation (typically named "reference governor" or "target optimisation" schemes, e.g. [Chisci, Falugi, and Zappa 2003; Köhler, Müller, and Allgöwer 2019]);
2. Then, we determine the control policy. By doing so, we are able to make the proposed NMPC optimisation only as complex as only two consecutive QPs.

[10]Due to the freedom provided by the artificial reference y_a, the feasibility property becomes independent of y_r [Limon et al. 2018].

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

$$y_a^o = \arg\min_{y_a \in \mathcal{Y}_a} V_O(y_a - y_r)\,. \tag{5.21}$$

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

Next, consider a *feasible* candidate artificial target y_a^c. In practice, this candidate variable is simply taken as the last artificial reference variable $y_a^\star$, which is ensured to be feasible due to the recursive feasibility property (Theorem 5, presented in the sequel) of the optimisation.

Remark 45. *We stress that y_a^c can only be selected from the second piece-wise constant reference change moment onward, since, at the first iteration of the MPC, the full nonlinear optimisation from Eq.* (5.20) *should be solved such that a recursively feasible candidate exists for the following samples.*

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

$$y_a^\alpha = (1-\alpha)\, y_a^c + \alpha y_a^o\,, \tag{5.22}$$

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

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

$$\begin{aligned} &\max_{V_k,\alpha} \quad \alpha \\ &\text{s.t.:} \quad \begin{cases} x(k+j+1|k) = A_\pi(\hat{\rho}(k+j|k))x(k+j|k) \\ \qquad\qquad + B(\hat{\rho}(k+j|k))v(k+j|k), j \in \mathbb{N}_{[0,N_p-1]}, \\ (x(k+j|k), v(k+j|k)) \in \mathcal{Z}_\pi(j), j \in \mathbb{N}_{[0,N_p-1]}, \\ \alpha \in [0,1], \\ (x(k+N_p|k), y_a^\alpha) \in \Gamma(g_\rho(y_a^o)). \end{cases} \end{aligned} \tag{5.23}$$

From the solution of the quadratic optimisation program in Eq. (5.23), we obtain the optimal value $\alpha^\star$, which is thus used to select the new artificial reference value $y_a = (1-\alpha^\star)y_a^c + \alpha^\star y_a^o$. Then, the proposed NMPC in Eq. (5.20) is solved without the constraints $x_a = g_x(y_a)$, $u_a = g_u(y_a)$, and $y_a \in \mathcal{Y}_a$ and without the offset cost $V_O(y_a - y_r)$, since y_a is no longer a decision variable, but an input to the optimisation. From it, the optimal control sequence $V_k^\star$. We stress that this two-step optimisation procedure is analogous to solving the NP in Eq. (5.20).

Remark 46. *Note that the maximisation in Eq.* (5.23) *is convex, quadratic and has a real positive scalar $\alpha \in [0,1]$ as its decision variable. Thus, an equivalent approach to solving is* *to perform a bisection search over the unit simplex*[11] *. By doing so, we can test the feasibility of* (5.20) *for a given* α*. If the problem is feasible, we use the corresponding artificial variable* y_a *and proceed to the MPC solution; otherwise, we pursue with the bisection.*

Remark 47. *Again, recall that the at the initial sample, the complete NP from Eq.* (5.20) *must be solved . Anyhow, if for any consecutive time sample* $k \geq 1$ *(when a reference change occurs), the solution of optimisation in Eq.* (5.23) *leads to* $\alpha^\star = 1$*, it is implied that* $y_a = y_a^o$*. In this case, the auxiliary reference governor optimisation becomes irrelevant until there is another variation of the set-point.*

5.4.4 Synthesis requirements

Next, we give some specific hypothesis on the form of the quadratic penalties costs $\ell(\cdot,\cdot)$, $V(\cdot)$, and the terminal TRPI set $\Gamma(\cdot)$, which are required to construct Eq. (5.20) with performance certificated. We note that the following requirements are similar to those presented in [Santos et al. 2019] and [Limon et al. 2018], which develop predictive control strategies for robust regulation and nominal tracking, respectively.

Assumption 15. *The MPC cost* $J = \sum_{j=0}^{N_p-1} \ell(x(k+j|k) - x_a, (v(k+j|k) + K_\pi x(k+j|k)) - u_a) + V(x(k+N_p|k) - x_a, \hat{\rho}(k+N_p-1|k))$ *satisfies the following requirements:*

1. *Its stage cost* $\ell(x,u)$ *quadratic, positive definite and uniformly continuous. Therefore, it follows that* $\ell(x,u) \geq \alpha_\ell(\|x\|)$ *and* $|\ell(x_1,u_1) - \ell(x_2,u_2)| \leq \lambda_x(\|x_1 - x_2\|) + \lambda_u(\|u_1 - u_2\|)$*, where* α_ℓ*,* λ_x *and* λ_u *are* $\mathcal{K}$*-functions.*

2. *The set of* ***reachable*** *admissible artificial references* $\mathcal{Y}_a = \{y_a \in \mathbb{R}^{n_y} \colon (g_x(y_a), y_a) \in \Gamma(g_\rho(y_a))\}$ *is a convex subset of the admissible tracking outputs reached within* N_p *steps, i.e.* $\{y_a \in \mathcal{Y}_T \colon (g_x(y_a), g_u(y_a)) \in \mathcal{Z}_\pi(N_p)\}$*.*

3. *Its output offset cost* $V_O(\cdot)$ *is quadratic, positive definite, uniformly continuous and convex, thus assuring that the minimiser* $y_a^o = arg\min_{y_a \in \mathcal{Y}_a} V_O(y_a - y_r)$ *is unique. Furthermore, for any* $y_r \in \mathbb{R}^{n_y}$ *and* $y_a \in \mathcal{Y}_a$*, we have* $V_O(y_a - y_r) - V_O(y_a^o - y_r) \geq \alpha_O(\|y_a - y_a^o\|)$*, where* α_O *is a* $\mathcal{K}$*-function*[12]*.*

4. *Its terminal cost* $V(\cdot)$ *is a control Lyapunov function for the unconstrained qLPV system in Eq.* (5.3)*, such that for all* $(x, y_a) \in \Gamma(\rho)$ *there exist constants* $b > 0$ *and* $\sigma > 1$ *such that* $V(x - x_a, \rho) \leq b|x - x_a|^\sigma$*. Also, due to the uniform continuity of* $V(x,\rho)$ *with respect to* x*, we have:* $V(A(\rho)(x - x_a) + B(\rho)\kappa_t(x - x_a, y), \rho^+) - V(x - x_a, \rho) \leq -\ell(x - x_a, \kappa_t(x,y)x - u_a)$*, where* $x_a = g_x(y_a)$ *and* $u_a = g_u(y_a)$*, and* $V(x_1 - x_a, \rho_1) - V(x_2 - x_a, \rho_2) \leq \lambda_r(|x_1 - x_2|)$*.*

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

---

[11] In order to preserve the ISS property, this bisection should also ensure that the terminal offset cost decays with the new artificial target, this is: $V(x(k+1|k) - g_x(y_a^\alpha)) - V(x(k|k) - g_x(y_a^c)) \leq -\ell(x(k|k) - g_x(y_a^c), \cdot)$.

[12] As long as the target output is admissible and reachable (i.e. $y_r \in \mathcal{Y}_a$), the minimiser is $y_a^o = y_r$ and the previous inequality can be reduced to $V_O(y) \geq \alpha_O(\|y\|)$.

1. *The terminal control law implies that* $\kappa_t(g_x(y_a), y_a) = g_u(y_a)$ *for all admissible equilibrium points*[13].

2. *The terminal set* $\Gamma(\rho)$ *is an admissible TRPI set. That is,* $\Gamma(\rho)$ *is a subset*[14] *of* $\Lambda(N_p) = \{(x, y) \in \mathbb{R}^{n_x} \times \mathcal{Y}_a \colon (x, \kappa_t(x, y) - K_\pi x) \in \mathcal{Z}(N_p)\}$*, thus satisfying Definition 5.2 for the terminal control law and disturbances* $w \in \mathcal{S}(N_p)$*. Equivalently, it is implied that* $(x, y_s) \in \Gamma \implies (x^+, y_s) \in \Gamma, x^+ = A(f_\rho(x))x + B(f_\rho(x))\kappa_t(x, y_s) + w,\ w \in \mathcal{S}(N_p)$, $(x, \kappa_t(x, y_s)) \in \mathcal{Z}$.

5.4.5 Terminal ingredients

Next, we provide a computationally elegant solution that can be used to compute parameter-dependent qLPV terminal ingredients, through the solution of matrix inequalities. The following Theorems provide recursive feasibility and exponential stability guarantees for the qLPV system in Eq. (5.3) subject to the proposed MPC control law from Eq. (5.7). For the sake of presentation clarity, the longer proofs have been moved in Appendix C.

Specifically, we synthesise ρ-dependent terminal ingredients, based on a positive-definite matrix $P(\rho)$. We use an ellipsoidal invariant $\mathbf{X}_f := \left\{x \,|\, x^T P(\rho) x \leq 1\right\}$, which is robust positively invariant regarding the closed-loop dynamics (Eq (5.8)), in such way that $\Gamma(\rho) := \mathbf{X}_f(\rho) \times \mathcal{Y}_a$ is a TRPI set. Complementary, we consider a sub-level terminal cost $V(x, \rho) = x^T P(\rho) x$ and a parameter-dependent terminal feedback $\kappa_t(x, y_a) = K_t(\rho)(x - g_x(y_a)) + g_u(y_a)$.

Consider the tracking error dynamics $e(k+j) = x(k+j|k) - g_x(y_a)$, being $x(k+j|k)$ being the nominal qLPV predictions from Eq. (5.11). We define $\theta(k+j|k) := w(k+j|k) + \Delta(k+j|k) \in \mathcal{S}(j)$ as the (bounded) uncertainties: result of the disturbance and the model-process mismatch due to the differences between the real scheduling variables $\rho(k+j)$ and $\hat{\rho}(k+j|k)$ (scheduling trajectory estimates from Eq. (3.43)).

From the nominal control law $u(k+j|k) = K_\pi x(k+j|k) + v(k+j|k)$ and the terminal condition $u_t = \kappa_t(x, y_a) - K_\pi x$, with $\kappa_t(x(k+j|k), y_a) = K_t(\hat{\rho}(k+j|k))e(k+j) + g_u(y_a)$, it follows for all $j \in \mathbb{N}_{[0, N_p - 1]}$ that:

$$e(k+j+1) = \overbrace{\left(A(\hat{\rho}(k+j|k)) + B(\hat{\rho}(k+j|k))K_t(\hat{\rho}(k+j|k))\right)}^{A_t(\hat{\rho}(k+j|k))} e(k+j) + \theta(k+j|k), \tag{5.24}$$

$$\theta(k+j|k) = (A(\hat{\rho}(k+j|k)) - I_{n_x})g_x(y_a) + B(\hat{\rho}(k+j|k))g_u(y_a) + w(k+j) \in \mathcal{S}(j). \tag{5.25}$$

Then, the following results ensure that the error dynamics from Eq. (5.24) converge to the origin (i.e. tracking is ensured). Again, recall that the total uncertainty set $\mathcal{S}(j)$ encompasses the disturbances and the model-mismatches (refer to Definition 5.1). Next, we provide a solution to compute terminal ingredients which satisfy Theorem 1 (taking into account the corresponding changes in the model).

[13] An evident explicit alternative for this terminal law is $\kappa_t(x, y_a) = K_t(f_\rho(x))(x - g_x(y_a)) + g_u(y_a)$.

[14] Note that $\mathcal{Z}(N_p)$ is the set of points (x, u) such that $u = v + K_\pi x$ with $(x, v) \in \mathcal{Z}_\pi(N_p)$.

$$\left[\begin{array}{cc|cc|cc|c} Y(\rho) & \star & \star & \star & \star & \star & \star \\ A(\rho)Y(\rho)+B(\rho)W(\rho) & Y(\rho^+) & \star & \star & \star & \star & \star \\ \hline 0 & \theta^T & I & \star & \star & \star & \\ A(\rho)Y(\rho)+B(\rho)W(\rho) & Y(\rho^+) & \theta & Y(\rho^+) & \star & \star & \star \\ \hline Y(\rho) & 0 & 0 & 0 & Q^{-1} & \star & \star \\ W(\rho) & 0 & 0 & 0 & 0 & R^{-1} & \star \\ \hline I & 0 & 0 & 0 & 0 & 0 & I \end{array}\right] \geq 0\,, \tag{5.28}$$

$$\text{(a)} \begin{bmatrix} \left(\overline{u}_i - I_{n_u,\{i\}}u_r\right)^2 & \star \\ (I_{n_u,\{i\}}W(\rho)) & Y(\rho) \end{bmatrix} \geq 0\,,\ \forall i \in \mathbb{N}_{[1,n_u]}\,, \tag{5.29}$$

$$\text{(b)} \begin{bmatrix} \left(\overline{x}_i - I_{n_x,\{j\}}x_r\right)^2 & \star \\ (I_{n_x,\{j\}}Y(\rho)) & Y(\rho) \end{bmatrix} \geq 0\,,\ \forall j \in \mathbb{N}_{[1,n_x]}\,,$$

$$\left[\begin{array}{cc|cc} \lambda Y(\rho) & \star & 0 & \star \\ (A(\rho)Y(\rho)+B(\rho)W(\rho)) & Y(\rho^+) & \star & \star \\ \hline 0 & \theta^T & (1-\lambda) & \star \\ (A(\rho)Y(\rho)+B(\rho)W(\rho)) & Y(\rho^+) & \theta & Y(\rho^+) \end{array}\right] > 0\,. \tag{5.30}$$

Theorem 4 (Tracking robust positive invariant set)**.** *Assume that there exists an ellipsoidal terminal set $\mathbf{X}_f(\rho)$. $\mathbf{X}_f$ is a robust positively invariant set iff, for any $e \in \mathbf{X}_f$ and $\rho \in \mathcal{P}$, i.e. $e^T P(\rho) e \leq 1$, it follows that $(e^+)^T P(\rho+\delta\rho)e^+ \leq 1$, i.e. the successor state e^+ is also inside $\mathbf{X}_f$, which implies in:*

$$(A_t(\rho)e+\theta)^T P(\rho+\delta\rho)(A_t(\rho)e+\theta) \quad \leq \quad 1\,. \tag{5.26}$$

Then, $\Gamma(\rho)$ is a TRPI for system (5.3) as follows:

$$\Gamma(\rho) \quad := \quad \{(x,y) \in \mathbb{R}^{n_x \times n_y} \mid (x - g_x(y)) \in \mathbf{X}_f(\rho), h(x, K_t(\rho)(x - g_x(y) + g_u(y)) \in \mathcal{Y}_a\}. \tag{5.27}$$

Proof. The validity of Eq. (5.26) follows directly from the following argument: $x(k), x(k+1) \in \mathbf{X}_f(\rho), \forall k \geq 0$ and $y = h(x, K_\pi x + K_t(\rho)(x - g_x(y)) + g_u(y)) \in \mathcal{Y}_a$. □

Theorem 5 (Terminal ingredients)**.** *Conditions (C1)-(C5) of Theorem 1 and the inequality of Theorem 4 are satisfied if there exist a symmetric parameter-dependent positive definite matrix $P(\rho) : \mathbb{R}^{n_p} \to \mathbb{R}^{n_x \times n_x}$, a parameter-dependent rectangular matrix $W(\rho) : \mathbb{R}^{n_p} \to \mathbb{R}^{n_u \times n_x}$, and a scalar $\lambda \in]0,1]$ such that $Y(\rho) = (P(\rho))^{-1} > 0$, $W(\rho) = K_t(\rho)Y(\rho)$ and that LMIs (5.28)-(5.29) and the BMI (5.30) hold under the minimisation of $\log det\{Y(\rho)\}$ using $\rho^+ = \rho + \delta\rho$ for all $\rho \in \mathcal{P}$ and $\delta\rho \in \delta\mathcal{P}$, considering θ as the vertices of $\mathcal{S}(N_p)$.*

Remark 48. *The terminal ingredients provided by Theorem 5 ensure recursive feasibility and ISS of the error trajectories (as verified in Propositions 5-6). Note that these ingredients are **robust** with respect to the mismatches between the nominal model from Eq. (5.11) and the real system trajectories from Eq. (5.10). The robustness is implied thanks to the bounds on uncertainty, which are derived from the bounds on the scheduling trajectory estimation* *error (Lemma 4.8.2). Moreover, constraint satisfaction is also enabled by robust design, when tightening the constraints along the horizon with the uncertainty propagation zonotopes $\mathcal{S}(j)$.*

Remark 49. *The BMI in Theorem 5 can be solved through simple bisection search over the optimisation plane since $0 < \lambda \leq 1$, by construction, as argues [Yang et al. 2016].*

Remark 50. *Theorem 5 provides infinite-dimensional inequalities, which must hold $\forall \rho \in \mathcal{P}$ and $\forall \delta\rho \in \delta\mathcal{P}$. In practice, the solution can be found by enforcing the inequalities over a sufficiently dense grid of points $(\rho, \delta\rho)$ along the $\mathcal{P} \times \delta\mathcal{P}$ plane. Then, the solution can be verified over a denser grid. The parameter-dependency of P may be dropped if the system is quadratically stabilisable, but this may result in quite conservative performances.*

5.4.6 Certificates

Proposition 5 (Recursive feasibility)**.** *Let there exist a solution $Y(\rho)$ to Theorem 5. Then, given any $x \in \mathcal{X}_a(N_p)$, $y_a \in \mathbb{R}^{n_y}$ and $u = K_\pi x + v$, $v = \kappa(x, y_a)$, we have $x^+ = A(f_\rho(x))x + B(f_\rho(x))u + w \in \mathcal{X}_a(N_p), \forall w \in \mathcal{W}$. Consider an optimal sequence $V_k^\star = \left[(v^\star(k|k))^T, (v^\star(k+1|k))^T, \dots, (v^\star(k+N_p-1|k))^T\right]^T$ and an optimal artificial target $y_a^\star$. Then $V_k^c = \left[(v^\star(k+1|k))^T, \dots, (v^\star(k+N_p-1|k))^T, \kappa_t(x(k+N_p|k), y_a^\star) - u_a\right]^T$ and $y_a^c = y_a^\star$ define a feasible (candidate) solution from the optimisation in Eq.* (5.20) *for any $\overline{y}_a \in \mathbb{R}^{n_y}$ and $w \in \mathcal{W}$, which means that Eq.* (5.20) *is recursively feasible.*

Proposition 6 (Error ISS)**.** *Let there exist a solution $Y(\rho)$ to Theorem 5. Then, the qLPV system in Eq.* (5.3) *in closed-loop with the MPC input from Eq.* (5.7) *has uniformly exponentially input-to-state stable error dynamics (as of Eq.* (5.24)*). That is, for any feasible initial condition x_0 and constant set-point $y_r \in \mathbb{R}^{n_y}$, with $w(k) \in \mathcal{W}$, it is implied that:*

$$\|x(k) - x_a(k)\| \leq \beta(\|x(0)\|, k) + \gamma(\overline{w}), \tag{5.31}$$

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

Remark 51. *Note that for an admissible equilibrium state x_a, the virtual control sequence $V_k^a = \left[v_a^T, \dots, v_a^T\right]^T$, where $u_a = v_a + K_\pi x_a$, is admissible since it maintains the system at x_a. Therefore, the set of corresponding admissible equilibrium states $\{x \in \mathbb{R}^{n_x} : \exists\, u_a \in \mathbb{R}^{n_u},\ (x_a, u_a) \in \mathcal{Z}_\pi(N_p), h(x_a, u_a) \in \mathcal{Y}_a\}$ is a subset of $\mathcal{X}_a(N_p)$ and feasibility is not lost for any set-point change.*

Remark 52. *In practice, we note that the feasible candidate solution (V_k^c, y_a^c) can be used as a warm-start to the optimisation in Eq.* (5.20)*.*

5.4.7 A summary

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

- **Offline procedure**:
  - Verify the baseline Assumption 12, required to apply the method;
  - Conceive a qLPV realisation of the nonlinear dynamics, in the form of Eq. (5.20);
  - Determine a (locally) stabilising state-feedback in the form of Eq. (5.7) (within the admissibility set $\mathcal{X}$);
  - Map the admissible equilibrium points determined by the output target y_r through Eq. (5.6), and determine the related maps $g_x(\cdot)$, $g_u(\cdot)$ and $g_\rho(\cdot)$;
  - Compute the estimation error bounds through Lemma 4.8.2;
  - Compute the uncertainty propagation sets $\mathcal{S}(j)$ through Theorem 3;
  - Determine the robust tracking terminal ingredients through Theorem 5.
- **Online procedure**:
- At each sample k:
  - Estimate the future sequence of scheduling variables, using the recursive extrapolation method (Eq. (3.43)), obtaining $\hat{P}_k$;
  - For every reference change in the piece-wise constant target signal y_r: solve the artificial reference choice (Eq. (5.23) together with the bisection selection), obtaining $y_a^\star$;
  - Solve the MPC optimisation in Eq. (5.20), using $y_a^\star$ and removing the related constraints and offset cost[15];
  - Apply the resulting control signal using $u(k) = K_\pi x(k) + v^\star(k|k)$.

5.5 Results

In this Section, we provide a simple case study in order to illustrate the features of the proposed Tracking NMPC algorithm. We debate the advantages and disadvantages of our method, comparing it to the state-of-the-art approach from [Limon et al. 2018], i.e. Eq. (5.18).

5.5.1 Nonlinear model and constraints

We consider the benchmark cascaded tank process from [Johansson 2000], considering the interconnection of two tanks, with an open hole at the bottom of the first tank, which leaks

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

fluid to the second. The latter has a pump at its end, regulated by a local proportional controller. The nonlinear level dynamics are:

$$\begin{cases} \frac{dh_1(t)}{dt} &= -\frac{a\sqrt{2gh_1(t)}}{A_1} + \frac{\gamma}{A_1}u(t)\,, \\ \frac{dh_2(t)}{dt} &= \frac{a\sqrt{2gh_1(t)}}{A_2} - \frac{k_p}{A_2}h_2(t)\,. \end{cases} \tag{5.32}$$

Each $h_i(t)$ represents the water level at the i-th tank, measured in centimeters; $u(t)$ represents the tension applied for the main pump in volts, for which the corresponding flow is $\gamma u(t)$. The tank cross sections A_i are of $10\,\text{cm}^2$, while the outlet hole cross section a is of $0.05\,\text{cm}^2$. The pump parameter γ is of $1.4\,\text{cm}^3/(\text{Vs})$. The proportional coefficient k_p is of $1.1\text{cm}^2/\text{s}$. Both level signals are considered the system state variables, i.e. $x(t) = [h_1(t)\,h_2(t)]^T$, whereas $y(t) = h_1(t)$ is the controlled output (the main variable of interest. The control input is the main pump tension signal $u(t)$. Also, we stress that this system should agree to the following admissibility constraints:

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

For coherence with the nonlinear model in Eq. (5.1), we assume that this process is also subject to bounded additive disturbances $w(k) \in \mathbb{R}^2$ such that $\|w(k)\| \leq 0.05$ cm. These disturbances perturb both level dynamics and could represent, for instance, unaccounted leaks or flows to each tank.

5.5.2 qLPV Embedding

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

$$\begin{cases} h_1(k+1) &= h_1(k) - T_s\frac{a\sqrt{2gh_1(t)}}{A_1} + T_s\frac{\gamma}{A_1}u(t)\,, \\ h_2(k+1) &= h_2(k) + T_s\frac{a\sqrt{2gh_1(t)}}{A_2} - T_s\frac{k_p}{A_2}h_2(t)\,. \end{cases} \tag{5.33}$$

Then, we choose the following nonlinear scheduling proxy:

$$\rho(k) \;:=\; f_\rho(x) \;:=\; (x_1)^{-0.5}, \tag{5.34}$$

which satisfies the requirement for differential inclusion and boundedness (Assumption 12). Thus, we obtain a qLPV-embedded model in the form of Eq. (5.3) with matrices:

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

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

$$\mathcal{P} \;=\; [0.3, 1]\,\mathrm{cm}^{-0.5}\,. \tag{5.35}$$

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

$$\delta\mathcal{P} := \{\delta\rho \in \mathbb{R}\,|\, -0.034 \leq \delta\rho \leq 0.0052\}\;. \tag{5.36}$$

5.5.3 Tracking

For tracking purposes, we consider the convex output tracking set of admissible references $\mathcal{Y}_T := [1\,, 10]$ cm. Moreover, we stress that the output steady-state condition from Eq. (5.6) implicitly defines the following functions:

$$\begin{cases} g_x(y_r) & := \begin{bmatrix} y_r \\ \frac{a\sqrt{2gy_r}}{k_p} \end{bmatrix}, \\ g_u(y_r) & := \frac{a\sqrt{2gy_r}}{\gamma}\,. \end{cases} \tag{5.37}$$

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

5.5.4 MPC synthesis and terminal ingredients

Using this generated qLPV embedding model, we compare the proposed NMPC method with the "NMPC for Tracking" algorithm from [Limon et al. 2018], i.e. Eq. (5.18). Note that this other algorithm operates on the basis of the original discrete-time nonlinear model of the system. In order to synthesise these predictive controllers, we use a prediction horizon of $N_p = 4$ steps and the quadratic stage cost $\ell(x,u) = \|x\|_Q^2 + \|u\|_R^2$ with $Q = I_{n_x}$ and $R = 1$. We note that the short size of the horizon is specifically chosen in order to emphasise the numerical capabilities of the proposed scheme which, even in these simpler cases, exhibits considerable decrease on the resulting time required to compute the control action, as shown in the sequel.

In order to synthesise the terminal ingredients via Theorem 5, we partition the output set $\mathcal{Y}_T$ in ten different partitions, thus finding one parameter-dependent RPI set $\Gamma(\rho)$ per partition, i.e. for $y_r \in [1, 2]$ or $(2, 3]$ or $(3, 4]$, and so on up to $(9, 10]$. For the NMPC algorithm, we use terminal ingredients synthesised through the procedure from [Limon et al. 2018, Appendix B].

Figure 5.1 shows the parameter-dependent TRPI sets and the quadratic TPI sets used for the NMPC algorithm [Limon et al. 2018]. The parameter-dependent sets (proposed in

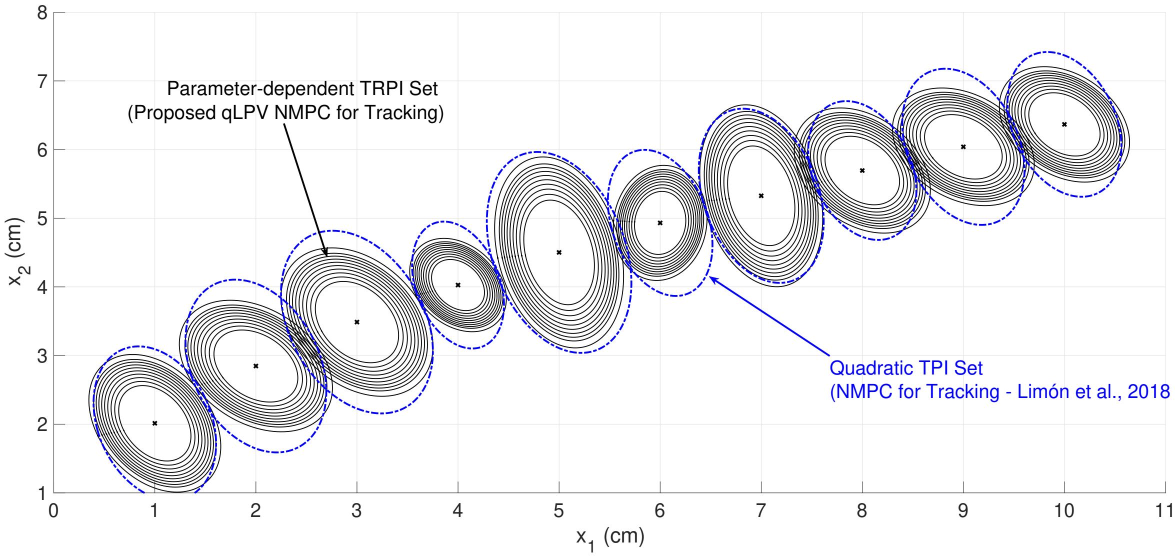


Figure 5.1: Synthesised TRPI set partitions and the quadratic TPI set partitions.

this Chapter) are shown for in bold black lines for frozen values of $\rho \in \mathcal{P}$, i.e. for $\rho = 0.3, 0.37, 0.44, \ldots, 1$. All sets are translated to from the error coordinates $(x - x_r)$ to the state coordinates x, centered at the different state targets x_r of each partition. We note that, albeit the parameter-dependent sets being slightly smaller than the TPI sets (due to the robustness considerations), Theorem 5 generates sufficiently large terminal regions for each reference partition $y_r \in \mathcal{Y}_T$.

5.5.5 Scheduling trajectory extrapolation

Before presenting the actual control results, we provide the scheduling sequence extrapolation estimates obtained with the recursive method presented in Chapter 3. As detailed in Lemma 3.6.2, convergence is indeed verified. The obtained bounds for the estimation error, using Lemma 4.8.2, are $\|\xi_\rho\| \leq \xi_\rho^{\text{bound}} = 0.015\,\text{cm}^{-0.5}$, as shown in Figure 5.2. The extrapolation mechanism offers very precise estimates $\hat{P}_k$, which means that the nominal qLPV predictions obtained through Eq. (5.11) are very close to the real system trajectories of Eq. (5.10) and thus the disturbance propagation along the horizon is reduced.

5.5.6 Disturbance propagation

The disturbance propagation reachable sets $\mathcal{S}(j), \forall j \in \mathbb{N}_{[0,N_p]}$ for the proposed algorithm were then obtained, considering the zonotopic disturbance propagation method and a closed-loop prediction paradigm defined by (5.7), with $K_\pi = \begin{pmatrix} -24.92 & 0 \end{pmatrix}$ calculated as proposed in [Cunha and Santos 2021]. Note that $\mathcal{S}(0)$ stands for the load disturbance set $\mathcal{W}$; the following zonotopes comprise the propagation of the load disturbances and the model-process mismatches along the prediction horizon. These sets are computed according to Theorem 3.

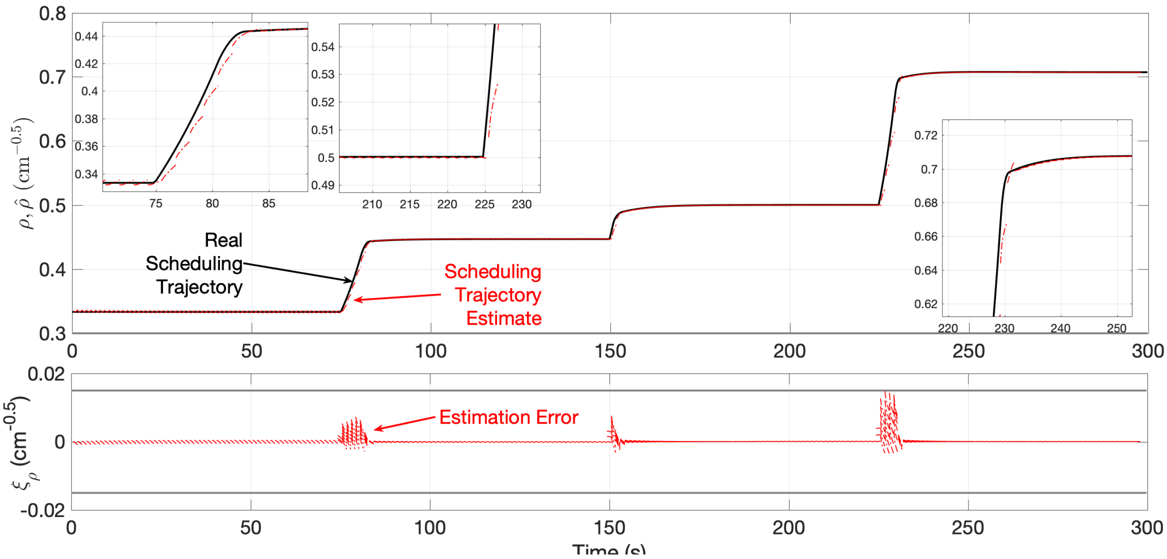


Figure 5.2: Scheduling trajectory estimates $\hat{P}_k$, corresponding estimation error $\xi_\rho(k+j|k)$, and error bounds.

In Figure 5.3, we show the collection of disturbance propagation sets $\mathcal{S}(j)$ over the $x_1 \times x_2$ plane (Definition 5.1). In this Figure, we can also see the original state admissibility set $\mathcal{X}$. Since the zonotopes $\mathcal{S}(j)$ are much smaller in size than $\mathcal{X}$, we can infer that the conservatism of the proposed method is quite reduced. We recall that the vertices of $\mathcal{S}(N_p)$ were used to construct the terminal ingredients through Theorem 5.

5.5.7 Simulation scenarios

We consider two different simulation scenarios. Since the NMPC algorithm from [Limon et al. 2018] is not robust by design, we first consider and compare the obtained tracking performances of both algorithms, without the presence of load disturbances (i.e. with w nil). Then, we consider the disturbance rejection robust performances solely of the proposed robust algorithm. Note that both controllers consider the same system (and also the same MPC synthesis weights), just represent via different realisations (nonlinear and qLPV models).

5.5.7.1 Nominal performances

Considering a step-like piece-wise constant output target signal which passes through $y_r = 2$, 4, 5 and 9 cm, the obtained tracking performances with both algorithms are shown in Figures 5.4 and 5.5. Figure 5.4 presents the resulting state, input and output trajectories, while Figure 5.5 shows the state phase plane and the terminal sets. Complementary, Figure 5.6 provides the values for the artificial reference tuning variable α for the proposed mechanism.

As one can see, the obtained tracking performances with both methods are offset-free

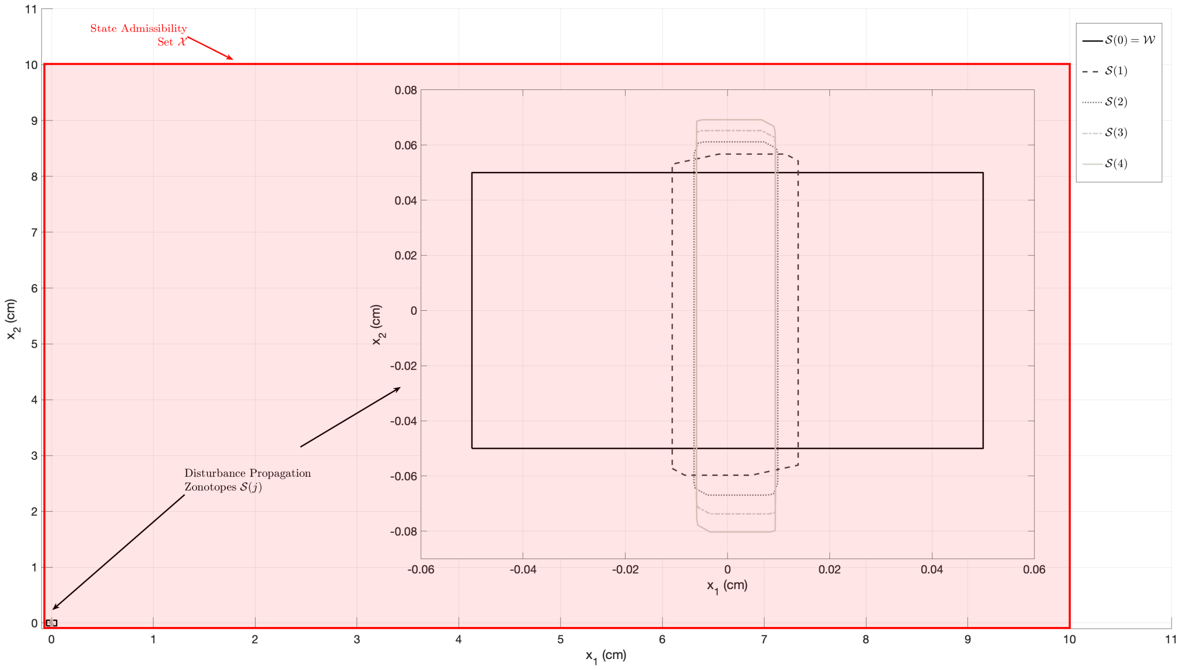


Figure 5.3: Zonotopic sets $\mathcal{S}(j)$.

steady-state output points. The proposed method ensures slightly faster convergence than the original NMPC for Tracking scheme from [Limon et al. 2018]. The main advantage resides in its simpler implementation, of QP-alike numerical burden, enable through the qLPV embedding. With the qLPV model, nonlinear constraints do not have to be solved internally by the optimisation procedure. The proposed qLPV NMPC mechanism requires only the operation of: one linear recursive law (Eq. (3.43)) and one QP problem (Eq. (5.20)). In the moments of reference changes, one bisection search (Remark 46) is also required, which increases the number of QPs to, at most, five iterations per sample. In contrast, the original NMPC for Tracking requires the solution of an NP optimisation problem per sampling instant, which is numerical-wise much harder.

n order to better compare the two tracking controllers, we assess the obtained performance results with performance indexes, presented in Table 5.1. The results are debated:

- Firstly, we stress that there is an overall performance enhancement with the proposed method: there is a small decrease on IAE index with respect to [Limon et al. 2018], of roughly 12 %. This performance enhancement can also be quantified through the RMS index of the cost function J_k, as well as in terms of the average tracking error (2.15 % with the proposed method, while 2.68 % using [Limon et al. 2018]). This performance enhancement is indeed an interesting feature of the proposed scheme, since the method from [Limon et al. 2018] has been exploited in many nonlinear applications presented in the literature.

- Complementary, we stress that the generated control input is smoother with the pro-

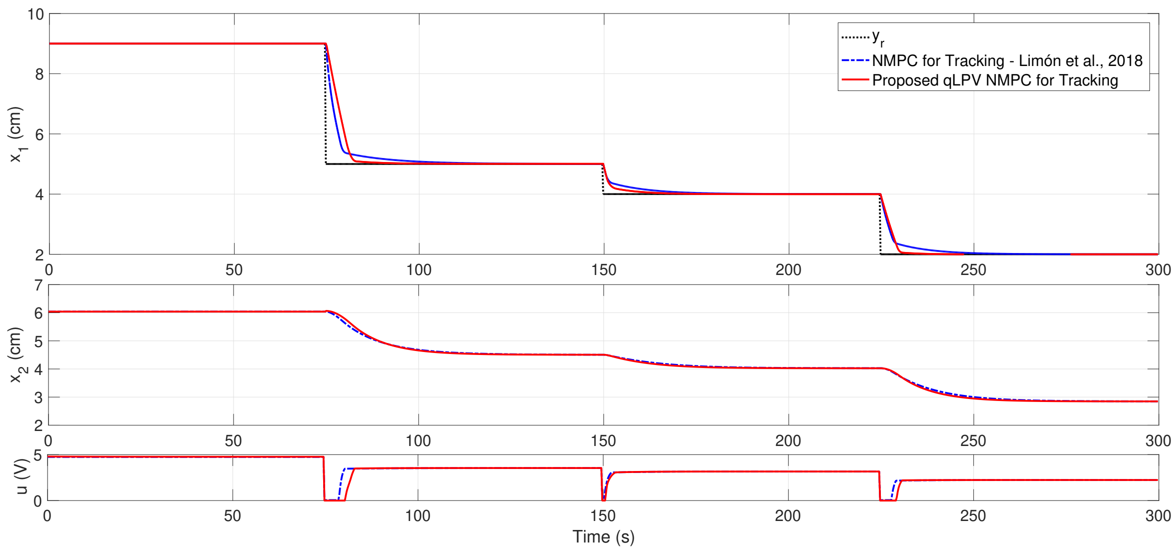


Figure 5.4: Nominal performances: State and input trajectories.

posed method: we obtain a control signal with 43 % reduction on its total variance (TV).

- Furthermore, the main advantage of the proposed method is that the average computational time needed to solve the control problem (t_c) is reduced over 44 %, as also exhibited in Figure 5.7. This is a strong and very significant feature, since the system order is small ($n_x = 2$) and so is the chosen control horizon. The complexity of the NP solution from [Limon et al. 2018] grows exponentially with ($N_p \times n_x$), which can a serious issue with time-critical systems. The proposed method has QP-alike burden, and thus t_c grows only linearly with ($N_p \times n_x$). This means it is readily-conceived for embedded applications.

- The model-process discrepancies (differences between Eqs. (5.10) and (5.11)) are very well-handled with the zonotope-based constraint-tightening approach, since the generated sets $\mathcal{S}(j)$ are arguably small (see Figure 5.3). This feature corroborates with prior discussions seen in the literature indicating this approach is promising, e.g. [Köhler, Müller, and Allgöwer 2020].

- Finally, we also stress that the terminal ingredients conceived with the proposed Theorem generate sufficiently large terminal sets, able to guarantee recursive feasibility for rather larger output-related sets.

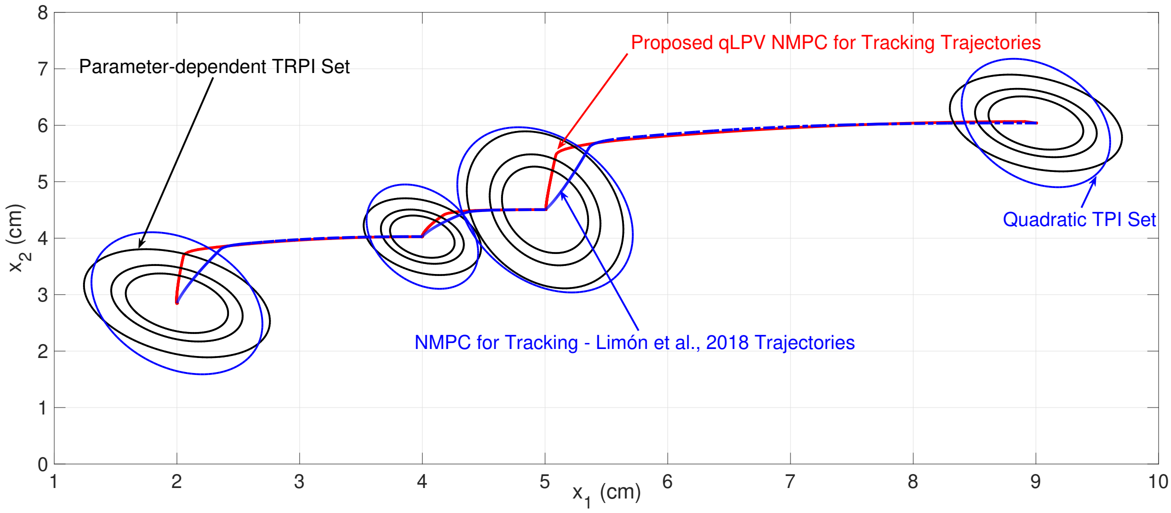


Figure 5.5: Nominal performances: State phase plane and TRPI sets.

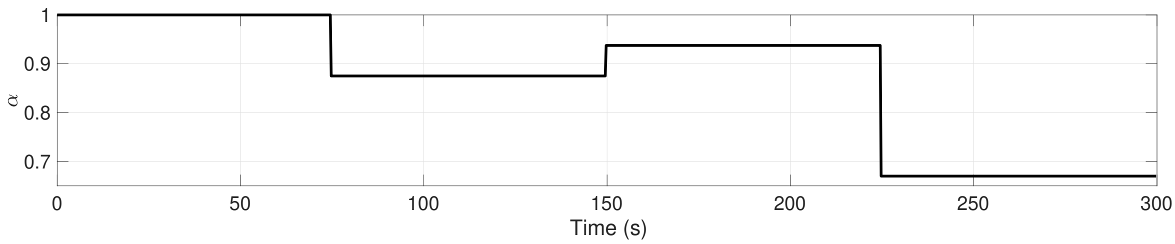


Figure 5.6: Artificial reference choice variable.

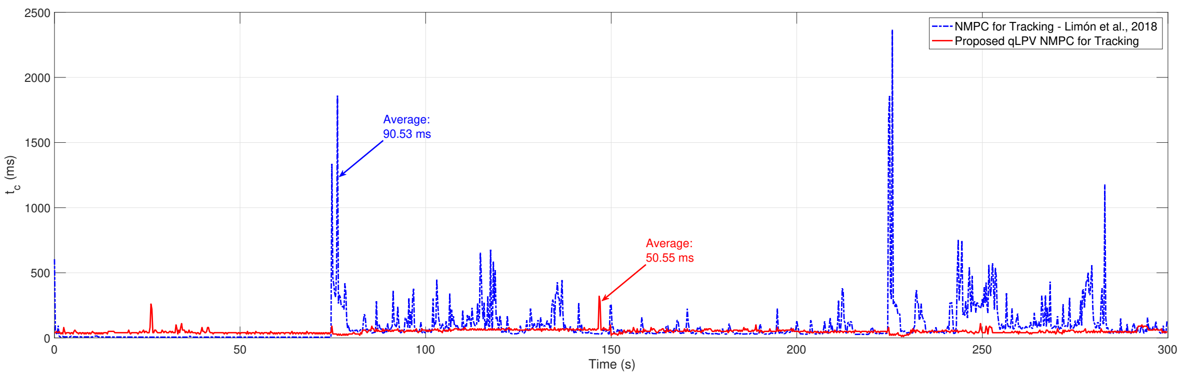


Figure 5.7: Nominal performances: Computational time.

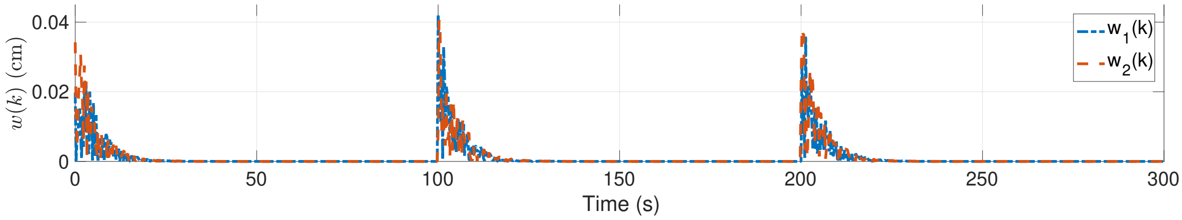


Figure 5.8: Disturbance scenario.

5.5.7.2 Robust performances

In order to illustrate the robustness properties of the proposed algorithm, we provide a second brief simulation scenario[16]. Consider three uniformly random disturbance sequences with unitary seeds, multiplied by decaying exponential terms, as illustrated in Figure 5.8.

In Figure 5.9, we show the state and output behaviours with respect to a step-like output target goal y_r. Clearly, robust stability is ensured: as the load disturbance sequences dissipate, the error trajectories $(x-x_r)$ converge to the origin; moreover, while $w(k)$ is non-null, the states stabilize at constant steady-states regimes, as close as possible to x_r. This is an additional nice feature of the proposed method, which is able to robustly tolerate bounded disturbances, which was not possible with competing NMPC techniques for tracking.

Table 5.1: Performance comparison.

Method	IAE	RMS$\{J\}$	TV	t_c
NMPC for Tracking	108.23	80.18	35.93	90.53 ms
Proposed qLPV NMPC	95.31	80.14	20.44	50.55 ms

5.5.8 Final debate

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

- Advantages:

  1. It is able to operate faster than state-of-the-art nonlinear MPCs for tracking, given that a reduced-complexity optimisation procedure is used (of QP-alike complexity);
  2. It includes artificial reference variables such that even unreachable reference goals are able to be directly taken into account by the controller;
  3. It includes (easy-to-compute) robustness arguments, written in terms of known the bounds of the load disturbances;

[16] We opt not to test the method from [Limon et al. 2018] against load disturbances since it is not a robust algorithm, which would in turn result in an unfair comparison.

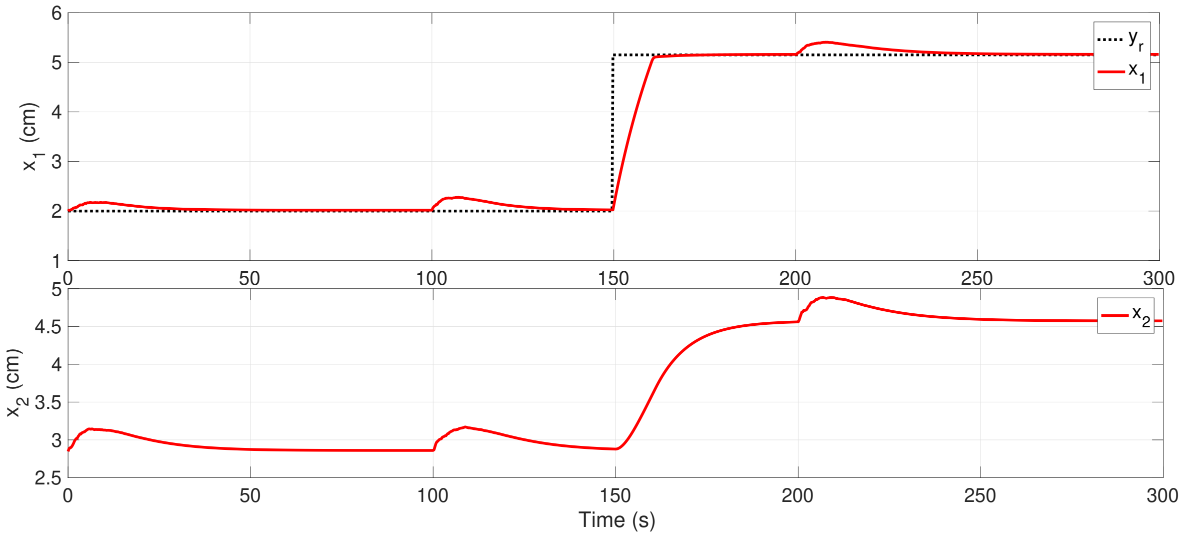


Figure 5.9: Robust performances.

4. Certificates of recursive feasibility and stability are available, which ensures an adequate behaviour of the resulting closed-loop.

- Disadvantages:

1. It requires a qLPV realisation of the nonlinear system, and thus the availability of a known proxy $f_\rho(\cdot)$ that generates bounded scheduling variables (and also abides to the hypothesis given in Assumption 12).
2. As in many Tracking NMPC algorithms, the state variables should be measurable, since the controller guarantees output tracking by steering the states to given steady-state variables.
3. TRPI sets must be computed offline, before the online implementation of the controller, in order to ensure correct behaviours of the resulting closed-loop.

5.6 Final comments

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

Considering a benchmark cascaded tank system, we thoroughly compare the proposed method against the nominal tracking NMPC framework from [Limon et al. 2018]. We are able to demonstrate that the proposed scheme achieved very similar tracking performances, with much smaller computational stress, benefiting from the linear predictions enabled by the qLPV realisation. The method is ready for embedded applications (the online stress is similar to that of a QP) and offers robustness towards bounded load disturbances with reduced conservatism.

Chapter 6

A dissipative approach

In this Chapter, we propose a robust, dissipative MPC scheme for nonlinear systems represented with qLPV models. The main novelty resides in benefiting from the recursive extrapolation approach (Chapter 3, approach (iv)) in order to fasten the (usually) sluggish performances achieved with the robust min-max schemes from the literature. The bounds on the estimation errors of the scheduling parameters through N_p are taken into account in order to formulate an online min-max problem with reduced uncertainties: firstly, a constrained CP is solved in order to determine the worst-case uncertainty propagation level and, subsequently, a second constrained QP is solved to minimise this worst-case cost function with respect to the control sequence vector. We discuss how, since the bounds on the estimation error for the scheduling parameters are usually much smaller than the bounds on the actual scheduling parameter, the conservativeness of the solution is quite reduced. Recursive feasibility and stability of the proposed algorithm are demonstrated with dissipativity arguments given in the form of an LMI remedy, which also determines the zone of attraction within which input-to-state stability is certified. The nonlinear temperature regulation problem of a flat solar collector is considered as a case study. Using a realistic simulation benchmark, the proposed technique is compared to other robust min-max LPV MPC algorithms from the literature, proving itself numerically efficient, whilst maintaining good performances.

Remark 53. *The developments herein presented correspond to those published in the following works: [Pipino et al. 2020b] (solar collector application), [Morato, Normey-Rico, and Sename 2021b] dissipativity arguments, and [Morato, Normey-Rico, and Sename 2021d; Morato, Normey-Rico, and Sename 2023c] (closed-loop induced robustness metrics). We note that in [Morato, Holicki, and Scherer 2023], one can find a novel synthesis formulation for MPC terminal ingredients based on convexly parametrised IQCs and dynamic multipliers.*

6.1 Introduction

In the previous Chapters, we discussed how MPC synthesis can be formulated in both gain-scheduled and robust settings, considering the problem of uncertainty propagation along the prediction horizon. As discussed in Chapter 5, an option to develop fast MPC solutions for nonlinear systems is to exploit qLPV embeddings in order to replace the nonlinear programs by linear ones. There are quite a few MPC algorithms specifically conceived for nonlinear systems represented by LPV realisations, as detailed in Chapters 1 and 2. With regard to this works, we highlight some issues:

- The sub-optimal and gain-scheduled methods, as detailed in Part II of this thesis, have a major drawback in the senses that the resulting QPs may find local minima of the original NP, which conversely may lead to poor or insufficient performances. Despite being able to run in real-time, these algorithms may lack performance certificates for the whole state admissibility regions, which thus compromises the obtained results (this is specially evident for systems with hard state transition nonlinearities).
- In the robust setting, the MPC optimisation is thence written with regard to the worst-case behaviour possibility, that is, considering (bounds of) the propagation of the uncertainty caused by the variation of the scheduling variables along the prediction horizon N_p. While Chapter 5 presented a constraints tightening framework to handle the uncertainty issue, there also exists another synthesis approach with large exploitation in the literature: we refer to the so-called "min-max" robust MPCs, which robustify the controller by first maximising the cost function with regard to the uncertainty (i.e. through $\overline{J_k}(\cdot) := \max_{\rho \in \mathcal{P}} J_k(\cdot)$ subject to constraints), and then minimising this worst-case induced cost $\overline{J_k}(\cdot)$ with regard to the sequence of control inputs U_k. In the original papers on robust min-max LPV MPC schemes [Cao and Lin 2005; Besselmann, Löfberg, and Morari 2009], the future scheduling parameters $\rho(k+j)$ are assumed to vary arbitrarily within the scheduling set $\mathcal{P}$. More recently, many works [Li and Xi 2010; Jungers, Oliveira, and Peres 2011; Bumroongsri 2014] have demonstrated that the min-max procedure can be simplified (and thus fastened) for the case of bounded rates of parameter variations (i.e. $\delta\rho(k+1) = \rho(k+1) - \rho(k)$ being bounded for all $k \geq 0$), which is standard in LPV applications. Nevertheless, we verify major drawbacks:
    1. They are usually not implementable for real-time applications, due to the complexity of solving the maximisation problem for the whole scheduling set $\mathcal{P}$ (or, the scheduling variation set $\delta\mathcal{P}$);
    2. The formulations through offline preparations (such as the tube paradigm) are often quite hard to design; the synthesis procedure is usually not trivial and hard to understand, which hinders industrial acceptance.

We argue that there is gap in the literature regarding this topic: there are only a few robust NMPC approaches, using the qLPV embedding framework, with direct and simple-enough online implementation. Therefore, in this Chapter, we propose a formulation that is able to run in real-time and, yet, maintains optimality concerns, leading to good performance. The bottleneck is that the algorithm should be able to run embedded (operating in real-time, in the range of milliseconds), whilst being able to take into account the model nonlinearities.

Remark 54. *Of course, in Chapter 5 we presented an elegant solution that is indeed able to operate in real-time, while offering robustness. Nevertheless, the constraint tightening mechanism is not simple to implement, and the zonotopic inclusion sets require relatively complex interval algebra. Therefore, the focus herein is cast to min-max algorithms, which are utterly easy to synthesise (arguably, almost as simple as LTI MPC algorithms).*

As debated in Chapter 3, there exist several estimation algorithms that can be been used to provide trajectory guesses for the future values of the scheduling parameters along the prediction horizon, e.g.[Cisneros, Voss, and Werner 2016; Cisneros and Werner 2017a; Morato, Normey-Rico, and Sename 2019]. Again, we exploit the Taylor-based recursive extrapolation from Chapter 3 (approach (iv)), taking into account the boundedness of the estimation, as done in Chapter 5. Herein, these bounds are used to formulate a robust min-max MPC. Motivated by the previous discussion and the literature gap, our contributions are as follows:

- Benefiting from Lemmas 3.6.2 and 4.8.2, we develop a min-max robust qLPV MPC framework, the use of the extrapolation of ρ to make model-based predictions;
- Then, resorting to dissipativity analyses, we employ an IQC to demonstrate input-to-state stability of the resulting closed-loop system. This analysis also serves to verify the recursive feasibility property of the proposed algorithm. An LMI-solvable remedy to estimate the ISS zone is obtained.

6.2 Formalities

In this Section, we provide the main assumptions for the considered robust control proposition. As done in previous Chapters, we consider, once again, a generic discrete-time nonlinear system:

$$x(k+1) \;=\; f\left(x(k), u(k), w(k)\right) , \tag{6.1}$$

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

Assumption 16. *The nonlinear map $f : \mathcal{X} \times \mathcal{U} \times \mathcal{W} \to \mathcal{X}$ is continuous and continuously differentiable with respect to x, i.e. class $\mathcal{C}^\infty$*

Assumption 17. *The (box-type) sets $\mathcal{X}$ and $\mathcal{U}$ define the feasibility constraints for the system states and the control vector, delimited by the operational (physical) limitations of these variables. These sets yield ultimate bounds on x and u, as follows:*

$$\mathcal{X} \;:=\; \left\{x \in \mathbb{R}^{n_x} : \|x_j\| \leq \overline{x}_j, \forall j \in \mathbb{N}_{[1,n_x]}\right\} , \; \mathcal{U} \;:=\; \left\{u \in \mathbb{R}^{n_u} : \|u_j\| \leq \overline{u}_j \, \forall j \in \mathbb{N}_{[1,n_u]}\right\} .$$

Assumption 18. *The set $\mathcal{W}$ defines the load disturbances. For regularity purposes, we consider that $\mathcal{W}$ is a priori an open set.*

Assumption 19. *The states are measurable at all sampling instants $k \in \mathbb{N}$, which means that control can be formulated under a state-feedback law.*

Assumption 20. *This nonlinear system agrees to a local LDI within $(\mathcal{X} \times \mathcal{U})$, which means Eq. (6.1) can be re-written using the following qLPV realisation:*

$$\left\{ \begin{array}{rcl} x(k+1) &=& A(\rho(k))x(k) + B(\rho(k))u(k) + B_w(\rho(k))w(k) \, , \\ \rho(k) &=& f_\rho(x(k), u(k)) \;\in\; \mathcal{P}, \end{array} \right. \tag{6.2}$$

where $f_\rho : \mathcal{X} \times \mathcal{U} \to \mathcal{P} \subset \mathbb{R}^{n_p}$ represents an endogenous nonlinear scheduling proxy. Once again, note that $\rho(k)$ is bounded and known online at each instant k, but generally unknown for any future instant $k + j \,\forall j \in \mathbb{N}_{[1,\infty]}$. Ultimate bounds are considered upon ρ, which are as follows:

$$\mathcal{P} \;:=\; \left\{\rho \in \mathbb{R}^{n_p} \,|\, \|\rho_j\| \leq \overline{\rho}_j \,, \forall j \in \mathbb{N}_{[1,n_p]}\right\} .$$

Remark 55. *The LDI property, as expressed in Assumption 20 is satisfied for closed sets $\mathcal{X}$ and $\mathcal{U}$ and not for the whole vector spaces $\mathbb{R}^{n_x}$ and $\mathbb{R}^{n_u}$. Therefore, any synthesised control law must ensure $x(k) \in \mathcal{X}$ and $u(k) \in \mathcal{U}$ for all $k \geq 0$.*

Remark 56. *Through the sequel, for simplicity, we drop the dependency of f_ρ on u, simply taking $\rho(k) = f_\rho(x(k))$. Nonetheless, we stress that all developments presented in the sequel can be easily extended to broader case. For the sake of simplicity, we also drop the parameter dependency from B_w, i.e. $B_w(\rho(k)) = B_w$. Note that this can always be done via the inclusion of the parameter dependency into the load disturbance signal, e.g. $B_w(\rho(k))w(k) = B_{w_2}w_2(k)$ with $w_2(k) = f_w(\rho(k), w(k))$.*

Complementary, we consider that Assumption 21 is satisfied[1]. This Assumption does not compromise the proposed approach; it serves only to analytically account for model-process mistmatch uncertainties. We also consider that the qLPV embedding of Eq. (6.2) satisfies Assumptions 22 (local Lipschitz property of $f_\rho(\cdot)$), 23 (bounded rates of variation for ρ) and 24 (stabilisability).

Assumption 21. *Matrices $A(\rho(k))$ and $B(\rho(k))$ are affine-dependent on $\rho(k)$, as in: $A(\rho(k)) = A_0 + A_1\rho(k)$ and $B(\rho(k)) = B_0 + B_1\rho(k)$.*

Assumption 22. *The nonlinear scheduling parameter map $f_\rho : \mathcal{X} \to \mathcal{P}$ agrees to a local Lipschitz condition around any arbitrary point $x \in \mathcal{X}$, this is:*

$$\|f_\rho(x) - f_\rho(\hat{x})\| \;\leq\; \gamma_\rho \|(x - \hat{x})\| \,, \forall\, x \in \mathcal{X} \,, \forall\, \hat{x} \in \mathcal{X} \,, \tag{6.3}$$

where the smallest constant γ_ρ that satisfies Eq. (6.3) is known as the Lipschitz constant for $f_\rho(\cdot)$.

Assumption 23. *The deviation of the scheduling parameters is bounded, i.e. $\delta\rho(k) = (\rho(k) - \rho(k-1)) \in \delta\mathcal{P}, \forall k \in \mathbb{N}$.*

Assumption 24. *The open-loop qLPV model $(A(\rho(k)), B(\rho(k)))$ is structurally stabilizable for all $\rho \in \mathcal{P}$.*

6.2.1 Model-based predictions and parameter extrapolation

Since the qLPV embedding in Eq. (6.2) retains the linearity property from inputs to outputs, we are able to fomulate numerically-efficient design procedures using these models. As debated, while LPV control is standard in both state-feedback and dynamics output-feedback

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

formulations[Mohammadpour and Scherer 2012; Shamma 2012; Sename, Gaspar, and Bokor 2013; Scorletti, Fromion, and De Hillerin 2015], the design predictive control algorithms for LPV systems is not trivial, since solving the inherent constrained $\arg\min_{U_k} J$ optimisation problem requires the knowledge of future values for the scheduling parameter.

As done in [Cisneros, Voss, and Werner 2016] and detailed in Chapter 3, we write the state predictions within a prediction horizon window of N_p steps using Eq. (3.8). Moreover, the scheduling trajectory is generated using an extrapolated guess $\hat{P}_k$ with bounded estimation errors (i.e. approach (iv) in Chapter 3). That is: $\xi_\rho(k+j|k) = (\rho(k+j) - \rho(k+j|k))$, $\forall j \in \mathbb{N}_{[0,N_p-1]}$. This estimation residual is bounded to a a convex set $\mathcal{Q} := \{\xi_\rho \in \mathbb{R}^{n_\rho} \,|\, \|\xi_\rho\| \leq \left(\gamma_\rho + \overline{f_\rho^\partial}\right) \overline{\delta x}\} = \xi_\rho^{\text{bound}}$ (thanks to Lemma 4.8.2).

Remark 57. *Generally, we obtain horizon-increasing extrapolation errors, i.e.* $\|\xi_\rho(k+j+1|k)\| \geq \|\xi_\rho(k+j|k)\|$ *for* $j \in \mathbb{N}_{[0,N_p-2]}$*, due to the fact the more information is available regarding the present instant than the future ones, which depend on future variables which haven't yet been defined (see Remark 20). Yet, as recalled next, we can use these estimates to generate the model-based predictions since* $\lim_{k\to+\infty} \xi_\rho(\cdot|k) \to 0$ *(Lemma 4.8.2). We stress that* $\xi_\rho^{bound} < \overline{\rho}$ *and, thus,* $\mathcal{Q} \subset \mathcal{P}$*.*

6.2.2 The generated sub-optimal MPC

Considering the qLPV prediction model from Eq. (3.8), obtained using the extrapolate scheduling trajectory estimate from Eq. (3.43), we detail the corresponding MPC application. For such, the following finite-horizon cost is considered:

$$J_k = J(x(k), U_k) = \left(\sum_{j=1}^{N_p} (\ell(x(k+j), u(k+j-1))) \right) , \tag{6.4}$$

where $\ell(x,u) := x^T Q x + u^T R u$ is the main (qadratic) stage cost. Note that, in this Chapter, we disregard the use of a terminal cost $V(\cdot)$ (refer to the discussions on stabilising MPC schemes without terminal ingredients in Chapter 2, Section 2.7.2).

If the extrapolation error is disregarded, a sub-optimal gain-scheduled MPC algorithm can be formulated, which would reside in solving the following constrained QP:

$$\min_{U_k} \quad J_k \tag{6.5}$$

$$\text{subject to} \quad X_k = \mathcal{A}(\hat{P}_k)x(k) + \mathcal{B}(\hat{P}_k)U_k \,, \tag{6.6}$$

$$x(k+j) \in \mathcal{X} \,,\, \forall j \in \mathbb{N}_{[1,N_p]} \,, \tag{6.7}$$

$$u(k+j-1) \in \mathcal{U} \,,\, \forall j \in \mathbb{N}_{[1,N_p]} \,. \tag{6.8}$$

As thoroughly debated in Chapters 3 and 4, such QP can lead to insufficient performances because local minima can be found since, although the qLPV embedding equivalently represents the nonlinear dynamics, the scheduling parameters are (partially) uncertain along the horizon due to the residual errors $\xi_\rho(k+j|k)$ Therefore, we proceed by adapting it in order to take into account the bounds on the estimation error of the scheduling sequence, such that performance certificates can be provided.

Along this Chapter, we use the Hessian notation from [Petsagkourakis, Heath, and Theodoropoulos 2020], that is: the previous QP is be re-stated as follows:

$$\begin{aligned} U_k^\star &= \arg\min_{U_k} \left(\frac{1}{2} U_k^T H(\hat{P}_k) U_k - U_k^T g(\hat{P}_k, x(k)) \right) , \\ \text{s.t.} \quad & A_{ineq} U_k \leq b_{ineq}(k) , \\ & C_{ineq} U_k = 0 , \end{aligned} \tag{6.9}$$

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

We recall that he MPC policy that results from the online solution of Eq. (6.9) is generated under a paradigm of a moving-window horizon, which slides along k as time evolves. This means that that at instant k the control sequence $U_k^\star$ is computed considering the system behaviour within the next N_p steps. At the following instant, $k+1$, the problem $\min J_{k+1}$ is solved considering the performances for N_p samples ahead of $k+1$, computing $U_{k+1}^\star$, and so forth. The control policy at each instant is the first entry of the solution the QP, this is:

$$u(k) = \overbrace{\left[\begin{array}{cccc} I_{n_u} & 0_{n_u} & \dots & 0_{n_u} \end{array}\right]_{n_u \times (n_u N_p)}}^{\mathcal{I}_1} U_k^\star = \kappa(k) x(k) . \tag{6.10}$$

6.2.3 Terminal ingredients and dissipativity constraints

In this Chapter, we use the concept of ISS in order to provide performance certificates. Specifically, we employ the generalised ISS condition for discrete-time nonlinear systems from [Jiang and Wang 2001] (Definition 2.6). We are concerned with ISS since the considered MPC generates a state-feedback control law, which means that the states should be stabilised. Recent results regarding ISS and ISpS, which is a weaker property[2], have been presented regarding min-max nonlinear MPCs, see [Limón et al. 2006a; Lazar et al. 2008; He, Huang, and Chen 2014]. In general, many robust MPC methods are not able ensure ISS (but simply ISpS) because the effect of non-null disturbance inputs is taken into account by the procedure even if the disturbance vanishes in reality. Anyhow, [Magni, Raimondo, and Scattolini 2006] demonstrates that only a local upper bound on the MPC cost function J_k (instead of a global one, which is more costly to demonstrate) is sufficient to ensure ISS. In this Chapter, we build from these previous results, specially concerning the feasibility property of the maximisation procedure.

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

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

logical. As detailed in Chapter 2, in order to verify that the MPC algorithm ensures closed-loop ISS and recursive feasibility of the optimisation procedure, we could require the so-called "terminal ingredients", as done in Chapter 5. Yet, we use a different approach herein: we verify dissipativity arguments regarding the MPC value function, as done in [Seiler, Packard, and Balas 2010]. The main characteristic of this approach is that LMIs are yielded for verification *a posterior* to MPC synthesis (the dissipativity arguments are not included to optimisation itself). This is the path followed in this Chapter, pursuing the advances from previous works [Seiler, Packard, and Balas 2010; Cisneros and Werner 2018].

6.3 Proposed min-max algorithm

Considering that $\xi_\rho(k+j|k) \in \mathcal{Q} \subset \mathcal{P}$, the sub-optimal MPC algorithm in Eq. (6.5) is now adapted in order to ensure robustness. Recall that we seek performances guarantees despite the uncertainties introduced by the scheduling sequence estimation error.

As previously discussed, solving a single QP with respect to a scheduling sequence guess as in Eq. (6.9) does not ensure performances, since the solution $U_k^\star$ may represent a local minima of J_k. Anyhow, we know that the actual nonlinear process model in Eq. (6.1) differs from the $\hat{P}_k$-based prediction in Eqs. (3.8)-(3.43) due to the discrepancy variable ξ_ρ. These model-process mismatches along the horizon can be treated robustly, providing a worst-case bound $J_k^{\text{bound}} > J_k$. Then, as done in robust min-max LPV MPC procedures, the QP is formulated with respect to J_k^{bound}, ensuring the overlap of local minima and robust performances.

Based on Assumption 21, we can thus expand the LPV model along the prediction horizon using:

$$\begin{aligned} x(k+j+1) &= A(\rho(k+j|k))x(k+j) + B(\rho(k+j|k))u(k+j) + B_w w(k+j) \\ &+ \underbrace{(A_1\xi_\rho(k+j|k)x(k+j) + B_1\xi_\rho(k+j|k)u(k+j))}_{\sigma(k+j|k)} . \end{aligned} \tag{6.11}$$

The uncertainties introduced due to the model-process mismatch (extrapolation of the scheduling sequence) are denoted henceforth as $\sigma(k+j|k)$, which belong to a compact set $\mathcal{S}$ whose bounds can be computed offline, with respect to ξ_ρ^{bound}, $\mathcal{X}$ and $\mathcal{U}$:

$$\mathcal{S} := \{\sigma \in \mathbb{R}^{n_x} \,|\, \|\sigma\| \leq \overline{\sigma}\} . \tag{6.12}$$

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

$$\Xi_k = \begin{bmatrix} \sigma(k|k)^T & \dots & \sigma(k+N_p-1|k)^T \end{bmatrix}^T . \tag{6.13}$$

Remark 58. *In regular min-max LPV MPC algorithms, e.g. [Cao and Lin 2005], $\mathcal{S}$ is computed with respect to much larger possible variations for ρ (usually, the whole set $\mathcal{P}$). In works [Bumroongsri 2014] that consider bounded rates of variations for ρ, the uncertainty set $\mathcal{S}$ is computed as if $\rho(k+j) = \rho(k) \pm j\delta\rho(k)$, which yields a smaller uncertainty set $\mathcal{S}$, but in general also larger than the set considered in this Chapter.*

Embedding the uncertainty to the process predictions, we obtain:

$$X_k = \mathcal{A}(\hat{P}_k)x(k) + \mathcal{B}(\hat{P}_k)U_k + \Xi_k \,. \tag{6.14}$$

Then, the core idea of the proposed method is quite simple, following the lines of the original min-max algorithms, but formulating the worst-case cost function with respect to the uncertainties introduced by the estimation errors ξ_ρ. As an abstraction[3], we can understand that there exists some $\Xi_k^\star$ which induces the worst-case bound on the cost function J_k^{bound}, i.e:

$$\begin{aligned} \max_{\Xi_k} \quad & J_k \\ \text{subject to} \quad & \text{constraints from Eqs. (6.6), (6.7) and (6.8)}\,. \end{aligned} \tag{6.15}$$

The complete solution achieved with the proposed tool is then given by; $U_k^\star = \arg\min_{U_k} \max_{\Xi_k} J_k$ subject to constraints (6.6)-(6.8). In our analyses, we proceed by viewing this min-max procedure as two steps, the maximisation CP with respect to Ξ_k and the minimisation with respect to U_k (QP).

6.3.1 Implementation

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

1. Offline Procedure:
   - Firstly, one should verify if the considered nonlinear process should satisfies Assumptions 16 to 20.
   - LDI should be performed, finding the qLPV model as given in Eq. (6.2);
   - Assumptions 21 to 24 should be checked;
   - The smallest Lipschitz constant γ_ρ in Eq. (22) should be defined and so should the bounds on Δx;
   - With the aid of simulation tools, the recursive extrapolation algorithm of Eq. (3.43) should be tested and the forgetting factors λ and ν should be adequately tuned ;
   - The worst-case bound on the estimation error should be computed as $\xi_\rho^{\text{bound}} = (\gamma_\rho + \overline{f_\rho^\partial})\overline{\delta x}$ (thanks to Lemma 4.8.2);
   - The compact uncertainty set $\mathcal{S}$ due to the wrong scheduling guess should be calculated using Eq. (6.12);
   - The MPC procedure should be prepared by tuning the cost weighting matrices Q and R;

[3]We emphasise that this is an argumentation abstraction only, given that the min-max operation is a single optimisation. Note that the vector of future control inputs U_k, decision variable to the minimisation problem, also influences the maximisation argument.

- Te nominal cost function J_k should be put in the Hessian form of Eq. (6.9).

2. Online Procedure: solve Algorithm 4.

Algorithm 4 Proposed robust min-max NMPC Algorithm

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

- Step (1): Measure the states $x(k)$ and get the scheduling parameters $\rho(k)$;
- Step (2): Evaluate the derivative f_ρ^∂;
- Step (3): Compute the extrapolation of the scheduling parameters along the horizon through Eq. (3.43);
- Step (4): Solve the min-max problem $\min_{U_k} \max_{\Xi_k} J_k$ subject to constraints (6.6)-(6.8), thus obtaining $U_k^\star$;
- Step (5) Apply the local control policy $u(k)$ as in Eq. (6.10);
- Step (6): Increment k, i.e. $k \leftarrow k + 1$.

end

6.4 Certificates

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

We stress than an abstraction is used: in practice, we solve a single min-max problem $\min_{U_k} \max_{\Xi_k} J_k$ subject to constraints (6.6)-(6.8), thus obtaining $U_k^\star$; yet, for our analyses, we decouple the maximisation CP and the minimisation QP. Accordingly, we consider $\Xi_k^\star$ as the maximiser of $\max_{\Xi_k} J_k$ subject to constraints (6.6)-(6.8), which, plugged to the minimisation problem $\min_{U_k} J_k$ using the model in Eq. (6.14) implies in the worst-case cost minimisation, i.e. $\min_{U_k} J_k^{\text{bound}}$.

6.4.1 The maximisation CP

In order to demonstrate the recursive feasibility property of the maximisation problem, we follow closely discussions of previous works [Magni, Raimondo, and Scattolini 2006; Limón et al. 2006a; Lazar et al. 2008].

Remark 59. *In these previous papers, ISS and ISpS properties are verified for the whole min-max CP through the use of terminal ingredients. In this Chapter, we follow a dissipativity formulation, since we do not make use of RPI sets as terminal constraints nor of terminal stage costs in our formulation. Anyhow, the analysis of recursive feasibility of the maximisation step can be maintained.*

Remark 60. *Some of the following steps are easier to follow if the weak duality property of CPs is considered [Löfberg 2012]: an maximisation CP can be equivalently written as a minimisation CP over the same variables with adjusted slack variables.*

The considered maximisation argument $\max_{\Xi_k} J_k(\cdot)$ subject to the inequality constraints (6.6)-(6.8), and based on the available scheduling sequence guess $\hat{P}_k$, considers Ξ_k as the sequence of uncertainties along the horizon, as gives Eq. (6.13). Note that it can be adequately re-written in a generalised formulation with respect to Ξ_k, this is:

$$\max_{\Xi_k} \left(\frac{1}{2} \Xi_k^T H_\sigma \Xi_k - \Xi_k^T g_\sigma(\hat{P}_k, x(k)) \right) . \tag{6.16}$$

Replacing X_k in Eq. (6.16), we obtain[4] $H_\sigma = 2\breve{Q}$ and $g_\sigma = -2\breve{Q}\left(\mathcal{A}(\hat{P}_k)x(k) + \mathcal{B}(\hat{P}_k)U_k\right)$.

Remark 61. *Note that, if constraints are disregarded, for rationale purposes only, it is direct to evaluate that the maximal value for J_k, with respect to Ξ_k would be found with $\Xi_k^\star = (H_\sigma)^{-1} g_\sigma$. Since g_σ is linear over U_k, the value for U_k that maximises J_k is $\overline{U} = col\{\overline{u}\}$ (a sequence of maximal control signals), e.g.: $\Xi_k^\star = -(2\breve{Q})^{-1}\left(\mathcal{A}(\hat{P}_k)x(k) + \mathcal{B}(\hat{P}_k)\overline{U}\right)$. Regarding the CP constraints, it follows that* (6.8) *adds no difference to this possible result. Moreover, constraints* (6.7) *and* (6.8) *are only box-type operations over x and u, respectively. Therefore, it follows directly that for any starting condition x_0 within the feasibility set $\mathcal{X}$, this maximisation CP is recursively feasible since J_k is never be unbounded with respect to Ξ_k due to its regular quadratic formulation on Ξ_k, operated through Eq.* (6.16). *Nonetheless, this property only remains true if and only if Q^{-1} exists, since $(H_\sigma)^{-1} = (2\breve{Q})^{-1}$.*

In order to demonstrate the recursive feasibility property of the CP in Eq. (6.16), we consider that:

- The minimisation QP is also feasible. This is quite logical because the min-max formulation resides in the operation of both these CPs consecutively.
- The stage cost $\ell(\cdot)$ is lower bounded for all $x \in \mathcal{X}_{\text{Max CP}}$ (this set denotes the feasibility region for the maximisation CP). Indeed, it follows directly from Eqs. (6.4) that:

$$\ell(\cdot) \quad \geq \quad a_\ell(\|x\|) \, , \tag{6.17}$$

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

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

Lemma 6.4.1. *Based on these previous conditions, of a feasible minimisation QP and of Eq.* (6.17)*, it follows that the worst-case cost function* J_k^{bound}*, computed with respect to* $\Xi_k^\star$*, is upper-bounded, considering that the uncertainties are described as of Eq.* (6.12)*, such that:*

$$J_k^{bound} \quad \leq \quad \alpha_J(\|x\|) + \beta_J(\overline{\sigma}) \, , \tag{6.18}$$

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

Proof. Refer to [Morato, Normey-Rico, and Sename 2021d, Appendix C]. □

Remark 62. *Since the previous Lemma requires the min. QP to be recursively feasible, it is implied that* $\mathcal{X}_{Max\ CP} := \mathcal{X}_{Min\ QP}$*, where* $\mathcal{X}_{Min\ QP}$ *is the feasibility set of the second CP. For simplicity, we henceforth denote the maximisation CP through an abstraction: we use an operator notation (*Υ*) applied over the states:* $\Xi_k^\star := \Upsilon(\hat{P}_k)x(k)$*.*

6.4.2 The minimisation QP

The analysis of the ISS property of the minimisation QP is more complex. This CP solves $U_k^\star = \arg\min_{U_k} J_k(\cdot)$ subject to constraints (6.6)-(6.8), based on the available scheduling sequence guess $\hat{P}_k$ and on the uncertainty vector $\Xi_k^\star$. Figure 6.1 gives a graphical block-diagram interpretation of the system, considering both CPs (6.5) and (6.15) and the extrapolation algorithm, where G represents the open-loop LPV embedding of Eq. (6.2).

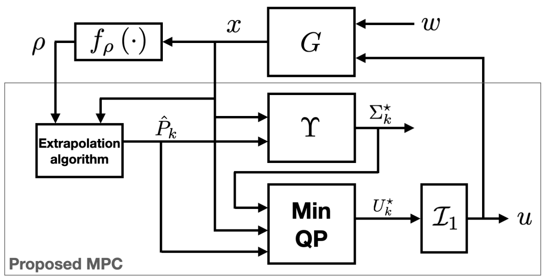


Figure 6.1: Graphical representation: qLPV-embedded nonlinear system and proposed algorithm.

Next, we replace X_k from Eq. (6.14) in Eq. (6.9), which leads to the following Hessian and linear term[5]:

$$H(\hat{P}_k) \quad = \quad 2\left(\breve{R} + \mathcal{B}(\hat{P}_k)^T \breve{Q} \mathcal{B}(\hat{P}_k)\right) \, , \tag{6.19}$$

$$g(\hat{P}_k, x(k)) \quad = \quad -2\mathcal{B}(\hat{P}_k)^T \breve{Q} \left(\mathcal{A}(\hat{P}_k)x(k) + \Xi_k^\star\right) \, . \tag{6.20}$$

Then, in order to verify ISS, we proceed by defining a nonlinear static map $\phi : g \to U_k^\star$ implied by the constrained minimisation QP in its regular form of Eq. (6.9). Specifically,

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

we use the formulation from [Heath, Wills, and Akkermans 2005; Petsagkourakis, Heath, and Theodoropoulos 2017]:

Lemma 6.4.2 (Sector bound [Petsagkourakis, Heath, and Theodoropoulos 2017])**.** *The nonlinearity $u^\star(k) = \mathcal{I}_1 U_k^\star$, which renders the MPC law via the optimisation in Eq.* (6.9)*, belongs to the sector* $[0, H(\hat{P}_k]$*,* $\forall \rho(k+j|k) \in \mathcal{P}$*.*

Proof. Consider the nonlinear static map $\phi_k : g(\hat{P}_k, x(k)) \rightarrow U_k^\star$ implied by the MPC constrained minimisation program from[6] Eq. (6.9). Then, the existence of a feasible control sequence $U_k^\star$ is ensured if the Karush-Kuhn-Tucker (KKT) conditions of Eq. (6.9) hold:

$$\begin{cases} H(\hat{P}_k)U_k^\star - g_k + A_{\text{in}}^T \lambda &= 0\,, \\ \tilde{\lambda}_j \left(A_{\text{ineq}_j} U_k^\star - b_{\text{ineq}(k)_j} \right) &= 0\,, \\ \tilde{\lambda}_j &\geq 0\,. \end{cases} \tag{6.21}$$

where $\tilde{\lambda}$ being the Lagrange multiplier. Multiplying the first KKT condition by U_k^T, being $U_k^\star = \phi_k$, leads to the following inequality, which should hold for all $\rho(k+j|k) \in \mathcal{P}$:

$$\phi_k^T H(\hat{P}_k)\phi_k - \phi_k^T g_k \quad \leq \quad 0\,, \forall g_k\,. \tag{6.22}$$

This concludes the proof. □

Note that the inequality argument in Eq. (6.22) derives directly from the Lagrange-KKT conditions of the QP, represented as a sector-bounded nonlinearities with regard to each scheduling parameter (thus, herein named parameter-dependent). Further discussions on this matter are available in the original paper [Heath, Wills, and Akkermans 2005].

Regarding Ineq. (6.22), we present a graphical interpretation of the considered system in Figure 6.2. The proposed MPC policy is divided by the upper ϕ block, which comprises the minimisation QP, and by the lower Υ block, which embeds the maximisation CP. Regarding, Figure 6.1, the output of the minimisation QP $U_k^\star$ is now replaced by the nonlinear operator ϕ_k. Moreover, the main open-loop process in Figure 6.2 is represented by $G_{\mathcal{I}_1}$, which is a compacted operator comprising the open-loop plant and matrix $\mathcal{I}_1$, since $u(k) = \mathcal{I}_1 U_k^\star$. It follows that $x := G(u, w)$, $u := \mathcal{I}_1 \phi$, and, thus, $x := G_{\mathcal{I}_1}(\phi, w)$.

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

$$\begin{bmatrix} 0 & 0 \\ 0 & H(\hat{P}_k) \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & \mathcal{B}(\hat{P}_k) \\ 0 & I \end{bmatrix}^T}_{H_P(\hat{P}_k)^T} \underbrace{\begin{bmatrix} 2\breve{Q} & 0 \\ 0 & 2\breve{R} \end{bmatrix}}_{H_0} \underbrace{\begin{bmatrix} 0 & \mathcal{B}(\hat{P}_k) \\ 0 & I \end{bmatrix}}_{H_P(\hat{P}_k)}\,. \tag{6.23}$$

[6] Seeking notation compactness, we use $g_k = g(\hat{P}_k, x(k))$ and $\phi_k = \phi\left(g(\hat{P}_k, x(k))\right)$.

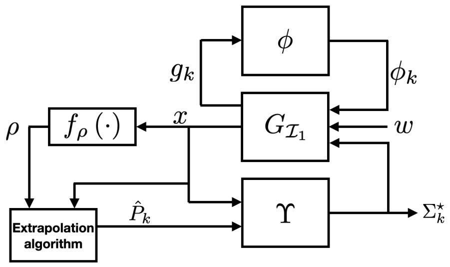


Figure 6.2: Graphical representation: Closed-loop system.

Based on the prior factorisation, we are able to re-write inequality (6.22) as follows:

$$[\ *\ ]^T \underbrace{\left(\left[\begin{array}{c|c} 0 & 0 \\ \hline -I & 0 \end{array}\right] + H_P(\hat{P}_k)^T H_0 H_P(\hat{P}_k)\right)}_{\Pi(\hat{P}_k)} \left[\begin{array}{c} g_k \\ \phi_k \end{array}\right] \leq 0\,. \tag{6.24}$$

As provided in previous works [Megretski and Rantzer 1997; Scherer 2001], the above parameter-dependent quadratic constraint can be cast into a regular multiplier form $z_g^T M_z z_g \leq 0$, where z_g is the output of a bounded linear operator $\Psi(\hat{P}_k)$ which factorises $\Pi(\hat{P}_k)$, this is: $\Pi(\hat{P}_k) = \left(\Psi(\hat{P}_k)\right)^* M_z \Psi(\hat{P}_k)$. The operator $\Pi(\cdot)$ stands for the "filling" of the previous inequality (6.24). Thus, we continue by using the previous factorisation to write $\Pi(\cdot)$ in a multiplier form, given as follows:

$$\Pi(\hat{P}_k) \;=\; \Psi^*(\hat{P}_k) \underbrace{\left[\begin{array}{c|c} 0 & \left[\begin{array}{cc} 0 & 0 \end{array}\right] \\ \hline \left[\begin{array}{c} 0 \\ -I \end{array}\right] & H_0 \end{array}\right]}_{M_z} \underbrace{\left[\begin{array}{c} \left[\begin{array}{cc} I & 0 \end{array}\right] \\ \hline H_P(\hat{P}_k) \end{array}\right]}_{\Psi(\hat{P}_k)}. \tag{6.25}$$

Figure 6.3 gives a graphical interpretation of the extraction of parameter-dependency through Ψ. It follows that the multiplier form of $\Pi(\cdot)$ is built with:

$$\left\{\begin{array}{rcl} M_z & = & \left[\begin{array}{c|cc} 0 & 0 & 0 \\ \hline 0 & 2\breve{Q} & 0 \\ -I & 0 & 2\breve{R} \end{array}\right], \\ \Psi(\hat{P}_k) & = & \left[\begin{array}{cc} I & 0 \\ \hline 0 & \mathcal{B}(\hat{P}_k) \\ 0 & I \end{array}\right]. \end{array}\right. \tag{6.26}$$

From the previous development, we express the dissipativity inequality (6.22) simply as:

$$[\ *\ ]^T \Psi(\hat{P}_k)^T M_z \Psi(\hat{P}_k) \left[\begin{array}{c} g_k \\ \phi_k \end{array}\right] \leq 0\,, \tag{6.27}$$

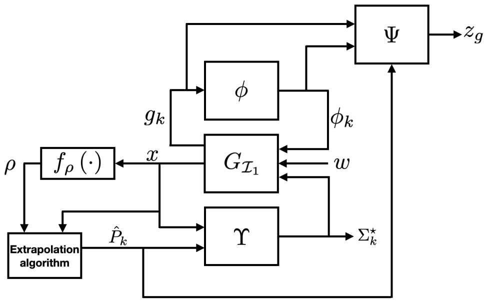


Figure 6.3: Graphical representation: Parameter-dependency extraction.

which is can be compacted as:

$$\left[\ *\ \right]^T M_z z_g \quad \leq \quad 0 \,\forall \hat{P}_k \in \breve{\mathcal{P}}\,, \tag{6.28}$$

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

The parameter-dependency has been dropped through the previous factorisation procedures. Therefore, we can perform an LFT to extract the LPV scheduling parameter dependency as an upper Θ-block (which is connected to an LTI nominal block). Considering z_g as an output the lifted system, we graphically illustrate the LFT in Figure 6.4, where $G^n_{\mathcal{I}_1}$ is an LTI nominal model of the augmented plant, as follows:

$$G^n_{\mathcal{I}_1} \; := \left\{ \begin{array}{lclclclcl} x(k+1) & = & A^n x(k) & + & B^n_w w(k) & + & B^n_\phi \phi_k & + & B^n_\rho u_\rho(k) \\ y_\rho(k) & = & C^n_\rho x(k) & + & D^n_{\rho,\phi}\phi_k & + & D^n_\rho u_\rho(k) & & \\ z_g(k) & = & C^n_z x(k) & + & D^n_{z,\phi}\phi_k & + & D^n_{z,\rho} u_\rho(k) & & \end{array} \right. , \tag{6.29}$$

where $u_\rho(k) := \Theta y_\rho(k)$ makes the interconnection between this nominal LTI block and the LPV-lifted upper Θ-block and $\Xi^\star_k$ appears now as an input to the $G^n_{\mathcal{I}_1}$ block, since it is present in g_k as gives Eq (6.20).

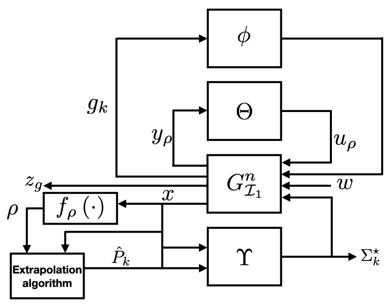


Figure 6.4: Graphical representation: Closed-loop with ρ-dependency extracted.

Finally, in order to check if the system is ISS, it remains to verify the following Lemma, adapted from [Cisneros and Werner 2018], which ensures that the lower $G^n_{\mathcal{I}_1}$-Θ block is stable despite the upper ϕ transfer.

Definition 6.1 (D/G Scalings, as gives [Cisneros and Werner 2018])
Let $\Theta = diag\{\rho_1(k)I_{size\{\rho_1\}}, \ldots, \rho_{n_p}(k)I_{size\{\rho_{n_p}\}}\}$*, with* $\rho_j \in \mathbb{R}$*. Accordingly, the set of D/G-Scalings is defined as follows:*

$$\mathcal{M}_{D/G} = \left\{ \begin{bmatrix} M_1 & M_2 \\ \star & -M_1 \end{bmatrix} : M_1 = M_1^T \succ 0, M_2 + M_2^T = 0, M_1\Theta = \Theta M_1, M_2\Theta = \Theta M_2 \right\} .$$

Lemma 6.4.3 (Adapted from [Cisneros and Werner 2018])**.** *The closed-loop system given in the LFT form in Eq.* (6.29)*, regulated under the proposed min-max MPC law in the form of* $U_k^\star = \arg\min_{U_k} J_k(\cdot)$ *subject to constraints* (6.6)-(6.8) *and based on the available scheduling sequence guess* $\hat{P}_k$ *and on the uncertainty* $\Xi_k^\star$*, is quadratically stable, verifying the dissipativity inequality* (6.22)*, if there exists a positive-definite matrix* $P = P^T > 0$ *and a constant* $\tau > 0$ *such that:*

$$\begin{bmatrix} (A^n)^T P A^n - P & (A^n)^T P B_\phi^n & (A^n)^T P B_\rho^n \\ \star & (B_\phi^n)^T P B_\phi^n & (B_\phi^n)^T P B_\rho^n \\ \star & \star & (B_\rho^n)^T P B_\rho^n \\ \star & \star & \star \end{bmatrix} - \tau \Pi_\phi + \Pi_\Theta \prec 0 \tag{6.30}$$

where

$$\Pi_\phi = [*]^T \left[\begin{array}{c|cc} 0 & 0 & 0 \\ \hline 0 & 2\breve{Q} & 0 \\ -I & 0 & 2\breve{R} \end{array}\right] \begin{bmatrix} C_z^n & D_{z,\phi}^n & D_{z,\rho}^n \end{bmatrix} , \tag{6.31}$$

$$\Pi_\Theta = [*]^T M_\Theta \begin{bmatrix} C_\rho^n & D_{\rho,\phi}^n & D_\rho^n \\ 0 & 0 & I \end{bmatrix} , \tag{6.32}$$

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

Proof. Consider the existence of a quadratic storage function $V = x^T P x$. From LMI (6.30), we obtain:

$$\begin{aligned} & (V(k+1) - V(k)) - \left(\tau z_g^T M_z z_g\right) \\ + & \left(\begin{bmatrix} u_\rho(k) \; y_\rho(k) \end{bmatrix}^T \begin{bmatrix} I \\ 0 \end{bmatrix}^T M_\Theta \begin{bmatrix} I \\ 0 \end{bmatrix} \begin{bmatrix} u_\rho(k) \; y_\rho(k) \end{bmatrix} \right) < 0 . \end{aligned} \tag{6.33}$$

Leveraging from the negativeness of the supply rate term $(z_g^T M_z z_g)$, M_Θ is implied as non-negative. Therefore, for any $\tau > 0$, it follows that $V(k+1) - V(k) < 0$, which means that the propose storage function is a Lyapunov function for the system and, thus, for any starting condition $x_0 \in \mathcal{X}_{ISS}$, local asymptotical stabilisation to origin of the state-space is ensured by the MPC policy. For full details refer to to [Morato, Normey-Rico, and Sename 2021d, Appendix D]. □

The positive definite matrix P found through Lemma 6.4.3 defines the following set:

$$\mathcal{X}_{\text{Min CP}} := \left\{ x \in \mathbb{R}^{n_x} \,|\, x^T P x \le 1 \right\} . \tag{6.34}$$

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

$$\mathcal{X}_{ISS} \; := \; \mathcal{X}_{\text{Max CP}} \cap \mathcal{X}_{\text{Min QP}} \,. \tag{6.35}$$

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

6.5 Application example

In this Section, we present a nonlinear case-study for which the proposed MPC method is applied. As discussed in energy systems literature, e.g. [Camacho et al. 2012; Pasamontes et al. 2013; Morato et al. 2020e], the addition of renewable energy sources to power plants can be a good route to reduce greenhouse gas emissions and environmental impact. Anyhow, an inherent problem to be solved is how to integrate these energy sources without loosing efficiency and dispatchability of energy plants.

6.5.1 Solar-thermal system, phenomenological model, and control problem

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

The focus of the control system is solely to the regulate the temperature of the ST collector panel. Accordingly, the collector outlet flow temperature signal must track a constant steady-state reference, despite instaneous variations on the solar irradiance or on the external temperature. Figure 6.5 illustrates the considered ST system.

Complete phenomenological models have previously been derived for ST collector fields, e.g. [Pasamontes et al. 2013], with according model-validation provided in [Ampuño et al. 2019]. These models are derived on the basis of the following set of assumptions:

1. The fluid flow through the solar collector is incompressible (with density ϵ_f), with uniform pressure along the field; the heat transfer capacity of the fluid is constant and denoted C_f;

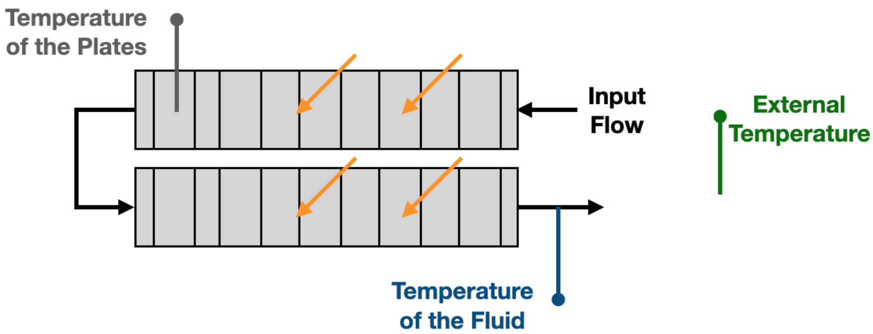


Figure 6.5: Schematic illustration of a solar-thermal collector field.

2. The heat transfer capacity of the collector plates is constant and denoted C_m; the density of these metal plates is also constant and denoted ϵ_m;

3. The balance of energy equations assume a constant thermal loss coefficient ν, with respect to the thermal energy that derives from the incident solar radiance;

4. The heat transfer coefficient of the absorver (external temperature to plates), denoted h_0, is constant, while the heat transfer coefficient of the fluid (fluid to plates), denoted $h_i(\cdot)$, varies positively according to the temperature of the plates.

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

$$\left\{ \begin{array}{rcl} \epsilon_m C_m A_e \frac{dT_p}{dt}(t) & = & -d_e \pi h_0 (T_p(t) - T_e(t)) - d_i \pi h_i(T_p(t))(T_p(t) - T_f(t)) \\ & + & d_e \pi \nu I(t)\,, \\ \epsilon_f C_f A_i \frac{\partial T_f}{\partial t}(t,s) & = & -u(t)\epsilon_f C_f \frac{\partial T_f}{\partial s}(t,s) + d_i \pi h_i(T_p(t))(T_p(t) - T_f(t))\,. \end{array} \right.$$

In these temperature gradient dynamics of Eq. (6.36), $I(t)$ stands for solar radiance focused upon the collectors (which is a load disturbance from a control viewpoint); T_p, T_e and T_f are, respectively, the collector plate, the external (load disturbance as well) and the fluid temperatures; u is the inlet fluid flow, which is the control input of the system; finally, A_i and A_e are, respectively, the internal and external surfaces of the pipes, that have (internal and external) diameters of d_i and d_e.

For application purposes, the space-derivative term $\frac{\partial T_f}{\partial s}(t,s)$ can be replaced, as debated in [Pasamontes et al. 2013; Ampuño et al. 2019; Pipino et al. 2020b], by either a nonlinear function or an apparent transport delay. In this Chapter, it is approximated by the following nonlinearity:

$$\frac{\partial T_f(t,s)}{\partial s} \approx \frac{1 - e^{-\frac{T_f(t)}{T_f^{\max}}}}{(1 - e^{-1})}\,, \tag{6.36}$$

which means that the diffusion of the thermal energy of the fluid flowing along the flat collectors increases with respect to its temperature $T_f(t)$ until a certain level is attained $T_f^{\max}$, after which the diffusion is constant, i.e. the whole fluid inside the flat collector is at the same temperature. This approximation is quite reasonable with respect to the ST application and in accordance with the literature, i.e. [Pasamontes et al. 2013].

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

$$h_i\left(T_p(t)\right) = \overline{h}_i\left(\frac{1-e^{\frac{-T_p(t)}{T_p^{\max}}}}{1-e^{-1}}\right), \tag{6.37}$$

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

Regarding the nonlinear model of Eq. (6.36) with the relaxations of Eqs. (6.36)-(6.37), the parameters have been identified and adjusted for the *CIESOL ST* plant, located in the *CIESOL-ARFR-ISOL R&D* Centre of the University of Almería, Spain. The numerical values for these parameters, from the work [Pasamontes et al. 2013], are given in Table 6.1.

Table 6.1: ST unit: Model parameters.

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

6.5.2 The control problem

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

In terms of performances, the temperature set-point tracking should be done as fast as possible, while respecting the maximal temperature of 300 °C that the inlet fluid can tolerate. Moreover, the temperature of the plates should not surpass 600 °C. These performances can be evaluated using usual reference-tracking indexes, such as the integral of the average tracking error. Through the sequel, we denote T_p^{SP} and T_f^{SP} as the constant steady-state temperature references to the collector plate and to flowing fluid, respectively. The considered steady-state targets for reference tracking are: $T_p^{\mathrm{SP}} = 109.93\,°\mathrm{C}$, and $T_f^{\mathrm{SP}} = 97\,°\mathrm{C}$.

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

We stress that the dynamics of this ST process exhibit average settling periods in the order of 100 s. In practice, many control schemes have been tuned considering a sampling period of a few seconds, e.g [Pasamontes et al. 2013; Ampuño et al. 2019]. Nevertheless, we choose a tighter sampling period for illustration purposes, in order to verify whether the proposed method could serve for embedded real-time applications.

The disturbances to this system (the solar radiance and external temperature variables) are assumed to be measurable from a control viewpoint. This is quite reasonable, given that accurate estimations for the future behaviour of these disturbances can be indeed obtained, see [Camacho et al. 2012]. These estimation results (for solar radiance and outside temperature) are easily provided with Neural Network tools, i.e. [Vergara-Dietrich et al. 2019].

Table 6.2 resumes the state and input constraints. Note that the fluid and plate temperatures are lower-bounded by external temperature to the ST system, $T_e(t)$. If there is no sun during the day, the ST system reaches a thermal equilibrium with $T_e(t)$. For simplicity, since $T_e(t) > 0$, the lower bounds on T_p and T_f can be taken as 0.

Table 6.2: Constraints of the considered ST system.

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

6.5.3 qLPV-embedded model

Since this Chapter is concerned with the application of MPC technique, the ST nonlinear phenomenological model of Eq. 6.36, with the relaxations of Eqs. (6.36)-(6.37), is Euler-discretised with the sampling period of $T_s = 0.01$ s. This procedure yields a nonlinear discrete-time model. Given that the proposed min-max MPC method is conceived for qLPV embedded nonlinear models, and due to the fact that the LDI property holds for the yielded discrete-time model, a qLPV model is obtained. We consider the following system states:

$$x(k) = \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} = \begin{bmatrix} T_p(k) - T_p^{\mathrm{SP}} \\ T_f(k) - T_f^{\mathrm{SP}} \end{bmatrix}, \tag{6.38}$$

and the scheduling parameters as $\rho = [\rho_1, \rho_2]^T$, which are respectively derived directly from the nonlinearities added to the balance of energy equations due to the time-varying thermal loss term given in Eq. (6.37) and due the partial derivative approximation given in Eq. (6.36):

$$\begin{bmatrix} \rho_1(k) \\ \rho_2(k) \end{bmatrix}^T = f_\rho(x(k)) = \begin{bmatrix} d_i \pi \overline{h_i} \left(\frac{1-e^{-\frac{x_1(k)}{(T_p^{\max}-T_p^{\mathrm{SP}})}}}{1-e^{-1}} \right) \\ \frac{1-e^{-\frac{x_2(k)}{(T_f^{\max}-T_f^{\mathrm{SP}})}}}{(1-e^{-1})A_i} \end{bmatrix}. \tag{6.39}$$

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

$$\rho_1 \in [\underline{\rho_1}\,,\, \overline{\rho_1}] = [0\,,\, d_i\pi\overline{h_i}] \text{ and} \tag{6.40}$$

$$\rho_2 \in [\underline{\rho_2}\,,\, \overline{\rho_2}] = \left[0\,,\, \frac{1}{A_i}\right], \tag{6.41}$$

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

$$\begin{cases} x(k+1) = A(\rho(k))x(k) + B(\rho(k))u(k) + B_w(\rho(k))w(k)\,, \\ \rho(k) = f_\rho(x(k))\,. \end{cases} \tag{6.42}$$

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

$$\begin{cases} A(\rho(k)) = I_{n_x} + T_s \begin{bmatrix} -\frac{d_e\pi h_0}{\epsilon_m C_m A_e} - \frac{1}{\epsilon_m C_m A_e}\rho_1(k) & \frac{1}{\epsilon_m C_m A_e}\rho_1 \\ \frac{1}{\epsilon_f C_f A_i}\rho_1 & -\frac{1}{\epsilon_f C_f A_i}\rho_1 \end{bmatrix} \\ B(\rho(k)) = T_s \begin{bmatrix} 0 \\ -\rho_2 \end{bmatrix}, \\ B_w(\rho(k)) = T_s \left[\begin{array}{cc|cc} \frac{d_e\pi\nu}{\epsilon_m C_m A_e} & \frac{d_e\pi h_0}{\epsilon_m C_m A_e} & -\frac{d_e\pi h_0}{\epsilon_m C_m A_e} - \frac{1}{\epsilon_m C_m A_e}\rho_1(k) & \frac{1}{\epsilon_m C_m A_e}\rho_1 \\ 0 & 0 & \frac{1}{\epsilon_f C_f A_i}\rho_1 & -\frac{1}{\epsilon_f C_f A_i}\rho_1 \end{array}\right]. \end{cases} \tag{6.43}$$

6.5.4 Offline preparations

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

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

$$\left\|\frac{\partial f_\rho}{\partial x}\right\| = \left\|\begin{bmatrix} \left(\frac{d_i\pi\overline{h_i}}{(1-e^{-1})}\right)\frac{1}{490}e^{-\frac{x_1}{490}} \\ \left(\frac{1}{A_i(1-e^{-1})}\right)\frac{1}{203}e^{-\frac{x_2}{203}} \end{bmatrix}\right\| \leq 0.3246 \;\forall\; x \in \mathcal{X}. \tag{6.44}$$

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

$$\left\|\begin{matrix} \left(\frac{d_i\pi\overline{h_i}}{(1-e^{-1})}\right)\left(e^{-\frac{x_1}{490}} - e^{-\frac{\hat{x}_1}{490}}\right) \\ \left(\frac{1}{A_i(1-e^{-1})}\right)\left(e^{-\frac{x_2}{203}} - e^{-\frac{\hat{x}_2}{203}}\right) \end{matrix}\right\| \leq \gamma_\rho \left\|\begin{pmatrix} x_1 - \hat{x}_1 \\ x_2 - \hat{x}_2 \end{pmatrix}\right\|, \tag{6.45}$$

where:

$$\gamma_\rho = \left| \begin{array}{c} \left(\frac{d_i \pi \overline{h_i}}{(1-e^{-1})}\right)\frac{e^1}{490} \\ \left(\frac{1}{A_i(1-e^{-1})}\right)\frac{e^1}{230} \end{array} \right| = 0.8825\,. \tag{6.46}$$

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

$$\xi_\rho^{\text{bound}} = \left(\gamma_\rho + \overline{f_\rho^\partial}\right)\overline{\Delta x} = \left[\begin{array}{cc} 0.046 & 0.0015 \end{array}\right]^T . \tag{6.47}$$

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

$$\mathcal{S} := \{\sigma \in \mathbb{R}^{n_x} \,|\, \|\sigma\| \leq 4.89^{\text{o}}\text{C}\} . \tag{6.48}$$

Notice, for comparison purposes, that the uncertainty set computed as if the scheduling parameters varied arbitrarily inside $\mathcal{P}$ (as done in the original min-max LPV MPC design algorithms, e.g. [Cao and Lin 2005]) is given by:

$$\mathcal{S}^{\text{Cao et al., 2005}} := \{\sigma \in \mathbb{R}^{n_x} \,|\, \|\sigma\| \leq 599^{\text{o}}\text{C}\} . \tag{6.49}$$

while the uncertainty set computed taking the rates of variations of the scheduling parameters ($\delta\rho$) into account, as done in [Li and Xi 2010], for a control horizon of $N_p = 30$ steps, is given by:

$$\mathcal{S}^{\text{Li et al., 2010}} := \{\sigma \in \mathbb{R}^{n_x} \,|\, \|\sigma\| \leq 489^{\text{o}}\text{C}\} . \tag{6.50}$$

Evidently, these two sets are much wider than the one with the proposed method. This means that the online computational effort to solve the maximisation CP is smaller with the proposed method. This issue is debated next.

6.6 Results and debate

Next, the proposed dissipative fast robust MPC method for nonlinear systems is applied to the ST collector system. The considered process is emulated through the nonlinear high-fidelity phenomenological partial-differential model given in Eq. (6.36), with parameters given by Table 6.1. The proposed control method is implemented with the uncertainties defined by the set $\mathcal{S}$ in Eq. (6.48). The solutions of the optimisation problems are obtained using Gurobi.

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

- A full-blown NMPC algorithm [Allgöwer and Zheng 2012], which embeds the complete nonlinear model predictions. To solve the resulting NP, fmincon solver is used; this method is referred to as “Full-Blown NMPC”.

- The original min-max LPV MPC algorithm [Cao and Lin 2005], defined with respect to the uncertainty set given in Eq. (6.49). It is henceforth denoted “min-max (Cao et. al, 2005)”.

- The min-max LPV MPC scheme considering bounded rates of parameter variations [Li and Xi 2010], defined with respect to the uncertainty set given in Eq. (6.50). This approach is denoted “min-max (Li et. al, 2010)”.

- The qLPV-embedding (SQP) MPC method from [Cisneros and Werner 2020] (approach (iii) in Chapter 3), which uses a scheduling sequence estimation and solves sequential QPs, solved via through iterated uses of Gurobi. This last method is henceforth marked as “qLPV MPC (Cisneros & Werner, 2020)”.

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

$$\begin{cases} Q &= \begin{bmatrix} 0.2 & 0 \\ 0 & 0.8 \end{bmatrix}, \\ R &= 10^{-6}. \end{cases} \tag{6.51}$$

We proceed by depicting the obtained results in terms of reference tracking, i.e. regulation of the system states to the origin. These results comprise 950 s of simulation of the considered solar-thermal unit. The load disturbances (solar irradiance and environment temperature) are shows in Figure 6.6.

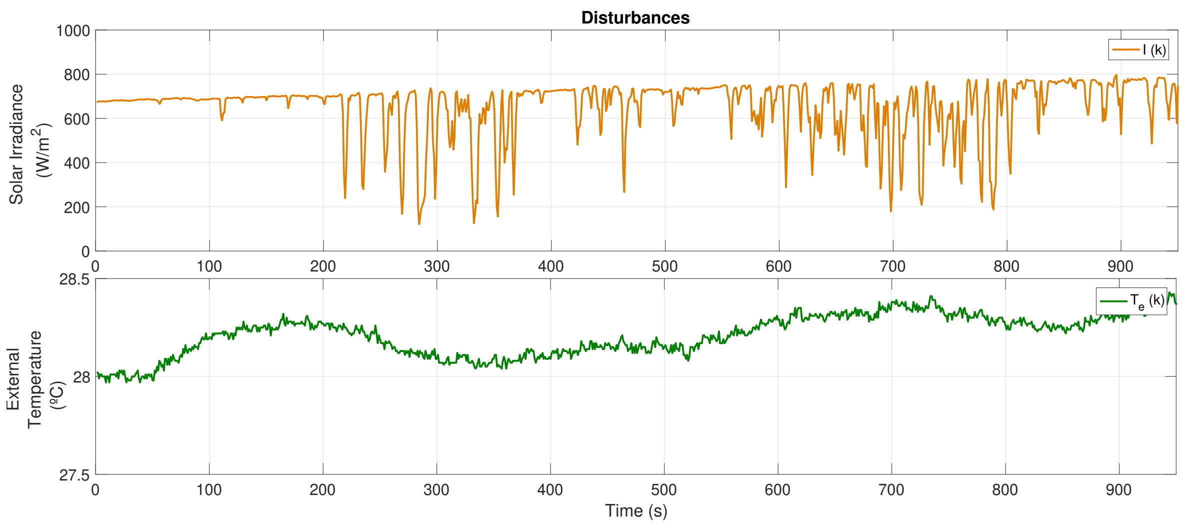


Figure 6.6: Disturbance scenario.

6.6.1 Region of attraction

Firstly, we aim to demonstrate that the proposed method is indeed recursively feasible, yielding an ISS region of attraction $\mathcal{X}_{ISS}$. According to the steps detailed in Sec. 6.4, the LMI in Lemma 6.4.3 yields a positive definite matrix P and a constant τ that verify the dissipativity conditions of the proposed min-max algorithm. This is, indeed there exist P and τ such that the cost function of the minimisation QP decays over the simulation run. These are:

$$\begin{cases} P &= \begin{bmatrix} 0.18746 & 0.00011 \\ 0.87199 & 24.00050 \end{bmatrix} 10^{-4}\,, \\ \tau &= 1.67939\,10^{-7}\,. \end{cases} \tag{6.52}$$

Thus, for whichever starting condition x_0 found inside the ellipsoidal set $\mathcal{X}_{ISS} := \{x_0 \in \mathbb{R}^{n_x} \,|\, x_0^T P x_0 \leq 0\}$, input-to-state stability is ensured. Accordingly, this is shown in Figure 6.7, where the elipsoid $\mathcal{X}_{ISS}$ is depicted altogether with the evolution of the systems states $x(k)$ (obtained with the proposed method).

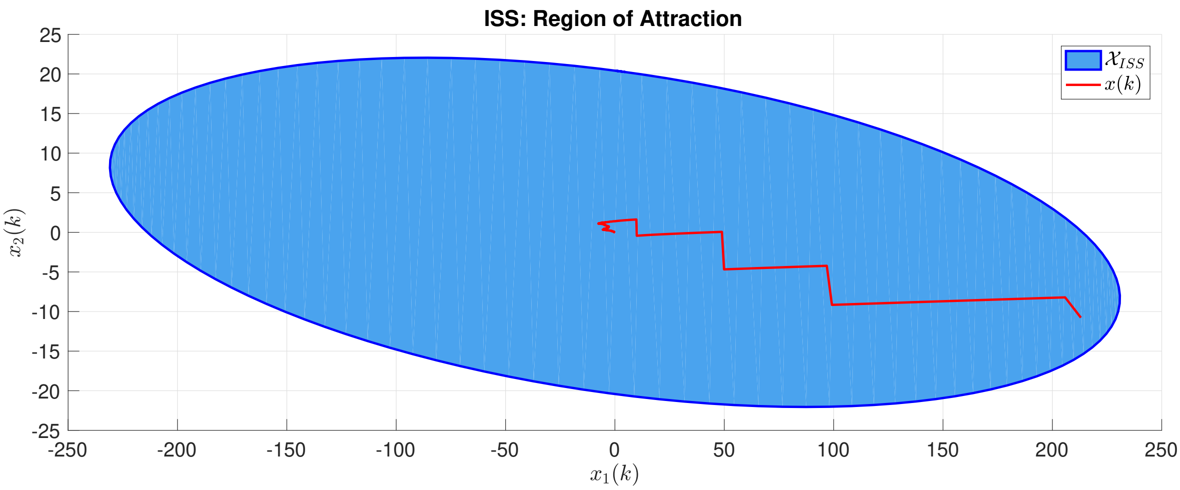


Figure 6.7: Region of attraction (ISS certificate).

6.6.2 Scheduling trajectory extrapolation

In Figure 6.8, we present the results concerning the extrapolation of the scheduling parameters ρ_1 and ρ_2 along the prediction horizon N_p. In this Figure, the dashed black line depicts the actual variation of $\rho(k)$, whilst the full blue line shows different snippets of scheduling sequences extrapolated according to the recursive algorithm in Eq. (3.43). The estimation error is quite small. Furthermore, the average time needed to solve the algorithm is of 0.41 ms, much smaller than the considered sampling period of 10 ms.

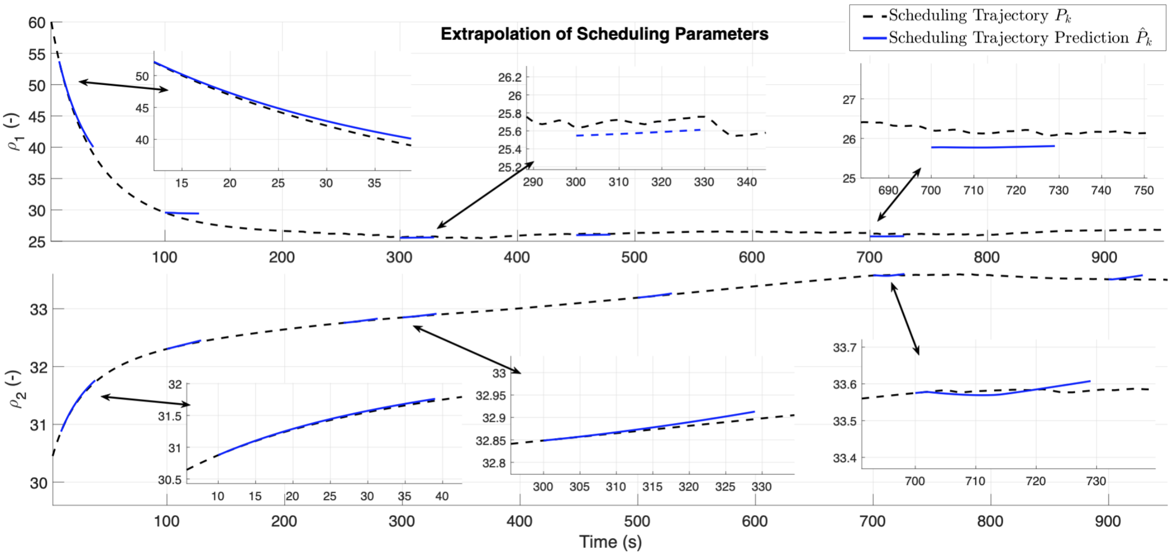


Figure 6.8: Results: Scheduling trajectory extrapolation.

6.6.3 Regulation results

The results concerning the regulation of $x(k)$, with all the tested methodologies, are presented in Figure 6.9. We stress that all methods ensure state and control constraints ($x \in \mathcal{X}$ and $u \in \mathcal{U}$). The regulation of the states to the origin is not thoroughly ensured by the min-max methods by Cao et al. and Li et al., since their respective uncertainty sets $\mathcal{S}^{\text{Cao et al., 2005}}$ and $\mathcal{S}^{\text{Li et al., 2010}}$ are too large with respect to $\mathcal{X}$. We note that the first min-max method stabilised x to $(-66.05\,,\,-96.7)\,°\text{C}$, while the second (bounded-rates) method brought the state trajectories to $(-65.95\,,\,-96.34)\,°\text{C}$. The smoother performances seem to be the ones attained the Full-blown MPC algorithm, while the proposed method and the one by Cisneros & Werner yield quite comparable performances. We remark that the control action also acts to attenuate the effect of the load disturbances; this is especially evident after $t = 500\,\text{s}$, when both disturbances vary abruptly (see Figure 6.6).

Table 6.3: Performances indexes: Plate temperature tracking (x_1).

Method	IAE ($.10^{-3}$)	RMS
Full-blown NMPC [Allgöwer and Zheng 2012]	7.6803	24.5295
Proposed qLPV MPC	11.5762	29.7982
min-max [Cao and Lin 2005]	60.5221	64.6373
min-max [Li and Xi 2010]	59.9912	64.2002
qLPV MPC [Cisneros and Werner 2020]	7.9062	24.6305

The proposed method is able to ensure adequate results since its uncertainty set $\mathcal{S}$ is relatively small. Moreover, the uncertainty vector $\Xi_k^\star$ computed through the maximisation CP norm-decreases over the simulation, as the extrapolation of the scheduling sequences gets better (see Figure 6.8).

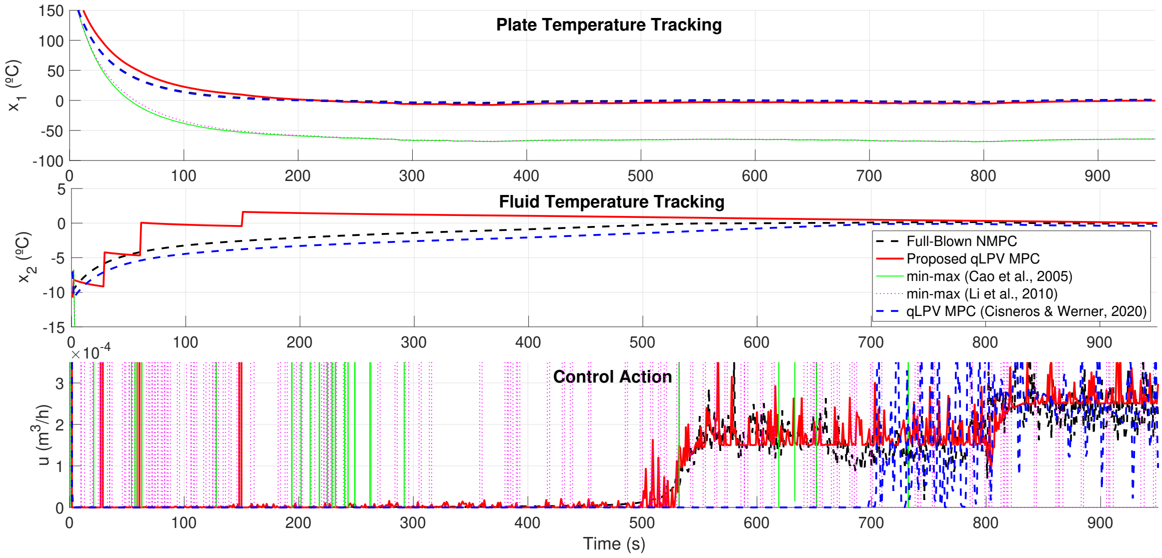


Figure 6.9: Results: Temperature tracking and control signals.

Table 6.4: Performances indexes: Fluid temperature tracking (x_2).

Method	IAE ($.10^{-3}$)	RMS
Full-blown NMPC [Allgöwer and Zheng 2012]	1.1509	2.0402
Proposed qLPV MPC	1.0354	1.8946
min-max [Cao and Lin 2005]	93.0361	96.2502
min-max [Li and Xi 2010]	91.5767	94.8112
qLPV MPC [Cisneros and Werner 2020]	1.9396	2.8030

We proceed by investigating these performances through performance indexes: Tables 6.3 and 6.4 show, respectively, the root-mean square (RMS) and integral-of-the-absolute-error (IAE) indexes applied to $x_1(k)$ (plate temperature tracking) and $x_2(k)$ (fluid temperature tracking). We stress that smaller IAE and RMS values indicate better performances, which conversely means that the references are tracked faster and with less steady-state error.

With respect to the regulation of x_1, these tables show that the performances achieved with the Full-Blown NMPC and the qLPV MPC by Cisneros & Werner are roughly equivalent in terms of RMS and IAE. The proposed method does not stay far behind, having slightly slower tracking in the first few seconds, which results in the settling seen by $t = 200\,$s in Figure 6.9. It is important to notice that this fact resides in the maximisation procedure, which implies the robustness by finding larger uncertainty vectors $\Xi_k^\star$ in these first moments, which reflect on the solution found by the minimisation QP and the slight difference to the other methods. Anyhow, we stress that the performances are comparable.

Owing to the regulation of x_2, it is seen that the IAE and RMS indexes indicate that the best tracking performances are obtained with the proposed method. As seen in Figure 6.9, the Full-Blown NMPC and the qLPV MPC method by Cisneros & Werner [Cisneros and Werner 2020] yield comparable results.

Table 6.5 presents the TV index for each control signal. It can be seen that the smoother control values are obtained by the Full-Blown NMPC method, with the proposed method and the method by Cisneros & Werner not standing far behind. The min-max methods by Cao et al. and Li et al. present negligible results, at least for this ST application for which ρ has a big variation set $\mathcal{P}$ with also large possible variation rates (i.e. $\delta\mathcal{P}$ is also large).

Table 6.5: Total variance of the control signal.

Method	TV
Full-blown NMPC [Allgöwer and Zheng 2012]	0.1987
Proposed qLPV MPC	0.2800
min-max [Cao and Lin 2005]	16.8449
min-max [Li and Xi 2010]	68.97871
qLPV MPC [Cisneros and Werner 2020]	0.2789

6.6.4 Computational stress

With respect to these results, we present a very important issue: the average computational time (t_c) needed to solve the optimisation procedure of the methods are synthesised in Table 6.6. We recall that the sampling period of the system is of 10 ms (which is the computational time upper bound). Evidently, the Full-Blown NMPC needs a lot of time to solve its inherent NP, which means that this method is not applicable in practice for processes with small sampling periods. The results obtained with this method are purely numeric and would not be able to be applied in practice. The qLPV MPC method by Cisneros & Werner solves, in average, **five** QPs (it iterates the QPs to compute the extrapolation guess $\hat{P}_k$). The proposed method operates, in average, within 6.3 ms, spending 0.41 ms to make the extrapolation guess $\hat{P}_k$. These are very interesting results, meaning that the proposed solution is indeed fast and able to operate for embedded applications. The performances of the proposed method are equivalent to the method by Cisneros & Werner, which operates in the millisecond range as well as the available modern NMPC solutions, such as ACADO and GRAMPC [Quirynen et al. 2015; Englert et al. 2019].

We highlight that the obtained time performance depends on the operating computer machine and on the size of the controlled system. In this Chapter, the considered system is a 2×2 system, for which the max. CP and the min. QP are evaluated simply enough. For larger order models, sub-optimal solutions might be necessary; refer to a previous discussion on this matter in [Zhang, Li, and Liao 2019].

6.7 Conclusions

In this Chapter, a novel MPC algorithm for nonlinear system is proposed. The nonlinear system is embedded into a qLPV formulation and its scheduling parameters ρ are extrapolated

Table 6.6: Computational load.

Method	t_c
Full-blown NMPC [Allgöwer and Zheng 2012]	776.5 ms
Proposed qLPV MPC	6.3 ms
min-max [Cao and Lin 2005]	7.2 ms
min-max [Li and Xi 2010]	7.5 ms
qLPV MPC [Cisneros and Werner 2020]	8.7 ms

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

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

- Full-blown nonlinear programming NMPC are not applicable for embedded applications of processes with fast sampling rates, since the average time needed to solve the NP is usually larger than the available sampling period. Recent literature has shown how approximated NMPC methods (such as CaSaDi, GRAMPC and ACADO [Quirynen et al. 2015]) and qLPV-embedding MPC algorithms [Cisneros and Werner 2020] are able to efficiently solve such complex control problem in the range of milliseconds.
- For the considered case study, the reference tracking performances obtained with the proposed qLPV-embedding min-max MPC method are equivalent to these fast modern nonlinear MPC methods [Cisneros and Werner 2020], as assessed by the RMS and IAE indexes. The numerical operability of the proposed. method is similar to previous works [Quirynen et al. 2015; Cisneros and Werner 2020]. We note that the complexity of the problem grows with the order of the system.
- The proposed method solves the maximisation convex programming problem with respect to the error regarding the estimation of the scheduling parameters along the prediction horizon. We stress that any kind of algorithm with bounded estimation errors could be used in the place of the Taylor expansion one proposed in this Chapter. An alternative and elegant option could be the use of the iterated mechanism [Cisneros and Werner 2020], which uses the state sequence computed with the minimisation QP to compute the evolution of ρ along the horizon.
- The proposed method is compared to two keystone min-max LPV MPC algorithms from the literature [Cao and Lin 2005; Li and Xi 2010], which consider, respectively, that ρ can vary arbitrarily inside $\mathcal{P}$ and considers bounded rates of variations for ρ. Since the variations of the scheduling parameters and its convex set are quite large for the considered application, the results obtained with these methods are quite poor. The uncertainty set with the proposed method is much smaller (by a factor of a hundredth).

Furthermore, as time control law progresses, the extrapolation method gets better estimations of ρ, which also makes the uncertainty output of the maximisation problem to converge to zero, as the state trajectories converge.

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

- As a note on perspectives, we emphasize that an alternative formulation to the LPV MPC min-max solution can be considered, in the sense that the scheduling set $\mathcal{P}$ can be taken as a time-varying projection related to the scheduling proxy $\rho = f_\rho(x)$. In this case, enhanced performances can be obtained since, as x is regulated, the variability of ρ is decreased and so is the corresponding scheduling set. We stress that in the proposed solution, a constant (full sized) scheduling set $\mathcal{P}$ was considered.

Part IV

Closure

Chapter 7

Conclusions and outlook

In this final Chapter, I present a broad overview of the developments and applications conceived along these (roughly) four years of Ph.D. Also, I recap the original thesis objectives, as presented in the Introduction, debating the achieved results in terms of limitations and advantages. Finally, I lay out the open threads and available investigation gaps that still remain open, providing perspectives and an outlook regarding future works. The complete list of publications that derived from this doctoral work is presented in Appendix A, with works being categorised by topic and considered application.

Before any of conclusion is drawn, nonetheless, I stress that the development of my thesis was directly affected by the COVID-19 pandemic (in many levels). As a start, I expected to spend the year of 2020-2021 in France, as accorded in the original joint supervision convention, in order to perform experimental essays on vehicular test-beds available in GIPSA-Lab. Secondly, I was financed by the German ministry of education (BMBF) for a three-months research stay in Stuttgart, supervised by Prof. Carsten Scherer (this is due to the Green Talent awarded to me my BMBF in 2019[1]). This research stay was planned to be carried out just before the start of my year-long stay in France. Due to the enacted quarantine and social distancing guidelines, these dates were postponed, which made to me aborting some of my initial plans for the thesis, while adapting and redirecting others. For instance, I had to replace many experimental essays by the means of validation using high-fidelity realistic numerical simulators (especially in the case of vehicle dynamic applications). Moreover, the developments on dissipativity theory applied to MPC were reduced, since I could only properly conducted in the last doctoral year. In the end, I could conduct some experimental essays during the period in France, along 2022, while many of these results were not included in this work but rather published in separate papers, e.g. [Morato et al. 2021c; Medero et al. 2022; Morato et al. 2022b].

Apart from all these unforeseen changes in the context of the "methodological script", planning, and initial calendar of my thesis, which, in some sense, disturbed my developments, I emphasise, as a personal note, that the uneasiness of the COVID-19 contagion in Brazil, corroborated by speeches of disbelief in science in a broad sense, instigated me to also ponder over possible implications of my scientific work (and of the algorithms investigated this thesis) with regard to mitigating the observed[2] social-economic effects of the pandemic in Brazil. Accordingly, I spent a considerable period of time, during the years of 2020 and 2021,

[1] See `https://www.greentalents.de/awardees-2019.php`.

[2] The challenges involved in controlling the SARS-CoV-2 epidemic are many. Brazil is a very large country, which registered over 35 million infections and over 690,000 deaths due to COVID-19. In the country, the

developing MPC applications with focus to COVID-19 contagion models (using nonlinear and LPV frameworks), seeking to assess on how optimisation and predictive control could be used to provide coherent (and socially applicable) guidelines for non-pharmaceutical interventions that helped diminish the effects of the contagion[3], taking into account what happened in Brazil (with special focus to the states of Santa Catarina and Bahia). In this sense, I was able to commit partial focus of my work, in some level, as a retribution to society in the context of the effervescent scientific debate that span these years, with recognised contributions published in international journals within the control community, e.g. Annual Reviews in Control, [Morato et al. 2020c], but also ones with much broader scope, e.g. Nature Scientific Reports, [Pataro et al. 2021a]. With respect to this matter, I conducted a joint collaboration[4] with the leading scientific institution for research and development in biological sciences in Latin America, Oswaldo Cruz Foundation, branch of the state of Bahia (*Instituto Gonçalo Moniz, Fiocruz Bahia*).

Also I remark that, along these years of thesis, I was able to collaborate with many colleagues from around the world, with connections enabled through my supervisors. Within the Brazilian scientific community, I had fruitful collaborations with researchers from the from the federal technical university of Paraná, from the university of Brasilia, from the federal universities of Minas Gerais, Rio Grande do Sul, and Bahia. In the international level, I could develop joint works with colleagues from the University of Seville, in Spain, from the Fraunhofer Institute and the University of Stuttgart, in Germany, from the University of the Philippines Diliman, in the Philippines, from the National Technologic University, in Argentina, from the University of Lorraine, in France, and from the University of Kragujevac, in Serbia.

---

contagion spread quickly: the first cases were registered in March, 2020; the first million was surpassed in June, 2020; one year later, in June, 2021, over 17 million cases had already been confirmed. In Brazil, first-wave mitigation strategies were largely decentralized, and the majority of governmental interventions occurred by the means local actions taken by each of 26 states (and the federal district), and their 5, 570 municipalities. Still, mitigation efforts were inadequate and the SARS-CoV-2 transmission was never under proper control: the collapse of health services was seen throughout the country, which directly influenced the number of fatal outcomes observed to date. Even in some areas with high seroprevalence, such as the city of Manaus, intense subsequent resurgent epidemic waves were seen along the year of 2021, in line with the recrudescence of the viral transmission, confirming that herd immunity was never a feasible or ethical route to tackle COVID-19.

[3]Mathematical models have played a key role in assessing the effectiveness of public health policies to contain the spread of SARS-CoV-2, as well as to evaluate the transmission dynamics of COVID-19 and how it is impacted by the movement of people. However, the bridging these models to socially applicable governmental actions is heavily limited by the inherent uncertainties surrounding the obtained estimates, interpretation difficulties by policy-makers, and the lack of full understanding of a model's predictive capabilities and limitations. Accordingly, I investigated how optimal control algorithms, coupled to (nonlinear and LPV) epidemiological models could provide intuitive means to derive coherent health policies from data.

[4]By drawing on the availability of widespread mobility traces from cell phones, and building on the premise that circulation of individuals is a chief contributing factor for SARS-CoV-2 transmission, the main results provided in [Pataro et al. 2021a] was the adaptation of a nonlinear MPC strategy that reliably predicts an optimal level of governmental interventions to decrease mobility, considering different degrees of social effects, thereby reducing COVID-19 cases, and resulting fatalities, while maintaining hospitalization requirements below their limits and averting the unnecessary extension of restrictive measures such as lock-downs. The developed algorithm was applied to study the disease dynamics in Bahia (before vaccine campaigns started), the largest and most populous state of northeast of the country, with territorial extension comparable to that of France.

These collaborations, and all works derived along the growth of this thesis, were carried out under strict scientific rigour, with exclusive dedication. In terms of the societal relevance of this thesis, as stated in the Introduction, I stress that I made an effort to conduct and align my research with focus in facilitating two concrete social issues: (*i*) affordable renewable energy generation, and (*ii*) efficient urban mobility technologies. By exploiting the application of model predictive control algorithms for these two classes of complex systems (described by the means of LPV models), I sought to contribute, in some level, by fostering practical possibilities in compass with Agenda 2030 and the endowment of Sustainable Development Goals 7 and 11, seeking to mitigate the ongoing social-environmental calamity derived from climate changes. With regard to this matter, I present two final messages:

(i) Regarding optimization and control for renewable energy generation systems, I studied how model predictive control can serve to maximise energy generation efficiency in different systems (especially microgrids and solar collector plants). Taking into the Brazilian context, I investigated how the national sugarcane industries can be explored as a practical (and economically plausible) paradigm to leverage the renewable energy generation in the country, if coupled to photovoltaic panels and wind turbines and coordinated using MPC. I also assessed on how MPC can be used as the main approach to enhance the performance of solar-colector systems, under real-time constraints. Accordingly, I demonstrated how this control method can lead to a maximised benefit of the solar availability. These developments thus addressed SDG 7, which deals with clean and accessible energy generation to all. Main works on this topic: [Morato et al. 2020e; Morato et al. 2020b; Pipino et al. 2020b; Bernardi et al. 2021; Morato et al. 2021c].

(ii) Regarding automated urban mobility, I recap that SDG 11 explicitly mentions how sustainable cities and communities require solutions that allow efficient control of car traffic, such as connected vehicles that can actively act in accident prevention. Accordingly, with regard to this topic, I developed novel MPC algorithms that improved the comfort of passengers in modern vehicles. By exploiting the LPV formulation, these algorithms are capable of operating in an embedded manner, in on-board microcontrollers, operating on a scale of a few milliseconds. Furthermore, I also proposed predictive control strategies for the assisted driving of semi-autonomous vehicles, with intelligent interventions avoiding loss of stability and accidents. The issue of automated driving under faulty situations was also investigated. Main works on this topic: [Morato, Normey-Rico, and Sename 2019; Morato, Normey-Rico, and Sename 2020c; Morato, Normey-Rico, and Sename 2021c; Medero et al. 2022; Morato et al. 2022a].

As a result of professional experience in classroom during the years thesis, I published with my supervisor Julio Elias Normey-Rico, at the invitation of the Brazilian Society of Automatics (SBA), a new textbook for basic courses on process control, with an approach focused on theoretical analysis without the need for advanced mathematical tools, see [Normey-Rico and Morato 2021; Normey-Rico and Morato 2022]. Furthermore, I also coordinated an outreach project that aimed to discuss the basic notions of process control and renewable energy applications in public elementary schools in Florianópolis[5] [Morato et al. 2019a]. Overall, I

[5]As a result of my contributions, the city council awarded me, in 2020, the Prof. João David Ferreira Lima

can confidently say that these years of Ph.D. were a very interesting research journey. I am satisfied with the obtained results, and already motivated for next scientific steps.

7.1 An overview

In this thesis, I addressed the problem of controlling linear parameter varying systems (and nonlinear ones too, under quasi-linear parameter varying structures). For such, I developed several model predictive control schemes, under three different kinds of exploitations: using gain-scheduling (Part II) and robust synthesis (Part III). Each of these categories of predictive controllers was synthesised and proposed with corresponding theoretical analysis tools which enable closed-loop stability and recursive feasibility of the controller optimisation.

Regarding the first contributions of this thesis (Part II), I focused on novel results using gain-scheduling (in terms of both analyses and design procedures). That is, procedures for which the formulated controller varied according to the instantaneous (sampled) gain of the controlled LPV system. In these methods, synthetically, the scheduling variable $\rho(k)$ is used to coordinate the optimisation (and, thus, the derived predictive control policy) at each sample k. The future values of the scheduling variables along the prediction horizon, i.e. $\rho(k+j), \forall j \in \mathbb{N}_{[1,N_p]}$, are generally neglected (considered frozen) or replaced by an accurate estimate $\rho(k+j|k)$. By the means of the developed algorithms, I could demonstrate how gain-scheduling can be quite useful in practice, and how nonlinear MPC can be efficiently solved by the means of appropriate qLPV embeddings. The major highlight is that a corresponding nonlinear predictive control can be rendered via the solution of, usually, only one QP per sample, which is much faster than solving a nonlinear program. Note that QPs have a complexity that grows linearly, in general, with regard to n_u, n_x and N_p (number of control inputs, system states and the prediction horizon size), while the growth in the nonlinear setting is **exponential**.

In Part III, I formulated robust MPC methods. These, in opposition to the gain-scheduling formulations, not only considered the future scheduling variables, but also the estimation error with regard to them, i.e. $\xi_\rho(k+j|k) = (\rho(k+j) - \rho(k+j|k))$. Thus, in the robust setting, the proposed algorithms included additional tools to handle the corresponding uncertainty propagation (maximisation program and constraint shrinking using zonotopes). By using these features, I could demonstrate guaranteed performance certificates and constraint satisfaction, despite the propagation of the uncertainties that arose due to the unknown scheduling variables. I point out that the issue of tracking (possibly unreachable) reference signals was also addressed in a robust fashion.

7.1.1 New results on gain-scheduled control synthesis

The first (main) contribution of this thesis was presented in Chapter 3. Therein, I discussed how the future scheduling variables $\rho(k+j)$ arise as uncertainties under MPC. Furthermore,

Medal, for services related to higher education in the city.

for the case of qLPV models, a new extrapolation algorithm was provided in order to generate a numerically non-demanding estimate for the whole scheduling trajectories. The proposed algorithm (Section 3.6), thus, enables the fast application of nonlinear MPC, comprised of only one QP per sample. The method is recursive, which means that the generated scheduling trajectory guess at a given sampling instant is used a basis for the construction of the estimate at the following sample (refer to Eq. (3.36)). Moreover, the resulting algorithm, formulated by exploiting a simple Taylor expansion argument, was shown to converge in finite time and five sufficient conditions for convergent extrapolation are presented (Section 3.6.1, Lemma 3.6.2). These conditions are quite simple and mainly refer to the form and class of the qLPV scheduling function and the robustness of the gain-scheduled MPC. Several different examples from the literature were provided in order to illustrate and discuss the algorithm (Section 3.7), including a semi-active suspension system and a pendubot benchmark. The method is also compared to state-of-the-art techniques that are used to generate estimated for qLPV scheduling trajectories, showing exceptional effectiveness under much reduced numerical burden. The application of the proposed recursive estimation algorithm (and the convergence establishment proofs) were celebrated in depth, in both theoretical [Morato, Normey-Rico, and Sename 2022b] and practical sense [Morato et al. 2021d; Morato et al. 2021a; Morato et al. 2022b]. Simulation essays comparing the resulting gain-scheduled MPC algorithms for nonlinear systems under qLPV-embeddings against to state-of-the-art NMPC algorithms were provided, demonstrating competitive performances of the proposed framework.

Secondly, in Chapter 4, two novel contributions were presented:

- First, a gain-scheduled qLPV MPC algorithm was proposed for the control of Semi-Active suspension systems, enabling the comfort enhancement of onboard passengers. The method considered a frozen scheduling trajectory formulation, and thus took into account the bounds on the variation rates of the scheduling parameters in order to consider the uncertainty propagation. Using set-based terminal ingredients, I could demonstrate closed-loop stability and recursive feasibility. Furthermore, successful nonlinear essays of scaled car were conducted, which served to illustrate the overall good operation of the vehicle under the proposed scheme: the comfort of the passengers was substantially improved, as measured through time and frequency domain indexes;

- Then, I elaborated on an MPC algorithm for qLPV systems represented in the IO form. The main idea is that, by considering such IO description, no state measurements are not necessary, which is interesting from an industrial and practical application perspective (no need for observer design, for instance). Taken into account the recursive Taylor-based extrapolation mechanism from Chapter 3, I developed quadratic terminal ingredients that enabled enable asymptotic IO stability in closed-loop (as well as a recursively feasible optimisation). Using an unstable nonlinear numerical example, I could demonstrate the advantages of the proposed method, as well as its real-time capabilities. I note that the IO qLPV MPC was formulated considering output tracking of piece-wise constant reference signals, using an explicitly-included integral feature.

7.1.2 Novel robust synthesis solutions

In Chapter 5, the first contribution regarding robust synthesis was presented. This contribution is a novel nonlinear MPC scheme conceived for the purpose of tracking (possibly unreachable) piece-wise constant reference signals. Specifically, I applied a qLPV translation and considered the scheduling trajectory extrapolation mechanism from Chapter 3. Then, the bounds on the scheduling parameters' estimation errors were considered in order to generate uncertainty propagation zonotopes and parameter-dependent terminal ingredients. Accordingly, I could provide closed-loop stability, and recursive feasibility certificates. Furthermore, the issue of avoiding feasibility losses due to large set-point variations was addressed using the so-called "artificial target variables". Assessments on the robustness qualities and real-time capabilities of the methods were illustrated using a two-tank system, for which the method was compared to state-of-the-art techniques, exhibiting comparable results under much faster computation.

Then, in Chapter 6, a different robust synthesis approach was considered. The developed scheme took into account the solution of an online min-max problem: firstly, a constrained CP is solved in order to determine the worst-case bound on the cost function and, subsequently, a second constrained QP is solved to minimise this worst-case cost function with respect to a control sequence vector. Since the bounds on the estimation error for the scheduling parameters are usually much smaller than the bounds on the actual scheduling parameter, the conservativeness of the solution was shown to be quite reduced. Again, I provided certificates on recursive feasibility of the proposed algorithm, as well as closed-loop stability. These certificates were conceived using dissipativity arguments given in the form of a an LMI remedy, which also determines the zone of attraction for which input-to-state stability is ensured. As a case study, the nonlinear temperature regulation problem of a flat solar collector was considered. Using a realistic simulator, the proposed technique is compared to other robust min-max LPV MPC algorithms from the literature, proving itself efficient and able to achieve good performances.

7.1.3 Recalling thesis' goals

With regard to the original objectives of my thesis (as presented in Chapter 1, Section 1.5), I stress that they have been thoroughly attained. I detail these advances:

- Objective (i), regarding an extrapolation method for future scheduling trajectories, with linear recursions and convergence properties, was established as of Chapter 3.
- Objective (ii), which is the development of new gain-scheduled predictive control algorithms, was addressed in in Chapter 4, considering both state-feedback and IO formulations;
- Finally, Objective (iii), of robust MPC alternatives for LPV systems, has been deployed along Chapters 5 and 6.

7.1.4 Main message

In general, I argue that this thesis serves and contributes to the context of nonlinear MPC design by supporting the following message: the approach of using LPV models (and qLPV embeddings) in order to model nonlinear and time-varying dynamics can serve as a support to design real-time capable NMPC algorithms. By exploiting the linearity features along the input-output channels of these LPV descriptions, I could show, with multiple examples, comparisons and synthesis options, how the resulting MPC algorithm is much similar to that of an LTI plant. Moreover, stability and robustness certificates for the resulting controller are also much similar to those in the LTI case, which are standard nowadays. I emphasised three main points:

1. When the scheduling trajectories are assumed constant (frozen), the resulting gain-scheduled LPV MPC algorithm is sub-optimal. Nevertheless, from practical and empirical perspectives, these controller are in much sense comparable to NMPC ones. Even though the accuracy of the gain-scheduled prediction deteriorates as the horizon size increases, the rolling-windows mechanism of the MPC inherently smoothes this issue. In practice, the approach is relevant, easy to implement, and can solve the issue of real-time applications. In many cases, this kind of approach is more than sufficient.
2. If one seeks to use estimation mechanisms for the future scheduling trajectories (as the Taylor-based approach proposed herein, or the iterative scheme from [Cisneros and Werner 2020]), the performances are enhanced. Since these mechanism ensure **convergence** of the estimated scheduling trajectories, the resulting closed-loop is **equivalent** to the one obtained with a NMPC algorithm (after a few discrete-time steps). This is a major advance since the optimisation is thus based on an **exact** prediction model, i.e. no approximation is required! Moreover, in comparison to linearisation-based techniques, the accuracy of the prediction is much better (while convergence hasn't been established), since the errors appear on the model scheduling parameters rather than on the full state trajectories, as what occurs with practical NMPC laws.
3. In the context of robust MPC, the LPV approach is also of interest: I assessed on how one can opt to use constraints tightening mechanisms, given with regard to the resulting uncertainty propagation (considering both bounded load disturbance and the scheduling parameter estimation error), thus leading to closed-loop performances that are not excessively conservative (yet the offline preparations are hard to synthesise), or to min-max approaches, with more conservative results but rather simple preparations.

7.2 Outlook

Even though quite a few contributions were derived from this thesis (refer to the full list of publications presented in Appendix A), there seem always to exist some open threads to be further investigated. Thus, I lay out some of these possible routes, in the sense of perspectives of future research:

- An interesting SA suspension control scheme using state-feedback qLPV MPC was proposed in Chapter 4. Yet, the main limitation of the method is that an online frozen estimate for the future scheduling trajectory is used, which render complex set-based ingredients. Moreover, in some cases, the considered bounds on the uncertainty propagation could be excessively large, which leads to empty terminal sets. Then, taking into account scheduling trajectory estimates, in the context of SA suspension control, is certainly an open for further assessments. Moreover, embedding a road profile (ISO) model to the control synthesis procedure could also be of interest, which could serve to transforms the load disturbances into noise variables, which can thus be treated with filtering techniques.

- With regard to the output-feedback qLPV MPC synthesis in Chapter 4, I stress that it did not take into account the issue of the errors on the scheduling parameter extrapolations. This was done because the residual errors $\xi_\rho(k+j|k)$ were shown to be bounded (Lemma 4.8.2) and, in practice, very small (Sec. 5.4), dissipating within a few samples. Thus, for the MPC formulation, It was assumed that $\xi_\rho(k+j|k)$ was null (i.e. that the scheduling trajectory estimates using Eq. (3.36) are exact). Nevertheless, assessments on a robustified version of the proposed IO MPC can be provided, taking into account the bounds scheduling prediction bias $\|\xi_\rho(\cdot|k)\|$. Such research could be of interest for other systems, when the bounds over $\|\xi_\rho(\cdot|k)\|$ are not negligible.

- For the case of reference tracking (Chapter 6), I stress that the bounds over the scheduling parameters' deviations over samples can be used as an additional synthesis parameter to enforce further conservatism (or aggressiveness) to the control law. That is, instead of considering these bounds as known and thus generating the zonotopes, the zonotopes can be considered to enforce a pre-specified rate of varation.

- As a final note, I mention that data-driven predictive control methods are also an interesting field of study, specially considering qLPV embedding. The only available results on data-driven predictive control consider that the processes have LTI behaviours [Berberich et al. 2020]. Nevertheless, embeddings could serve to generate data-driven predictive control laws for nonlinear system within confined scheduling sets, extending the idea of the scheduling proxy representation from [Cisneros, Voss, and Werner 2016; Morato, Normey-Rico, and Sename 2022b] to the data-driven context.

This is the end.

Bibliography

Abbas, Hossam S et al. (2014). "Embedding of nonlinear systems in a linear parameter-varying representation." In: *IFAC Proceedings Volumes* 47.3, pp. 6907–6913 (cit. on p. 26).

Abbas, Hossam S et al. (2015). "An MPC approach for LPV systems in input-output form." In: *2015 54th IEEE Conference on Decision and Control (CDC)*. IEEE, pp. 91–96 (cit. on pp. 15, 111, 114, 119, 239).

Abbas, Hossam S et al. (2016). "A robust MPC for input-output LPV models." In: *IEEE Transactions on Automatic Control* 61.12, pp. 4183–4188 (cit. on pp. 15, 92, 111, 239).

Abbas, Hossam S et al. (2018). "A new approach to robust MPC design for LPV systems in input-output form." In: *IFAC-PapersOnLine* 51.26, pp. 112–117 (cit. on pp. 111, 114, 119).

Abbas, Hossam Seddik et al. (2019). "Tube-based model predictive control for linear parameter-varying systems with bounded rate of parameter variation." In: *Automatica* 107, pp. 21–28 (cit. on pp. 11, 67, 88, 234, 251, 255).

Alamir, Mazen (2013). *A pragmatic story of model predictive control: self-contained algorithms and case-studies*. Createspace Independent Publishing Platform (cit. on pp. 4, 37, 231).

Alamo, Teodoro, José Manuel Bravo, and Eduardo F Camacho (2005). "Guaranteed state estimation by zonotopes." In: *Automatica* 41.6, pp. 1035–1043 (cit. on p. 134).

Alcalá, Eugenio, Vicenç Puig, and Joseba Quevedo (2019). "LPV-MPC control for autonomous vehicles." In: *IFAC-PapersOnLine* 52.28, pp. 106–113 (cit. on pp. 14, 51, 58, 133, 136, 238, 245, 248).

Alessio, Alessandro et al. (2007). "Squaring the circle: An algorithm for generating polyhedral invariant sets from ellipsoidal ones." In: *Automatica* 43.12, pp. 2096–2103 (cit. on p. 34).

Allan, Douglas A et al. (2017). "On the inherent robustness of optimal and suboptimal nonlinear MPC." In: *Systems & Control Letters* 106, pp. 68–78 (cit. on pp. 5, 232).

Allgöwer, Frank and Alex Zheng (2012). *Nonlinear model predictive control*. Vol. 26. Birkhäuser (cit. on pp. 5, 9, 23, 52, 177, 180–183, 233, 245).

Ampuño, Gary et al. (2019). "Apparent delay analysis for a flat-plate solar field model designed for control purposes." In: *Solar Energy* 177, pp. 241–254 (cit. on pp. 172, 173, 175).

Andersson, Joel AE et al. (2019). "CasADi: a software framework for nonlinear optimization and optimal control." In: *Mathematical Programming Computation* 11.1, pp. 1–36 (cit. on pp. 10, 233).

Apkarian, Pierre and Hoang Duong Tuan (2000). "Parameterized LMIs in control theory." In: *SIAM journal on control and optimization* 38.4, pp. 1241–1264 (cit. on p. 222).

Ayala, Claudio O et al. (2011). "Local model predictive controller in a solar desalination plant collector field." In: *Renewable energy* 36.11, pp. 3001–3012 (cit. on pp. 14, 238).

Bachnas, AA et al. (2014). "A review on data-driven linear parameter-varying modeling approaches: A high-purity distillation column case study." In: *Journal of Process Control* 24.4, pp. 272–285 (cit. on pp. 13, 15, 111, 235, 239).

Bastos, Saulo B et al. (2021). "The COVID-19 (SARS-CoV-2) Uncertainty Tripod in Brazil: Assessments on model-based predictions with large under-reporting." In: *Alexandria Engineering Journal* 60.5, pp. 4363–4380 (cit. on p. 212).

Beal, Craig Earl and J Christian Gerdes (2013). "Model predictive control for vehicle stabilization at the limits of handling." In: *IEEE Transactions on Control Systems Technology* 21.4, pp. 1258–1269 (cit. on p. 93).

Berberich, Julian et al. (2020). "Data-driven model predictive control with stability and robustness guarantees." In: *IEEE Transactions on Automatic Control* (cit. on p. 194).

Bernardi, Emanuel (2021). "Fault-tolerant Model-based Predictive Control Applied to Industrial Processes." PhD thesis. Universidad Nacional del Litoral (cit. on p. 59).

Bernardi, Emanuel et al. (2021). "Fault-tolerant energy management for an industrial microgrid: A compact optimization method." In: *International Journal of Electrical Power & Energy Systems* 124, p. 106342 (cit. on pp. 17, 189, 212, 241, 262).

Besselmann, Thomas, Johan Löfberg, and Manfred Morari (2009). "Explicit LPV-MPC with bounded rate of parameter variation." In: *IFAC Proceedings Volumes* 42.6, pp. 7–12 (cit. on p. 158).

Besselmann, Thomas, Johan Lofberg, and Manfred Morari (2012). "Explicit MPC for LPV systems: Stability and optimality." In: *IEEE Transactions on Automatic Control* 57.9, pp. 2322–2332 (cit. on pp. 37, 41).

Bitsoris, Georges (1988). "Positively invariant polyhedral sets of discrete-time linear systems." In: *International Journal of Control* 47.6, pp. 1713–1726 (cit. on p. 34).

Blanchini, Franco and Stefano Miani (2008). *Set-theoretic methods in control.* Springer (cit. on p. 32).

Boccia, Andrea, Lars Grüne, and Karl Worthmann (2014). "Stability and feasibility of state constrained MPC without stabilizing terminal constraints." In: *Systems & control letters* 72, pp. 14–21 (cit. on p. 127).

Boyd, Stephen et al. (1994). *Linear matrix inequalities in system and control theory.* SIAM (cit. on pp. 23, 26, 221).

Branco, Alice F et al. (2019). "Tools for the control of modern solar-thermal heating plants." In: *Proceedings of the* 14^{th} *Brazilian Symposium of Inteligent Automation (SBAI).* SBA, p. 104234 (cit. on p. 209).

Brezas, Panos, Malcolm C Smith, and Will Hoult (2015). "A clipped-optimal control algorithm for semi-active vehicle suspensions: Theory and experimental evaluation." In: *Automatica* 53, pp. 188–194 (cit. on p. 93).

Briat, Corentin (2008). "Robust control and observation of LPV time-delay systems." PhD thesis. Institut National Polytechnique de Grenoble (cit. on pp. 23, 24, 28, 223).

Brunner, Florian D, Mircea Lazar, and Frank Allgöwer (2013). "An explicit solution to constrained stabilization via polytopic tubes." In: *52nd IEEE Conference on Decision and Control.* IEEE, pp. 7721–7727 (cit. on pp. 14, 238).

Bumroongsri, Pornchai (2014). "An offline formulation of MPC for LPV systems using linear matrix inequalities." In: *Journal of Applied Mathematics* 2014 (cit. on pp. 14, 158, 163, 238).

Camacho, Eduardo F and Carlos Bordons (2007). "Nonlinear model predictive control: An introductory review." In: *Assessment and future directions of nonlinear model predictive control.* Springer, pp. 1–16 (cit. on pp. 52, 245).

Camacho, Eduardo F. and Carlos Bordons (2013). *Model Predictive Control.* Springer Science & Business Media (cit. on pp. 4, 23, 231).

Camacho, Eduardo F et al. (2012). “Control Issues in Solar Systems.” In: *Control of Solar Energy Systems*. Springer, pp. 25–47 (cit. on pp. 172, 175).

Cao, Y-Y and Zongli Lin (2005). “Min-max MPC algorithm for LPV systems subject to input saturation.” In: *IET Proceedings-Control Theory and Applications* 152.3, pp. 266–272 (cit. on pp. 158, 163, 177, 178, 180–183, 261).

Casavola, Alessandro, Domenico Famularo, and Giuseppe Franzè (2003). “Predictive control of constrained nonlinear systems via LPV linear embeddings.” In: *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal* 13.3-4, pp. 281–294 (cit. on pp. 58, 248).

Casella, Francesco and Marco Lovera (2008). “LPV/LFT modelling and identification: overview, synergies and a case study.” In: *IEEE International Conference on Computer-Aided Control Systems*. IEEE, pp. 852–857 (cit. on pp. 13, 235).

Cavanini, Luca, Gianluca Ippoliti, and Eduardo F Camacho (2021). “Model predictive control for a linear parameter varying model of an UAV.” In: *Journal of Intelligent & Robotic Systems* 101.3, pp. 1–18 (cit. on pp. 58, 248).

Chisci, Luigi, Paola Falugi, and Giovanni Zappa (2003). “Gain-scheduling MPC of nonlinear systems.” In: *International Journal of Robust and Nonlinear Control* 13.3-4, pp. 295–308 (cit. on pp. 128, 139).

Cisneros, P G et al. (2019). *Plenary at the* 3^{rd} *IFAC Workshop on Linear Parameter Varying Systems: Fast Nonlinear MPC via quasi-LPV Models*. IFAC. URL: `https://lpvs2019.tue.nl/plenary-speakers/` (cit. on pp. 69, 252).

Cisneros, Pablo Gonzalez and Herbert Werner (2017a). “Fast nonlinear MPC for reference tracking subject to nonlinear constraints via quasi-LPV representations.” In: *IFAC-PapersOnLine* 50.1, pp. 11601–11606 (cit. on pp. 67, 116, 159, 251).

Cisneros, Pablo S G, Aadithyan Sridharan, and Herbert Werner (2018). “Constrained Predictive Control of a Robotic Manipulator using quasi-LPV Representations.” In: *IFAC-PapersOnLine* 51.26, pp. 118–123 (cit. on pp. 51, 67, 246, 251).

Cisneros, Pablo S G, Sophia Voss, and Herbert Werner (2016). “Efficient nonlinear model predictive control via quasi-LPV representation.” In: *IEEE Conference on Decision and Control*. IEEE, pp. 3216–3221 (cit. on pp. 15, 67, 159, 161, 194, 238, 251).

Cisneros, Pablo S G and Herbert Werner (2017b). “Parameter-dependent stability conditions for quasi-LPV Model Predictive Control.” In: *American Control Conference*. IEEE, pp. 5032–5037 (cit. on pp. 15, 58, 238, 239, 248).

— (2018). “A dissipativity formulation for stability analysis of nonlinear and parameter dependent MPC.” In: *American Control Conference*. IEEE, pp. 3894–3899 (cit. on pp. 15, 46, 80, 163, 170, 171, 239).

— (2020). “Nonlinear model predictive control for models in quasi-linear parameter varying form.” In: *International Journal of Robust and Nonlinear Control* 30.10, pp. 3945–3959 (cit. on pp. 15, 16, 51, 54, 67, 69, 71, 76, 78, 82–84, 88, 128, 178, 180–183, 193, 237, 239, 240, 246, 251, 252, 255, 260, 261, 264).

Cisneros, Pablo SG and Herbert Werner (2019). “Wide range stabilization of a pendubot using quasi-LPV predictive control.” In: *IFAC-PapersOnLine* 52.28, pp. 164–169 (cit. on pp. 15, 51, 67, 81, 86, 87, 238, 246, 251).

Clarke, David W, Coorous Mohtadi, and PS Tuffs (1987). "Generalized predictive control: Part I. The basic algorithm." In: *Automatica* 23.2, pp. 137–148 (cit. on pp. 4, 15, 231).

Cunha, Victor M and Tito Luis Maia Santos (2021). "Robust Nonlinear Model Predictive Control with Bounded Disturbances Based on Zonotopic Constraint Tightening." In: *Journal of Control, Automation and Electrical Systems*, pp. 1–14 (cit. on pp. 104, 128, 134–136, 148).

Cutler, Charles R and Brian L Ramaker (1980). "Dynamic matrix control: a computer control algorithm." In: *Joint Automatic Control Conference*. 17, p. 72 (cit. on pp. 4, 231).

Darby, Mark L and Michael Nikolaou (2012). "MPC: Current practice and challenges." In: *Control Engineering Practice* 20.4, pp. 328–342 (cit. on p. 15).

Doyle, John C, Joseph E Wall, and Gunter Stein (1982). "Performance and robustness analysis for structured uncertainty." In: *1982 21st IEEE conference on decision and control*. IEEE, pp. 629–636 (cit. on pp. 12, 234).

Duan, Guang-Ren and Hai-Hua Yu (2013). *LMIs in control systems: analysis, design and applications*. CRC press (cit. on p. 222).

Emedi, Zlatko and Alireza Karimi (2016). "Fixed-structure LPV discrete-time controller design with induced $\mathcal{L}_2$-norm and $\mathcal{H}_2$ performance." In: *International Journal of Control* 89.3, pp. 494–505 (cit. on pp. 13, 235).

Englert, Tobias et al. (2019). "A software framework for embedded nonlinear model predictive control using a gradient-based augmented Lagrangian approach (GRAMPC)." In: *Optimization and Engineering* 20.3, pp. 769–809 (cit. on pp. 10, 182, 233).

Ferramosca, Antonio (2011). "Model Predictive Control of Systems with Changing Set-points." PhD thesis. Universidad de Sevilla (cit. on pp. 23, 37).

Fiacchini, Mirko, Teodoro Alamo, and Eduardo F Camacho (2010). "On the computation of convex robust control invariant sets for nonlinear systems." In: *Automatica* 46.8, pp. 1334–1338 (cit. on p. 33).

Fischer, Daniel and Rolf Isermann (2004). "Mechatronic semi-active and active vehicle suspensions." In: *Control engineering practice* 12.11, pp. 1353–1367 (cit. on pp. 92, 100).

Froisy, J Brian (2006). "Model predictive control—building a bridge between theory and practice." In: *Computers & chemical engineering* 30.10-12, pp. 1426–1435 (cit. on pp. 15, 239).

Gros, Sébastien et al. (2020). "From Linear to Nonlinear MPC: Bridging the Gap via the Real-time Iteration." In: *International Journal of Control* 93.1, pp. 62–80 (cit. on pp. 10, 52, 233, 245).

Grüne, Lars and Vryan Gil Palma (2015). "Robustness of performance and stability for multistep and updated multistep MPC schemes." In: *Discrete & Continuous Dynamical Systems* 35.9, p. 4385 (cit. on p. 47).

Grüne, Lars and Jürgen Pannek (2017). "Nonlinear model predictive control." In: *Nonlinear Model Predictive Control*. Springer, pp. 45–69 (cit. on pp. 5, 46).

Grüne, Lars, Simon Pirkelmann, and Marleen Stieler (2018). "Strict dissipativity implies turnpike behavior for time-varying discrete time optimal control problems." In: *Control systems and mathematical methods in economics*. Springer, pp. 195–218 (cit. on p. 47).

Grüne, Lars et al. (2010). "Analysis of unconstrained nonlinear MPC schemes with time varying control horizon." In: *SIAM Journal on Control and Optimization* 48.8, pp. 4938–4962 (cit. on p. 47).

Guo, Shuqi, Shaopu Yang, and Cunzhi Pan (2006). "Dynamic modeling of magnetorheological damper behaviors." In: *Journal of Intelligent material systems and structures* 17.1, pp. 3–14 (cit. on p. 97).

Hanema, Jurre, Mircea Lazar, and Roland Tóth (2020). "Heterogeneously parameterized tube model predictive control for LPV systems." In: *Automatica* 111, p. 108622 (cit. on pp. 14, 238).

Hanema, Jurre, Roland Tóth, and Mircea Lazar (2017). "Stabilizing non-linear MPC using linear parameter-varying representations." In: 56^{th} *Annual Conference on Decision and Control*. IEEE, pp. 3582–3587 (cit. on p. 136).

— (2021). "Stabilizing non-linear model predictive control using linear parameter-varying embeddings and tubes." In: *IET Control Theory & Applications* 15.10, pp. 1404–1421 (cit. on pp. 56, 88, 116, 128, 255).

He, De-Feng, Hua Huang, and Qiu-Xia Chen (2014). "Quasi-min–max MPC for constrained nonlinear systems with guaranteed input-to-state stability." In: *Journal of the Franklin Institute* 351.6, pp. 3405–3423 (cit. on p. 162).

Heath, W P, Adrian G Wills, and J A G Akkermans (2005). "A sufficient condition for the stability of optimizing controllers with saturating actuators." In: *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal* 15.12, pp. 515–529 (cit. on p. 168).

Hermelin, Brita and Malin Henriksson (2022). *Transport and Mobility Planning for Sustainable Development* (cit. on pp. 18, 242).

Hespe, Christian and Herbert Werner (2021). "Convergence Properties of Fast quasi-LPV Model Predictive Control." In: *2021 60th IEEE Conference on Decision and Control (CDC)*. IEEE, pp. 3869–3874 (cit. on pp. 15, 69–72).

Hoffmann, Christian and Herbert Werner (2014). "A survey of linear parameter-varying control applications validated by experiments or high-fidelity simulations." In: *IEEE Transactions on Control Systems Technology* 23.2, pp. 416–433 (cit. on pp. 12, 26, 52, 59, 112, 234).

Houska, Boris, Hans Joachim Ferreau, and Moritz Diehl (2011). "An Auto-generated Real-time Iteration Algorithm for Nonlinear MPC in the Microsecond Range." In: *Automatica* 47.10, pp. 2279–2285 (cit. on pp. 10, 233).

International Organization for Standardization (Nov. 2016). *ISO 8608:2016: Measurement and Evaluation of mechanical vibration and shock as applied to machines, vehicles and structures*. Standard (cit. on pp. 100, 108).

Jiang, Zhong-Ping and Yuan Wang (2001). "Input-to-state stability for discrete-time nonlinear systems." In: *Automatica* 37.6, pp. 857–869 (cit. on pp. 30, 162).

Johansson, Karl Henrik (2000). "The Quadruple-tank Process: A Multivariable Laboratory Process with an Adjustable Zero." In: *IEEE Transactions on Control Systems Technology* 8.3, pp. 456–465 (cit. on pp. 18, 145).

Jungers, Marc and Eugênio B Castelan (2011). "Gain-scheduled output control design for a class of discrete-time nonlinear systems with saturating actuators." In: *Systems & Control Letters* 60.3, pp. 169–173 (cit. on p. 46).

Jungers, Marc, Ricardo CLF Oliveira, and Pedro LD Peres (2011). "MPC for LPV systems with bounded parameter variations." In: *International Journal of Control* 84.1, pp. 24–36 (cit. on pp. 14, 94, 158, 238).

Jungers, Marc et al. (2009). "Model predictive control for linear parameter varying systems using path-dependent Lyapunov functions." In: *IFAC Proceedings Volumes* 42.2, pp. 97–102 (cit. on p. 76).

Käpernick, Bartosz and Knut Graichen (2014). "The gradient based nonlinear model predictive control software GRAMPC." In: *European Control Conference*. IEEE, pp. 1170–1175 (cit. on pp. 10, 233).

Kay Lup, Andrew Ng et al. (2023). "Sustainable energy technologies for the Global South: Challenges and solutions toward achieving SDG 7." In: *Environmental Sciences: Advances*, pp. – (cit. on p. 213).

Khalil, Hassan K and Jessy W Grizzle (2002). *Nonlinear systems*. Vol. 3. Prentice hall Upper Saddle River, NJ (cit. on pp. 23, 32, 219, 220).

Köhler, Johannes (2021). "Analysis and design of MPC frameworks for dynamic operation of nonlinear constrained systems." PhD thesis. Universitat Stuttgart (cit. on p. 23).

Köhler, Johannes, Frank Allgöwer, and Matthias A Müller (2019). "A simple framework for nonlinear robust output-feedback MPC." In: 18^{th} *European Control Conference (ECC)*. IEEE, pp. 793–798 (cit. on p. 45).

Köhler, Johannes, Matthias A Müller, and Frank Allgöwer (2018). "A novel constraint tightening approach for nonlinear robust model predictive control." In: *American Control Conference (ACC)*. IEEE, pp. 728–734 (cit. on pp. 11, 104, 128, 137, 234).

— (2019). "A nonlinear model predictive control framework using reference generic terminal ingredients." In: *IEEE Transactions on Automatic Control* 65.8, pp. 3576–3583 (cit. on pp. 130, 139).

— (2020). "A nonlinear tracking model predictive control scheme for dynamic target signals." In: *Automatica* 118, p. 109030 (cit. on pp. 127, 128, 136, 137, 151).

Köhler, Johannes et al. (2020). "A computationally efficient robust model predictive control framework for uncertain nonlinear systems." In: *IEEE Transactions on Automatic Control* (cit. on pp. 11, 233).

Lazar, Mircea et al. (2008). "On input-to-state stability of min–max nonlinear model predictive control." In: *Systems & Control Letters* 57.1, pp. 39–48 (cit. on pp. 162, 166).

Li, Dewei and Yugeng Xi (2010). "The feedback robust MPC for LPV systems with bounded rates of parameter changes." In: *IEEE Transactions on Automatic Control* 55.2, pp. 503–507 (cit. on pp. 158, 177, 178, 180–183, 261).

Limón, Daniel et al. (2006a). "Input to state stability of min–max MPC controllers for nonlinear systems with bounded uncertainties." In: *Automatica* 42.5, pp. 797–803 (cit. on pp. 11, 162, 165, 234).

— (2006b). "On the stability of constrained MPC without terminal constraint." In: *IEEE transactions on automatic control* 51.5, pp. 832–836 (cit. on p. 41).

Limon, Daniel et al. (2018). "Nonlinear MPC for Tracking Piece-wise Constant Reference Signals." In: *IEEE Transactions on Automatic Control* 63.11, pp. 3735–3750 (cit. on pp. 5, 104, 117, 127, 130, 131, 136–139, 141, 145, 147, 149–151, 153, 155, 259).

Ljung, Lennart (1987). *System identification: theory for the user*. Prentice-hall (cit. on p. 64).

Löfberg, Johan (2012). "Oops! I cannot do it again: Testing for recursive feasibility in MPC." In: *Automatica* 48.3, pp. 550–555 (cit. on pp. 11, 166, 234).

Löwy, Michael (2015). *Ecosocialism: A radical alternative to capitalist catastrophe*. Haymarket Books (cit. on pp. 18, 241).

Magni, Lalo, Davide Martino Raimondo, and Riccardo Scattolini (2006). "Regional input-to-state stability for nonlinear model predictive control." In: *IEEE Transactions on automatic control* 51.9, pp. 1548–1553 (cit. on pp. 31, 162, 165).

Marruedo, Daniel Limón (2002). "Control predictivo de sistemas no lineales con restricciones: estabilidad y robustez." PhD thesis. Universidad de Sevilla (cit. on pp. 23, 42, 43, 45, 219).

Mate, S. et al. (2019). "A Stabilizing Sub-Optimal Model Predictive Control For Quasi-Linear Parameter Varying Systems." In: *IEEE Control Systems Letters* (cit. on pp. 14, 238).

Mayne, David Q et al. (2000). "Constrained model predictive control: Stability and optimality." In: *Automatica* 36.6, pp. 789–814 (cit. on pp. 41–44, 92).

Medero, Ariel et al. (2022). "MPC-based optimal parameter scheduling of LPV controllers: Application to Lateral ADAS Control." In: *Proceedings of the* 30^{th} *Mediterranean Conference of Control and Automation* (cit. on pp. 25, 187, 189, 211, 262).

Megretski, Alexandre and Anders Rantzer (1997). "System analysis via integral quadratic constraints." In: *IEEE Transactions on Automatic Control* 42.6, pp. 819–830 (cit. on p. 169).

Mohammadpour, Javad and Carsten W. Scherer (2012). *Control of linear parameter varying systems with applications*. Springer Science & Business Media (cit. on pp. 12, 23, 24, 161, 234).

Morato, Marcelo Menezes (2023). "A Robust Model Predictive Control Algorithm for Input-Output LPV Systems using Parameter Extrapolation." [Pre-Print, Submitted] (cit. on pp. 51, 91, 116, 118, 214, 246, 256).

Morato, Marcelo Menezes, Emanuel Bernardi, and Vladimir Stojanovic (2021). "A qLPV Nonlinear Model Predictive Control with Moving Horizon Estimation." In: *Complex Engineering Systems* 1.1, p. 5 (cit. on p. 213).

Morato, Marcelo Menezes, Tobias Holicki, and Carsten W. Scherer (2023). "Stabilizing Model Predictive Control Synthesis using Integral Quadratic Constraints and Full-Block Multipliers." [Pre-Print, Submitted] (cit. on pp. 157, 214).

Morato, Marcelo Menezes, Julio Normey-Rico, and Olivier Sename (2021a). "Short-Sighted Robust LPV Model Predictive Control: Application to Semi-Active Suspension Systems." In: *Proceedings of the European Control Conference 2021 (ECC21)*. European Control Association (cit. on p. 210).

— (2022a). "A Tracking Model Predictive Control for Input-Output LPV Systems using Parameter Extrapolation." In: *Proceedings of the* 5^{th} *IFAC Workshop on Linear Parameter-Varying Systems (LPVS)*. IFAC (cit. on pp. 51, 91, 211, 246, 256).

Morato, Marcelo Menezes and Julio Elias Normey-Rico (2021). "A novel unified method for time-varying dead-time compensation." In: *ISA transactions* 108, pp. 78–95 (cit. on p. 212).

Morato, Marcelo Menezes, Julio Elias Normey-Rico, and Olivier Sename (2019). "Novel qLPV MPC design with least-squares scheduling prediction." In: *IFAC-PapersOnLine* 52.28, pp. 158–163 (cit. on pp. 51, 52, 63, 65, 81, 91, 94, 111, 128, 136, 159, 189, 209, 245, 246, 248, 256, 262).

— (2020a). "Model predictive control design for linear parameter varying systems: A survey." In: *Annual Reviews in Control* 49, pp. 64–80 (cit. on pp. 3, 13, 14, 52, 56, 58, 94, 128, 212, 235, 238, 245, 248).

Morato, Marcelo Menezes, Julio Elias Normey-Rico, and Olivier Sename (2020b). "Sub-optimal recursively feasible Linear Parameter-Varying predictive algorithm for semi-active suspension control." In: *IET Control Theory & Applications* 14.18, pp. 2764–2775 (cit. on pp. 85, 128, 212).

— (2020c). "Sub-optimal recursively feasible Linear Parameter-Varying predictive algorithm for semi-active suspension control." In: *IET Control Theory & Applications* 14.18, pp. 2764–2775 (cit. on pp. 91, 104, 105, 189, 256, 262).

— (2021b). "A fast dissipative robust nonlinear model predictive control procedure via quasi-linear parameter varying embedding and parameter extrapolation." In: *International Journal of Robust and Nonlinear Control* 31.18, pp. 9619–9651 (cit. on pp. 51, 111, 157, 213, 246, 260).

— (2021c). "An input-to-state stable model predictive control framework for Lipschitz nonlinear parameter varying systems." In: *International Journal of Robust and Nonlinear Control* 31.17, pp. 8239–8272 (cit. on pp. 6, 17, 51, 91, 189, 212, 241, 246, 256, 262).

— (2021d). "On the Robustness Properties of Gain-scheduled Unconstrained MPC for LPV Systems." In: *Simpósio Brasileiro de Automação Inteligente-SBAI*. Vol. 1. 1 (cit. on pp. 157, 167, 171, 210, 260).

— (2021e). "Síntese Estruturada para Controladores Preditivos LPV." In: *Simpósio Brasileiro de Automação Inteligente-SBAI*. Vol. 1. 1 (cit. on p. 211).

— (2022b). "Sufficient Conditions for Convergent Recursive Extrapolation of qLPV Scheduling Parameters along a Prediction Horizon." In: *IEEE Transactions on Automatic Control*, pp. 1–12 (cit. on pp. 51, 52, 72, 76, 77, 128, 191, 194, 213, 246, 252).

— (2023a). "Nonlinear Data-Driven Control Part I: Trajectory Representation under quasi-Linear Parameter Varying Embeddings." [Pre-Print, Submitted] (cit. on pp. 51, 214, 246).

— (2023b). "Nonlinear Data-Driven Control Part II: qLPV Predictive Control using Parameter Extrapolation." [Pre-Print, Submitted] (cit. on p. 214).

— (2023c). "Robustness Analysis of Gain-scheduled MPC for LPV Systems: An Integral Quadratic Constraints Approach." [Pre-Print, Submitted] (cit. on pp. 157, 214, 260).

— (2023d). "Sufficient Conditions for Convergent Recursive Extrapolation of qLPV Scheduling Parameters along a Prediction Horizon." In: *IEEE Transactions on Automatic Control*. [Accepted, To Appear] (cit. on pp. 115, 116).

Morato, Marcelo Menezes, Tito Santos, and Julio Elias Normey-Rico (2022). "Explicit Dead-Time Compensation in Linear Parameter Varying Model Predictive Control." In: *18th IFAC Workshop on Control Applications of Optimization* (cit. on p. 211).

Morato, Marcelo Menezes, Olivier Sename, and Luc Dugard (2018). "LPV-MPC Fault Tolerant Control of Automotive Suspension Dampers." In: *IFAC-PapersOnLine* 51.26, pp. 31–36 (cit. on pp. 51, 93, 133, 245).

— (2019a). "Design and Analysis of Several State-Feedback Fault-Tolerant Control Strategies for Semi-Active Suspensions." In: *IFAC-PapersOnLine* 52.17, pp. 48–53 (cit. on pp. 107, 209).

— (2019b). "Nonlinear Fault Estimation Methods for Semi-Active Suspension Dampers." In: *Proceedings of the* 14^{th} *Brazilian Symposium of Inteligent Automation (SBAI)*. SBA, pp. 13–18 (cit. on pp. 91, 209, 256).

Morato, Marcelo Menezes and Vladimir Stojanovic (2021). "A robust identification method for stochastic nonlinear parameter varying systems." In: *Mathematical Modelling and Control* 1.1, pp. 35–51 (cit. on p. 213).

Morato, Marcelo Menezes et al. (2018a). "Design of a fast real-time LPV model predictive control system for semi-active suspension control of a full vehicle." In: *Journal of the Franklin Institute* (cit. on pp. 6, 9, 17, 85, 93–95, 100, 107, 109, 211, 241).

Morato, Marcelo Menezes et al. (2018b). "Future hybrid local energy generation paradigm for the brazilian sugarcane industry scenario." In: *International Journal of Electrical Power & Energy Systems* 101, pp. 139–150 (cit. on pp. 18, 241).

Morato, Marcelo Menezes et al. (2019a). "Automation and Renewable Energy: Outreach Efforts in Brazilian Public Schools." In: *Proceedings of the ISES Solar World Congress 2019*. ISES (cit. on pp. 189, 210).

Morato, Marcelo Menezes et al. (2019b). "Fault estimation for automotive Electro-Rheological dampers: LPV-based observer approach." In: *Control Engineering Practice* 85, pp. 11–22 (cit. on pp. 17, 97, 98, 102, 211, 241).

Morato, Marcelo Menezes et al. (2020a). "A parametrized nonlinear predictive control strategy for relaxing COVID-19 social distancing measures in Brazil." In: *ISA transactions* (cit. on pp. 6, 212).

Morato, Marcelo Menezes et al. (2020b). "A Two-Layer EMS for Cooperative Sugarcane-based Microgrids." In: *International Journal of Electrical Power & Energy Systems* 118, p. 105752 (cit. on pp. 7, 8, 189, 212, 262).

Morato, Marcelo Menezes et al. (2020c). "An optimal predictive control strategy for COVID-19 (SARS-CoV-2) social distancing policies in Brazil." In: *Annual Reviews in Control* 50, pp. 417–431 (cit. on pp. 6, 188, 212).

Morato, Marcelo Menezes et al. (2020d). "Development of a simple ER damper model for fault-tolerant control design." In: *Journal of the Brazilian Society of Mechanical Sciences and Engineering* 42.10, pp. 1–22 (cit. on p. 212).

Morato, Marcelo Menezes et al. (2020e). "LPV-MPC fault-tolerant energy management strategy for renewable microgrids." In: *International Journal of Electrical Power & Energy Systems* 117, p. 105644 (cit. on pp. 7, 51, 56, 172, 189, 211, 245, 246, 262).

Morato, Marcelo Menezes et al. (2020f). "Optimal Control Concerns Regarding the COVID-19 (SARS-CoV-2) Pandemic in Bahia and Santa Catarina, Brazil." In: *Proceedings of the XXIII Brazilian Congress of Automatica (CBA)*. SBA (cit. on p. 210).

Morato, Marcelo Menezes et al. (2020g). "Sub-optimal linear parameter varying model predictive control for solar collectors." In: *Proceedings of the* 21^{rst} *IEEE International Conference on Industrial Technology (ICIT)*. IEEE, pp. 95–100 (cit. on p. 210).

Morato, Marcelo Menezes et al. (2021a). "A Sequential Quadratic Programming Approach for the Predictive Control of the COVID-19 Spread." In: *Proceedings of the* 11^{th} *IFAC Symposium on Biological and Medical Systems (BMS)*. IFAC (cit. on pp. 191, 210).

Morato, Marcelo Menezes et al. (2021b). "Assessing Demand Compliance and Reliability in the Philippine Off-Grid Islands with Model Predictive Control Microgrid Coordination." In: *Renewable Energy* (cit. on pp. 7, 213).

Morato, Marcelo Menezes et al. (2021c). “Fault-Tolerant Energy Management in Renewable Microgrids using LPV MPC.” In: *Proceedings of the* 5^{th} *IFAC Workshop on Linear Parameter-Varying Systems (LPVS)*. IFAC (cit. on pp. 187, 189, 211, 262).

Morato, Marcelo Menezes et al. (2021d). “NMPC Through qLPV Embedding: A Tutorial Review of Different Approaches.” In: *Proceedings of the* 7^{th} *IFAC Conference on Nonlinear Model Predictive Control (NMPC)*. IFAC (cit. on pp. 90, 191, 210, 256).

Morato, Marcelo Menezes et al. (2021e). “Robust Nonlinear Predictive Control through qLPV embedding and Zonotope Uncertainty Propagation.” In: *Proceedings of the* 4^{th} *IFAC Workshop on Linear Parameter-Varying Systems (LPVS)*. IFAC (cit. on pp. 127, 210, 258).

Morato, Marcelo Menezes et al. (2022a). “A Predictive Fault Tolerant Control Method for qLPV Systems Subject to Input Faults and Constraints.” In: *Journal of the Franklin Institute* 359.16, pp. 9129–9167 (cit. on pp. 189, 213, 262).

Morato, Marcelo Menezes et al. (2022b). “NMPC via qLPV models and Taylor-based Scheduling Parameter Extrapolation: A Cartesian Robot Case Study.” In: *Proceedings of the* 30^{th} *Mediterranean Conference of Control and Automation* (cit. on pp. 90, 187, 191, 211, 256).

Morato, Marcelo Menezes et al. (2023a). “A Robust Nonlinear Tracking MPC using qLPV Embedding and Zonotopic Uncertainty Propagation.” [Pre-Print, Submitted] (cit. on pp. 51, 127, 214, 246, 258).

Morato, Marcelo Menezes et al. (2023b). “An Event-Triggered MPC Scheme for Nonlinear systems using qLPV Embeddings and Parameter Extrapolation.” [Pre-Print, Submitted] (cit. on p. 214).

Naspolini, Amir, Marcelo Menezes Morato, and Julio Elias Normey-Rico (2022). “LPV Predictive Control for Renewable Hydrogen Generation.” In: *Revista Electrónica de divulgación de STEM de la Facultad de Ingeniería de la Universidad Nacional de la Patagonia San Juan Bosco* 4.2, pp. 3–22 (cit. on p. 213).

Nathwani, Jatin and Daniel M. Kammen (2019). “Affordable Energy for Humanity: A Global Movement to Support Universal Clean Energy Access.” In: *Proceedings of the IEEE* 107.9, pp. 1780–1789 (cit. on pp. 18, 242).

Nguyen, Manh Quan (2016). “LPV approaches for modelling and control of vehicle dynamics: application toa small car pilot plant with ER dampers.” PhD thesis. Université Grenoble Alpes (cit. on p. 9).

Nguyen, Manh Quan et al. (2015). “A state feedback input constrained control design for a 4-semi-active damper suspension system: a quasi-LPV approach.” In: *IFAC-PapersOnLine* 48.14, pp. 259–264 (cit. on p. 93).

Nguyen, Manh-Quan et al. (2016). “A Model Predictive Control approach for semi-active suspension control problem of a full car.” In: *55th Conference on Decision and Control*. IEEE, pp. 721–726 (cit. on pp. 93, 100).

Normey-Rico, J.E. and E. Camacho (2007). *Control of dead-time processes*. Springer Science & Business Media (cit. on p. 39).

Normey-Rico, Julio Elias and Marcelo Menezes Morato (2021). *Introdução ao Controle de Processos*. Editora Blucher (cit. on p. 189).

— (2022). *Introducción al control de procesos*. Editora Blucher (cit. on p. 189).

Pasamontes, M et al. (2013). “Hybrid modeling of a solar-thermal heating facility.” In: *Solar Energy* 97, pp. 577–590 (cit. on pp. 172–175).

Pataro, Igor M L et al. (2021a). "A control framework to optimize public health policies in the course of the COVID-19 pandemic." In: *Scientific reports* 11, p. 13403 (cit. on pp. 188, 213).

Pataro, Igor M L et al. (2021b). "Optimal Control Approach for the COVID-19 Pandemic in Bahia and Santa Catarina, Brazil." In: *Journal of Control, Automation and Electrical Systems* (cit. on p. 213).

Peaucelle, Dimitri et al. (2012). "Evaluating regions of attraction of LTI systems with saturation in IQS framework." In: *IFAC Proceedings Volumes* 45.13, pp. 242–247 (cit. on p. 45).

Petsagkourakis, Panagiotis, William Heath, and Constantinos Theodoropoulos (2017). "IQC analysis of constrained MPC of large-scale systems." In: *Computer Aided Chemical Engineering*. Vol. 40. Elsevier, pp. 1627–1632 (cit. on p. 168).

Petsagkourakis, Panagiotis, William Paul Heath, and Constantinos Theodoropoulos (2020). "Stability analysis of piecewise affine systems with multi-model predictive control." In: *Automatica* 111, p. 108539 (cit. on p. 162).

Pipino, Hugo A et al. (2020a). "Formulación de un LPV-MPC Adaptativo para Procesos Industriales No Lineales." In: *Proceedings of the Congreso Argentino de Control Automático 2020*. AADECA (cit. on pp. 59, 210).

Pipino, Hugo A et al. (2020b). "Nonlinear temperature regulation of solar collectors with a fast adaptive polytopic LPV MPC formulation." In: *Solar Energy* 209, pp. 214–225 (cit. on pp. 17, 56, 157, 173, 189, 212, 241, 246, 260, 262).

Pluymers, Bert et al. (2005). "The efficient computation of polyhedral invariant sets for linear systems with polytopic uncertainty." In: *Proceedings of the 2005, American control conference, 2005.* IEEE, pp. 804–809 (cit. on p. 34).

Poussot-Vassal, Charles et al. (2008). "A new semi-active suspension control strategy through LPV technique." In: *Control Engineering Practice* 16.12, pp. 1519–1534 (cit. on p. 93).

Poussot-Vassal, Charles et al. (2012). "Survey and performance evaluation on some automotive semi-active suspension control methods: A comparative study on a single-corner model." In: *Annual Reviews in Control* 36.1, pp. 148–160 (cit. on pp. 92, 93, 95, 100, 109).

Primbs, James A, Vesna Nevistić, and John C Doyle (1999). "Nonlinear optimal control: A control Lyapunov function and receding horizon perspective." In: *Asian Journal of Control* 1.1, pp. 14–24 (cit. on p. 33).

Quirynen, Rien et al. (2015). "Autogenerating Microsecond Solvers for Nonlinear MPC: A Tutorial using ACADO Integrators." In: *Optimal Control Applications and Methods* 36.5, pp. 685–704 (cit. on pp. 10, 182, 183, 233, 260).

Rakovic, Saša V et al. (2012). "Parameterized tube model predictive control." In: *IEEE Transactions on Automatic Control* 57.11, pp. 2746–2761 (cit. on pp. 14, 238).

Rathai, Karthik Murali Madhavan et al. (2018). "A Parameterized NMPC scheme for embedded control of semi-active suspension system." In: *IFAC-PapersOnLine* 51.20, pp. 301–306 (cit. on pp. 52, 245).

Reble, Marcus and Frank Allgöwer (2012). "Unconstrained model predictive control and suboptimality estimates for nonlinear continuous-time systems." In: *Automatica* 48.8, pp. 1812–1817 (cit. on p. 47).

Rego, Brenner S et al. (2020). “Guaranteed methods based on constrained zonotopes for set-valued state estimation of nonlinear discrete-time systems.” In: *Automatica* 111, p. 108614 (cit. on p. 136).

Ren, Hongbin et al. (2016). “State observer-based sliding mode control for semi-active hydro-pneumatic suspension.” In: *Vehicle System Dynamics* 54.2, pp. 168–190 (cit. on p. 97).

Richter, Stefan, Colin Neil Jones, and Manfred Morari (2011). “Computational complexity certification for real-time MPC with input constraints based on the fast gradient method.” In: *IEEE Transactions on Automatic Control* 57.6, pp. 1391–1403 (cit. on pp. 10, 233).

Rodriguez-Guevara, Daniel et al. (2021). “Active suspension control using an MPC-LQR-LPVcontroller with attraction sets and quadratic stability conditions.” In: *Mathematics* 9.20, p. 2533 (cit. on pp. 58, 248).

Rotondo, Damiano (2016). “Advances in gain-scheduling and fault tolerant control techniques.” PhD thesis. Universitat Politecnica de Catalunya (cit. on p. 23).

— (2017). *Advances in gain-scheduling and fault tolerant control techniques.* Springer (cit. on p. 12).

Rotondo, Damiano, Vicenç Puig, and Fatiha Nejjari (2016). “On the analogies in control design of non-linear systems using LPV and Takagi-Sugeno models.” In: *5th International Conference on Systems and Control.* IEEE, pp. 225–230 (cit. on p. 12).

Sala, Antonio (2017). “Generalising quasi-LPV and CDI models to quasi-convex difference inclusions.” In: *IFAC-PapersOnLine* 50.1, pp. 7560–7565 (cit. on p. 26).

Santos, Tito Luis Maia (2011). “Contribuições para o controle preditivo com compensação de atraso robusta.” PhD thesis. Universidade Federal de Santa Catarina (cit. on pp. 23, 37).

Santos, Tito Luis Maia and Victor Moreira Cunha (2021). “Robust MPC for linear systems with bounded disturbances based on admissible equilibria sets.” In: *International Journal of Robust and Nonlinear Control* (cit. on pp. 11, 233).

Santos, Tito Luis Maia et al. (2019). “A constraint-tightening approach to nonlinear model predictive control with chance constraints for stochastic systems.” In: *American Control Conference (ACC).* IEEE, pp. 1641–1647 (cit. on pp. 11, 104, 128, 135, 136, 141, 234).

Savaresi, Sergio M et al. (2010). *Semi-active suspension control design for vehicles.* Butterworth-Heinemann. Elsevier (cit. on pp. 6, 17, 18, 92, 97, 99, 241).

Scherer, Carsten W. (2001). “LPV control and full block multipliers.” In: *Automatica* 37.3, pp. 361–375 (cit. on pp. 119, 169, 222).

— (2006). “LMI relaxations in robust control.” In: *European Journal of Control* 12.1, pp. 3–29 (cit. on pp. 12, 222, 223, 234).

— (2022). “Dissipativity, Convexity and Tight O’Shea-Zames-Falb Multipliers for Safety Guarantees.” In: *Proceedings of the* 25^{th} *International Symposium on Mathematical Theory of Networks and Systems (MTNS)*, pp. 1–8 (cit. on p. 34).

Scherer, Carsten W. and Siep Weiland (2000). “Linear matrix inequalities in control.” In: *Lecture Notes, Dutch Institute for Systems and Control, Delft, The Netherlands* 3.2 (cit. on p. 23).

Scokaert, Pierre OM, David Q Mayne, and James B Rawlings (1999). “Suboptimal model predictive control (feasibility implies stability).” In: *IEEE Transactions on Automatic Control* 44.3, pp. 648–654 (cit. on p. 41).

Scorletti, Gérard, Vincent Fromion, and Safta De Hillerin (2015). "Toward nonlinear tracking and rejection using LPV control." In: *IFAC-PapersOnLine* 48.26, pp. 13–18 (cit. on pp. 13, 161, 235).

Seiler, Peter, Andrew Packard, and Gary J Balas (2010). "A dissipation inequality formulation for stability analysis with integral quadratic constraints." In: *Proceedings of the* 49^{th} *IEEE Conference on Decision and Control*. IEEE, pp. 2304–2309 (cit. on p. 163).

Sename, Olivier, Peter Gaspar, and József Bokor (2013). *Robust control and linear parameter varying approaches: application to vehicle dynamics*. Vol. 437. Springer (cit. on pp. 12, 13, 23, 161, 218, 234, 235).

Shamma, Jeff S (2012). "An overview of LPV systems." In: *Control of linear parameter varying systems with applications*. Springer, pp. 3–26 (cit. on pp. 13, 25, 161).

Skibik, Terrence et al. (2021). "A Feasibility Governor for Enlarging the Region of Attraction of Linear Model Predictive Controllers." In: *IEEE Transactions on Automatic Control* (cit. on p. 137).

Skogestad, Sigurd and Ian Postlethwaite (2007). *Multivariable feedback control: analysis and design*. Vol. 2. Citeseer (cit. on p. 23).

Souza, Alex dos Reis de, Denis Efimov, and Tarek Raïssi (2021). "Robust output feedback MPC for LPV systems using interval observers." In: *IEEE Transactions on Automatic Control* 67.6, pp. 3188–3195 (cit. on p. 45).

Szabó, Zoltán and József Bokor (2018). "Transformations for linear parameter varying systems." In: *IFAC-PapersOnLine* 51.26, pp. 87–93 (cit. on p. 12).

Takagi, Tomohiro and Michio Sugeno (1985). "Fuzzy identification of systems and its applications to modeling and control." In: *IEEE transactions on systems, man, and cybernetics* 1, pp. 116–132 (cit. on p. 12).

Torrico, Bismark Claure et al. (2010). "Robust nonlinear predictive control applied to a solar collector field in a solar desalination plant." In: *IEEE Transactions on Control Systems Technology* 18.6, pp. 1430–1439 (cit. on p. 18).

Tóth, Roland (2010). *Modeling and identification of linear parameter-varying systems*. Vol. 403. Springer (cit. on pp. 12, 52, 234).

Tóth, Roland, Hossam Seddik Abbas, and Herbert Werner (2011). "On the state-space realization of LPV input-output models: Practical approaches." In: *IEEE transactions on control systems technology* 20.1, pp. 139–153 (cit. on p. 111).

Tóth, Roland et al. (2011). "The behavioral approach to linear parameter-varying systems." In: *IEEE Transactions on Automatic Control* 56.11, pp. 2499–2514 (cit. on p. 111).

Tseng, H Eric and Davor Hrovat (2015). "State of the art survey: active and semi-active suspension control." In: *Vehicle system dynamics* 53.7, pp. 1034–1062 (cit. on p. 93).

Tudón-Martínez, Juan C et al. (2015). "Adaptive road profile estimation in semiactive car suspensions." In: *IEEE Transactions on Control Systems Technology* 23.6, pp. 2293–2305 (cit. on p. 102).

Unger, Andreas et al. (2013). "Application of LQ-based semi-active suspension control in a vehicle." In: *Control Engineering Practice* 21.12, pp. 1841–1850 (cit. on pp. 93, 97, 102).

United Nations (2018). "The 2030 Agenda and the Sustainable Development Goals: An opportunity for Latin America and the Caribbean (LC/G.2681-P/Rev.3)." In: (cit. on pp. 18, 241).

United Nations General Assembly (2015). *Transforming our world : the* 2030 *Agenda for Sustainable Development.* Tech. rep. 21 October 2015, A/RES/70/1. URL: https://www.refworld.org/docid/57b6e3e44.html (cit. on pp. 18, 241).

Vergara-Dietrich, J. D. et al. (2019). "Advanced Chance-Constrained Predictive Control for the Efficient Energy Management of Renewable Power Systems." English. In: *Journal of Process Control* (cit. on pp. 7, 175).

Yang, Weilin et al. (2016). "An optimal approach to output-feedback robust model predictive control of LPV systems with disturbances." In: *International Journal of Robust and Nonlinear Control* 26.15, pp. 3253–3273 (cit. on p. 144).

Yu, Shuyou et al. (2013). "Tube MPC scheme based on robust control invariant set with application to Lipschitz nonlinear systems." In: *Systems & Control Letters* 62.2, pp. 194–200 (cit. on pp. 11, 234).

Zhang, Yinyan, Shuai Li, and Liefa Liao (2019). "Near-optimal control of nonlinear dynamical systems: A brief survey." In: *Annual Reviews in Control* 47, pp. 71–80 (cit. on pp. 52, 182, 245).

Zhou, Kemin and John Comstock Doyle (1998). *Essentials of robust control.* Vol. 104. Prentice hall Upper Saddle River, NJ (cit. on p. 78).

Zuo, L and SA Nayfeh (2003). "Low order continuous-time filters for approximation of the ISO $2631-1$ human vibration sensitivity weightings." In: *Journal of sound and vibration* 265.2, pp. 459–465 (cit. on pp. 100, 108).

Appendix A

List of publications

In this Appendix, I present a broad overview of all scientific works that were derived from the developments of this thesis. Accordingly, I categorise the complete set of publications (and also those works that are already submitted, and are currently under review) in terms of their main content and application purpose. This is presented in Table A.1, whose notation refers to the list of works presented below[1].

The works that are marked in navy are those that are **directly** related to this thesis, meaning that they are publications linked to the thesis objectives (see Section 1.5). The other papers are "side works" that were developed along the course of my doctorate. Nevertheless, I stress that these complementary works are also related to this thesis in a way, since they also assess the development of MPC algorithms using LPV models, with different degrees of complexity. Again, I note that many of these papers were also developed jointly with fellow colleges from other research groups which I collaborated with along the four years of thesis.

- Conference proceedings:
  - 2019:
    - [C1] **7$^{\text{th}}$ IFAC Symposium on System Structure and Control (SSSC)**, 2019: Design and Analysis of Several State-Feedback Fault-Tolerant Control Strategies for Semi-Active Suspensions [Morato, Sename, and Dugard 2019a].
    - [C2] 14**º Simpósio Brasileiro de Automação Inteligente**[2] **(SBAI)**, 2019: Nonlinear Fault Estimation Methods for Semi-Active Suspension Dampers [Morato, Sename, and Dugard 2019b].
    - [C3] 14**º Simpósio Brasileiro de Automação Inteligente (SBAI)**, 2019: Tools for the control of modern solar-thermal heating plants [Branco et al. 2019].
    - [C4] **3$^{\text{rd}}$ IFAC Workshop on Linear Parameter Varying Systems (LPVS)**, 2019: Novel qLPV MPC Design with Least-Squares Scheduling Prediction[3] [Morato, Normey-Rico, and Sename 2019].

[1]For brevity, the following publications are presented just by title, journal or conference, and date. Many have been developed in collaboration with other colleges, and also with my supervisors. The complete bibliographical details, including all author names, is indicated at the references of this thesis.

[2]14$^{\text{th}}$ Brazilian Symposium on Intelligent Automation.

[3]Finalist in the Young Author Award category.

[C5] **ISES Solar World Congress**, 2019: Automation and Renewable Energy: Outreach Efforts in Brazilian Public Schools
[Morato et al. 2019a].

– 2020:

[C6] **IEEE International Conference on Industrial Technology (ICIT)**, 2020:
Sub-optimal Linear Parameter Varying Model Predictive Control for Solar Collectors
[Morato et al. 2020g].

[C7] **Argentinian Congress of Automatic Control (AADECA)**, 2020:
Formulación de un LPV-MPC Adaptativo para Procesos Industriales No Lineales
[Pipino et al. 2020a].

[C8] **XXIII Congresso Brasileiro de Automática**[4] **(CBA)**, 2020:
Optimal Control Concerns Regarding the COVID-19 (SARS-CoV-2) Pandemic in Bahia and Santa Catarina, Brazil[5]
[Morato et al. 2020f].

– 2021:

[C9] **European Control Conference (ECC)**, 2021:
Short-Sighted Robust LPV Model Predictive Control: Application to Semi-Active Suspension Systems
[Morato, Normey-Rico, and Sename 2021a].

[C10] **4th IFAC Workshop on Linear Parameter Varying Systems (LPVS)**, 2021: Robust Nonlinear Predictive Control through qLPV embedding and Zonotope Uncertainty Propagation
[Morato et al. 2021e].

[C11] **7th IFAC Conference on Nonlinear Model Predictive Control (NMPC)**, 2021:
NMPC Through qLPV Embedding: A Tutorial Review of Different Approaches
[Morato et al. 2021d].

[C12] **11th IFAC Symposium on Biological and Medical Systems (BMS)**, 2021:
A Sequential Quadratic Programming Approach for the Predictive Control of the COVID-19 Spread
[Morato et al. 2021a].

[C13] **Brazilian Symposium on Intelligent Automation (SBAI)**, 2021:
On the Robustness Properties of Gain-scheduled Unconstrained MPC for LPV Systems
[Morato, Normey-Rico, and Sename 2021d].

[4] XXIII Brazilian Congress of Automatica.
[5] Best Paper Award on the doctoral category.

[C14] **Brazilian Symposium on Intelligent Automation (SBAI)**, 2021:
Síntese Estruturada para Controladores Preditivos LPV
[Morato, Normey-Rico, and Sename 2021e].

- 2022:

[C15] **30th Mediterranean Conference on Control and Automation (MED)**, 2022:
NMPC via qLPV models and Taylor-based Scheduling Parameter Extrapolation: A Cartesian Robot Case Study
[Morato et al. 2022b].

[C16] **30th Mediterranean Conference on Control and Automation (MED)**, 2022:
MPC-based optimal parameter scheduling of LPV controllers: Application to Lateral ADAS Control
[Medero et al. 2022].

[C17] **18th IFAC Workshop on Control Applications of Optimization (CAO)**, 2022:
Explicit Dead-Time Compensation in Linear Parameter Varying Model Predictive Control
[Morato, Santos, and Normey-Rico 2022].

[C18] **5th IFAC Workshop on Linear Parameter Varying Systems (LPVS)**, 2022:
A Tracking Model Predictive Control for Input-Output LPV Systems using Parameter Extrapolation
[Morato, Normey-Rico, and Sename 2022a].

[C19] **5th IFAC Workshop on Linear Parameter Varying Systems (LPVS)**, 2022:
Fault-Tolerant Energy Management in Renewable Microgrids using LPV MPC
[Morato et al. 2021c].

- Journal papers:

  - 2019:

[J1] **Control Engineering Practice**, 2019:
Fault estimation for automotive Electro-Rheological dampers: LPV-based observer approach
[Morato et al. 2019b].

[J2] **Journal of the Franklin Institute**, 2019:
Design of a fast real-time LPV model predictive control system for semi-active suspension control of a full vehicle
[Morato et al. 2018a].

  - 2020:

[J3] **International Journal of Electrical Power & Energy Systems**, 2020:
LPV-MPC fault-tolerant energy management strategy for renewable microgrids
[Morato et al. 2020e].

[J4] **International Journal of Electrical Power & Energy Systems**, 2020:
A Two-Layer EMS for Cooperative Sugarcane-based Microgrids
[Morato et al. 2020b].

[J5] **Annual Reviews in Control**, 2020:
Model predictive control design for linear parameter varying systems: A survey
[Morato, Normey-Rico, and Sename 2020a].

[J6] **IET Control Theory & Applications**, 2020:
Sub-optimal recursively feasible Linear Parameter-Varying predictive algorithm for semi-active suspension control
[Morato, Normey-Rico, and Sename 2020b].

[J7] **International Journal of Robust and Nonlinear Control**, 2020:
An input-to-state stable model predictive control framework for Lipschitz nonlinear parameter varying systems
[Morato, Normey-Rico, and Sename 2021c].

[J8] **Solar Energy**, 2020:
Nonlinear temperature regulation of solar collectors with a fast adaptive polytopic LPV MPC formulation
[Pipino et al. 2020b].

[J9] **Annual Reviews in Control**, 2020: An optimal predictive control strategy for COVID-19 (SARS-CoV-2) social distancing policies in Brazil
[Morato et al. 2020c].

[J10] **ISA Transactions**, 2020: A parametrized nonlinear predictive control strategy for relaxing COVID-19 social distancing measures in Brazil
[Morato et al. 2020a].

[J11] **Journal of the Brazilian Society of Mechanical Sciences and Engineering**, 2020:
Development of a simple ER damper model for fault-tolerant control design
[Morato et al. 2020d].

– 2021:

[J12] **ISA Transactions**, 2021:
A novel unified method for time-varying dead-time compensation
[Morato and Normey-Rico 2021].

[J13] **Alexandria Engineering Journal**, 2021: The COVID-19 (SARS-CoV-2) Uncertainty Tripod in Brazil: Assessments on model-based predictions with large under-reporting
[Bastos et al. 2021].

[J14] **International Journal of Electrical Power & Energy Systems**, 2021:
Fault-tolerant energy management for an industrial microgrid: A compact optimization method
[Bernardi et al. 2021].

[J15] **Mathematical Modelling and Control**, 2021:
A robust identification method for stochastic nonlinear parameter varying sys-

tems
[Morato and Stojanovic 2021].

[J16] **Journal of Control, Automation and Electrical Systems**, 2021: Optimal Control Concerns Regarding the COVID-19 Pandemic in Bahia and Santa Catarina, Brazil
[Pataro et al. 2021b].

[J17] **Nature: Scientific Reports**, 2021:
A control framework to optimize public health policies in the course of the COVID-19 pandemic
[Pataro et al. 2021a].

[J18] **Renewable Energy**, 2021:
Assessing Demand Compliance and Reliability in the Philippine Off-Grid Islands with Model Predictive Control Microgrid Coordination
[Morato et al. 2021b].

[J19] **International Journal of Robust and Nonlinear Control**, 2021:
A fast dissipative robust nonlinear model predictive control procedure via quasi-linear parameter varying embedding and parameter extrapolation
[Morato, Normey-Rico, and Sename 2021b].

[J20] **Complex Engineering Systems**, 2021:
A qLPV Nonlinear Model Predictive Control with Moving Horizon Estimation
[Morato, Bernardi, and Stojanovic 2021].

– 2022:

[J21] **IEEE Transactions on Automatic Control**, 2022:
Sufficient Conditions for Convergent Recursive Extrapolation of qLPV Scheduling Parameters along a Prediction Horizon
[Morato, Normey-Rico, and Sename 2022b].

[J22] **Journal of the Franklin Institute**:
A Predictive Fault Tolerant Control Method for qLPV Systems Subject to Input Faults and Constraints
[Morato et al. 2022a].

[J23] **Revista Electrónica de divulgación de STEM de la Facultad de Ingeniería de la Universidad Nacional de la Patagonia San Juan Bosco**, 2022: LPV Predictive Control for Renewable Hydrogen Generation
[Naspolini, Morato, and Normey-Rico 2022].

– 2023:

[J24] **Environmental Science: Advances**, 2023: Sustainable energy technologies for the Global South: Challenges and solutions toward achieving SDG 7
[Kay Lup et al. 2023]

- List of submissions (under peer review):

[S1] **International Journal of Robust and Nonlinear Control**:
A Robust Nonlinear Tracking MPC using qLPV Embedding and Zonotopic Uncer-

tainty Propagation
[Morato et al. 2023a].

[S2] **Journal of Process Control**:
A Robust Model Predictive Control Algorithm for Input-Output LPV Systems using Parameter Extrapolation
[Morato 2023]

[S3] **International Journal of Robust and Nonlinear Control**:
Robustness Analysis of Gain-scheduled MPC for LPV Systems: An Integral Quadratic Constraints Approach
[Morato, Normey-Rico, and Sename 2023c].

[S4] **Journal of Control, Automation and Electrical Systems**:
Nonlinear Data-Driven Control Part I: Trajectory Representation under quasi-Linear Parameter Varying Embeddings
[Morato, Normey-Rico, and Sename 2023a]

[S5] **Journal of Control, Automation and Electrical Systems**:
Nonlinear Data-Driven Control Part II: qLPV Predictive Control using Parameter Extrapolation
[Morato, Normey-Rico, and Sename 2023b]

[S6] **International Journal of Robust and Nonlinear Control**:
Stabilizing Model Predictive Control Synthesis using Integral Quadratic Constraints and Full-Block Multipliers
[Morato, Holicki, and Scherer 2023].

[S7] **Nonlinear Analysis: Hybrid Systems**:
An Event-Triggered MPC Scheme for Nonlinear systems using qLPV Embeddings and Parameter Extrapolation
[Morato et al. 2023b].

Table A.1: Overview of the doctoral scientific scope.

Application: Topic:	Robotics, industrial applications	Solar plants	Renewable energy	Vehicle dynamics	Epidemics (COVID-19)	Numerical examples
Synthesis						
Gain-scheduled design	[J20, S5]	[C6,C7,C15] [C17,J8]	[J3 ,J4, J14] [J18, J23]	[C4,C16,J6] [J21]	[C12, J2]	[C8, C14]
Robust design	[J12, S1, S6]	[J19]	[J18]	[C9, J22]	[J7]	[C10, C14, S2]
Tracking	[S1]	-	-	[C5]	[C18, S2]	-
Fault tolerant	[J11]	-	[C19, J3, J14]	[C1, C2, J1]	-	-
control		-	[J22]		-	-
Data-driven	[S4, S5]	-	-	-	-	-
Analysis						
Robustness	[J21, S4, S6]	[J19]	[J18]	[J15, J21]	[J13, J17]	[C13, S3]
Dead-time	[J12]	[C17]	-	-	-	-
Dissipativity	-	[J19]	-	-	-	[C13, S3]
Classes						
Nonlinear systems	[C11,J11, J20, S7] [S1]	-	-	[C2, J2]	[C8, J9, J10,] [J13, J16, J17]	-
Polytopic	-	[C6, C7, J8]	-	[C9]	-	-
NLPV	-	-	-	[J7, J15]	-	-
Surveys						
MPC design	[C15,J5,J11, S6]	[C3]	[J4]	[C16]	[C8, J9, J10] [J16, J17]	- -
Comparisons	[C11, C15, J5]	-	[C5, J23]	[C1,C11]	[C8, J9, J10]	-

Appendix B

Maths: Spaces, algebrae, norms, and functions

In this Appendix, complementary background on mathematical formalism is presented: basic notions on vector spaces, norms, and algebra.

B.1 Norms, sets, and spaces

Norms and vector spaces are required for the computation of closed-loop performance metrics, and in order to well characterise some maps and functions that appear when considering nonlinear and LPV state transitions. In the sequel, the main vector spaces, corresponding algebraic operators, norms, sets, set operations, and some special functions used along this thesis are recalled.

In practice, Banach, Hilbert, and $\mathcal{L}_p$ vector spaces are used for the developments of the MPC algorithms for LPV systems. These spaces, in much, are used to characterise the sets, functions, and elements that enable closed-loop stability using an MPC algorithm. Also, the notions of $\mathcal{L}_p$ vector spaces are necessary to compute the corresponding induced $\mathcal{L}_p$-norms of some important signals. This can be used to characterise input-to-output energy ratios, for instance.

Definition B.1 (*Banach* space)
A Banach space is a real (or complex) complete normed vector space B, with all Cauchy sequence of points in this space have a limit that is also inside B. A Banach space has norm $\| \cdot \|_p$, such that every Cauchy sequence in B has a limit defined within B itself.

Definition B.2 (*Hilbert* space)
A Hilbert space is a (real or complex) vector space H with an inner product that is complete under the norm defined by itself, being this internal product denoted $< \cdot , \cdot >$. The norm of $f \in H$ is as gives Eq. (B.1)*. Every Hilbert space is also a Banach space, since a Hilbert space is complete with respect to the norm associated with its inner product.*

$$\|f\| = \sqrt{< f , f >} . \tag{B.1}$$

Definition B.3 (*Hardy* space)
A Hardy space, denoted $\mathcal{H}_p$, is a certain space of holomorphic functions on the unit circle or

upper half plane. Remark that these holomorphic functions are defined on an open subset of the complex plane $\mathbb{C}$ *with values in* $\mathbb{C}$ *that are complex-differentiable at all and every point.*

Definition B.4 ($\mathcal{L}_p$ space)
$\mathcal{L}_p$ spaces define the spaces of p-power integrable functions (that posses existing Lebesgue integrals), and the corresponding sequence spaces.

Example 8 (Vector spaces, from [Sename, Gaspar, and Bokor 2013])**.** *Consider the* n*-power real and complex sets* $\mathbb{R}^n$ *and* $\mathbb{C}^n$*, respectively. These spaces with the spatial p-norm,* $\|\cdot\|_p$ *for* $1 \leq p < \infty$*, are Banach spaces. Take* $p = 2$ *and thus find that these are also Hilbert spaces. All functions within these sets that are integrable derive the (real and complex) subsets* $\mathcal{L}_2$ *.*

Consider two finite real signals $v : [0, N_v] \rightarrow \mathbb{R}^{n_v}$ and $u : [0, N_u] \rightarrow \mathbb{R}^{n_u}$. Accordingly, the (discrete-time) $\mathcal{L}_2$ space is characterised by following norm and inner product rules:

$$\begin{cases} \|v\|_2, \forall v \in \mathcal{L}_2 & := \sqrt{\sum_{i=0}^{N_v} v^T(i)v(i)} < \infty \, . \\ < v \, , \, u >, \forall v, u \in \mathcal{L}_2 & := \sqrt{\sum_{i=0}^{\min(N_v, N_u)} v^T(i)u(i)} \, . \end{cases} \tag{B.2}$$

We stress that normed spaces are those with exact and explicit defined norm. This is: let $\mathcal{S}$ be a vector space with finite dimensions, then $\forall p \geq 1$, the application $\|\cdot\|_p$ is denoted a norm over $\mathcal{S}$, with $\|s\|_p := (\sum_i |s_i|^p)^{\frac{1}{p}}$ for all $s \in \mathcal{S}$. Then, let $\mathcal{S}$ be a vector space over $\mathbb{C}$ and let $\|\cdot\|$ be a norm defined over $\mathcal{S}$. In this case, we say that $\mathcal{S}$ is a normed space. Throughout this thesis, all considered sets (and subsets) are assumed to be normed.

Along this thesis, specific norms are used to measure and quantify some important signals of controlled systems. Specifically, consider a given causal finite discrete-time signal $x : [0, N_x] \rightarrow \mathbb{R}$. Accordingly, the following norms are used:

- The 1-norm of this mapping of this discrete function is defined as:

$$\|x\|_1 := \sum_{i=0}^{N_x} |x(i)| \, .$$

- The 2-norm of this same function is defined as:

$$\|x\|_2 := \sqrt{\sum_{i=0}^{N_x} x^T(i)x(i)} \, .$$

- Finally, the corresponding ∞-norm of this map is:

$$\|x\|_\infty := \sup_k |x(k)| \, .$$

B.2 Functions and operators

Along this work, in order to assess the stability of the controlled systems, some special functions mapped within normed vector spaces are required. Specifically, these mappings are used to correctly apply Lyapunov stability theory. In the sequel, some functions and classes of functions of interest are presented, following the definitions presented in [Khalil and Grizzle 2002].

Definition B.5 ($\mathcal{C}^n$ function)
Consider two normed vector spaces $A \subseteq \mathbb{R}^n$ and $B \subseteq \mathbb{R}^m$. Then, a function $f : A \to B$ is said of class $\mathcal{C}^n$, for any integer $n \geq 1$, if and only if it is n times differentiable with continuous n-order derivatives. That is, all Jacobian matrices $\left(\nabla^T\right)^n f : A \to (A \times B)$ are properly defined.

Definition B.6 (Convex function)
A function $f : A \to B$, with $A \subseteq \mathbb{R}^n$ and $B \subseteq \mathbb{R}$ is said to be convex if and only if for all $x, y \in A$ and $\lambda \in [0\,,\,1]$, it follows that $f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda) f(y)$. Equivalently, f is convex if and only if its epigraph is convex.

Remark 63. *The epigraph (or supergraph) of a function $f : A \to B$, with $A \subseteq \mathbb{R}^n$ and $B \subseteq \mathbb{R}$, is defined as the set of points lying on or above its graph. This is:*

$$epi(f) \;=\; \{(x,y) \,:\, x \in A,\, y \in B \,|\, f(x) \leq y\}\,. \tag{B.3}$$

Definition B.7 ($\mathcal{K}$ functions)
$\mathcal{K}$ refers to the class of positive and strictly increasing scalar functions that pass through the origin. Thus, a given continuous scalar function $f : A \to B$, with $A, B \subseteq \mathbb{R}_+$ is said of class $\mathcal{K}$ if and only if $f(0) = 0$ and $\lim_{\xi \to +\infty} f(\xi) \to +\infty$, i.e. for any $a, b \in A$ with $a > b$, it follows that $f(a), f(b) \in B$ and $f(a) > f(b)$.

Definition B.8 ($\mathcal{K}_\infty$ functions)
$\mathcal{K}_\infty$ refers to the class of positive and strictly increasing scalar functions that pass through the origin with unbounded limits. Thus, a real-valued scalar function $\phi : A \to B$, with $A, B \subseteq \mathbb{R}_+$ is said of class $\mathcal{K}_\infty$ if and only if it is of class $\mathcal{K}$ and it is radially unbounded, i.e. $\lim_{s \to +\infty} \phi(s) \to +\infty$.

Definition B.9 ($\mathcal{KL}$functions)
$\mathcal{KL}$ refers to the class of double-scalar functions that are strictly increasing with regard to one entry, while decreasing with regard to the other, passing through the origin via both entries. Thus, a function $\beta : A \times B \to C$, with $A, B, C \subseteq \mathbb{R}_+$, is said of class $\mathcal{KL}$ if and only if, $\beta(a,b)$ is of class $\mathcal{K}$ with regard to $a \in A$, for all fixed $0 \leq b \in B$, and $\beta(a,b)$ is decreasing with regard to $b \in B$, for all fixed $0 \leq a \in A$, i.e. $\lim_{b \to +\infty} \beta(a,b) \to 0$.

Some properties of the prior functions should be emphasised, as recalled in [Marruedo 2002]. For such, consider $f_1(\cdot)$ and $f_2(\cdot)$ as $\mathcal{K}$ class functions defined for $[0, a)$. Furthermore, consider $\phi_1(\cdot)$ and $\phi_2(\cdot)$ as $\mathcal{K}_\infty$ class functions, and $\beta(\cdot,\cdot)$ as a $\mathcal{KL}$ class function. Thus, it follows that:

- The inverse function $f_1^{-1}(\cdot)$ is a $\mathcal{K}$ class function defined for $[0, \alpha_1(a))$. Furthermore, the inverse mapping $\phi_1^{-1}(\cdot)$ is a $\mathcal{K}_\infty$ class function.
- The cascaded functions $f_1(f_2(\cdot))$ and $\phi_1(\phi_2(\cdot))$ are, respectively, of class $\mathcal{K}$ and $\mathcal{K}_\infty$. Furthermore, the composition $\beta(f_1(\cdot),\cdot)$ is of class $\mathcal{KL}$.
- The maximiser $\max_x(f_1(x), f_2(x))$ is a $\mathcal{K}$ class function.
- For any $\mathcal{K}_\infty$ class function, there exists another associated $\mathcal{K}_\infty$ such that $\phi_2(x) \leq \phi_1(x)\,, \forall x \geq 0$ and that $\phi_3(s) = x - \phi_2(x)$ is of class $\mathcal{K}$, i.e. $\phi_2(x) < x\,, \forall x > 0$.

Definition B.10 (Positive definite function)
A given function $V : A \to B$, with $A \subseteq \mathbb{R}^n$ and $B \subseteq \mathbb{R}_+$, is said (locally) positive definite (within A) if there exists a corresponding $\mathcal{K}$ function $f(x)$, $x \in A$ such that $f(\|x\|) \leq V(x)\,, \forall x \in A_r$, being $A_r = \{x \in A \,|\, \|x\| \leq r\}$.

Remark 64. *A function $V(\cdot)$ is globally positive definite if the previous definition holds for $A \equiv \mathbb{R}^n$ and $r \to +\infty$. Moreover, if $V(\cdot)$ is indeed globally positive definite, then its corresponding function $f(\cdot)$ is of class $\mathcal{K}_\infty$, i.e. its radially unbounded.*

Lemma B.2.1 (Positive definite functions, from [Khalil and Grizzle 2002])**.** *Let there exist a continuous function $V : \mathbb{R}^n \to \mathbb{R}_+$. Let $V(\cdot)$ be positive definite within the closed ball $A_r = \{x \in \mathbb{R}^n \,|\, \|x\| \leq r\}$. Then, there also exists a corresponding $\mathcal{K}$ function $g(\cdot)$, defined within $[0, r]$, such that $V(x) \leq g(\|x\|)\,, \forall x \in A_r$. Furthermore, as long as $V(\cdot)$ is radially unbounded, then A_r has no frontiers and $g(\cdot)$ is of class $\mathcal{K}_\infty$.*

B.3 Optimisation

This thesis deals specifically with the application of predictive control. As discussed thoroughly in Chapters 1 and 2, this control strategy is based on the recurrent solution of optimisation programs, during the implementation. For such, the different kinds of optimisation that are generated.

Definition B.11 (Nonlinear Programming Problem)
Consider an arbitrary real-valued nonlinear function $f_c(x_c) \in B \subseteq \mathbb{R}$. A Nonlinear Programming Problem (NP) finds the vector $x_c \in A \subseteq \mathbb{R}^n$ that minimises $f_c(x_c)$ subject to $g_i(x_c) \leq 0$, $h_j(x_c) = 0$ and $x_c \in \mathcal{X}_c$, where g_i and h_j are also nonlinear.

Definition B.12 (Convex Programming Problem)
A Convex Programming Problem is a linearly constrained optimisation problem of a convex function. A CP is a particular type of nonlinear programming problem, for which the function $f_c(x_c) \in B \subseteq \mathbb{R}$ is inherently convex with respect to $x_c \in A \subseteq \mathbb{R}^n$ and the constraints $g_i(x_c) \leq 0$ and $h_j(x_c) = 0$ are linear on x_c. Any CP can be formulated as

$x_c^\star = \arg\min_{x_c \in \mathcal{X}_c} f_c(x_c)$ subject to constraints $A_{ineq}x_c \leq b_{ineq}$ and $A_{eq}x_c = b_{eq}$. It follows that: $A_{ineq}, A_{eq} \in \mathbb{R}^{m\times n}$, $b_{ineq}, b_{eq} \in \mathbb{R}^m$. The solution $x_c^\star$ to this kind of problem is found through interior-point algorithms.

Definition B.13 (Quadratic Programming Problem)
A Quadratic Programming Problem (or simply Quadratic Problem, QP) is a linearly constrained mathematical optimisation problem of a quadratic function. A QP is a particular type of convex programming problems. The quadratic function may be defined with respect to several variables, all of which may be subject to linear contraints. Considering a $c \in \mathbb{R}^n$ gradient vector and a symmetric Hessian matrix $H_c \in \mathbb{R}^{n\times n}$, the goal of a QP is to determine the vector $x_c \in \mathbb{R}^n$ that minimises a regular quadratic function of form $\frac{1}{2}\left(x_c^T H_c x_c + c^T x\right)$ subject to constraints $A_{ineq}x_c \leq b_{ineq}$ and $A_{eq}x_c = b_{eq}$. The solution $x_c^\star$ to this kind of problem is found by many solvers seen in the literature, based on Interior Point algorithms, quadratic search, etc.

Definition B.14 (Semi-Definite Programming Problem)
A Semi-Definite Programming Program (SDP) is defined as follows:

$$\begin{aligned} min \quad & c^T x\,, \\ s.t.\ F(x) \quad & \geq \quad 0\,, \end{aligned} \tag{B.4}$$

being $F(x) \in \mathbb{R}^m$ an affine symmetric matrix function of $x \in \mathbb{R}^n$ and $c \in \mathbb{R}^n$ is a given real vector that defines the problem's objective.

B.4 Linear matrix inequalities

In many of the developments presented in this thesis, specially for the computation of terminal and invariant sets, the use of Linear Matrix Inequalities (LMIs) is exploited in order to render computationally realisable solutions. A thorough review of the application of LMIs in control theory can be found on [Boyd et al. 1994]. Nevertheless, some basic definitions and operations are recalled in the sequel. Note that, in any symmetric matrix or matricial operation, $(\star)$ denotes a symmetric term.

Definition B.15 (Linear Matrix Inequality)
For two sets mapped within normed vector spaces $A \subseteq \mathbb{R}^n$ and $B \subseteq \mathbb{R}^m$, a strict LMI constraint over any vector $x \in A$ is defined as $F(x) \in B$, for $F(x) = F_0 + \sum_{i=1}^n F_i x_i \geq 0\ (\succ 0)$, where $F_0 = F_0^T$ and $F_i = F_i^T \in \mathbb{R}^{m\times n}$.

Example 9. *Let one assess the stability verification of the following autonomous system $x(k+1) = Ax(k)$. For such, consider the following storage function: $V(k) = x^T(k)Px(k)$. Based on the direct Lyapunov stability method, we can infer wether this system is stable by verifying if $V(k)$ dissipates along time (i.e. $V(k) > 0$ and $V(k+1) - V(k) < 0, \forall k \geq 0$. Thus:*

$$\begin{aligned} x^T(k)Px(k) \quad & > \quad 0 \\ x^T(k)(A^TPA - P)x(k) \quad & < \quad 0\,, \end{aligned}$$

Left- and right-multiplying the previous inequalities by $x(k)$ and its transpose, respectively, generates:

$$F(P) \;=\; \begin{bmatrix} A^T P A - P & 0 \\ 0 - P & \end{bmatrix} \;<\; 0\,,$$

being $P = P^T$ a positive-definite decision variable. The inequality $F(P) < 0$ is linear with regard to P, and, thus, it is an LMI.

Remark 65. *Reader must bear in mind how LMI-based optimisation falls within the context of CPs. This property is fundamental as it guarantees the global (or optimal) solution $x^\star$ of a given minimization problem under LMI constraints on x (such as an SDP) can be found efficiently, in polynomial time.*

The use of LMIs allows the designer to re-state complex optimisation problems into the form of linear structures, which thus allows for the use of convex optimisation tools. Usually, the conversion from a nonlinear optimisation into a convex one, by the means of LMIs, requires differents transformations (such as changes of variables, exclusions, etc). The applicability of LMIs is wide in control theory. Next, some examples of criteria that can be handled with the use of LMIs are recalled: control synthesis with $\mathcal{H}_2$ and $\mathcal{H}_\infty$ performance goals, robustness analyses, LQR synthesis, Ricatti equations and inequalities, robust set computation, and so forth.

Some specific transformations are of special interest in this thesis, being applied in multiple chapters that deal with LMIs. These transformations are given in the form of lemmas. For further details and other transformations, refer to [Apkarian and Tuan 2000; Scherer 2001; Scherer 2006; Duan and Yu 2013].

Lemma B.4.1 (Schur complement)**.** *Let $Q = Q^T$ and $R = R^T$ be affine matrices of compatible size. Then, the following LMI constraint:*

$$\begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \;\geq\; 0\,,$$

is equivalent to the following set of inequalities:

$$\begin{aligned} R \;&\geq\; 0\,, \\ Q - SR^{-1}S^T \;&\geq\; 0\,. \end{aligned}$$

Lemma B.4.2 ($\mathcal{S}$-Procedure)**.** *Let F and G be symmetrical matrices, f and g be vectors, and a and b be real scalars. Suppose that there exists some vector x_0 such that the strict inequality $x_0^T F x_0 + 2f^T x_0 + a < 0$ holds. Furthermore, suppose that there exists a scalar $\lambda > 0$ such that the following LMI constraint holds:*

$$\lambda \begin{bmatrix} F & f \\ f^T & a \end{bmatrix} - \begin{bmatrix} G & g \\ g^T & b \end{bmatrix} \;\succ\; 0\,.$$

Then, it is implied that $x^T F x + 2f^T x + a \leq 0$ imposes $x^T G x + 2g^T x + b \leq 0$.

B.4.1 Relaxing parametrised LMIs

In many cases, LMIs are expressed with parameter-dependent variables. In the analysis of the stability of LPV systems, for instance, LMIs are written with regard to positive-definite variables $P(\rho)$, which are analytically dependent of the scheduling parameters $\rho \in \mathcal{P}$. This means that LMI problem becomes, *a priori* infinite-dimensional.

Anyhow, when the parameter-dependency is given with regard to parameters which assume finite values in a given set, the corresponding LMIs can be *gridded*, which is a direct relaxation to make their solution feasible. Gridding refers, then, to the repeated instances of an infinite-dimensional LMI problem in finite dimensions of a chosen grid.

Consider an LMI problem: solve $\mathcal{L}(\rho) \leq 0, \forall \rho \in \mathcal{P}$. This problem can be relaxed (gridded) by selecting a grid of n_g points over $\mathcal{P}$, i.e. $\mathcal{P}_{\mathcal{G}} := \left[\begin{array}{ccc} \rho_1 & \dots & \rho_{n_g} \end{array} \right]$, and thus re-writing the original problem as: solve $\mathcal{L}(\rho_1) \leq 0$ and $\mathcal{L}(\rho_2) \leq 0$ and $\dots$ and $\mathcal{L}(\rho_{n_g}) \leq 0$.

The gridding approach is based on the fact that, by discretising the parameter space along a finite set of values, there should exist a sufficiently dense grid which evaluates all critical points for the considered problem. In practice, gridding is often solved over a given grid and re-verified (if the solution holds) for a denser grid, to make sure the solution is a feasible candidate. Full discussions on this matter are available in [Scherer 2006; Briat 2008].

Appendix C

Appendix to Chapter 5

C.1 Proof of Theorem 5: Terminal ingredients

We begin by showing the positive invariance of the ellipsoid. Applying the $\mathcal{S}$-procedure, with $\lambda > 0$ to the inequality in Eq. (5.26) and $\left(1 - e^T P(\rho) e \leq 1\right)$, we get:

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

which can be rewritten as:

$$\left(\begin{array}{cc} e^T P(\rho) & I_{n_x} \end{array} \right) \underbrace{\left[\begin{array}{cc} N_{11} & N_{12} \\ N_{21} & N_{22} \end{array} \right]}_{N} \left(\begin{array}{c} P(\rho)e \\ I_{n_x} \end{array} \right) \;>\; 0\,, \tag{C.1}$$

with $N > 0$ and:

$$N_{11} = \lambda Y(\rho) - Y^T(\rho) A_t^T(\rho) P(\rho^+) A_t(\rho) Y(\rho)\,, \tag{C.2}$$
$$N_{12} = -Y^T(\rho) A_t^T(\rho) P(\rho^+)\theta\,, \tag{C.3}$$
$$N_{21} = -\theta^T P(\rho^+) A_t(\rho) Y(\rho)\,, \tag{C.4}$$
$$N_{22} = (1-\lambda) - \theta^T P(\rho^+)\theta\,. \tag{C.5}$$

Applying a Schur complement over $P(\rho^+)$ for each N_{ij} leads to BMI (5.30), which ensures the requirements of Theorem 4.

Complementary, we proceed by demonstrating that the resulting $P(\rho)$ satisfies all five conditions of Theorem 1. (C1) trivially holds due to the ellipsoidal form of $\mathbf{X}_f$. (C2) is verified due to the fact that $\mathbf{X}_f$ is a sub-level set of the terminal cost $V(\cdot)$. Therefore, if condition (C3) is verified, (C2) is consequently ensured.

The discrete Ricatti condition (C3) in the form of Eq. (2.23) (due to the parameter-dependency of $P(\rho)$) is verified through the solution of BMI (5.28). Since $Q^{-1} > 0$, $R^{-1} > 0$ and $Y(\rho^+) > 0$, we apply[1] three consecutive Schur complements to this BMI, which leads to:

$$\left(\begin{array}{cc} e^T P(\rho) & I_{n_x} \end{array} \right) \underbrace{\left[\begin{array}{cc} M_{11} & M_{12} \\ M_{21} & M_{22} \end{array} \right]}_{M} \left(\begin{array}{c} P(\rho)e \\ I_{n_x} \end{array} \right) \;\geq\; 0\,, \tag{C.6}$$

[1]Here, we use $\frac{dP(\rho)}{d\rho} = e^T A_t^T(\rho)\left(P(\rho^+) - P(\rho)\right) A_t(\rho) e$.

with $M \geq 0$ and:

$$\begin{aligned} M_{11} &= Y(\rho) - Y^T(\rho)QY(\rho) - W^T(\rho)RW(\rho) \\ &- Y^T(\rho)A_t^T(\rho)P(\rho^+)A_t(\rho)Y(\rho)\,, \end{aligned} \tag{C.7}$$

$$M_{12} = -Y(\rho)A_t^T(\rho)P(\rho^+)\theta\,, \tag{C.8}$$

$$M_{21} = -\theta^T P(\rho^+)A_t(\rho)Y(\rho)\,, \tag{C.9}$$

$$M_{22} = -\theta^T P(\rho^+)\theta\,. \tag{C.10}$$

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

$$\begin{aligned} (A_t(\rho)e + \theta)^T P(\rho)(A_t(\rho)e + \theta) - e^T P(\rho)e + \frac{\partial V(\cdot)}{\partial \rho} \\ + e^T Qe + e^T K^T(\rho)RK_t(\rho)e \leq 0\,. \end{aligned} \tag{C.11}$$

which is a a sufficient condition for (C3), begin $V(\cdot)$ as a sub-level of $\mathbf{X}_f$.

The fourth and fifth conditions (C4-C5) are verified by the direct application of the Schur complement to Eq. (5.29a) and Eq. (5.29b), respectively, using $W(\rho) = K_t(\rho)Y(\rho)$. These lead, respectively, to:

$$\left(\overline{u}_i - I_{n_u,\{i\}}u_r\right)^2 \geq \left(I_{n_u,\{i\}}K_t(\rho)\right) Y(\rho)P(\rho)Y(\rho) \left(I_{n_u,\{i\}}K_t(\rho)\right)^T\,, \tag{C.12}$$

$$\left(\overline{x}_i - I_{n_x,\{i\}}x_r\right)^2 \geq \left(I_{n_x,\{i\}}\right) Y(\rho)P(\rho)Y(\rho) \left(I_{n_x,\{i\}}\right)^T\,. \tag{C.13}$$

Since the maximum normed Fe of an e that belongs to some ellipsoid $e^T Pe \leq 1$ is given by $\sqrt{F^T\left(P^{-1}\right)F}$, it follows that Ineq. (C.12) implies that the projection $I_{n_u,\{i\}}K_t(\rho)e$ (i.e. i-th control signal) is upper-bounded, in norm, by $\overline{u}_i - I_{n_u,\{i\}}u_r$, which satisfies (C4). Analogously, Ineq. (C.13) ensures that the projection $I_{n_x,\{j\}}x$ (i.e. j-th state) is norm-bounded by $\overline{x}_j - I_{n_x,\{j\}}x_r$, which satisfies condition (C5). This concludes the proof. □

C.2 Proof of Proposition 5: Recursive feasibility

Let Assumption 15 hold. Consider there exists a solution $Y(\rho), W(\rho)$ to Theorem 5 which generate a terminal state-feedback law $u_t = K_t(\rho)(x - g_x(y)) + g_u(y)$ and a TRPI set $\Gamma(\rho)$. Furthermore, consider an initial state condition $x(0) \in \mathcal{X}$, with a corresponding initial scheduling variable $\rho(0) \in \mathcal{P}$ and a scheduling trajectory estimate $\hat{P}_0$. Recall that $\rho(N_p - 1|0)$ gives the last entry of this vector.

Then, denote $(V_0^\star, y_a^\star)$ as the optimal solution of the NMPC optimisation from Eq. (5.20), related to these initial conditions. Furthermore, consider $\hat{X}_0^\star$ as the corresponding optimal state sequence, from which the first entry is $x^\star(0|0) = x(0)$. It is implied that $(x^\star(N_p|0), y_a^\star) \in \Gamma(\rho(N_p - 1|0))$.

Let the successor state be defined as follows: $x^+ = A(f_\rho(x))x + B(f_\rho(x))K_t(f_\rho(x))(x - g_x(y_a^\star))$. Since the scheduling parameters' deviations are bounded (Assumption 12) and the estimation error is also bounded (Lemma 4.8.2), we know that the following scheduling sequence estimate $\hat{P}_1$ has its last entry $\rho(N_p|1)$ close to $\rho(N_p-1|0)$, that is: $(\rho(N_p|1)-\rho(N_p-1|0)) \in \mathcal{Q} \subset \delta\mathcal{P}$. Complementary, consider $\tilde{y}_a^+ = y_a^\star$, and $V_1 = \left[(v^\star(1|0))^T \ldots, (v^\star(N_p-1|0))^T, (K_t(\rho(N_p|0))(x^\star(N_p|0)-g_x(y_a^\star)) + g_u(y_a^\star) - K_\pi x^\star(N_p|0))^T\right]^T$.

Then, the predicted sequence of candidate states for the successor step is given by: $\hat{X}_1 = \left[(x^\star(1|0))^T, \ldots, (x^\star(N_p|0))^T, x_t^T(N_p+1|0)\right]^T$, where the last entry $x_t(N_p+1|0) == A(\rho(N_p-1|0)x^\star(N_p|0) + B(\rho(N_p-1|0))K_t(\rho(N_p-1|0))(x^\star(N_p|0)-g_x(y_a^\star)) + B(\rho(N_p-1|0))g_u(y_a^\star)$. This last successor state implies in $x_t(N_p+1|0) - g_x(y_a^\star) = A_t(\rho(N_p-1|0))\,(x^\star(N_p|0)-g_x(y_a^\star))$ and, since $(x^\star(N_p|0), y_a^\star) \in \Gamma(\rho(N_p-1|0))$ and $(\rho(N_p|1)-\rho(N_p-1|0)) \in \mathcal{Q}$, it follows that $(x_t(N_p|0), y_a^+) \in \Gamma(\rho(N_p|1)-\xi_\rho), \forall \xi_\rho \in \mathcal{Q}$. Therefore, it follows that $x^\star(1|0) = x^+$, and thus that $(V_1, \tilde{y}_a^+)$ is a feasible solution for the NMPC optimisation in Eq. (5.20) at the successor step, i.e. $k=1$.

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

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

Finally, from conditions (C1), (C2), (C4), and (C5) from Theorem 1, we can infer that the generated control signal is well defined. Thus, being Γ a parameter-dependent TRPI set for the closed-loop system evolution, the constraints are fulfilled as the horizon slides and the optimisation is indeed recursively feasible. This concludes the proof. □

C.3 Proof of Proposition 6: Error ISS

Let there be a terminal stage cost $V(\cdot)$ such that Assumption 15 holds. Leveraging from Lemma 4.8.2, assume that $\lim_{k\to+\infty} \hat{P}_k \to P_k$, i.e. the extrapolated scheduling trajectory converges to the **true** scheduling parameter values. Furthermore, let Proposition 5 be satisfied. Note that since $\ell(x-x_a, u-u_a)$ is a quadratic stage cost $\|x-x_a\|_Q^2 + \|u-u_a\|_R^2$, α_ℓ, γ_x and γ_u indeed exists.

Next, consider there exists a solution $Y(\rho)$ to Theorem 5. Then, the closed-loop is stable due to (C3) of Theorem 1, which conversely ensures that:

$$\begin{aligned} \delta V(k) &= V(x(k) - x_a(k), \rho(k)) - V(x(k-1) - x_a(k-1), \rho(k-1)) \\ \delta V(k) &\leq -\|x(k) - x_a(k)\|_Q + \gamma_V(\overline{w}) \,. \end{aligned} \tag{C.14}$$

Here, we assume that $\rho(k+j|k) = \rho(k+j)$, for simplicity. If the scheduling uncertainty is considered, one can account for the related model uncertainty $\Delta(k+j|k)$, as gives Eq. (5.16), and, thanks to the robust constraint satisfaction, consider that the deviation of the nominal predicted state trajectory from the real one is bounded.

Assume that $\lim_{k\to+\infty} y_a(k) \to y_a^o$. Analogously, use $\lim_{k\to+\infty} x_a(k) \to x_a^o := g_x(y_a^o)$. Then, thanks to the error dynamics in Eq. (5.24), we obtain:

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

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

Now we consider the convergence of $V_O(y_a(k)-y_r)$ such that the limit $\lim_{k\to+\infty} y_a(k) \to y_a^o$ holds. Let us define $\hat{y}_a = (1-\alpha)y_a(k) + \alpha y_a^o$, where $\alpha \in [0,1]$ is the optimal solution from Eq. (5.23). From the convexity of $V_O(\cdot)$, we obtain:

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

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

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

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

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

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

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

$$\begin{aligned} V_O(y_a(k) - y_r) \leq a_1 L_x^\sigma \alpha^{\sigma-1}\|y_a(k) - y_a^o\|^\sigma &+ \gamma_{V_O}(\overline{w}) \\ +\left(\frac{1-\alpha}{\alpha}\right) V_O(y_a(k) - y_r) &+ V_O(y_a^o - y_r) \,. \end{aligned}$$

Finally, since $\lim_{k\to+\infty}\left(\frac{1-\alpha}{\alpha}\right)V_O(y_a(k)-y_r)\;\to\;\gamma_y(\overline{w})$, we obtain $V_O(y_a(k)-y_r)-V_O(y_a^o-y_r)\;\leq\;a_1L_x^\sigma\alpha^{\sigma-1}\|y_a(k)-y_a^o\|^\sigma+\gamma_n(\overline{w})$, with $\sigma>1$ and $a_1>0$ as a constant scalars. Taking the limit at both sides of this inequality leads to:

$$V_O\left(\left(\lim_{k\to+\infty}y_a(k)\right)-y_r\right)\;\;\leq\;\;V_O(y_a^o-y_r)+\gamma_n(\overline{w})\,.$$

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

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

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

Appendix D

Résumé étendu

Méthodes de Commandes Prédictive pour les Systèmes Linéaires à Paramètres Variants

---

Le principal sujet étudié tout au long de ce travail est l'exploitation de schémas de commande prédictive pour les systèmes linéaires à paramètres variants. Plus précisément, je suis confronté au problème concevoir des algorithmes précis sans nécessiter la connaissance complète des trajectoires futures des paramètres d'ordonnancement.

Dans la suite de ce résumé étandu, je rappelle le contexte principal de l'ouvrage et distingue ses principaux apports. En conséquence, je détaille la structure de la thèse, expliquant quels sujets sont débattus dans chacun de ses chapitres. Je commence par récapituler le contexte général de mon travail.

Pourquoi le MPC est-il pertinent ?

La littérature sur les systèmes de contrôle s'est profondément épanouie au cours des cinq dernières décennies. Néanmoins, jusqu'à la fin des années 80, la pratique industrielle de la théorie du contrôle consistait à utiliser des contrôleurs Proportionnels-Intégraux (PIs) et Proportionnels-Intégraux-Dérivatifs (PIDs) simples pour la grande majorité des applications, malgré toutes les avancées théoriques. Ce n'est qu'avec le développement de la Commande Prédictive basée sur Modèle (MPC), tel que proposé dans les articles originaux par Cutler, Clark et leurs collègues [Cutler and Ramaker 1980; Clarke, Mohtadi, and Tuffs 1987], que des techniques de commande plus avancées ont été mises en œuvre dans des contextes industriels réels. Depuis ses débuts, la commande prédictive est devenu un domaine actif de recherche, avec une large acceptation industrielle [Camacho and Bordons 2013]: une large gamme des implémentations MPC réussies sont aujourd'hui reconnues (pour les types de systèmes les plus divers !), c.f. [Alamir 2013].

La principale raison pour laquelle la commande prédictive est si bien établi réside dans le fait qu'elle a la capacité de considérer conjointement l'optimisation des performances et la satisfaction des contraintes dans un cadre de synthèse relativement simple (et intuitif). Pour justifier l'argument, considérons un système avec un comportement dynamique connu ($\mathfrak{B}$).

Ensuite, lorsqu'on applique un algorithme de commande prédictive pour réguler ce système, on extrait une action de commande optimale u de la solution d'un problème d'optimisation qui inclut les objectifs de performance et les contraintes du système. Cette solution d'optimisation est répétée en ligne lors de l'implémentation : à chaque unité de temps discrète (échantillon), une nouvelle optimisation est résolue, donc conduisant à une nouvelle entrée de commande correspondante.

Considérons qu'un système Σ est linéaire invariant dans le temps (stationnaire, LTI). De plus, supposons que son comportement $\mathfrak{B}$ puisse être partitionné dans un canal à entrée unique et sortie unique (SISO) $u \Leftrightarrow y$ tel que $\mathfrak{B}_{u \Leftrightarrow y} := \left\{ \exists y \in \mathbb{R}\,,\, (u, y) \in \mathfrak{B} \,|\, y = \sum_{i=1}^{n_y} a_i z^{-i} y + \sum_{i=1}^{n_u} b_i z^{-i} u \right\}$. En plus, tenez compte de l'objectif et des contraintes de performances suivants : (a) la sortie y doit être ramenée à une consigne de régime d'équilibre noté y_r, tandis que (b) le signal de commande doit être limité à l'ensemble opérationnel convexe $[0, 1]$. En conséquence, on peut formuler un MPC basé sur l'optimisation suivante, à résoudre à chaque instant discret k :

$$\mathfrak{P}_k \;:=\; \left\{ \begin{array}{lccc} \min_{U_k} & & \sum_{i=1}^{N_p} \|z^i y - y_r\| & \\ \text{t.q.} & (z^i y, z^{i-1} u) & \in & \mathfrak{B}_{u \Leftrightarrow y}, \quad \forall i \in \mathbb{N}_{[1,N_p]}\,, \\ & z^{i-1} u & \in & [0, 1], \quad \forall i \in \mathbb{N}_{[1,N_p]}\,, \end{array} \right. \tag{D.1}$$

étant $U_k := \text{col}\{u, \dots z^{N_p-1} u\}$ la trajectoire de la commande le long de l'horizon de prédiction futur N_p. Notez que le problème d'optimisation $\mathfrak{P}_k$ minimise le déviance de la sortie y par rapport au régime permanent y_r, cherchant ainsi à assurer la satisfaction de l'objectif de performance "(a)".

La solution statique de l'optimisation $\mathfrak{P}_k$ dans l'Éq. (D.1) (i.e. le résoudre une seule fois, au lieu de le faire à plusieurs reprises, à chaque échantillon) est, en soi, déjà d'un intérêt pratique. L'application de la séquence de contrôle correspondante $U_k^\star$ est généralement appelée contrôle optimal. Régulateurs quadratiques linéaires (LQRs), par exemple, peuvent être représentés dans ce cadre.

Contrairement à une solution statique de l'optimisation, MPC fonctionne sous un paradigme d'horizon glissant. Cela signifie qu'à chaque instant discret k, le problème d'optimisation contraint $\mathfrak{P}_k$ est résolu, et la première entrée de son minimiseur $U_k^\star$ est appliquée au procédé (le reste du vecteur est ignoré). Une telle procédure de mécanisme d'horizon roulant fournit des propriétés de robustesse inhérentes au MPC, comme le soutient [Allan et al. 2017], une fois que $\mathfrak{P}_k$ est mis à jour en fonction des mesures disponibles du procédé.

Le principe de base de MPC repose sur le fait d'avoir un modèle du procédé adéquat afin de prédire le comportement futur des variables d'état (ou des sorties). Si un modèle fiable n'est pas disponible, la loi de commande dérivée peut simplement être irréaliste et, par conséquent, le contrôleur peut être insuffisamment robuste pour contrer les incertitudes causées par les inadéquations de prédiction (même la stabilité peut être perdue, dans certains contextes dramatiques).

Donc, la recherche sur MPC a constamment débattu de ce qui se passe lorsque des modèles de procédé imparfaits sont utilisés dans l'optimisation. En conséquence, la théorie Les schémas

MPC en boucle fermée ont été étudiés de manière approfondie au cours des dernières décennies. Ensuite, je récapitule certaines des avancées récentes de la recherche dans le contexte des synthèses MPC non linéaires et robustes.

MPC non linéaire

Les algorithmes de commande prédictive basée sur des modèles non linéaires (NMPC) sont d'une grande pertinence lorsque les systèmes non linéaires sont contrôlés dans des conditions de fonctionnement plus larges ou lorsque les réponses du procédé dépendent fortement de paramètres externes. Cependant, l'inclusion de prédictions de modèles non linéaires n'est pas triviale et augmente la la complexité de l'algorithme résultant [Allgöwer and Zheng 2012]. L'augmentation de la charge numérique devient un obstacle pour certaines applications en temps réel (c'est-à-dire : l'algorithme MPC n'arrive pas a calculer la loi de commande en temps).

Afin d'alléger la complexité de calcul de NMPC, la recherche sur des formulations rapides et leur garanties théoriques a pris de l'ampleur (et de la concrétisation) au cours des dernières années. Reportez-vous, par exemple, à la revue [Gros et al. 2020]. L'un des principaux objectifs de cette ligne d'investigation a été de développer des solutions basées sur des *solvers*, qui se rapprochent le programme non linéaire résultant par des programmes plus simples (tout en conservant la précision de la loi de commande résultante). Ceci est fait viable principalement grâce à des méthodes d'itération en temps réel, telles que ACADO et CasADi [Houska, Ferreau, and Diehl 2011; Quirynen et al. 2015; Andersson et al. 2019], et Lagrange et solutions basées sur le gradient, telles que GRAMPC [Richter, Jones, and Morari 2011; Käpernick and Graichen 2014; Englert et al. 2019]. L'intérêt principal de ces algorithmes est qu'ils permettent une implémentation en temps réel, servant ainsi à la commande des procédés non linéaires avec des taux d'échantillonnage (très) rapides.

L'objet de cette thèse n'est pas le développement de schéma NMPC basé sur un *solver* comme détaillé précédemment, mais je les mentionne comme références de référence pour les algorithmes que je développe. Le principal avantage de ces approches orientées implémentation est qu'elles sont capables de récupérer des lois MPC embarquées (pour des systèmes d'ordre relativement important) dans la plage de la milliseconde, ce qui est plutôt impressionnant. De plus, ils se traduisent par des performances quasi optimales avec des contraintes satisfaction, proche de ce que l'on obtiendrait avec une mise-en-œuvre “à part entière” du NMPC.

MPC robuste

Pourtant, avec une grande valeur pratique, la conception MPC standard (linéaire et non linéaire) manque de garanties de faisabilité récursive ou de stabilité en boucle fermée en présence de perturbations. Par conséquent, parallèlement à la recherche sur des algorithmes NMPC rapides, il y a également eu un grand enthousiasme pour les schémas MPC robustes avec des certificats de performance, c.f. [Köhler et al. 2020; Santos and Cunha 2021].

Depuis les années 2000, il y a eu une croissance du corps de recherche sur les techniques MPC robustes, c'est-à-dire celles qui se concentrent sur l'obtention de performances optimales en boucle fermée, la faisabilité et la satisfaction des contraintes malgré les effets causés sur les sorties par des perturbations non mesurables, ou par des incertitudes sur le modèle de prédiction. La prémisse de base pour la conception de robuste schémas MPC est que les incertitudes doivent être bornées (et les bornes sont connues ou, au moins, estimées). Ainsi les certificats de performance peuvent être fournis en robustifiant l'optimisation MPC vis-à-vis des bornes d'incertitudes. Cela peut être fait via l'optimisation min-max [Limón et al. 2006a; Löfberg 2012], resserrement des contraintes [Köhler, Müller, and Allgöwer 2018; Santos et al. 2019], propagation des perturbations par tubes [Yu et al. 2013; Abbas et al. 2019], et ainsi de suite.

Un bref aperçu des systèmes LPV

Dans le précédent, les idées principales de MPC et les frontières de la littérature sur le sujet ont été revissées. Ensuite, je détaille brièvement le deuxième sujet principal de cette thèse : les Systèmes Linéaires à Paramètres Variants (LPV). Parallèlement à la mise en place de schémas MPC modernes, un examen concerté du contrôle robuste s'est développé, y compris une analyse de robustesse structurée et la généralisation du théorème du petit gain, c.f [Doyle, Wall, and Stein 1982; Scherer 2006]. En conséquence, à partir du tissu de la théorie du contrôle robuste, le cadre du système LPV a été développé [Mohammadpour and Scherer 2012; Sename, Gaspar, and Bokor 2013]. Cette boîte à outils d'analyse et de contrôle est devenue populaire et largement utilisée pour gérer des procédés avec des dynamiques complexes (non linéaires, variant dans le temps), comme débattu dans [Tóth 2010; Hoffmann and Werner 2014].

En synthèse, on peut comprendre les systèmes LPV comme une classe **spéciale** de procédés non linéaires, bien adaptés au contrôle de la dynamique avec des variations programmées de paramètres. Dans un certain sens, les systèmes LPV se trouvent quelque part *entre* les formalismes non linéaires et LTI, car ils sont linéaires dans l'espace d'états, tandis que non linéaires dans l'espace des paramètres. Contrairement au cas LTI, les transitions d'état et de sortie dépendent de paramètres variant dans le temps, appelés **variables d'ordonnancement**, notés ρ. Ces paramètres sont supposés **connus** et implicitement **bornés**, et peuvent donc être utilisés pour la commande.

Considérons un système générique Σ, dont le comportement est donné par :

$$\left\{ \begin{array}{rcl} x(k+1) & = & A(\rho)x(k) + B(\rho)u(k)\,, \\ y(k) & = & C(\rho)x(k) + D(\rho)u(k)\,, \end{array} \right. \tag{D.2}$$

étant $A(\cdot)$, $B(\cdot)$, $C(\cdot)$ and $D(\cdot)$ fonctions matricielles par rapport aux paramètres d'ordonnancement ρ. Dans ce modèle, x représente l'état, u représente les entrées et y représente les sorties contrôlées.

Selon la nature des paramètres ρ sur modèle et la dépendance des matrices sur ces paramètres, l'Éq. (D.2) peut représenter différents types de dynamique : (a) si les paramètres sont statiques (constants et invariants dans le temps), alors le modèle est LTI ; (b) si ρ est une fonction **inconnue** du temps, i.e. $\rho(k)$, alors le modèle représente la dynamique LTV (Linear Time Varying) ; (c) lorsque les paramètres varient dans le temps, mais **mesuré** et **limité**, le modèle devient LPV. Dans la philosophie LPV, le concepteur ne s'intéresse pas à la dépendance explicite des matrices d'état au temps, mais plutôt à la dépendance de ces matrices aux paramètres d'ordonnancement eux-mêmes. Ensuite, la méthodologie d'analyse et de conception LPV découle de l'exploitation de la forme de ces dépendances paramétriques bornées.

De manière générale, la caractéristique la plus pertinente des techniques LPV, dans le contexte de la synthèse de contrôle, est qu'ils fournissent une procédure de conception systématique pour les commandes multivariables auto-coordonnées, bénéficiant de la disponibilité des paramètres d'ordonnancement du procédé. La méthodologie LPV permet, donc, d'intégrer la performance et la robustesse dans un cadre unifié, et c'est pourquoi le sujet est d'une grande pertinence scolaire.

En comblant les lacunes

La boîte à outils LPV peut, en effet, être utilisée pour représenter des dynamiques non linéaires avec fiabilité. Ainsi, la synthèse de contrôle et les analyses des caractéristiques non linéaires deviennent beaucoup plus faciles en exploitant la disponibilité des variables d'ordonnancement connues et la dynamique LPV correspondante. De nos jours, la synthèse de contrôle LPV est standard en utilisant des transformées fractionnelles [Casella and Lovera 2008], objectives H_2 et H_∞ [Sename, Gaspar, and Bokor 2013; Emedi and Karimi 2016], ainsi que pour l'asservissement et le suivi de consigne [Scorletti, Fromion, and De Hillerin 2015]. Néanmoins, le cas n'est certainement pas vrai pour les applications de contrôle prédictif. L'étude des schémas LPV MPC a correctement commencé au milieu des années 00. Quoi qu'il en soit, il reste encore des lacunes complexes à approfondir sur ce sujet, comme indiqué dans [Bachnas et al. 2014; Morato, Normey-Rico, and Sename 2020a].

Étant donné que les fonctions non linéaires peuvent être refondus en tant que LPV modèles, il semble naturel de développer des algorithmes NMPC en exploitant les réalisations LPV. Il est particulièrement intéressant d'utiliser des modèles LPV dans la synthèse de MPC car ces représentations conservent la linéarité propriété le long des canaux d'entrées-sorties, ce qui signifie que des procédures de conception efficaces en termes de calcul peuvent être rendues. Inversement, cela signifie que les inconvénients des algorithmes NMPC complets sont évités (l'utilisation de programmes non linéaires), sans qu'il soit nécessaire d'approximer la solution du problème d'optimisation (comme le font les solutions NMPC rapides les plus modernes, telles que l'itération en temps réel et les méthodes basées sur le gradient, par exemple).

Dans le contexte de MPC, un modèle de prédiction à horizon complet est requis (c'est-à-dire pour décrire les variables du système le long des futures étapes N_p avant chaque échantillon). Néanmoins, lorsqu'un modèle de prédiction LPV est utilisé, ce problème dépend non seulement des entrées futures (à déterminer par l'optimisation), mais aussi des paramètres d'ordonnancement futurs $\rho(k+i), \forall i \in \mathbb{N}_{[0,N_p-1]}$, qui sont vraisemblablement **inconnu**.

En conséquence, considérons qu'un problème MPC à retour d'état est formulé. Cela signifie qu'à chaque instant d'échantillonnage en temps discret k, le problème d'optimisation suivant doit être résolu :

$$\begin{aligned}
\min_{U_k} \quad & \overbrace{V(x(k+N_p|k)) + \left(\sum_{i=0}^{N_p-1} \ell\left(x(k+i+1|k), u(k+i|k)\right) \right)}^{J_k = J(x(k),U_k)} && \text{(D.3)} \\
\text{sous réserve de:} \quad & \text{Modèle de procédé}, \forall i \in \mathbb{N}_{[1,N_p]}, && \text{(D.4)} \\
& x(k+i|k) \in \mathcal{X} \, \forall \, i \in \mathbb{N}_{[1,N_p]} \, , && \\
& u(k+i-|k) \in \mathcal{U} \, \forall \, i \in \mathbb{N}_{[1,N_p]} \, , && \\
& x(k+N_p|k) \in \mathbf{X}_f \, , && \text{(D.5)}
\end{aligned}$$

en tenant compte que $x \in \mathcal{X}$ désigne les contraintes d'admissibilité de l'état, $u \in \mathcal{U}$ l'admissibilité de l'entrée, $x \in \mathbf{X}_f$ l'admissibilité terminale, et U_k donne le vecteur des efforts de commande prédites au fil de l'horizon de prédiction, soit :

$$U_k = \begin{bmatrix} u(k|k)^T & u(k+1|k)^T & \dots & u(k+N_p-1|k)^T \end{bmatrix}^T . \qquad \text{(D.6)}$$

Tout algorithme MPC, comme le précédent, prend en compte la dynamique du système contrôlé sur un horizon de prédiction de N_p (la contrainte du modèle du procédé). Ainsi, dans le cadre non linéaire général, en considérant une dynamique d'état non linéaire générique à temps discret $x(k+1) = f\left(x(k), u(k)\right)$, la séquence suivante de règles basées sur un modèle est évaluée (en interne par l'optimiseur, c'est-à-dire le *solver* du problème d'optimisation) :

$$\begin{aligned}
x(k+1|k) &= f\left(x(k), u(k|k)\right) , && \text{(D.7)} \\
x(k+2|k) &= f\left(f\left(x(k), u(k|k)\right), u(k+1|k)\right) , && \\
x(k+3|k) &= f\left(f\left(f\left(x(k), u(k|k)\right), u(k+1|k)\right), u(k+2|k)\right) , &&
\end{aligned}$$

et ainsi de suite, jusqu'à la N_p-ième prédiction.

Notez que la solution (originale) du problème $\min_{U_k} J_k$ (en utilisant un modèle non linéaire), nécessite l'évaluation des prédictions données dans l'Éq. (D.7), ce qui fait de l'optimisation un programme non linéaire (qui peut même être non convexe).

Néanmoins, lorsque le modèle non linéaire est remplacé par un modèle LPV, c'est-à-dire en utilisant l'Éq. (D.2), la séquence de règles de prédiction basées sur un modèle qui sont évaluées par l'optimiseur devient :

$$\begin{aligned} x(k+1|k) &= A(\rho(k))x(k) + B(\rho(k))u(k|k)\,, && \text{(D.8)} \\ x(k+2|k) &= A(\rho(k+1))A(\rho(k))x(k) + A(\rho(k+1))B(\rho(k))u(k|k) && \text{(D.9)} \\ &+ B(\rho(k+1))u(k+1|k)\,, \\ x(k+3|k) &= A(\rho(k+2))A(\rho(k+1))A(\rho(k))x(k) && \text{(D.10)} \\ &+ A(\rho(k+2))A(\rho(k+1))B(\rho(k))u(k|k) \\ &+ A(\rho(k+2))B(\rho(k+1))u(k+1|k) \\ &+ B(\rho(k+2))u(k+2|k)\,, \end{aligned}$$

et ainsi de suite, jusqu'à la N_p-ième prédiction.

$$\begin{aligned} x(k+N_p|k) &= A(\rho(k+N_p-1))\ldots A(\rho(k))x(k) && \text{(D.11)} \\ &+ A(\rho(k+N_p-1))\ldots A(\rho(k+1))B(\rho(k))u(k|k) \\ &+ A(\rho(k+N_p-1))\ldots A(\rho(k+2))B(\rho(k+1))u(k+1|k) + \ldots \\ &+ B(\rho(k+N_p-1))u(k+N_p-1|k)\,. \end{aligned}$$

Notez que, dans le cas LPV, ces prédictions nécessitent les valeurs des futures variables d'ordonnancement, c'est-à-dire $\rho(k+1)$, $\rho(k+2)$, et ainsi de suite. Cette "trajectoire d'ordonnancement future" peut être compactée, en format de vecteur, comme suit :

$$P_k = \begin{bmatrix} \rho(k)^T & \rho(k+1)^T & \ldots & \rho(k+N_p-1)^T \end{bmatrix}^T\,, \tag{D.12}$$

où seul $\rho(k)$ est, en pratique, connu. Pourtant, le vecteur de trajectoire d'ordonnancement complet P_k peut être utilisé pour exprimer analytiquement le vecteur complet des prédictions d'état, c'est-à-dire $X_k = [x(k+1|k)^T\,, \ldots\,, x(k+N_p|k)^T]^T$. Comme le montre [Cisneros and Werner 2020], il s'ensuit que:

$$X_k = \mathcal{A}(P_k)x(k) + \mathcal{B}(P_k)U_k\,, \tag{D.13}$$

oú $\mathcal{A}(P_k) \in \mathbb{R}^{(n_xN_p)\times n_x}$ et $\mathcal{B}(P_k) \in \mathbb{R}^{(n_xN_p)\times(n_uN_p)}$. En pratique, ces matrices conservent leur forme à chaque échantillon et peuvent donc être calculées rapidement. Ainsi, la loi MPC est activée via la solution en ligne de $\min_{U_k} J_k$ sous réserve de l'Éq. (D.13) et contraintes, qui est un QP, tant que P_k est connu.

Notamment, ces prédictions LPV nécessitent implicitement la connaissance de la trajectoire dite d'ordonnancement ($\rho(k)$, $\rho(k+1)$, jusqu'à $\rho(k+N_p-1)$). Cependant, ces variables sont indisponibles à chaque instant k, alors que seul $\rho(k)$ est connu. En raison de ce problème d'indisponibilité, la conception de MPC utilisant des modèles LPV devient particulièrement compliquée, essentiellement parce que la faisabilité récursive et la stabilité en boucle fermée de la stratégie exigent théoriquement que le MPC tolère les incertitudes qui surviennent en raison de l'indisponibilité de la future trajectoire d'ordonnancement.

Énoncé du problème :
Ainsi, par rapport au contexte discuté, le principal problème étudié dans cette thèse est :

Comment concevoir des algorithmes de commande prédictive (précis) pour les systèmes LPV sans la connaissance réelle de la future trajectoire d'ordonnancement ?

État-de-l'art : Techniques disponibles

Il y a eu quelques travaux récents sur la conception de MPC pour les systèmes LPV. Dans notre article de revue [Morato, Normey-Rico, and Sename 2020a], un aperçu détaillé de l'état de l'art est présenté. Ici, par souci de brièveté, je récapitule certaines des techniques les plus pertinentes, qui peuvent être classées en deux groupes principaux :

- Méthodes robustes, c.f. [Jungers, Oliveira, and Peres 2011; Rakovic et al. 2012; Bumroongsri 2014; Hanema, Lazar, and Tóth 2020], qui considèrent les pires performances en boucle fermée impliquées par rapport aux paramètres d'ordonnancement futurs, que sont inconnus, en tenant compte leurs bornes. En conséquence, l'optimisation est réécrite afin de prendre en compte les limites de toutes les variations futures possibles des paramètres, ce qui peut donner des résultats généralement conservateurs.
- Méthodes basés sur l'ordonnancement par gain, c.f. [Ayala et al. 2011; Brunner, Lazar, and Allgöwer 2013; Mate et al. 2019; Alcalá, Puig, and Quevedo 2019]. Dans ces articles, le modèle LPV est remplacé, à chaque instant d'échantillonnage, par un modèle LTI (ou une séquence de modèles LTI) basé sur une estimation de la trajectoire d'ordonnancement. Dans de nombreux cas, cette supposition est simplement une hypothèse selon laquelle les paramètres d'ordonnancement resteront constants tout au long de l'horizon de prédiction. Bien que ces méthodes fonctionnent assez rapidement (elles présentent une charge numérique réduite), une sous-optimalité peut être implicite. Néanmoins, lorsque la trajectoire d'ordonnancement est précise (comme on le voit pour le cas qLPV dans [Cisneros, Voss, and Werner 2016; Cisneros and Werner 2017b; Cisneros and Werner 2019]), une solution MPC non linéaire exacte est obtenue au moyen de programmes d'optimisation quadratique, rendant ainsi une solution comparable à l'état-des solutions NMPC basées sur des *solvers* de pointe (telles que ACADO et CasADi).

Lacunes de recherche liées au problème

Bien qu'il existe, de nos jours, des formulations NMPC généralisées, la synthèse d'algorithmes MPC utilisant le formalisme LPV est d'un intérêt académique et pratique total, étant donné que la propriété de linéarité peut être exploitée de telle manière que l'algorithme résultant a allégé la complexité numérique.

En accord avec les travaux qui ont déjà été proposés sur ce sujet, tels que récapitulés dans la précédente, il reste encore quelques fils d'investigation disponibles pour des recherches plus approfondies :

- Tout d'abord, je souligne que les formulations LPV sous la forme entrée-sortie (IO), bien qu'étant soutenues par de fortes prétentions théoriques dans le sens de l'identification, c.f. [Bachnas et al. 2014], n'ont qu'une poignée d'homologues de synthèse de contrôle. La grande majorité de la synthèse de contrôle LPV (y compris MPC) est réglée pour des descriptions en espace-d'état. Néanmoins, l'industrie est beaucoup plus encline à accepter les formulations IO, comme le discute [Froisy 2006]. Par conséquent, le pont entre théorie et applications industrielles pour le cas du LPV MPC la conception sera encore soutenue lorsqu'un organisme de recherche théorique sur les formulations IO deviendra disponible. Bien que ce sujet soit fondamental et prometteur, il existe assez peu d'articles qui l'élaborent, c.f. [Abbas et al. 2015; Abbas et al. 2016], manquant ainsi d'autres évaluations.

- Dans le contexte des méthodes LPV MPC à gain programmé, il existe un manque d'exploitation supplémentaire des certificats de performance de ces algorithmes. L'utilisation de la dépendance paramétrique LPV sur les arguments de stabilité et ses implications n'ont été que brièvement évaluées, c'est-à-dire dans [Cisneros and Werner 2017b; Cisneros and Werner 2020].

- Comme indiqué, les algorithmes LPV MPC à gain programmé, basés sur une estimation précise de la trajectoire de paramètres d'ordonnancement futurs, sont capables en temps réel et tout à fait comparables aux techniques NMPC modernes. Néanmoins, l'exploitation de la fonction d'ordonnancement (des modèles qLPV) pour générer avec précision des estimations de trajectoire d'ordonnancement (et ainsi synthétiser un LPV MPC avec une optimalité proche de celle d'un NMPC) n'a été que brièvement évaluée, sans preuves supplémentaires au sens théorétique.

- Même si l'application de schémas MPC robustes pour les systèmes LPV est plutôt établie, des travaux récents, c.f. [Cisneros and Werner 2018], ont souligné comment la théorie de la dissipativité peut être utilisée pour lisser la dureté en ligne de l'optimisation résultante. Pourtant, un débat plus approfondi sur cette question, avec des résultats d'application correspondants, peut encore être fourni.

Objectifs de la thèse

Au cours des dernières décennies, il y a eu une intérêt croissant pour le développement d'algorithmes MPC rapides pour les systèmes non linéaires. En conséquence, la boîte à outils de variation de paramètres linéaires apparaît comme une alternative intéressante pour aborder ce sujet sans la nécessité d'une solution basée sur des *solvers* approchés (tel comme ACADO), puisque les incorporations ou les réalisations planifiées peuvent être utilisées pour décrire la dynamique non linéaire avec exactitude (ou bonne précision).

Néanmoins, les schémas d'application LPV MPC nécessitent la connaissance de la trajectoire d'ordonnancement future, qui n'est généralement pas disponible. L'état de l'art et les lacunes de recherche correspondantes indiquent deux alternatives pour traiter cette question :

(i) la synthèse robuste ; et (ii) conception à gain programmé basée sur des estimations de trajectoire. Chacune de ces branches a certaines limites et avantages, mais elles sont toutes d'enthousiasme philosophique et théorique.

Par conséquent, en gardant à l'esprit l'énoncé du problème de cette thèse et compte tenu de l'état de l'art disponible, les objectifs de cette thèse coïncident, en majorité, avec l'investigation disponible fils dans ce nouveau domaine. Dans un souci de rigueur de présentation, je considère trois objectifs principaux, qui sont corroborés par quelques buts spécifiques. Ci-dessous, je les énumère et les discute en détail, étant les éléments marqués en chiffres romains les objectifs principaux, et ceux marqués en majuscules, les spécifiques :

(i) **Fournir de nouveaux algorithmes afin d'estimer les futures trajectoires d'ordonnancement LPV, du point de vue de chaque instant d'échantillonnage, en envisageant des prédictions précises pour le MPC.**

  (i.A) Assurer la charge de calcul allégée de ces outils, les rendant plus simples à mettre en œuvre que méthodes basées sur la fonction d'ordonnancement (QPs séquentiels), c.f. [Cisneros and Werner 2020].

  (i.B) Démontrer les propriétés de convergence des méthodes développées par rapport à la véritable séquence d'ordonnancement.

(ii) **Développer de nouveaux algorithmes de contrôle prédictif à gain programmé avec des certificats de faisabilité recursive et de stabilité.**

  (ii.A) Considérez la solution en utilisant une formulation de retour d'état, corroborée par la contraction des contraintes terminales.

  (ii.B) Fournissez la formulation correspondante pour le cas de retour de sortie dynamique, lorsque les mesures d'état ne sont pas disponibles, en tenant compte des descriptions d'entrée-sortie.

(iii) **Proposer des algorithmes de contrôle prédictifs robustes pour les systèmes LPV, avec un calcul rapide lors de la mise en œuvre, adaptés aux applications embarquées en temps réel.**

  (iii.A) Envisagez des formulations de suivi avec des certificats de satisfaction de contraintes robustes pour des signaux de référence constants par morceaux (avec des valeurs éventuellement inaccessibles).

  (iii.B) Envisagez des formulations robustes basées sur des arguments de dissipativité, ne nécessitant donc pas l'utilisation d'ingrédients terminaux et soulageant le stress informatique global de l'implémentation en ligne résultante.

Méthodologie et applications

Les contributions scientifiques issues de cette thèse, qui se rapportent aux objectifs susmentionnés, sont répertoriées en annexe A. De plus, je souligne que ce doctorat a été réalisé selon une méthodologie de recherche scientifique, bibliographique et documentaire rigoureuse afin de répondre aux objectifs de la thèse. Pour chacun des objectifs principaux (et spécifiques correspondants) listés dans la précédente, en bouclant des phases de spécifications, de conception et des dissertations.

Complémentairement, puisque l'on ne peut pas dissocier la recherche menée de son chercheur correspondant, une personne avec des subjectivités individuelles, je considère que, pour des raisons personnelles et particulières, et aussi en raison de leur pertinence scientifique et sociale, un bon nombre des algorithmes MPC proposés étaient axés sur la contrôle de deux classes spécifiques de systèmes complexes : (i) la production d'énergie renouvelable et (ii) les technologies de mobilité urbaine. Les deux principaux problèmes qui apparaissent avec l'application de la commande prédictive pour ces procédés sont de savoir comment gérer les non-linéarités du modèle et la complexité numérique accrue de la loi de commande résultante, qui devient trop numérique pour être implémentée en temps réel, sous échantillonnage strict. seuils de période. Pour répondre à ces préoccupations, le formalisme LPV fournit un ensemble d'outils bien adaptés, comme discuté précédemment.

Le problème qui apparaît avec l'application de la commande prédictive pour ces systèmes est précisément lié à la complexité numérique accrue des algorithmes de contrôle, qui ne peuvent se permettre d'être mis en œuvre sous des seuils de période d'échantillonnage stricts, dans de nombreux cas.

Étant donné que les procédés d'énergie renouvelable, c.f.[Pipino et al. 2020b; Bernardi et al. 2021], sont, en général, des systèmes non linéaires. Ainsi, ils peuvent être directement représentés par des structures LPV en utilisant des plongements appropriés. La principale préoccupation en matière de contrôle est de maximiser l'efficacité des énergies renouvelables, et donc de réduire les émissions de gaz à effet de serre et l'utilisation de combustibles fossiles. Pour cela, les algorithmes de contrôle prédictif LPV sont des solutions adéquates, car ils sont capables d'intégrer des données météorologiques dans les prédictions et d'améliorer adéquatement les performances qui en résultent.

Les véhicules autonomes sont au centre des préoccupations sociétales, compte tenu de l'importance du sujet de la mobilité urbaine moderne. Ces systèmes peuvent souvent être représentés à l'aide de modèles LPV, en particulier lorsque l'on considère la dynamique verticale, comme détaillé dans [Savaresi et al. 2010; Morato et al. 2018a; Morato et al. 2019b; Morato, Normey-Rico, and Sename 2021c]. De plus, ils sont contrôlés, en général, par le moyen de microcontrôleurs embarqués qui génèrent de nouvelles actions de commande toutes les 1 à 10 millisecondes, ce qui rend le MPC non linéaire "à part entière" peu pratique. En ce sens, l'approche LPV MPC apparaît comme une alternative abordable en termes de calcul.

Je souligne que ces deux sujets sont au cœur de l'Agenda 2030 des Nations Unies [United Nations General Assembly 2015; United Nations 2018], qui propose les Objectifs de Développement Durable (ODD) afin de prévenir une catastrophe socio-environnementale de grande ampleur et à l'échelle mondiale [Löwy 2015]. Je n'aborderai pas ce sujet avec beaucoup d'attention, car il est discuté dans d'autres travaux parallèles, c.f. [Morato et al. 2018b]. Néanmoins, je souligne que l'ODD 7 concerne la production d'énergie propre et abordable

[Nathwani and Kammen 2019], tandis que l'ODD 11 traite des villes et des communautés durables avec des liens cohésifs et universels mobilité [Hermelin and Henriksson 2022]. Par conséquent, il me semble évident que ces sujets sont d'importance sociale et, par conséquent, élargissent également la portée possible de ce travail doctoral.

Comme dernier commentaire concernant les applications envisagées, je note que des résultats ont été fournis en considérant également d'autres types de systèmes, tels que des procédés industriels (par exemple, un réacteur à cuve à agitation continue, Chapitre 3), des systèmes robotiques (par exemple, un pendule inversé, Chapitre 3), jouets, systèmes de référence orientés vers l'éducation (par exemple, réservoirs en cascade, Chapitre 5) et installations de capteurs solaires thermiques (suivi de la température, Chapitre 6).

Disposition du manuscrit

Tenant compte du débat précédent, je présente les grandes lignes de cette thèse, en détaillant ses principaux apports et la manière dont chaque partie et chapitre est structuré. Le travail se compose de quatre parties, de telle sorte que chacun des objectifs principaux de cette thèse est abordé par une partie individuelle. La première partie comprend l'introduction, les préliminaires et les bases développements, les deux parties intermédiaires portent sur les objectifs premiers, tandis que la dernière partie rappelle et discute l'ensemble des travaux fournis ici. Ces parties comprennent les sujets suivants :

- Partie I: "Preamble":
  - Dans cette première partie de la thèse, je présente le contexte principal du travail, les objectifs envisagés et sa structure. Aussi, dans le chapitre 2, j'aborde les préliminaires de tout ce qui suit, établissant le cadre théorique de cette thèse, rappelant les concepts de base, les définitions et les résultats de la littérature. Plus précisément, ce chapitre commence par présenter une représentation non linéaire unifiée, caractérisant les notions de base, tels que la stabilité. Je présente les normes et espaces de signaux utilisés au cours des développements qui en découlent, ainsi que certaines procédures de synthèse, opérations et les transformées. Ensuite, je détaille les procédures d'intégration qui peuvent être utilisées pour générer des modèles LPV et qLPV (en utilisant des inclusions différentielles), et discute de certaines propriétés de cette représentation caractéristique. Concernant l'application de MPC pour ces systèmes, je donne plus de détails sur la pertinence de synthétiser de tels algorithmes, en discutant des principales complications qui en découlent. En conséquence, je récapitule les résultats clés sur la faisabilité récursive et la stabilité du résultat en boucle fermée, qui servent à générer ce que l'on appelle l'optimisation des "ingrédients terminaux", ainsi que la présentation les notions de base de la dissipativité.
- Partie II: "Gain-scheduled formulations". Dans cette partie, j'aborde le sujet des approches de contrôle prédictif à gain programmé, ciblant les objectifs (i) et (ii) de cette thèse. Plus précisément, les sujets suivants sont débattus :

- Comme discuté précédemment, l'application des algorithmes LPV MPC devient assez compliquée car la trajectoire d'ordonnancement, le long de l'horizon de prédiction MPC, est, *a priori*, inconnue. Ainsi, dans le chapitre 3, je discute des stratégies disponibles qui peuvent être utilisées pour générer des estimations de la trajectoire d'ordonnancement, et des avantages (et inconvénients) qui découlent de chaque approche, dans le contexte de MPC. De plus, je présente un nouveau algorithme d'extrapolation, qui estime les valeurs futures de ces paramètres d'ordonnancement qLPV pour un horizon de prédiction fixe de N_p étapes. La méthode est dérivée d'un simple argument de Taylor, et des conditions suffisantes sont présentées pour des estimations convergentes (répondant ainsi à l'objectif (i)). De plus, en utilisant différents exemples de référence de simulation tirés de la littérature, j'illustre et discute de la manière dont les différentes approches d'estimation peuvent être appliquées dans la pratique et je les compare.

- Dans le chapitre 4, je développe deux algorithmes différents de contrôle prédictif à ordonnancement de gain formalisés à l'aide de structures de retour d'état et de sortie, ce qui concerne l'objectif (ii) de la thèse:

  * Dans un premier temps, je propose un schéma de contrôle pour l'amélioration du confort des passagers à bord d'un véhicule à suspensions semi-actives, dans le cadre du formalisme MPC. Pour cela, la dynamique verticale de la voiture modélisée dans un cadre qLPV et, par conséquent, le contrôlé découle de la solution d'une optimisation sous contraintes, qui intègre un indice de performance de confort. La méthode est sous-optimale car la synthèse considère une approche de trajectoire d'ordonnancement figée. Quoi qu'il en soit, en supposant des limites sur les taux de variation des paramètres d'ordonnancement, la méthode permet de remplacer l'optimisation non linéaire complexe d'origine par un programme quadratique beaucoup plus simple, qui comprend un coût décroissant de Lyapunov et des ingrédients terminaux basés sur des ensembles. Des simulations non linéaires réalistes réussies d'une voiture à l'échelle un cinquième avec des suspensions électro-rhéologiques sont présentées, pour lesquelles la méthode proposée est testée et comparée à d'autres contrôleurs optimaux. Les résultats illustrent le bon fonctionnement global du véhicule ; le confort des passagers est considérablement amélioré, mesuré par des indices temporels et fréquentiels.

  * Dans un second temps, je développe un output-feedback approche à gain programmé pour les systèmes non linéaires représentés sous les modèles qLPV d'entrée-sortie. Pour cela, je détaille comment le modèle input-output peut générer une prédiction basée sur un estimation de la trajectoire d'ordonnancement future. Ensuite, je démontre la stabilité asymptotique du système en boucle fermée (et l'établissement du suivi de sortie correspondant). Le procédé comprend une action intégrale pour chaque canal d'entrée-sortie, assurant ainsi un suivi avec une erreur en régime permanent nul. Avec l'aide d'un benchmark de simulation numérique, je démontre l'efficacité de la solution.

- Partie III: "Robust synthesis". Dans cette partie de la thèse, ma principale préoccupation est de présenter de nouveaux algorithmes de contrôle prédictif robustes pour les systèmes LPV, c'est-à-dire l'objectif (iii). En conséquence, je considère des formulations avec à la fois des ingrédients terminaux et des arguments de dissipativité utilisés pour assurer la faisabilité récursive et la stabilité entrée-état (les formulations sont sous représentation de retour d'état). La question de la suivi des signaux de consigne constants par morceaux est également abordée. Plus précisément, ces sujets sont structurés comme suit :

  – Dans le chapitre 5, je présente une formulation de suivi NMPC pour des signaux de référence constants par morceaux utilisant l'incorporation de qLPV et l'extrapolation de trajectoire d'ordonnancement. Le cadre proposé est capable d'éviter les pertes de faisabilité dues à de grandes variations de consigne, qui sont suivies grâce à une variable cible réalisable artificielle, dont la distance à la consigne réelle est minimisée par un coût de décalage supplémentaire. A chaque période d'échantillonnage, un problème d'optimisation est résolu sur la base de prédictions linéaires (programmées); la ténacité numérique moyenne est comparable à un programme d'optimisation quadratique. La satisfaction robuste des contraintes est obtenue avec des zonotopes qui propagent l'incertitude. Ces ensembles sont calculés par rapport aux limites d'avance de l'erreur d'estimation de la séquence d'ordonnancement qLPV, offrant un conservatisme réduit. La stabilité en boucle fermée et la faisabilité récursive sont fournies avec des ingrédients terminaux robustes dépendant des paramètres. Afin d'illustrer les performances de la méthode, je fournis un exemple de référence, qui démontre que l'algorithme est en effet capable d'assurer un suivi de référence avec une demande numérique réduite par rapport aux techniques de l'état-de-l'art.
  – Bénéficiant de l'algorithme d'extrapolation récursive de Chapitre 3 et théorie de la dissipativité, dans le Chapitre 6, je propose un MPC robuste méthode capable de rattraper les performances poussives obtenues avec les schémas robustes de la littérature en incorporant les bornes sur les erreurs d'estimation des paramètres d'ordonnancement. Des arguments complémentaires de dissipativité sont utilisés pour démontrer la faisabilité récursive et stabilité du système en boucle fermée, démontrée par la solution d'un remède à l'inégalité matricielle linéaire, qui détermine la zone d'attraction pour laquelle la stabilité entrée-état est assurée. Je considère la température non linéaire problème de régulation d'un capteur solaire plat comme étude de cas. Par le biais d'une simulation numérique réaliste, je compare la méthode proposée à d'autres algorithmes robustes de la littérature, le démontrant comme une alternative intéressante, avec des calculs allégés en charge numérique.

- Partie IV: "Closure": dans cette dernière partie, je présente un panorama global de mon travail doctoral, où les objectifs sont rappelés et mis en regard des avancées présentées et des résultats obtenus. Un aperçu général de tout ce qui a été développé dans ce travail est rappelé et les développements sont analysés individuellement en termes d'avantages et de limites. Dans cette dernière partie du manuscrit, je taquine également les perspectives des fils d'investigation ouverts mentionnés dans cette introduction, mettant en lumière des voies qui restent à poursuivre dans les recherches futures.

Apports et principales propositions

Les principales contributions de cette thèse ont été centrées sur les chapitres 3 à 6. Ainsi, je détaille les principaux thèmes, propositions et avancées présentés dans chacun de ces chapitres.

Chapitre 3

Il existe, de nos jours, plusieurs algorithmes NMPC efficaces basés sur des *solvers* [Zhang, Li, and Liao 2019; Rathai et al. 2018; Gros et al. 2020]. Néanmoins, ces méthodes sont inventées en utilisant des approximations du problème d'optimisation non linéaire qui se pose lors de l'application de MPC non linéaire. Les formulations NMPC exactes nécessitent la solution en ligne de ces NP, qui ne peut généralement pas être résolue pour les petits périodes d'échantillonnage (dans le cas de systèmes de contrôle embarqués rapides). En somme, l'application en temps réel des algorithms NMPC pour les procédés a un obstacle inhérent à la résolution du NP exact lors de la mise en œuvre en ligne, comme l'ont souligné toutes les principales revues systématiques sur le sujet [Camacho and Bordons 2007; Allgöwer and Zheng 2012].

Tout au long de ce travail, je soutiens avec insistance qu'une approche élégante pour résoudre ce problème (revivre la dureté de calcul de NMPC) est de remplacer le modèle de système non linéaire par un modèle LPV. Ainsi, comme indiqué dans [Morato, Normey-Rico, and Sename 2020a], en général, l'optimisation résultante peut être exécutée beaucoup plus rapidement, puisque le NP d'origine est remplacé par un QP (ou un SQP, dans certains cas).

En effet, l'application de MPC aux systèmes LPV devient d'un intérêt total pour la commande de procédés non linéaires variant dans le temps. Cependant, il s'accompagne d'une complication intrinsèque : les paramètres dits d'ordonnancement, qui coordonnent la dynamique LPV, sont *a priori* inconnus sur un horizon de prédiction futur.

Dans ce chapitre, je débat de cette question, dans le contexte de la conception à gains programmés. Plus précisément, je détaille les quatre principales approches disponibles qui peuvent être utilisées pour calculer des estimations pour les futures trajectoires d'ordonnancement (et également comment faire fonctionner l'algorithme MPC correspondant). Ces méthodes sont :

(i) Estimations *figées*, lorsque le contrôleur suppose une prédiction comme si les paramètres resteraient constants le long de l'horizon, comme déployé dans [Morato, Sename, and Dugard 2018; Alcalá, Puig, and Quevedo 2019; Morato et al. 2020e] ;

(ii) Les estimations basées sur l'identification, pour lesquelles des équations auto-régressives sont utilisées, modélisent le comportement des futures variables d'ordonnancement, comme proposé dans [Morato, Normey-Rico, and Sename 2019];

(iii) Ides règles tératives, qui génèrent des estimations basées sur des itérations séquentielles de l'optimisation MPC, par échantillon, exploitant la relation connue entre les variables d'ordonnancement et les variables endogènes du système futur (uniquement possible pour le cas qLPV). Cette approche est l'état-de-l'art actuel dans la littérature qLPV MPC, c.f.[Cisneros, Sridharan, and Werner 2018; Cisneros and Werner 2019; Cisneros and Werner 2020];

(iv) Schémas d'extrapolation, qui génèrent des estimations basées sur une simple condition d'expansion de Taylor (encore une fois, uniquement possible pour le cas qLPV), comme proposé à l'origine dans [Morato, Normey-Rico, and Sename 2022b] et également appliqué dans [Morato, Normey-Rico, and Sename 2021b; Morato et al. 2023a; Morato 2023].

A cet égard, les principaux éléments abordés dans ce chapitre sont les suivants :

1. Je discute des quatre alternatives disponibles pour planifier les estimations de trajectoire, pour les modèles LPV et qLPV, en discutant de leurs principales caractéristiques, des hypothèses requises et des propriétés.
2. Plus de détails sont donnés concernant la méthode d'extrapolation pour les trajectoires des paramètres d'ordonnancement qLPV. Plus précisément, je présente des conditions suffisantes qui permettent un schéma d'extrapolation convergent, conçu en ce qui concerne la forme et la classe de la fonction d'ordonnancement et la robustesse de l'algorithme MPC correspondant.
3. Je présente également plusieurs exemples de référence de simulation différents issus de la littérature, afin d'illustrer et de discuter de leurs principales caractéristiques. Ces résultats sont également comparés aux techniques de pointe.

Je souligne que les développements présentés dans ce chapitre correspondent (en partie) aux travaux publiés dans [Morato, Normey-Rico, and Sename 2021c; Morato, Normey-Rico, and Sename 2022a; Morato 2023; Morato et al. 2023a; Morato, Normey-Rico, and Sename 2023a]. Plus précisément, la méthode d'extrapolation (approche (iv)) est l'une des principales contributions dérivées de cette thèse, telle que formalisée en profondeur dans [Morato, Normey-Rico, and Sename 2022b]. Ce sujet comprend, en fait, l'objectif (i) de cette thèse. Je note également qu'une autre contribution de ce travail est la procédure récursive des moindres carrés présentée dans [Morato, Normey-Rico, and Sename 2019] (approche (i)).

Approche (i) : Trajectoires d'ordonnancement figées

La première approche vue dans la littérature pour traiter le problème d'indisponibilité de la trajectoire d'ordonnancement P_k repose sur une idée assez simple : du point de vue du fonctionnement MPC, à chaque instant d'échantillonnage, je suppose que les variables d'ordonnancement resteront constantes (donc, *figées*). Autrement dit, P_k est remplacé par $\left[\begin{array}{ccc} \rho(k)^T & \dots & \rho(k)^T \end{array}\right]^T$ pour générer les prévisions d'état futur X_k. Dans de nombreuses applications pratiques avec des paramètres d'ordonnancement à variation lente, comme le cas des systèmes d'énergie renouvelable [Pipino et al. 2020b; Morato et al. 2020e], cette méthode est plutôt intéressante, car la propagation de l'incertitude le long de l'horizon de prédiction est négligeable.

Pourtant pertinent dans la pratique, le schéma MPC résultant qui est généré lors de l'utilisation de trajectoires d'ordonnancement figées peut souffrir de graves problèmes liés à la robustesse. Rappelons que les paramètres d'ordonnancement rentrent dans le modèle de prédiction de manière multiplicative. Ainsi, lorsque le MPC est basé sur un modèle nominal sous forme de $x(k+j|k) = A(\rho_k)x(k+j-1|k) + B(\rho_k)u(k+j-1|k)$, $\forall j \in \mathbb{N}_{[1,N_p]}$. Alors, l'incertitude de prédiction modèle-procédé correspondante est :

$$\begin{aligned} \mu(k+j) &= A(\rho(k+j))x(k+j-1|k) - A(\rho_k)x(k+j-1|k) \\ &+ B(\rho(k+j))u(k+j-1|k) - B(\rho_k)u(k+j-1|k), \forall j \in \mathbb{N}_{[1,N_p-1]}, \end{aligned} \tag{D.14}$$

qui dépend non seulement des paramètres d'ordonnancement futurs réels $\rho(k+j)$, mais aussi de l'état futur et des trajectoires d'entrée.

À des fins de discussion plus approfondie, en exploitant la dépendance affine sur les matrices du système, on obtient d'incertitude de modèle suivante:

$$\begin{aligned} \mu(k+j) &= A(\rho(k+j) - \rho_k)x(k+j-1|k) \\ &+ B(\rho(k+j) - \rho_k)u(k+j-1|k), \forall j \in \mathbb{N}_{[1,N_p-1]}, \end{aligned} \tag{D.15}$$

où, pour le cas des variations des paramètres d'ordonnancement bornés sur des échantillons, c'est-à-dire $\delta\rho(k) = \rho(k+1) - \rho(k) \in \delta\mathcal{P}$, il s'ensuit que :

$$\|\rho(k+j) - \rho_k\| \leq j\overline{\delta\rho}, \tag{D.16}$$

qui se traduit par :

$$\|\mu(k+j)\| \leq \left(A((N_p-1)\overline{\delta\rho})\overline{x} + B((N_p-1)\overline{\delta\rho})\overline{u}\right). \tag{D.17}$$

En pratique, le MPC correspondant, afin de stabiliser le système contrôlé, devrait tolérer, d'une façon robuste, ce terme d'incertitude borné $\mu(k+j)$. Néanmoins, sous des hypothèses relativement simples, l'incertitude qui survient augmente non seulement en ce qui concerne toutes les variables endogènes, mais aussi en ce qui concerne la taille de l'horizon de prédiction, comme le donne l'Éq. (D.17). Assez facilement, les bornes sur $\mu(k+j)$ peuvent être encore plus grandes que l'espace d'état disponible, mettant ainsi fin à tout certificat de performance possible d'un MPC synthétisé sur la base d'un modèle LPV figée.

Synthétiquement, l'alternative de conception dite figée LPV MPC présente un inconvénient théorique majeur : pour qu'elle assure la stabilité en boucle fermée, une robustesse excessive peut devoir être impliquée, ce qui rétrécit ainsi la région d'attraction correspondante activée par les contrôleurs. D'un point de vue philosophique, afin de concevoir un algorithme LPV MPC figé qui assure la stabilité, même sous des hypothèses standard, je cherche en fait à déterminer un contrôleur unique capable de stabiliser (quadratiquement) **tous** les modèles LTI dans le de $x(k+1) = A(\rho_k)x(k) + B(\rho_k)u(k)$, généré par des valeurs fixes de $\rho_k \in \mathcal{P}$. De plus, le contrôleur doit s'assurer que la stabilité est maintenue même si ces modèles changent dans les échantillons, ce qui est, bien sûr, un problème difficile.

Dans tous les cas, les algorithmes MPC pour les systèmes LPV décrits sur la base de trajectoires d'ordonnancement figées ont une grande valeur pratique. Pour le cas des systèmes LPV avec des paramètres d'ordonnancement variant lentement, c'est-à-dire pour de petites bornes sur $\|\delta\rho(k)\|$, cette approche est standard, et largement exploitée dans la littérature. Ce qui est fait, dans de nombreux travaux, est simplement de négliger l'existence de la propagation d'incertitude $\mu(k)$, en supposant que le MPC permettra une boucle fermée robustement stable vis-à-vis de cette variable.

Datant d'un article théorique original de 2003, [Casavola, Famularo, and Franzè 2003], de nombreux travaux axés sur la pratique ont été présentés en utilisant cette méthode de prise en compte de modèles LPV figés à chaque échantillon, c.f. les articles récents suivants [Cisneros and Werner 2017b; Alcalá, Puig, and Quevedo 2019; Rodriguez-Guevara et al. 2021; Cavanini, Ippoliti, and Camacho 2021]. Je peux même affirmer, à partir de la récente enquête sur les algorithmes MPC utilisant des modèles LPV, telle que présentée dans [Morato, Normey-Rico, and Sename 2020a], que l'approche figée à gain programmé est la méthode **standard** vue dans la littérature, avec le plus grand nombre de applications pratiques en cours d'enregistrement.

Approche (ii) : Estimations basées sur l'identification

L'approche du modèle LPV gelé pour la synthèse MPC à gain programmé est assez standard dans la pratique. Néanmoins, comme mis en évidence, les performances peuvent être compromises lorsque les paramètres varient rapidement entre les périodes d'échantillonnage (et, dans le cas qLPV, si le contrôleur est trop agressif, obligeant les trajectoires d'état à se déplacer trop rapidement et, donc implicitement, faisant également varier les paramètres d'ordonnancement brusquement). Par conséquent, je détaille maintenant une deuxième alternative pour générer des estimations pour les trajectoires d'ordonnancement, qui est basée sur une procédure d'identification en ligne. Cette méthode a été proposée dans le cadre de cette thèse, et présentée dans [Morato, Normey-Rico, and Sename 2019]. Son objectif principal est de fournir une meilleure estimation que de simplement considérer les paramètres comme étant constants tout au long de l'horizon de prédiction.

Encore une fois, je considère que les variables d'ordonnancement ρ ne sont connues (ou mesurées) qu'à l'instant d'échantillonnage actuel k, ce qui signifie que l'ensemble du comportement futur de la trajectoire d'ordonnancement P_k est inconnu. Quoi qu'il en soit, je suppose que ces paramètres conviennent à un comportement dynamique linéaire auto-régressif dans le temps. Autrement dit, je considère que le signal $\rho(k)$ vérifie l'hypothèse suivante.

Assumption 25. *Il existe un modèle linéaire auto-régressif* Π*, sous la forme de l'Éq.* (D.18)*, qui cartographie le comportement des paramètres d'ordonnancement* $\rho(k)$ *du système LPV* *contrôlé.*

$$\begin{aligned} \rho(k+N_p) &= a_1\rho(k-(N_p-1)) + \cdots + a_{N_p}\rho(k) \\ &+ b_1\xi(k-(N_p-1)) + \cdots + b_{N_p}\xi(k)\,. \end{aligned} \tag{D.18}$$

En pratique, l'hypothèse 25 implique qu'il existe un modèle à temps discret avec N_p délais d'échantillonnage, qui donne la relation entre une entrée ξ et les variables d'ordonnancement ρ, c'est-à-dire qu'il existe un modèle linéaire transfer $\rho := \Pi\xi$, étant Π donne le modèle d'ordonnancement auto-régressif. Quelques commentaires sont présentés ci-dessous en ce qui concerne la variable d'entrée de ce transfert, dit ξ :

- Dans le cas du cas LPV pur, Π est activé par une variable exogène ξ, telle qu'un signal de coordination ou une entrée auxiliaire. Lorsque ces signaux ne sont pas disponibles, l'Éq. (D.18) peut être converti en un modèle auto-régressif pur, sans entrées, ou écrit par rapport à un bruit d'entrée donné dans une gamme de fréquences connue.
- Dans le cadre qLPV, l'hypothèse 25 est tout à fait raisonnable, puisque les paramètres d'ordonnancement, à chaque instant d'échantillonnage, sont fortement liés aux variables endogènes du système. Ainsi, selon quelles variables endogènes sont incluses dans la fonction d'ordonnancement, le signal ξ est choisi. Autrement dit, pour les fonctions liés à l'état $f_\rho(x(k))$, on peut simplement prendre $\xi(k) = x(k)$. De manière équivalente, pour le cas des fonctions liés à la sortie et à l'entrée $f_\rho(u(k), y(k))$, je prends $\xi(k) = \begin{bmatrix} u(k)^T & y(k)^T \end{bmatrix}^T$, et ainsi de suite.

Sous forme compacte, notons $\Xi_k = \begin{bmatrix} \xi(k)^T & \ldots & \xi(k+N_p-1) \end{bmatrix}^T$. Ainsi, en exploitant l'hypothèse 25, je peux écrire la relation compacte suivante pour Π :

$$\rho(k) = \Theta \overbrace{\begin{bmatrix} P^T_{(k-2N_p+1)} & \Xi^T_{(k-2N_p+1)} \end{bmatrix}}^{\Psi_{(k-2N_p+1)}}, \tag{D.19}$$

étant les paramètres du modèle auto-régressif compactés dans

$$\Theta := \begin{bmatrix} a_1 & \ldots & a_{N_p} & b_1 & \ldots & b_{N_p} \end{bmatrix}. \tag{D.20}$$

Ensuite, si Θ est connu, la relation auto-régressive linéaire peut être exploitée afin de couvrir l'ensemble des trajectoires d'ordonnancement futures, comme suit :

$$\left\{ \begin{aligned} \rho(k+1) &= a_1\rho(k-2N_p+2) + \cdots + a_{N_p}\rho(k-N_p+1) \\ &+ b_1\xi(k-2N_p+2) + \cdots + b_{N_p}\xi(k-N_p+1) \\ &= \Theta\Psi_{(k-2N_p+2)}, \end{aligned} \right. \tag{D.21}$$

$$\left\{ \begin{aligned} \rho(k+2) &= a_1\rho(k-2N_p+3) + \cdots + a_{N_p}\rho(k-N_p+2) \\ &+ b_1\xi(k-2N_p+3) + \cdots + b_{N_p}\xi(k-N_p+2) \\ &= \Theta\Psi_{(k-2N_p+3)}, \end{aligned} \right. \tag{D.22}$$

et ainsi de suite jusqu'à :

$$\begin{cases} \rho(k+N_p-1) & = & a_1\rho(k-N_p)+\cdots+a_{N_p}\rho(k-1) \\ & + & b_1\xi(k-N_p)+\cdots+b_{N_p}\xi(k-1) \\ & = & \Theta\Psi_{(k-N_p)}. \end{cases} \tag{D.23}$$

Afin de considérer des comportements plus génériques, avec des dynamiques variables dans le temps liées aux variables d'ordonnancement, j'implique que les paramètres du modèle auto-régressif Π varient dans le temps, et donc que $\rho(k)$, = $\Theta(k)\Psi_{(k-2N_p+1)}$. Ensuite, le mécanisme basé sur l'identification fonctionne comme suit :

1. A chaque instant d'échantillonnage, on collecte le vecteur empilé des variables d'ordonnancement précédentes et des entrées auxiliaires, c'est-à-dire $\Psi_{(k-2N_p+1)}$;
2. Ensuite, une procédure de minimisation en ligne récursive des moindres carrés est résolue afin d'estimer les paramètres $\Theta(k)$, c'est-à-dire :

$$\Theta(k) \quad = \quad \Theta(k-1)+\lambda Q_\theta\left(\Psi_{(k-2N_p+1)},\rho(k),\xi(k)\right) , \tag{D.24}$$

   où λ est un paramètre de mise à jour (facteurs d'oubli moindres carrés) et Q_θ est une fonction de mise à jour.

3. La trajectoire future des paramètres d'ordonnancement P_k est estimée sur la base de :

$$P_k \quad = \quad \begin{bmatrix} \rho(k) \\ 0 \\ \vdots \\ 0 \end{bmatrix} + \Theta(k) \begin{bmatrix} 0 \\ \Psi_{(k-2N_p+2)} \\ \vdots \\ \Psi_{(k-N_p)} \end{bmatrix} . \tag{D.25}$$

Discutons quelques aspects de cette approche basée sur l'identification, avant de présenter des résultats de simulation :

- La méthode est conçue sur la base de l'hypothèse 25, qui peut être partiellement fausse pour de nombreux systèmes. Dans de nombreuses applications LPV pures, le concepteur n'a pas accès à un certain signal d'activation ξ et donc la seule alternative est de considérer le modèle Π comme auto-régressif et autonome, ou sujet à des entrées bruitées ;
- En pratique, lorsque les paramètres sont considérés comme variant dans le temps et ré-identifiés en ligne, au moyen d'une procédure LS récursive, des estimations précises peuvent être formulées ;
- Les estimations sont générées au moyen d'opérations linéaires, les Eqs. (D.24)-(D.25), qui permettent des applications rapides et en temps réel de l'algorithme MPC correspondant ;

- Pourtant, avec une grande valeur pratique et des résultats empiriques assez satisfaisants (comme indiqué ci-dessous), la méthode manque de preuves rigoureuses de convergence (c'est-à-dire que je ne peux pas garantir que les estimations pour $\rho(k+j)$, $\forall j \in \mathbb{N}_{[1,N_p-1]}$ suivent les valeurs correctes des futurs paramètres d'ordonnancement), et l'erreur d'estimation est, *a priori*, illimitée (je n'ai pas des moyens appropriés pour calculer les limites d'erreur ou pour garantir un modèle générique auto-régressif, puisque cela change d'un système à l'autre).

Approche (iii) : Mécanisme d'estimation itératif

Les méthodes précédentes (approche figée et mécanisme auto-régressif basé sur les moindres carrés) représentent des alternatives pratiques intéressantes pour résoudre le problème de l'indisponibilité des trajectoires d'ordonnancement, nécessaires pour calculer les prédictions MPC. Ces deux approches ont des contreparties empiriques avec de bons résultats, comme enregistré dans la littérature. Néanmoins, avec la méthode basée sur le gel, approche (i), j'obtiens une propagation d'incertitude corrélée qui peut croître de manière significative, tandis qu'avec le mécanisme basé sur LS, approche (ii), je suis incapable de quantifier l'ampleur de l'erreur d'estimation (et la validité et l'existence d'un modèle auto-régressif).

Ensuite, je détaille une autre alternative largement utilisée pour générer P_k en ligne, proposée à l'origine dans [Cisneros, Voss, and Werner 2016] et appliqué pour de nombreuses applications depuis lors, c.f. [Cisneros and Werner 2017a; Cisneros, Sridharan, and Werner 2018; Cisneros and Werner 2019; Abbas et al. 2019; Cisneros and Werner 2020]. Le mécanisme est basé sur l'opération itérative de l'optimisation MPC comme base pour générer la séquence d'ordonnancement. L'approche nécessite que le modèle soit quasi-LPV, puisque la fonction d'ordonnancement $f_\rho(\cdot)$ est utilisé pour générer les estimations P_k. Étant donné que le mécanisme opère n_{iter} itérations par échantillon, le MPC correspondant présente une complexité de calcul d'un SQP.

L'idée centrale du mécanisme d'estimation itérative est la suivante :

- À chaque échantillon, j'ai besoin de P_k pour résoudre l'optimisation MPC, qui donne en sortie la séquence de contrôle future optimale $U_k^\star$ et la séquence d'état prédite $X_k^\star$;
- Ensuite, au lieu de résoudre l'optimisation MPC une seule fois par échantillon, elle est résolue n_{iter} fois : les sorties de l'optimisation d'une itération (U_k^l et X_k^l, où le super-indice l désigne l'instance d'itération) sont utilisés pour générer la trajectoire d'ordonnancement de l'exemple suivant, en utilisant : $P_k^l = f_\rho\left((X_k^{l-1})^\perp, U_k^{l-1}\right)$, où le vecteur $(X_k^{l-1})^\perp = \begin{bmatrix} x(k)^T & (X_k^{l-1})^T \end{bmatrix}^T$ collecte la mesure de l'état actuel et les prédictions d'état à la l-ième itération du schéma, la dernière entrée étant supprimée ;
- Les itérations se poursuivent jusqu'à l'obtention de la convergence (un certain seuil limite est atteint, c'est-à-dire $\|P_k^l - P_k^{l-1}\| \leq \xi_P$) ou o nombre maximal d'itérations est atteint, soit $l = n_{\text{iter}}$;

- Notez que, si la convergence des estimations de trajectoire d'ordonnancement est effectivement atteinte, c'est-à-dire que $\lim_{l\to+\infty} P_k^l$ est égal à la trajectoire d'ordonnancement **réele** P_k, puis la solution de la MPC correspondante exacte (pas d'écarts de prédiction du modèle), et donc la même que celle qui aurait été obtenue avec une NMPC « complète » (sous convexité du coût et des contraintes d'optimisation) ;

Comme discuté en détail dans [Cisneros and Werner 2020], les preuves empiriques montrent que cette approche garantit la convergence de la trajectoire d'ordonnancement prédite P_k, à chaque échantillon k, vers le véritable comportement d'ordonnancement. De plus, cette propriété est généralement obtenue en un nombre relativement faible d'itérations par échantillon (environ 5-10 itérations internes 1), pour une grande variété de systèmes. Des preuves pratiques ont été présentées dans [Cisneros et al. 2019], qui explique comment l'algorithme MPC résultant devient une alternative compétitive à la mise en œuvre de NMPC (fonctionnant embarqué aussi rapidement que ACADO, ou CasADi, pour de nombreux procédés). Contrairement à ce qui a été vu à propos de la méthode précédente, ces propriétés ont également été vérifiées théoriquement.

La principale contrainte de cette approche d'estimation d'ordonnancement est que la boucle interne peut prendre plusieurs itérations pour converger. Pour les systèmes rapides, cela n'est pas souhaitable car le nombre d'itérations internes l nécessaires à la convergence (ou jusqu'à ce que le critère d'arrêt $l \leq n_{\text{iter}}$ soit atteint) peut théoriquement nécessiter plus temps supérieur au seuil de période d'échantillonnage disponible (et donc la convergence peut ne pas être garantie). De plus, selon le type de non-linéarité présenté dans la fonction d'ordonnancement $f_\rho(\cdot)$, le calcul de chaque entrée de P_k^l, en utilisant $f_\rho(x(k+j-1|k), u(k+j-1|k))$, $\forall j \in \mathbb{N}_{[1,N_p]}$, peuvent également être numériquement coûteux.

J'insiste sur les principaux avantages du mécanisme:

- À chaque échantillon, l'optimisation MPC est formulée comme un QP, et donc la complexité de calcul de l'algorithme MPC complet est équivalente à celle d'un SQP, par échantillon ;

- L'application du mécanisme est relativement simple : il ne nécessite que le décalage vectoriel et l'application de la fonction d'ordonnancement sur N_p entrées vectorielles ;

- Des preuves empiriques montrent que la méthode converge en un nombre relativement petit d'itérations, pour de nombreux systèmes différents.

Approche (iv) : Schéma d'extrapolation basé sur Taylor

Ensuite, je présente la dernière approche discutée dans ce chapitre, qui est, en fait, l'une des principales contributions de cette thèse, discutée en profondeur dans [Morato, Normey-Rico, and Sename 2022b]. Cette méthode n'est, comme la précédente, possible que pour le paramètre qLPV, puisque lea fonction d'ordonnancement $f_\rho(\cdot)$ est utilisé pour générer l'estimation de la trajectoire d'ordonnancement.

L'approche proposée est **récursive**, dans le sens où les trajectoires d'ordonnancement sont générées au moyen de la loi suivante :

$$\hat{P}_k = \Phi\left(\hat{P}_{k-1}, \rho(k), x(k)\right) . \quad \text{(D.26)}$$

Ainsi, dans la suite, j'explique comment l'opérateur $\Phi(\cdot)$ peut être linéaire, et comment les vecteurs estimés convergent vers l'ordonnancement correct comportement de trajectoire, dans un nombre fini d'échantillons. Notez que la procédure dans l'Éq. (D.26) génère une nouvelle extrapolation pour les trajectoires d'ordonnancement P_k, à l'instant k, basée sur l'extrapolation précédente et le nouveau jeu de données disponible ($\rho(k)$ et $x(k)$). Pour cela, l'approche récursive proposée est conçue à partir de l'hypothèse suivante :

Assumption 26. *La fonction statique $f_\rho(x)$ peut être approchée par une expansion de Taylor de premier ordre autour de $\overline{x}$:*

$$f_\rho(x) \approx f_\rho(x)|_{\overline{x}} + \left.\frac{\partial f_\rho}{\partial x}\right|_{\overline{x}} (x - \overline{x}) , \quad \text{(D.27)}$$

étant $\overline{x}$ un point de linéarisation arbitraire. La fonction réelle peut être exprimée analytiquement par la somme de cette approximation à un signal résiduel ξ_ρ, qui hérite de l'écart entre la carte statique réelle et son approximation de Taylor :

$$f_\rho(x) = f_\rho(x)|_{\overline{x}} + \left.\frac{\partial f_\rho(x)}{\partial x}\right|_{\overline{x}} (x - \overline{x}) + \xi_\rho . \quad \text{(D.28)}$$

Considérez que l'hypothèse 26 est valide. Alors, l'expression suivante est valide, en considérant la linéarisation à un instant donné $k+j-1$ et l'incrément le long de x jusqu'à l'instant suivant $k+j$, soit $\Delta x(k+j-1)$:

$$\begin{aligned} f_\rho(x(k+j)) &= f_\rho(x(k+j-1)) + \xi_\rho(k+j-1) \\ &+ \left.\frac{\partial f_\rho(x)}{\partial x}\right|_{x(k+j-1)} \Delta x(k+j-1) . \end{aligned} \quad \text{(D.29)}$$

Je désigne désormais $f_\rho^\partial(k+j-1) = \left.\frac{\partial f_\rho}{\partial x}\right|_{x(k+j-1)}$. En développant l'expression dans l'Éq. (D.29) le long de l'horizon de prédiction fixe de N_p étapes et en l'intégrant à la fonction d'ordonnancement $\rho(k) = f_\rho(x(k))$ donne :

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

Puisque $\rho(k)$ et $\Delta x(k)$ sont connus, alors que $f_\rho^\partial(k)$ peut être évalué numériquement sur la base du courant mesure d'état $x(k)$. Néanmoins, en pratique, $f_\rho^\partial(k+j)$ pour $j \in \mathbb{N}_{[1,N_p-2]}$ est inconnu, qui nécessite une seconde supposition:

Assumption 27. *Pour simplifier, à chaque instant d'échantillonnage k, on suppose que la dérivée partielle $f_\rho^\partial(k)$ reste constante le long de la prédiction horizon, c'est-à-dire $f_\rho^\partial(k+j) = f_\rho^\partial(k)\,,\, \forall\, j \in \mathbb{N}_{[1,N_p-2]}$.*

Les termes des dérivées partielles $f_\rho^\partial(k+j)$ pourraient être calculés sur la base de la prédiction de trajectoire d'état X_k (généré par l'algorithme MPC). Néanmoins, le faire c'est numériquement coûteux. Ainsi, j'exploite l'hypothèse 27 afin de rendre notre procédure d'extrapolation rapide et bon marché numériquement, prenant ainsi $f_\rho^\partial(k+j) = f_\rho^\partial(k)$. Dans la suite, je montre que même en utilisant une telle approximation, la convergence est toujours assurée.

Notez que les expansions le long de l'horizon de prédiction peuvent être données en fonction de la valeur du paramètre de programmation précédent et d'un terme de correction, comme suit :

$$\rho(k+j|k) = \rho(k+j-1|k-1) + f_\rho^\partial(k)\Delta x(k+j-1) + \xi_\rho(k+j|k)\,, \tag{D.30}$$

Par conséquent, il retient que :

$$\hat{P}_k = \hat{P}_{k-1}^\star + f_\rho^\partial(k)\Delta X_k^\star + \Xi_k\,, \tag{D.31}$$

qui est une loi d'estimation récursive de la mode dans l'Éq. (D.26).

Basé sur l'Éq. (D.31), je fournis des résultats théoriques pour la convergence de l'estimation générée (i.e. $\hat{P}_k$ convergeant vers P_k) et pour la délimitation de l'erreur d'estimation (i.e. chaque $\rho(k+j|k) - \rho(k+j)$).

Remarques générales

Par le biais d'une large utilisation de la simulation non linéaire (haute fidélité), en considérant les systèmes de référence de la littérature, l'efficacité et les principales caractéristiques de chacun des mécanismes d'estimation de trajectoire d'ordonnancement sont débattues. Les remarques générales sont :

- L'approche (i), basée sur une estimation figée des trajectoires d'ordonnancement, fournit des performances suffisantes, dans de nombreux cas. Il est capable de fonctionner rapidement en ligne, puisque le MPC résultant est basé uniquement sur un seul QP. Néanmoins, le contrôleur doit être excessivement robuste pour tolérer les incertitudes de prédiction, ce qui peut ainsi conduire à des performances conservatrices.

- L'approche (ii) est une alternative pour améliorer les performances du mécanisme de modèle figé. Il est plus approprié pour le réglage qLPV, car le mécanisme d'identification nécessite, en général, un signal d'activation du transfert auto-régressif (comme des variables endogènes qui affectent les paramètres d'ordonnancement). Quoi qu'il en soit, on est incapable de s'assurer que la méthode converge effectivement, et les bornes sur l'erreur d'estimation ne sont pas non plus disponibles. Le MPC résultant a la complexité d'un QP et d'une solution récursive LS (linéaire), étant possible pour les applications en temps réel.

- L'approche (iii) a été largement appliquée dans des travaux récents, y compris des variations MPC à base de tubes plus robustes, c.f. [Abbas et al. 2019; Hanema, Tóth, and Lazar 2021]. La méthode a de nombreuses preuves empiriques de la convergence des estimations de trajectoire d'ordonnancement, ayant ainsi une importance précieuse. Néanmoins, il est garanti que la méthode converge rapidement (dans le seuil de la période d'échantillonnage), et il ne peut pas non plus être généré de bornes d'erreur d'estimation. Comme en témoignent les résultats précédents, ainsi que dans [Cisneros and Werner 2020], le MPC résultant est comparable aux solutions NMPC de pointe de référence, tout en fonctionnant avec une complexité en ligne de SQP.

- L'approche (iv), l'une des principales contributions de cette thèse, est une procédure d'estimation récursive basée sur un argument d'expansion de Taylor assez simple. Avec cette méthode, on est capable d'assurer la convergence de l'estimation, tant que cinq conditions suffisantes sont satisfaites. De plus, des limites théoriques sur l'erreur d'estimation pendant le comportement transitoire sont disponibles, ce qui peut être pris en compte dans le contexte d'une conception MPC robuste. De plus, le procédé est capable d'améliorer les performances du MPC par rapport à l'approche (i), tout en maintenant une charge de calcul proche de celle d'un seul QP, étant ainsi plus rapide que l'approche (iii). Le mécanisme d'extrapolation récursive repose sur de simples opérateurs linéaires, avec une dureté numérique croissante linéairement avec la taille de l'horizon de prédiction N_p et le nombre de variables d'ordonnancement n_ρ.

- Étant donné que l'approche (iii) a une ténacité numérique qui croît de manière exponentielle avec la taille de l'horizon de prédiction N_p et avec le nombre d'états du système n_x, dans le cas de systèmes en temps réel d'ordre supérieur, l'approche (iv) peut être beaucoup plus approprié, car il nécessite beaucoup moins de charge de calcul.

En résumé, le Chapitre 3 a présenté un aperçu de la méthode disponible qui peut être utilisée pour estimer les futurs paramètres d'ordonnancement LPV, afin de construire des lois de prédiction dans le contexte de MPC. Quatre méthodes ont été détaillées, avec les résultats de simulation correspondants présentés et discutés. Plus important encore, une nouvelle approche est présentée pour le cas des systèmes qLPV, résidant dans une loi récursive simple et rapide, qui ne nécessite que l'évaluation d'un calcul de dérivée partielle à chaque instant d'échantillonnage. De plus, cinq conditions suffisantes simples à vérifier sont présentées pour la convergence de ce mécanisme.

Comme indiqué dans la précédente, l'approche développée est comparée au mécanisme de pointe d'estimation des paramètres d'ordonnancement via les SQP (bouclage du MPC plusieurs fois), montrant des estimations équivalentes et un taux de convergence similaire. Des comparaisons et des discussions corroborantes sont également disponibles dans [Morato et al. 2021d; Morato et al. 2022b].

Dans l'ensemble, je note que les schémas d'estimation de trajectoire d'ordonnancement décrits peuvent certainement servir à la conception d'algorithmes rapides LPV MPC. De plus, ceux-ci peuvent être exploités pour la conception de NMPC au moyen d'intégrations LPV : le contrôleur résultant a une complexité de calcul inférieure à celle d'un programme non linéaire, car les non-linéarités des contraintes de prédiction du modèle sont supprimées de manière al LPV.

Chapter 4

Dans le Chapitre 4, je discute de l'exploitation de la conception de MPC LPV à gain programmé sous de nouvelles formulations de retour d'état et de sortie. En conséquence, je présente les deux principales évaluations[1]:

- Dans un premier temps, je propose un algorithme d'amélioration du confort des passagers d'un véhicule à suspensions semi-actives, synthétisé sous une formulation de contrôle prédictif. L'application MPC considérée prend en compte un modèle qLPV de la dynamique de la voiture et intègre un indice de performance de confort comme fonction de coût. La méthode proposée est sous-optimale du fait qu'elle est basée sur une estimation figée des paramètres d'ordonnancement le long de l'horizon. Les bornes sur les taux de variation des paramètres d'ordonnancement sont prises en compte et, ainsi, la propagation de l'incertitude est atténuée. Le problème d'incertitude est traité à l'aide d'ingrédients terminaux basés sur des ensembles. Des simulations non linéaires réalistes réussies d'une voiture à l'échelle sont présentées, comparant les solutions développées à d'autres contrôleurs optimaux. Les résultats illustrent le bon fonctionnement général du véhicule ; le confort des passagers est sensiblement amélioré, tel que mesuré par des indices temporels et fréquentiels.
- Deuxièmement, je présente un algorithme MPC pour les systèmes qLPV représentés sous la forme entrée-soritie (IO). La méthode est basée sur le mécanisme d'extrapolation récursive basé sur Taylor du chapitre 3. La principale innovation est que, en utilisant une description IO de la dynamique du système, les mesures d'état ne sont pas nécessaires, ce qui est intéressant du point de vue de l'application industrielle et pratique (pas besoin de conception d'observateur, par exemple). Afin d'assurer un suivi de référence sans décalage, l'algorithme comprend une formulation d'action intégrale explicite qui, couplée à des ingrédients terminaux quadratiques, permet également une stabilité IO asympto-

[1] Les développements présentés dans ce chapitre correspondent (en partie) aux travaux publiés dans [Morato, Normey-Rico, and Sename 2019; Morato, Sename, and Dugard 2019b; Morato, Normey-Rico, and Sename 2020c; Morato, Normey-Rico, and Sename 2021c] (formulation retour d'état) et [Morato, Normey-Rico, and Sename 2022a; Morato 2023] (formulation retour de sortie).

tique. Un exemple de référence numérique est utilisé pour illustrer les avantages de la méthode proposée, ainsi que ses capacités en temps réel.

En résumé, je souligne les principales remarques concernant chaque approche (formulations LPV MPC à retour d'état et de sortie):

1. Retour d'état:

   - Il est montré comment un algorithme MPC peut être utilisé pour l'amélioration du confort des passagers à l'aide de systèmes de suspension semi-actifs. La suspension est modélisée dans un cadre qLPV, et la force d'amortissement est modélisée par une fonction tangente hyperbolique non linéaire, comme suggéré par la littérature. Le procédé intègre les non-linéarités dans un paramètre d'ordonnancement, qui est estimé à travers l'horizon de prédiction à chaque instant d'échantillonnage. La conjecture d'évolution d'ordonnancement fixe (c'est-à-dire, figée du point de vue de la commende) est utilisée pour transformer le problème de prédiction non linéaire en un QP linéaire, qui peut être résolu en quelques millisecondes. Un ensemble de contrôle invariant positif terminal robuste est utilisé dans le contexte de MPC afin qu'il soit capable de maintenir la stabilité (quadratique) et la faisabilité récursive, malgré les inadéquations modèle-procédé (dans le cas de la régulation). Ces propriétés sont démontrées analytiquement. Il est démontré que la fonction de coût d'optimisation du MPC intègre des contraintes de confort, en ce qui concerne les indices de performance de la littérature. L'algorithme est appliqué avec succès au contrôle d'un système de suspension SA via une simulation réaliste, obtenant de bons résultats par rapport aux méthodes de contrôle optimales existantes.
   - La principale limite de la méthode réside dans la prise en compte de l'estimation figée et la complexité du calcul des ingrédients basés sur un ensemble. En outre, la méthode utilise des variables d'état, qui ne sont généralement pas disponibles dans des situations pratiques.

2. Retour de sortie:

   - Tel qu'élaboré et débattu tout au long de cette thèse, on peut concrètement affirmer que le Model Predictive Control est aujourd'hui une technique largement utilisée, avec une pertinence pratique et un intérêt théorique. Les schémas MPC utilisant des modèles de procédés en espace-d'états ont fait l'objet de recherches considérables au cours des dernières années, avec de nombreux résultats enregistrés. Des MPC basés sur des modèles d'espace d'état peuvent être élaborés pour répondre à des exigences de performances rigoureuses. De plus, sous des hypothèses fades, ces algorithmes permettent des performances optimales en boucle fermée. Pourtant, avec une grande valeur théorique, la conception standard des MPC à retour d'état nécessite la disponibilité de mesures d'état en temps réel. À leur tour, les variables d'état sont souvent difficiles à mesurer ou à estimer avec précision. De plus, les schémas d'estimation d'état peuvent détériorer considérablement les performances en boucle fermée de MPC en présence de perturbations et de contraintes. En raison de ces problèmes, les réalisations de l'espace d'états sont en deçà des attentes industrielles, qui sont rarement ancrées dans les descriptions de procédé du type entrée-sortie.

- En conséquence, ce problème est résolu par la proposition d'un nouvel algorithme MPC pour les systèmes LPV représentés sous la forme entrée-sortie. Les futurs paramètres d'ordonnancement LPV sont extrapolés à l'aide d'une loi d'expansion récursive de Taylor, qui génère les matrices de prédiction MPC à chaque période d'échantillonnage.
- La méthode proposée possède des capacités temps réel, puisque sa mise en œuvre en ligne est uniquement liée à un opérateur linéaire (extrapolation des paramètres d'ordonnancement) et à la solution d'un seul QP. Comme le montrent les exemples, la loi de commande peut être générée en quelques millisecondes avec des *solvers* standards.
- La méthode est capable de résoudre le problème de contrôle de suivi des systèmes hautement non linéaires, tant qu'un modèle d'entrée-sortie peut être généré. De plus, la méthode ne nécessite aucun outil de suivi de référence supplémentaire (comme les variables de référence artificielles).
- Les ingrédients terminaux proposés, qui garantissent la faisabilité récursive et la stabilité asymptotique de la sortie, sont activés par des solutions LMI quadratiques et peuvent donc être facilement calculés avec des *solvers* standard.

Chapitre 5

Dans le Chapitre 5, je présente un NMPC pour le suivi de signaux de référence constants par morceaux. Ce schéma est efficace en termes de calcul en raison de l'utilisation d'une réalisation d'intégration qLPV pour décrire la dynamique non linéaire. En conséquence, je profite de la procédure d'extrapolation basée sur Taylor (Chapitre 3, approche (iv)) afin d'estimer le comportement futur des paramètres d'ordonnancement avec une erreur d'estimation bornée. A chaque période d'échantillonnage, le problème d'optimisation utilise des prédictions linéaires et présente donc une ténacité numérique allégée (comparable à un QP). Bénéficiant de variables cibles artificielles, la méthode est également capable d'éviter les pertes de faisabilité dues à de grandes variations de consigne. Des certificats robustes de satisfaction des contraintes, de stabilité en boucle fermée et de faisabilité récursive sont fournis, grâce aux zonotopes de propagation d'incertitude et aux ingrédients terminaux dépendant des paramètres. Enfin, un exemple de référence est utilisé pour illustrer l'efficacité de la méthode, qui est comparée aux techniques de l'état de l'art.

Je souligne que les développements présentés dans ce chapitre correspondent à ceux présentés dans [Morato et al. 2021e] (asservissement) et [Morato et al. 2023a] (suivi de consigne). Les principales nouveautés incluses sont résumées ci-dessous :

1. Le NMPC de suivi proposé est basé sur des intégrations qLPV, qui permettent des prédictions de modèles linéaires. Le modèle est exploité par un mécanisme d'extrapolation

qui fournit la séquence complète des futurs paramètres d'ordonnancement, à chaque instant d'échantillonnage. En conséquence, je calcule des limites simples sur l'erreur de prédiction à partir de ces réalisations qLPV. De plus, je propose des zonotopes qui bornent la propagation d'incertitude correspondante, qui sont ensuite utilisés pour la satisfaction de contraintes robustes.

2. Je propose des ingrédients terminaux robustes dépendant des paramètres pour le NMPC proposé. Ces outils assurent la faisabilité récursive de la procédure d'optimisation et la stabilité de la dynamique de l'erreur de poursuite, en considérant toute valeur de consigne dans un ensemble prédéfini. De plus, je propose une optimisation supplémentaire pour le choix de la variable de référence artificielle, avec une complexité allégée.

3. En considérant un système de réservoir en cascade de référence, je compare minutieusement la méthode proposée au cadre de suivi nominal NMPC de [Limon et al. 2018]. Il est démontré que le schéma proposé atteint des performances de suivi très similaires, avec un stress de calcul beaucoup plus faible, bénéficiant des prédictions linéaires permises par la réalisation de qLPV. La méthode est prête pour les applications embarquées (la contrainte en ligne est similaire à celle d'un QP) et offre une robustesse vis-à-vis des perturbations de charge bornées avec un conservatisme réduit.

En résumé des résultats obtenus par rapport au MPC LPV développé dans ce chapitre, je list les principaux caractéristiques :

- Avantages:
  1. Il est capable de fonctionner plus rapidement que les MPC non linéaires de pointe pour le suivi, étant donné qu'une procédure d'optimisation à complexité réduite est utilisée (de complexité similaire à QP);
  2. Il inclut des variables de référence artificielles telles que même des objectifs de référence inaccessibles peuvent être directement pris en compte par le contrôleur ;
  3. Il inclut des arguments de robustesse (faciles à calculer), écrits en termes de limites connues des perturbations de charge ;
  4. Des certificats de faisabilité et de stabilité récursives sont disponibles, ce qui garantit un comportement adéquat de la boucle fermée résultante.
- Désavantages:
  1. It nécessite une réalisation qLPV du système non linéaire, et donc la disponibilité d'une fonction connue $f_\rho(\cdot)$ qui génère des variables d'ordonnancement bornées (et respecte également l'hypothèse de Lipschitz locale).
  2. Comme dans de nombreux algorithmes de suivi NMPC, les variables d'état doivent être mesurables, car le contrôleur garantit le suivi de la sortie en orientant les états vers des variables d'état stable données.
  3. Le suivi d'ensembles d'invariants positifs robustes doit être calculé hors ligne, avant l'implémentation en ligne du contrôleur, afin d'assurer des comportements corrects de la boucle fermée résultante.

Chapitre 6

Dans le chapitre 6, je propose un schéma MPC robuste et dissipatif pour les systèmes non linéaires représentés avec des modèles qLPV. La principale nouveauté réside dans le fait de bénéficier de l'approche d'extrapolation récursive (Chapitre 3, approche (iv)) afin de rattraper les performances (habituellement) poussives obtenues avec les schémas min-max robustes de la littérature.

Les bornes sur les erreurs d'estimation des paramètres d'ordonnancement par N_p sont prises en compte afin de formuler un problème min-max en ligne avec des incertitudes réduites : dans un premier temps, un CP contraint est résolu afin de déterminer le niveau de propagation d'incertitude le plus défavorable et, par la suite, un deuxième QP contraint est résolu pour minimiser cette fonction de coût dans le cas le plus défavorable par rapport au vecteur de séquence de commande. J'examine comment, puisque les bornes sur l'erreur d'estimation pour les paramètres d'ordonnancement sont généralement beaucoup plus petites que les bornes sur le paramètre d'ordonnancement réel, la prudence de la solution est assez réduite.

La faisabilité récursives et la stabilité de l'algorithme proposé sont démontrées avec des arguments de dissipativité donnés sous la forme d'un remède LMI, qui détermine également la zone d'attraction dans laquelle la stabilité entrée-état est certifiée.

Le problème de régulation non linéaire de la température d'un capteur solaire plat est considéré comme une étude de cas. En utilisant un benchmark de simulation réaliste, la technique proposée est comparée à d'autres algorithmes robustes min-max LPV MPC de la littérature, se révélant numériquement efficaces, tout en conservant de bonnes performances.

Les développements présentés dans ce chapitre correspondent à ceux publiés dans les ouvrages suivants : [Pipino et al. 2020b] (application capteur solaire), [Morato, Normey-Rico, and Sename 2021b] arguments de dissipativité, et [Morato, Normey-Rico, and Sename 2021d; Morato, Normey-Rico, and Sename 2023c] (métriques de robustesse induites en boucle fermée) . Au regard des résultats obtenus, quelques points clés sont soulignés :

- Les NMPC de programmation non linéaire à part entière ne sont pas applicables aux applications embarquées de procédés avec des taux d'échantillonnage rapides, car le temps moyen nécessaire pour résoudre le NP est généralement supérieur à la période d'échantillonnage disponible. La littérature récente a montré comment les méthodes NMPC approchées (telles que CaSaDi, GRAMPC et ACADO [Quirynen et al. 2015]) et les algorithmes MPC intégrant qLPV [Cisneros and Werner 2020] sont capables de résoudre efficacement un problème de contrôle aussi complexe en quelques millisecondes.
- Pour l'étude de cas considérée, les performances de suivi de référence obtenues avec la méthode MPC min-max d'intégration qLPV proposée sont équivalentes à ces méthodes MPC non linéaires modernes rapides [Cisneros and Werner 2020], telles qu'évaluées par les indices RMS et IAE. L'opérabilité numérique de la méthode proposée est similaire aux travaux précédents [Quirynen et al. 2015; Cisneros and Werner 2020]. Je constate que la complexité du problème croît avec l'ordre du système.

- La méthode proposée résout le problème de programmation convexe de maximisation par rapport à l'erreur concernant l'estimation des paramètres d'ordonnancement le long de l'horizon de prédiction. J'insiste sur le fait que n'importe quel type d'algorithme avec des erreurs d'estimation bornées pourrait être utilisé à la place du développement de Taylor proposé dans ce chapitre. Une option alternative et élégante pourrait être l'utilisation du mécanisme itéré [Cisneros and Werner 2020], qui utilise la séquence d'états calculée avec la minimisation QP pour calculer l'évolution de ρ le long de l'horizon.
- La méthode proposée est comparée à deux algorithmes keystone min-max LPV MPC de la littérature [Cao and Lin 2005; Li and Xi 2010], qui considèrent respectivement que ρ peut varier arbitrairement à l'intérieur de $\mathcal{P}$ et considèrent des taux bornés de variations pour ρ. Comme les variations des paramètres d'ordonnancement et de son ensemble convexe sont assez importantes pour l'application considérée, les résultats obtenus avec ces méthodes sont assez médiocres. L'incertitude fixée avec la méthode proposée est beaucoup plus faible (d'un facteur centième). De plus, au fil du temps, la méthode d'extrapolation obtient de meilleures estimations de ρ, ce qui fait également converger vers zéro la sortie d'incertitude du problème de maximisation, à mesure que les trajectoires d'état convergent.
- Enfin, la méthode a assuré la stabilité entrée-état pour un plus grand domaine régional $\mathcal{X}_{ISS}$. Cette propriété est assurée ainsi que la faisabilité récursive à travers un cadre de vérification de la dissipativité, résolu via des LMI. Je note que l'avantage de ce cadre est qu'il ne nécessite pas l'utilisation d'ingrédients terminaux (contraintes et coûts) sur le problème d'optimisation, ce qui peut être assez difficile à calculer en ligne pour les systèmes LPV. Par conséquent, la fonction de coût MPC est quadratique sur x et u (et assez simple), ce qui permet son fonctionnement rapide.

Remarques finales

Afin de conclure ce résumé, je présente un large aperçu des développements et applications conçus tout au long de cette thèse. Je souligne que tous les travaux dérivés le long de la croissance de cette thèse, ont été réalisés sous une rigueur scientifique stricte, avec un dévouement exclusif. En termes de pertinence sociale de cette thèse, j'ai fait un effort pour mener et aligner mes recherches en mettant l'accent sur la facilitation de deux problèmes sociaux concrets : (i) la production d'énergie renouvelable abordable et (ii) l'efficacité technologies de la mobilité urbaine. En exploitant l'application d'algorithmes de contrôle prédictif de modèles pour ces deux classes de systèmes complexes (décrits au moyen de modèles LPV), j'ai cherché à contribuer, à un certain niveau, en favorisant des possibilités pratiques en boussole avec l'Agenda 2030 et la dotation du Développement Durable Objectifs 7 et 11, visant à atténuer la calamité socio-environnementale en cours découlant des changements climatiques. À ce sujet, je présente deux derniers messages :

(i) En ce qui concerne l'optimisation et la commande des systèmes de production d'énergie renouvelable, j'ai étudié comment la commande prédictive des modèles peut servir à maximiser l'efficacité de la production d'énergie dans différents systèmes (en particulier les micro-réseaux et les centrales solaires). Dans le contexte brésilien, j'ai étudié comment les industries nationales de la canne à sucre peuvent être explorées en tant que paradigme pratique (et économiquement plausible) pour tirer parti de la production d'énergie renouvelable dans le pays, si elles sont couplées à des panneaux photovoltaïques et des éoliennes et coordonnées à l'aide de MPC. J'ai également évalué comment MPC peut être utilisé comme approche principale pour améliorer les performances des systèmes de capteurs solaires, sous des contraintes de temps réel. En conséquence, j'ai démontré comment cette méthode de contrôle peut conduire à un bénéfice maximisé de la disponibilité solaire. Ces développements ont ainsi abordé l'ODD 7, qui traite de la production d'énergie propre et accessible à tous. Principaux ouvrages sur ce sujet : [Morato et al. 2020e; Morato et al. 2020b; Pipino et al. 2020b; Bernardi et al. 2021; Morato et al. 2021c].

(ii) En ce qui concerne la mobilité urbaine automatisée, je rappelle que l'ODD 11 mentionne explicitement à quel point les villes et les communautés durables nécessitent des solutions permettant un contrôle efficace du trafic automobile, telles que des véhicules connectés qui peuvent agir activement dans la prévention des accidents. En conséquence, en ce qui concerne ce sujet, j'ai développé de nouveaux algorithmes MPC qui améliorent le confort des passagers dans les véhicules modernes. En exploitant la formulation LPV, ces algorithmes sont capables de fonctionner de manière embarquée, dans des microcontrôleurs embarqués, fonctionnant à l'échelle de quelques millisecondes. Par ailleurs, j'ai également proposé des stratégies de contrôle prédictif pour la conduite assistée de véhicules semi-autonomes, avec des interventions intelligentes évitant les pertes de stabilité et les accidents. La question de la conduite automatisée dans des situations défectueuses a également été étudiée. Principaux ouvrages sur ce sujet :[Morato, Normey-Rico, and Sename 2019; Morato, Normey-Rico, and Sename 2020c; Morato, Normey-Rico, and Sename 2021c; Medero et al. 2022; Morato et al. 2022a].

Dans cette thèse, j'ai abordé le problème de la commande des systèmes linéaires à paramètres variants (et non linéaires aussi, sous des structures à paramètres variables quasi-linéaires). Pour cela, j'ai développé plusieurs schémas de contrôle prédictif de modèles, sous trois types d'exploitation différents : en utilisant l'ordonnancement de gain et la synthèse robuste. Chacune de ces catégories de contrôleurs prédictifs a été synthétisée et proposée avec des outils d'analyse théorique correspondants qui permettent une stabilité en boucle fermée et une faisabilité récursive de l'optimisation du contrôleur.

En ce qui concerne les premières contributions de cette thèse, je me suis concentré sur de nouveaux résultats utilisant l'ordonnancement des gains (en termes d'analyses et de procédures de conception). C'est-à-dire des procédures pour lesquelles le contrôleur formulé variait en fonction du gain instantané (échantillonné) du système LPV contrôlé. Dans ces méthodes, synthétiquement, la variable d'ordonnancement $\rho(k)$ est utilisée pour coordonner l'optimisation (et donc la politique de contrôle prédictif dérivée) à chaque échantillon k. Les valeurs futures des variables d'ordonnancement le long de l'horizon de prédiction, c'est-à-dire $\rho(k+j), \forall j \in \mathbb{N}_{[1,N_p]}$, sont généralement négligées (considérées comme figées) ou remplacé par une estimation précise $\rho(k+j|k)$. Au moyen des algorithmes développés, j'ai pu démontrer comment l'ordonnancement du gain peut être très utile dans la pratique, et comment le MPC non linéaire peut être efficacement résolu au moyen d'intégrations qLPV appropriées. Le principal point fort est qu'un contrôle prédictif non linéaire correspondant peut être rendu via la solution de, généralement, un seul QP par échantillon, ce qui est beaucoup plus rapide que la résolution d'un programme non linéaire. Notez que les QP ont une complexité qui croît linéairement, en général, en ce qui concerne n_u, n_x et N_p (nombre d'entrées de commande, états du système et taille de l'horizon de prédiction), tandis que la croissance du paramètre non linéaire est **exponentielle**.

Comme deuxième branche d'avancées, j'ai formulé des méthodes MPC robustes. Celles-ci, contrairement aux formulations d'ordonnancement du gain, considéraient non seulement les variables d'ordonnancement futures, mais aussi l'erreur d'estimation à leur égard, c'est-à-dire $\xi_\rho(k+j|k) = (\rho(k+j) - \rho(k+j|k))$. Ainsi, dans le cadre robuste, les algorithmes proposés incluaient des outils supplémentaires pour gérer la propagation de l'incertitude correspondante (programme de maximisation et réduction des contraintes à l'aide de zonotopes). En utilisant ces fonctionnalités, j'ai pu démontrer des certificats de performance garantis et une satisfaction des contraintes, malgré la propagation des incertitudes dues aux variables d'ordonnancement inconnues. Je souligne que la question du suivi des signaux de référence (éventuellement inaccessibles) a également été résolue de manière robuste.

Message principal

En général, je soutiens que cette thèse sert et contribue au contexte de la conception MPC non linéaire en soutenant le message suivant : l'approche consistant à utiliser des modèles LPV (et des intégrations qLPV) afin de modéliser des dynamiques non linéaires et variant dans le temps peut servir de support. pour concevoir des algorithmes NMPC capables de fonctionner en temps réel. En exploitant les caractéristiques de linéarité le long des canaux d'entrée-sortie de ces descriptions LPV, j'ai pu montrer, avec de multiples exemples, comparaisons et options de synthèse, comment l'algorithme MPC résultant est très similaire à celui d'une usine LTI. De plus, les certificats de stabilité et de robustesse pour le contrôleur résultant sont également très similaires à ceux du cas LTI, qui sont standard de nos jours. J'ai insisté sur trois points principaux :

1. Lorsque les trajectoires d'ordonnancement sont supposées constantes (figées), l'algorithme LPV MPC à gain programmé résultant est sous-optimal. Néanmoins, d'un point de vue pratique et empirique, ces contrôleurs sont en grande partie comparables à ceux du NMPC. Même si la précision de la prédiction à gain programmé se détériore à mesure que la taille de l'horizon augmente, le mécanisme de fenêtres glissantes du MPC atténue intrinsèquement ce problème. En pratique, l'approche est pertinente, facile à mettre en œuvre, et permet de résoudre la problématique des applications temps réel. Dans de nombreux cas, ce type d'approche est plus que suffisant.

2. Si l'on cherche à utiliser des mécanismes d'estimation pour les trajectoires

d'ordonnancement futures (comme l'approche basée sur Taylor proposée ici, ou le schéma itératif de [Cisneros and Werner 2020]), les performances sont améliorées. Comme ces mécanismes assurent la **convergence** des trajectoires d'ordonnancement estimées, la boucle fermée résultante est **équivalente** à celle obtenue avec un algorithme NMPC (après quelques pas en temps discret). C'est une avancée majeure puisque l'optimisation est ainsi basée sur un modèle de prédiction **exact**, c'est-à-dire qu'aucune approximation n'est nécessaire ! De plus, par rapport aux techniques basées sur la linéarisation, la précision de la prédiction est bien meilleure (alors que la convergence n'a pas été établie), puisque les erreurs apparaissent sur les paramètres d'ordonnancement du modèle plutôt que sur les trajectoires d'état complètes, comme ce qui se produit avec la pratique Lois NMPC.

3. Dans le contexte de MPC robuste, l'approche LPV est également intéressante : j'ai évalué comment on peut choisir d'utiliser des mécanismes de resserrement des contraintes, compte tenu de la propagation de l'incertitude résultante (en tenant compte à la fois de la perturbation de charge bornée et de l'erreur d'estimation des paramètres d'ordonnancement), conduisant ainsi à des performances en boucle fermée qui ne sont pas excessivement conservatrices (pourtant les préparations hors ligne sont difficiles à synthétiser), ou à des approches min-max, avec des résultats plus conservateurs mais des préparations plutôt simples.

En guise de commentaire final, je souligne que de nombreux travaux scientifiques sont issus des développements de cette thèse. Ceux-ci comprennent plus de vingt articles publiés dans des revues internationales, dix-neuf actes de conférence et quelques autres soumissions actuellement en cours d'examen.

Appendix E

Abstracts

Predictive Control Methods for Linear Parameter Varying Systems

Abstract — This thesis discusses in detail the application of Model Predictive Control (MPC) strategies for Linear Parameter Varying (LPV) models. LPV models, just as standard Linear Time-Invariant (LTI) ones, exhibit linear relationships along the input and output channels, together with time-varying state transitions, which are coordinated by the so-called scheduling parameters. Thanks to input-output linearity along suitable coordinates, the theoretical analysis of LPV dynamics falls within the scope of the LTI toolkit, represented by Lyapunov storage functions, Ricatti equalities and inequalities. In addition, by the means of differential inclusions, LPV realisations are able to describe a wide variety of nonlinear and time-varying behaviours. Therefore, the synthesis of LPV controllers is of great scholastic interest.

Model predictive control, in turn, is a widely recognised and established approach, both in the academic and industrial contexts. This control paradigm is enabled by the solution of an optimization problem at each discrete-time instant. In general, MPC schemes have been based on LTI models, since when nonlinear dynamics are accounted for, the resulting optimisation problem becomes numerically complex (NP-hard), with an exponential growth of the numerical burden with respect to the number of states and size of the prediction horizon. As of this, recent research has been focused on solver-based approximations in order to apply nonlinear MPC in real-time (under the millisecond range), such as real-time iteration methods (ACADO) or gradient and Lagrange-based frameworks (GRAMPC).

With regard to this context, this thesis focuses on the exploitation of LPV realisations in order to develop exact (nonlinear) MPC strategies with fact computation, without the need for any kind of optimisation approximation. Nevertheless, the main issue that arises is that, when LPV models are used for the synthesis of predictive controllers, the optimization program derived at each sampling instant requires the future trajectories of the scheduling parameters, along the prediction horizon.

Accordingly, one of the main contributions of this thesis is a novel strategy to estimate these scheduling trajectories, based on a simple Taylor expansion mechanism. The proposed extrapolation mechanism offers convergence guarantees and bounded residuals. Thus, as a second contribution, by benefiting from these scheduling parameter estimates, novel gain-scheduled LPV MPC controllers are proposed. Specifically, the following elements are debated: (a) a state-feedback algorithm for semi-active suspension systems, and (b) an output-feedback formulation using input-output descriptions, with stability guarantees.

As a third contribution, robustified MPC algorithms are developed, considering the issue of uncertainties and disturbances. Specifically, two new frameworks are debated: (a) a reference tracking algorithm with robustness guarantees enabled through constraint tightening with zonotopes that propagate the uncertainties along the horizon; and (b) min-max approach with input-to-state stability assessments derived based on dissipativity arguments.

Along this work, several different case studies and realistic examples are used to demonstrate the effectiveness of the developed predictive control schemes. The resulting LPV MPC algorithms are shown to be of have great practical relevance: they allow optimal performances, comparable to those obtained with nonlinear predictive controllers, although with low computational cost, thus enabling for complex control applications in real-time.

Keywords: Model Predictive Control, Linear Parameter Varying Systems, Robust Control, Nonlinear Control, Tracking.

Méthodes de Commandes Prédictive pour les Systèmes Linéaires à Paramètres Variants

Résumé — Cette thèse discute en détail l'application des stratégies de Commande Prédictive basée sur Modèle (CPM) pour les procédés Linéaires à Paramètres Variables (LPV). Modèles LPV, tel comme les modèles linéaires stationnaires (LS), ont des relations linéaires entre les canaux d'entrées-sorties. Toutefois, ils ont des transitions d'état variantes dans le temps, coordonnées par les paramètres de planification. Grâce à la linéarité entrées-sorties, l'analyse théorique des dynamiques LPV s'inscrit dans le cadre des outils conçues originellement pour les modèles LS, représenté en grande partie par les fonctions de Lyapunov, les égalités et les inégalités de Ricatti. En plus, à travers de l'inclusion différentielle, les réalisations LPV modèles peuvent décrire une grande variétés des comportements non linéaires et variantes. Par conséquent, la synthèse de la commande LPV est d'un grand intérêt scolaire.

La commande prédictive basée sur modèle, quant à elle, représente une approche largement reconnue et établie, tant dans le contexte académique qu'industriel. Ces lois de commande sont pilotées à travers de la résolution d'un problème d'optimisation, à chaque échantillon discret. En général, les schémas CPM sont basés sur des modèles LTI, car lorsque la dynamique non linéaire est prise en compte, le problème d'optimisation résultant devient numériquement complexe (programmes de difficulté non linéaire), avec une croissance exponentielle de la charge numérique par rapport au nombre d'états et à la taille de horizon de prédiction. Pour cette raison, des recherches récentes se sont concentrées sur des approximations basées sur des outils afin d'appliquer MPC non linéaire en temps réel (sous la plage de la milliseconde), tel comme les méthodes d'itération (ACADO) ou les cadres de gradient et de Lagrange (GRAMPC).

Dans ce contexte, cette thèse se concentre sur l'exploitation des réalisations LPV afin de développer des stratégies CPM (non linéaires) exactes, avec un faible temps de calcul, sans avoir besoin d'aucune sorte d'approximation sur l'optimisation. Néanmoins, le principal problème qui se pose est que, lorsque des modèles LPV sont utilisés pour la synthèse de régulateurs prédictifs, le programme d'optimisation dérivé à chaque échantillonnage requière les trajectoires futures des paramètres d'ordonnancement, au fil de l'horizon de prédiction.

En conséquence, l'une des principales contributions de cette thèse est une nouvelle stratégie pour estimer ces trajectoires d'ordonnancement, basée sur un simple mécanisme d'expansion de Taylor. Le mécanisme d'extrapolation proposé offre des garanties de convergence et des résidus bornés. Ainsi, comme deuxième contribution, en bénéficiant de ces estimations de paramètres d'ordonnancement, de nouveaux algorithmes LPV MPC à gain programmé sont proposés. Plus précisément, les éléments suivants sont débattus : (a) l'approche retour-d'états pour les systèmes de suspension semi-actifs, et (b) une formulation retour de sortie en utilisant des descriptions d'entrée-sortie, avec des garanties de stabilité.

Comme troisième contribution, des algorithmes MPC robustes sont développés, prenant en compte la question des incertitudes et des perturbations. Plus précisément, deux nouveaux cadres sont débattus : (a) un algorithme de suivi de référence avec des garanties de robustesse rendues possibles grâce au resserrement des contraintes avec des zonotopes qui propagent les incertitudes le long de l'horizon ; et (b) l'approche min-max avec des évaluations de stabilité entrée-état dérivées sur la base d'arguments de dissipativité.

Tout au long de ce travail, plusieurs études de cas différentes et des exemples réalistes sont utilisés pour démontrer l'efficacité des schémas de commande développés. Ainsi, les algorithmes LPV MPC se révélent d'une grande pertinence pratique : ils permettent des performances optimales, comparables à celles obtenues avec des contrôleurs prédictifs non linéaires, bien qu'avec un faible coût de calcul, ce qui permet des applications embarquées des commandes complexes, tournant en temps réel.

Mots clés : Commande Prédictive basée sur Modèle, Systèmes Linéaires à Paramètres Variants, Commande Robuste, Commande Non linéaire, Suivi de Consigne.

Métodos de Controle Preditivo para Sistemas Lineares a Parâmetros Variantes

Resumo — Nesta tese, discute-se, em minúcia, a aplicação de estratégias de Controle Preditivo baseado em Modelo (CPM) para processos representados através de modelos Lineares a Parâmetros Variantes (LPV). Modelos LPV, tais como modelos lineares e invariantes no tempo (LIT), apresentam relações lineares entre os canais de entrada e de saída. Todavia, possuem transições de estados não lineares, coordenadas pelos parâmetros variantes. Graças a linearidade entradas-saídas, a análise teórica de dinâmicas LPV recai sobre o arcabouço ferramental LIT, representado em grande parte por funções de armazenamento de Lyapunov, igualdades e desigualdades de Ricatti. Ademais, através da inclusão diferencial, realizações LPV são capazes de descrever uma vasta gama de comportamentos não-lineares e variantes no tempo. Portanto, a síntese de controladores LPV é de sumo interesse escolástico.

Controle preditivo, por sua vez, é uma abordagem amplamente reconhecida e estabelecida, tanto no contexto acadêmico e industrial. Estas leis de controle são implementadas através da solução de um problema de otimização a cada amostra discreta. Em geral, as principais estratégias MPC têm sido baseados em modelos LTI, pois quando a dinâmica considerada é não linear, o problema de otimização resultante torna-se numericamente complexo (programas de dificuldade n ão linear), com um crescimento exponencial da carga numérica em relação ao número de estados e tamanho do horizonte de previsão. Por esta razão, pesquisas recentes têm se concentrado em aproximações baseadas em *solucionadores* que permitem a aplicação de de leis MPC não lineares em tempo real (dentro da faixa dos milissegundos), através de aproximações do problema de otimização, tais como métodos de iteração (ACADO) ou baseados em gradiente e argumentos de Lagrange (GRAMPC).

Com relação a este contexto, este tese tem como foco a exploração das diversas representações LPV para o desenvolvimento de algoritmos MPC (não lineares) exatos, com cômputo rápido, sem que haja a necessidade de qualquer tipo de aproximação da otimização. Entretanto, o principal problema que surge é que, quando modelos LPV são utilizados para a síntese de controladores preditivos, o programa de otimização resultante a cada instante requer as trajetórias futuras dos parâmetros de agendamento, ao longo do horizonte de predição

Portanto, uma das principais contribuições desta tese é uma nova estratégia para estimar tais trajetórias de agendamento, com base em um mecanismo simples de expansão de Taylor. O mecanismo de extrapolação proposto oferece garantias de convergência e termos residuais com norma limitada. Ademais, como uma segunda contribuição, beneficiando-se dessas estimativas dos parâmetros de agendamento, são propostos novos controladores LPV MPC com ganho escalonado. Especificamente, os seguintes elementos são debatidos: (a) um algoritmo de realimentação de estado para sistemas de suspensão semi-ativos, e (b) uma formulação de realimentação de saída usando representações entrada-saída, com garantias de estabilidade.

Como terceira contribuição, algoritmos de MPC robustos são desenvolvidos, considerando a questão das incertezas e de perturbações. Especificamente, dois novos arcabouços teóricos são discutidos: (a) um algoritmo de rastreamento de referência com garantias de robustez possibilitadas pelo aperto de restrições com base em zonotopos que propagam as incertezas ao longo do horizonte; e (b) uma abordagem min-max com avaliações de estabilidade de entrada para estado derivadas com base em argumentos de dissipatividade.

Ao longo deste trabalho, diversos estudos de caso e exemplos realistas são usados para demonstrar a eficácia das estratégias de controle desenvolvidas. Os algoritmos LPV MPC resultantes mostram-se de grande relevância prática: estes permitem desempenhos ótimos, comparáveis aos obtidos com controladores preditivos não lineares, porém com baixo custo computacional, possibilita, portanto, aplicações complexas embarcadas, operando em tempo-real.

Palavras-chave: Controle Preditivo. Sistemas Lineares a Parâmetros Variantes. Controle Robusto. Controle Não-Linear. Rastreamento de referência.

Métodos de Control Preditivo para Sistemas Lineales a Parámetros Variantes

Resumen — En esta tesis, se analiza en detalle la aplicación de estrategias de Control Predictivo basado em Modelo (CPM) a procesos representados a través de modelos Lineales con Parámetros Variantes (LPV). Los modelos LPV, como los modelos Lineales e Invariantes en el Tiempo (LIT), muestran relaciones lineales entre los canales de entrada y salida. Sin embargo, tienen transiciones de estado no lineales, coordinadas por los parámetros variantes. Gracias a la linealidad de entrada-salida, el análisis teórico de la dinámica LPV cae en el marco de herramientas LIT, representado en gran parte por las funciones de Lyapunov y las igualdades y desigualdades de Ricatti. Además, a través de la inclusión diferencial, las realizaciones de LPV pueden describir una amplia gama de comportamientos no lineales y variables en el tiempo. Por lo tanto, la síntesis de los controladores LPV es de gran interés escolástico.

El control predictivo basado en modelo, a su vez, es un enfoque ampliamente reconocido y establecido, tanto en el contexto académico como industrial. Este paradigma de control se implementa a través de la resolución de un problema de optimización a cada muestra discreta. En general, las principales estrategias de CPM se han basado en modelos LTI, ya que cuando se tiene en cuenta la dinámica no lineal, el problema de optimización resultante se convierte en numéricamente complejo (programas de dificultad no lineal), con un crecimiento exponencial de la carga numérica con respecto al número de estados y el tamaño de la horizonte de predicción. A partir de esto, la investigación reciente se ha centrado en soluciones basadas en aproximaciones del programa de optimización para aplicar CPM no lineal en tiempo real (en el rango de milisegundos), como métodos de iteración (ACADO) o gradientes y marcos basados en Lagrange (GRAMPC).

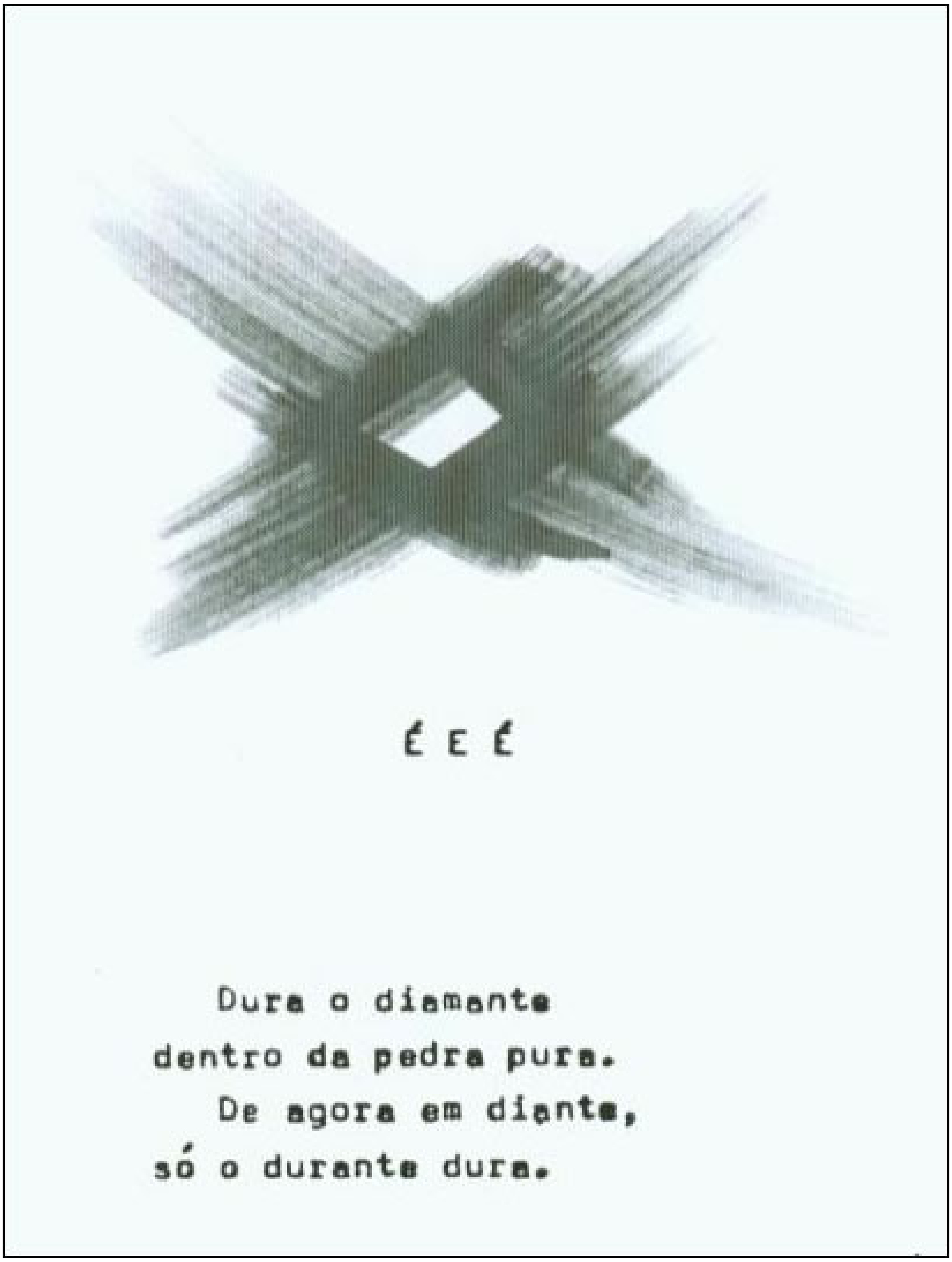
En este contexto, esta tesis se centra en la explotación de realizaciones LPV para desarrollar estrategias de CPM (no lineales) exactas con computado rápido y sin necesidad de ningún tipo de aproximación de optimización. Sin embargo, el problema principal que surge es que, cuando se utilizan modelos LPV para la síntesis de controladores predictivos, el programa de optimización derivado en cada muestreo requiere las trayectorias futuras de los parámetros variantes, a lo largo del horizonte de predicción.

En consecuencia, una de las principales contribuciones de esta tesis es una estrategia nueva para estimar estas trayectorias de los parámetros variantes, basada en un mecanismo simple de expansión de Taylor. El mecanismo de extrapolación propuesto ofrece garantías de convergencia y errores acotados. Por lo tanto, como segunda contribución, al beneficiarse de estas estimaciones de los parámetros variantes, se proponen nuevos controladores LPV MPC con ganancia programada. Específicamente, se debaten los siguientes elementos: (a) un algoritmo de retroalimentación de estado para sistemas de suspensión semiactivos, y (b) una formulación de retroalimentación de salida utilizando descripciones de entrada-salida, con garantías de estabilidad.

Además, como tercera contribución de esta tesis, se desarrollan algoritmos MPC robustos, considerando el tema de las incertidumbres y de las perturbaciones. Específicamente, se debaten dos nuevos marcos: (a) un algoritmo de seguimiento de referencia con garantías de robustez habilitadas mediante el ajuste de restricciones con zonotopos que propagan las incertidumbres a lo largo del horizonte; y (b) enfoque min-max con evaluaciones de estabilidad de entrada a estado derivadas en base a argumentos de disipatividad.

En esta obra, a lo largo de su desarrollo, se utilizan varios estudios de casos y ejemplos realistas para demostrar la eficacia de las estrategias de control estudiadas. Los algoritmos CPM LPV resultantes muestran una gran relevancia práctica: permiten un rendimiento óptimo, comparable al obtenido con controladores predictivos no lineales, pero con un bajo costo computacional, lo que permite aplicaciones complejas embarcadas, operando en tiempo-real.

Palabras Clave: Control predictivo. Sistemas lineales con parámetros variantes. Control robusto. Control No lineal. Seguimiento de referencia.



LEMINSKI, P.; SUPLICY, J. **Winterverno**. São Paulo: Iluminuras, 2001.